Measurement-Driven Simulation of Complex Engineering Systems

PhD Thesis Universiteit van Amsterdam

Arnoud Visser

23 October 2007

Lay out: typeset by the author using $Iate{T}_{E}X 2_{\varepsilon}$. Cover design: René Staelenberg. Cover illustration: Jeroen Bons.

ISBN: 978-90-5629-500-4 NUR: 980

© Vossiuspers UvA - Amsterdam University Press, 2007

All rights reserved. Without limiting the rights under copyright reserved above, no part of this book may be reproduced, stored in or introduced into a retrieval system, or transmitted, in any form or by any means (electronic, mechanical, photocopying, recording or otherwise) without the written permission of both the copyright owner and the author of the book.

Measurement-Driven Simulation of Complex Engineering Systems

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Universiteit van Amsterdam op gezag van de Rector Magnificus prof. dr. D.C. van den Boom ten overstaan van een door het college voor promoties ingestelde commissie, in het openbaar te verdedigen in de Aula der Universiteit op dinsdag 23 oktober 2007, te 10:00 uur

door

Arnoud Visser

geboren te Castricum.

Promotiecommissie

Promotor: Prof. dr. L.O. Hertzberger Overige leden: Prof. dr. ir. F. C. A. Groen Prof. dr. P. W. Adriaans Prof. drs. M. Boasson Prof. dr. ir. B. van Arem Dr. G. M. Meijer Dr. ir. S. J. Gelsema

Faculteit der Natuurwetenschappen, Wiskunde en Informatica



This research was carried out in the context of the Virtual Laboratory for e-Science project (www.vl-e.nl). This project is supported by a BSIK grant from the Dutch Ministry of Education, Culture and Science (OC&W) and is part of the ICT innovation program of the Ministry of Economic Affairs (EZ).



The work reported here is part of the research performed for the Interactive Collaborative Information Systems (ICIS) project, supported by the Dutch Ministry of Economic Affairs, grant nr: BSIK03024.

Contents

1	Intr	oductio	on	1
	1.1	Conte	ext	1
	1.2	Challe	enges	3
	1.3	Struct	ure of the thesis	6
2	Con	nplex S	bystems	7
	2.1	Introc	luction	7
	2.2	Cyber	metics	8
	2.3	Syster	m Theory	10
	2.4	Mode	ling and Simulation	13
	2.5	Concl	usion	16
3	Арр	olicatio	n	17
	3.1	Introc	luction	17
	3.2	Arche	etypes	19
		3.2.1	Example 1: distance-based charging	
			i.e. "Kilometerheffing"	21
		3.2.2	Example 2: regional and time dependent charging	
			i.e. "Rekening Rijden"	22
		3.2.3	Example 3: congestion tracks	
			i.e. "Almere - Amstelveen"	24
	3.3	Techn	ical Solutions	26
		3.3.1	Odometry	26
		3.3.2	License plate reading	27
		3.3.3	Dedicated Short-Range Communication	27
		3.3.4	Wireless Local Area Networks	27
		3.3.5	Cellular Phone Networks	28
		3.3.6	Global Navigation Satellite Systems	28
		3.3.7	Correspondence of archetypes and technical solutions .	28
		3.3.8	Technical solution for the "Rekening Rijden" system	30
	3.4	Sensit	ive Zones	31

	3.5 3.6	Statistical analysis	34 35
4	Мо	deling Methodology	37
	4.1	Introduction	37
	4.2	Modeling methodology	38
		4.2.1 Characteristics of the System	38
		4.2.2 The Virtual Sensor model	39
		4.2.3 Simulation Concept	41
	4.3	Simulation environment 1	44
	4.4	Formal description	48
		4.4.1 Communication subsystem	49
		4.4.2 Detection and tracking subsystem	50
		4.4.3 Coordination and Registration subsystem	50
		4.4.4 Coupling of the subsystems	51
	4.5	Discussion	52
5	Cali	ibration of the Traffic Model	53
0	5.1	Introduction	53
		5.1.1 Why should a microscopic Random Traffic Generator be	
		built?	54
		5.1.2 What can be learned from the development and calibra-	-
		tion of the RTG?	55
		5.1.3 What is the scope of the results?	57
	5.2	Traffic generation	58
		5.2.1 Basic structure of the Traffic Generator	59
	5.3	Calibration and Validation of the subroutines	60
		5.3.1 Vehicle Type Generator	60
		5.3.2 InterArrival Time generator	62
		5.3.3 Drawing Speeds with the Speed Generator	64
	5.4	Hierarchical Aggregation	65
	5.5	Conclusions	68
6	Μο	deling of Dedicated Short Range Communication	69
Ū	61	Introduction	69
	6.2	Modeling	69
	6.3	Implementation	71
	0.0	6.3.1 Transmitter Geometry model	71
		6.3.2 Transmitter Field model	73
		6.3.3 Single Receiver model	75
		634 Single Vehicle model	77
		635 Multiple Vehicle model	79
	64	Validation	81
	65	Reculte	83
	0.0	incounto	00

	6.6	Conclusions	85				
7	An 7.1 7.2 7.3 7.4 7.5 7.6 Disc 8.1	Architecture for a Virtual Traffic Laboratory Introduction The laboratory Analysis results Implementation Discussion Conclusions Introduction	 83 87 87 90 92 93 94 96 97 97 				
	8.2 8.3 8.4 8.5 8.6	Data fusion and uncertainty propagation Integration approach Integration approach Measurement representation Multi-level modeling Integration Synopsis Integration	97 98 99 100 101				
9	Con 9.1 9.2	clusion 1 Overview 1 Lessons learnt 1 9.2.1 Applicability of Complex System theory 1 9.2.2 Value of the Virtual Sensor concept 1 9.2.3 Benefit of the multi-level modeling and simulation 1 9.2.4 Role of the Problem Solving Environment in the validation process 1 9.2.5 Benefit of the measured driven simulation 1 9.2.6 Scientific objectivity and a political setting 1	103 104 104 104 105 105 105 106				
Su	Summary						
Sa	Samenvatting						
Bi	bliog	raphy 1	119				
Αı	ıthor	's publications 1	129				
Pr	ojecť	s deliverables	135				
Ac	Acknowledgments						

CONTENTS

Chapter 1

Introduction

1.1 Context

While most sciences concentrate on how they can decompose a system into separate measurable elements upon which they can focus their research, computer science often concentrates on combining elements into versatile computing environments. The last century science has progressed fast, and scientists had to specialize on more and more extreme elements. The extremity can be in small sizes (atoms visible by a tunnelling microscope), large sizes (old galaxies visible by the Hubble telescope), high energies (Higgs-boson visible by the Large Hadron Collider) or in complexity (cell parts studied in the system-biology).

In the 'The end of science revisited' [47], John Horgan claims that, due to this progress, fundamental discoveries are becoming increasingly difficult. Science will only progress in small steps, in contrast to the developments in the previous centuries. Yet, this doesn't mean that scientists can end their work: the only way to truly know the limits of science is to keep trying to overcome them.

For computer scientists it is difficult to accept a limit on the progress, having grown up in a period of explosive scientific and technological progress, reflected by such measures as Moore's law. The American historian Henry Adams already observed a century ago that science & technology were accelerating at that age with an unprecedented momentum, and pointed out that at the underlying force was that more and more scientists dare to trust instruments which superseded their senses [2]. Knowledge gives better techniques; better techniques increase knowledge. Technological advances often enable researchers to overcome seemingly insurmountable obstacles. Computers in particular have vastly increased scientists' capacity for data acquisition, analysis, storage, and communication. Innovations such as optical and quantum computing may extend the reign of Moore's law indefinitely, but this still makes computers no magic wands that will simply solve the toughest puzzles. Computers remain instruments that have to be used wisely. Finding the methodologies to use the computer wisely is task of the computional scientist, and the subject of this thesis for a specific domain.

Our domain is very complex dynamical systems; systems with a very large number of interacting "objects" exhibiting many complicated phenomena. It is intriguing to trace these complicated phenomena back to their source, because chaos theory has shown that interaction of many automata with simple behaviors can already generate very complicated phenomena [31]. When the interacting "objects" are more than automata, but rather entities that can adjust their behavior dynamically (think of the entities that are part of a stock market, a neuron system, an ecosystem, or a traffic system), we are sure that we can never predict their behavior individually, and can only estimate their behavior with a coarse model

Simulation can be used to gain insight into the emergent behavior of the combination of many interacting "objects". Simulation models are capable of mimicking the dynamics at any required level of detail, although this mimicking remains an approximation of reality. Further, there is a trade-off in selecting the appropiate level of detail. First, the lower the level of detail, the higher the computational demands of the simulation. Second, the lower the level of detail, the higher the number of parameters that have to be validated. Yet, there are also benefits to be gained. First, the level of control the scientist has over the circumstances that are evaluated. The scientist can design clean experiments to find dependencies between parameters by scanning over complex surfaces in parameter space. Second, a large variety of phenomena can be tracked. With experiments in the real world, one has to wait till a phenomenon occurs, if one can detect it with the sensors available. In the simulated world a domain scientist has full control and can create infrequent phenomena on request, and observe these phenomena with virtual sensors which are designed to follow precisely those features that are of interest at that moment. So, simulation is a powerful instrument for computer experiments, when used properly, and in combination with experiments in the real world.

Computer and real world experiments are two important methodologies to gain insight in complex systems, when applied individually or iteratively. But when both methodologies are combined more intimately [25], even more insight can be gained, as demonstrated in this thesis. An application simulation can guide the measurement process, creating more effective data acquisition. Vise versa, the measuring process can support the simulation process. New data sources can enhance or refine the simulation by counterbalancing

2

incompleteness in the model. New data sources can be on-line measurements, or archival data that is collected by making the appropriate query. We will call this paradigm MDS-SDM. This acronym stands for Measurement Driven Simulations – Simulation Driven Measurements. We will look what generic methodology can be derived from the experiments we have performed during the evaluation of the "Rekening Rijden" system.

1.2 Challenges

To enable the Measurement Driven Simulations – Simulation Driven Measurements paradigm (MDS-SDM), the following challenges have to be addressed:

Multi-level modeling

Measurements can be made of a large variety of features of the system. To be able to incorporate this large variety of measurements, application models are needed that describe the application systems at different levels of detail. Simulators have to be able to invoke those models dynamically, to be able to make transformations between different levels of detail.

Data fusion and uncertainty propagation

Major problems occur when data (measured or computed) has to be combined that is collected on different spatial or temporal scales. Further precautions have to be taken to prevent small sample sizes and extreme events. With data generated dynamically, methods are needed that estimate the quality and propagation of errors and uncertainty.

Measurement representation / transformation

There is a need developing globally accessible interfaces to complex measurement systems. This can for instance mean that other researchers can have a view on current and archived measurements. This requires new approaches for information management systems, allowing different naming schemas and complex joins of information to support different scientific views on the same 'raw' measurements.

Integration environment

To couple the variety of measurements and models, an environment is needed that supports dynamic selection and coupling of application components. This means new interfaces to measurements systems and models simulators, interfaces that guarantee stable data streams by dynamically match computational and data requirements to the appropriate resources. These are ambitious challenges. Still, it is worth taking on those challenges and creating environments with some of the features introduced here. As stated by researchers of the RAND institute [26]; it is very useful to already have a part of these features available. Different components, models and information systems with each their own detail, grow slowly together into a family. This integration process is difficult [71], because it requires translations between those components that are not trivial.

Essential is not to have a hierarchical view on the components; to classify one model as high level, and the other one as low level (see figure 1.1) and only to look at the aggregation transformation. Both the high- and low-level models are good ways to describe the same recorded process of going from an initial state (i/I) to a final state (f/F). The process is typically a function over time, and the time is typically represented with a higher resolution in the lowlevel simulation/measurements. The transformation between the low-level and high-level is seen as an aggregation, a process where information is lost. The aggregation is typically a statistical operation, which can be a simple average, or the fitting of a distribution function. When enough information of the distribution is known, the process can be reversed, a process that is called *dis*aggregation. The information that is discarded during the aggregation process is recreated based on domain knowledge (as for instance the shape of the distribution function). This transformation is as fundamental as its reverse. With this transformation it is possible to couple models at multiple levels, to provide multiple interfaces (both to high and low resolution models) and maintain the consistency internally.



Figure 1.1: Hierarchical view on high-level and low-level application models

This hierarchical view of models is misleading. Each model is build for a certain purpose, which means that some aspects are worked out in detail, while other aspects are represented in an abstract way. Typically, for certain applications, the complexity of high- and low-level models is equivalent, where the complexity of low-level models is concentrated in grasping the dynamics of the physical world, while the complexity of high-level models can be found in strategies and policies. This means that low-level models can benefit from

4

aggregating complex rules into typical usage, while high-level models can aggregate dependencies into typical performance. I.e., transformations are not top-down or bottom-up, but peer-to-peer transformations with at both sides aggregation of unnecessary details and disaggregation of essential distributions. This is illustrated in figure 1.2.



Figure 1.2: Balanced view on high- (strategic) level and low- (operational) level application models

Essential for the transformation, both the aggregation and disaggregation, is that statistics can only be used with care. When for instance a phase-transition can be expected during the simulated/measured time, there exist two independent models (before and after the phase-transition). The parameters of both models have to be estimated separately. This doubles the validation effort, but prevents the distributions of the parameters from being smeared out over an unrealistically large domain.

Further it is important not to forget that both models are approximations of the real world. It is unfair to compare the equivalence of one model with another model after transformation in the context of the former model, because such a transformation is always crude. If one has to compare the applicability of two models, one has to do that based on their predictions of the performance of the system that is of interest of the users. So, both final states have to be projected on users quality measures, and on that level both models have to come with equivalent answers.

Central in this study is the evaluation of the reliability of the "Rekening Rijden" system¹ for the Dutch government. The quality measures imposed by the Dutch government focused on a number of failure events, which should have a very low probability of occurring (as low as 10^{-6}). The evaluation of the "Rekening Rijden" system was performed in several phases. During the first phase of the evaluation it became clear that the requirements could be met during 'normal' circumstances, but that study was needed to characterize the circumstances under which failures were made. This resulted in more detailed models of the "Rekening Rijden" system, which gave equivalent answers for the quality measures for 'normal' Dutch traffic, but also incorporated different answers when the system was tested under 'specific' circumstances. This level of detail required a tight coupling between the modeling & simulation and the measurements & analysis. Special care had to be taken, to ensure that 'specific' circumstances occurred enough in 'normal' traffic to contribute to the failure events reported in the first phase of the project. For a balanced view two-way transformations between the different variables of the models are needed, as we show in this thesis.

1.3 Structure of the thesis

Chapter 1 has introduced the context of this study in a wide perspective. Chapter 2 will survey the literature about complex systems, concentrating on multilevel modeling. The details of the application studied in this thesis are explained more elaborate in chapter 3.

In chapter 4 the methodology that we applied to the model and simulation the system is introduced. Chapter 5 highlights the effort to create realistic stimuli for our system. Multi-level simulation is worked out for one of our subsystems in chapter 6. The environment combine simulation and measurement analysis is described in chapter 7. Our progress is compared with comparable initiatives in chapter 8. The thesis is concluded in chapter 9.

¹"Rekening Rijden" is the name of a specific Road Pricing variant which was extensively studied from 1995 until 2001 for the Dutch government

Chapter 2

Complex Systems

2.1 Introduction

The domain of this thesis is the study of a complex dynamic system; a system consisting of many subsystems, with complicated interaction between those subsystems. Each subsystem is a macroscopic system in its own right, which can be described by many variables and controlled by many parameters. The subsystems are responding to the stimuli from the environment, which are in the case of the "Rekening Rijden" application the passage of vehicles. The vehicles are steered by their drivers, living beings from which the behavior is not simple to predict.

Yet, biologists have already modeled the behavior of living beings for a long time with success, which evolved into contributions to statistics [72], self-organizing systems [9] and cybernetics [10]. These contributions had indirectly a great impact on the engineering science. The contribution of biology can be explained [69] by the fact that dissipative systems, systems that show irreversible processes, can be associated in biology with progression (e.g. evolution). For physics, dissipative systems are associated with degradation of the system (e.g. friction). Cybernetics evolved later into general system theory [53], where a scientific discipline such as biology is seen as just a particular class of systems. In system theory the informational, relational and structural aspects of a model are dominant, the domain knowledge only a component. Examples of such domain knowledge would be biology, sociology, medicine, physics or chemistry.

The Modeling and Simulation field has its own ambitions. Taking advantage of improvements in software (object-oriented programming) and hardware (faster processors), the creation and execution of impressive simulation models is now possible [98]. The fundamental issues currently under study are model credibility (e.g. validation, verification, model family consistency) and interoperation (e.g. repositories, reuse of components, resolution matching) [24]. This thesis concentrates more on the model credibility [45]. This thesis depends heavily on the complexity concept as introduced in all three fields in the past century, especially on the relationships of multiple models inside a family.

2.2 Cybernetics

Cybernetics was a field introduced at the Second World War by Wiener [94]. In Cybernetics [10] the relationship with complexity is two-fold. On the one hand it is important, because only when systems become complex the methods to cybernetics do show their power. On the other hand the complexity of a system can be hidden, because the focus is on the emergent behavior of the system and the way to steer that behavior. The system is seen as a black box; a central concept in cybernetics.

The black box can be used as model for a system in the real world, which can have many aspects. Only some aspects of the real world will be of interest of the investigator and are represented in the black box. In that sense complexity is not something intrinsic of the system, but related to the model of the system. Complexity is expressed by the interaction between the investigator and the object of investigation. Complexity in that sense is the investigator's view on the system. This point is well-characterized by Ross Ashby [11]:

I shall measure the degree of "complexity" by the quantity of information required to describe the vital system. To the neurophysiologist the brain, as a feltwork of fibres and a soup of enzymes, is certainly complex; and equally the transmission of a detailed description of it would require much time. To a butcher the brain is simple, for he has to distinguish it from only about thirty other "meats", so not more than log^230 , i.e. about 5 bits, are involved. This method admittedly makes a system's complexity purely relative to a given observer; it rejects the attempt to measure an absolute, or intrinsic, complexity; but this acceptance of complexity as something in the eye of the beholder is, in my opinion, the only workable way of measuring complexity.

So, in Ashby's eyes, no assumptions are made about the nature of the black box and its contents. However, the investigator should have certain given resources for acting on it and certain given resources for observing its behavior. By thus acting on the box, and by allowing the box to affect him and his recording apparatus, the investigator is coupled to the box, so that the two together form an experiment; a system with feedback, as illustrated in figure 2.1.

Black Box	Investigator
--------------	--------------

Figure 2.1: An experiment defines the relation between an investigator and a black box

To make the experiment well defined and reproducible, the box's 'input' and 'output' have to be specified. Every real system has an indefinite number of possible inputs (ways to exert some action on the box) and outputs (means to observe and record some behavior of the box). For an orderly experiment the inputs and outputs to be used have to be selected, which means that all other possible inputs and outputs are left alone.

At the moment that the experimenter starts to alter the inputs of the box, some of the outputs will start to change (otherwise the inputs are incorrectly chosen [63]). At a certain moment the experimenter recognizes repeatable patterns in the behavior, and defines states and transitions between those states. States can be described by the instantiation of multiple variables. Some variables show correlations, and the scientist starts to deduce parts and connections within the black box. Yet, Shannon has shown that such behavior can be produced by an indefinitely number of possible networks of parts and connections inside the black box. Multiple system models can show the same behavior. A behavior doesn't specify the parts and the connections uniquely. Occam's razor says that for models with equal predictive powers, we should choose the simplest model. For models with a different number of parts and connections this seems decidable, but shifts at the end the question to what the predictive power of one connection relative to another connection is. In System Theory this is called the Reconstruction Problem.

Because of this ambiguity, whole families of system models can describe the box's behavior, and it is important to be able distinguish if two system models are equivalent (isomorphism) or that one system is a simplification of the other system (homomorphism).

- Two systems are *isomorphic* when a one-to-one transformation can be found for each state and variable.
- A system is a *homomorphism* of another system when multiple states or variables transform into a single state or variable.

The distinction between homomorphism and isomorphism makes it possible to specify the granularity of the knowledge of a system and to create multilevel models. Yet, for real systems this knowledge is only a small selection of what could be known from the system. Worse, at the moment that the experimenter has the impression that a consistent view of the behavior of the system is acquired, the tendency is to make this model a component in a larger system. A compound system is created by combining a model of a 'known' black box with another 'known' black box. For a combination of a limited number of boxes this approach can work, and the behavior of the overall system can be predicted from the behavior of its parts. Yet, as the compound systems grows, the moment will come that it is realized that an essential property of the overall system is missed, that there is a correlation between states of components not modeled before.

Another issue is the relation between the dynamics of the system versus the dynamics of the environment [69]. On the one hand the behavior of the system has to show long-range order, to be able to identify it as a system with certain macroscopic properties. On the other hand the behavior has to show short-scale randomness; otherwise it would be a deterministic system. The natural spread of responses allows a scientist to explore the state-space of the system. So, variance in its behavior is a natural prerequisite for a complex system, and finding the invariants in the underlying dynamics is the core of Cybernetics.

2.3 System Theory

System Theory [54] distinguishes two generic measures for complexity:

- **Descriptive complexity** is to measure the number of entities involved in the system (variables, states, components) and the variety of interdependence among the entities, as observed by the investigator.
- **Uncertainty complexity** is to measure uncertainty in the prediction of the behavior of the system, when the model is compared to measurements.

When we simplify a system, we want to reduce both the complexity based on descriptive information and the complexity based on the uncertainty information. Unfortunately, these two complexities conflict with each other. Often, when we reduce one, the other increases or, at best, remains unchanged. In System Theory, a set is created of models that can be used to describe a system. The descriptive complexity can be used to select a number of simple models from the set. Each simple model is represented with system behavior function f(c), is compared with the measurements $f_h(c)$, to check if the system is still well enough represented. The measure for this check is the normalized information divergence [23] between the model and the measurements. The information divergence is defined to indicate how well the system behavior reproduces the measured behavior; the information divergence is used as mea-

sure for the uncertainty complexity. The information divergence is defined as the loss of information when the behavior of a real system is replaced by the behavior of model system. The loss of information can be expressed by

$$D(f, f^{h}) = \frac{1}{\log_2 \|C\|} \sum_{c \in C} f(c) \log_2 \frac{f(c)}{f^{h}(c)}$$
(2.1)

for a probabilistic system.

The variable f represents the behavior function of the real system, the variable f^h represents the behavior function of a hypothetical model system, C is the set of behavior states c. A behavior function f(c) defines if a state c occurs (f(c) = 1) or not (f(c) = 0). The behavior function f(c) can be learned from a set of observations of the real system.

Decomposing a system into subsystems is a common methodology in System Theory, a process that creates Structured Systems. Decomposing in System Theory is actually only a partial solution to battle the complexity. To be able to define a whole-part relation in System Theory, first a common set of the states and variables has to be defined. So the model of a subsystem is only a subset of the set of its supersystem: both systems remain compatible, only supersystem is larger than the other. The benefit of decomposition of a system is that the number of connections between states and variables can be reduced, by ignoring weak and indirect correlations. So decomposition in System Theory is not as powerful as decomposition in Software Engineering, where also states and variables can be made private to the subsystem.

The power of System Theory is that it generalizes systems to such a high abstraction level, that a family of structured system models could be identified that could represent the real system (a set of reconstruction hypotheses). The selection of the best structured system is often impossible, because there are many comparable systems. It is possible to define structures that have equivalence classes (same number of connections), and to define an ordering between those classes (refinement ordering). Refinement is seen as making the system more general, to reduce the number of assumptions, to eliminate as many couplings between subsystems as possible. The ultimate goal of refinement is to reduce the system into independent parts, with no connections between them.

Structures with a certain equivalence class can be refined or coarsened. Unfortunately, in many cases several refinements and coarsenings are possible, each leading to structures in different equivalences classes. When one is looking for the simplest model with the maximum descriptive power, the family of structured systems can grow very rapidly. Yet, the family relation can be used to limit the number of structures that have to be evaluated for their descriptive power. For a number of structures in an equivalence class the descriptive power can be calculated by a measure like the information divergence. The structures with the lowest information divergence are further refined, the other structures are coarsened. To be a measure, the information divergence has to be monotonic, which means that the information divergence can only increase by refinement. Coarsening has the stop criterion that the information divergence is less than the current minimum. Refinement can continue until a threshold, the maximum acceptable information divergence is reached. The result is the simplest model that describes the behavior of the system well enough (according the information divergence measure, which compares the behavior of the model with the behavior of the real system).

To summarize, the refinement ordering is used as a relative measure for the descriptive complexity, while the information divergence is used as measure for the uncertainty complexity. Davis and Bigelow [27] amend that descriptive complexity is not only located in the number of variables and their relations, but that the cognitive aspect is also important. A model needs a good 'story-line' that makes it comprehensive why a model behaves as it does. Otherwise the model remains a black box to other scientists. This is close to a pragmatic interpretation of Occam's razor, where simplicity is a valuable aspect of a model because it makes the model easier to understand and work with.

The System Theory also provides another way to describe complex systems, namely metasystems. A metasystem is set of system models and a replacement function, a function that indicates in which situation which system model is appropriate. The transition from one system model to another is like a phase transition in physics. Two metasystems describing the same real system can be described with two sets of variables and states that are not compatible. Such a transition can be described by a homomorphism.

Yet, the variance of behaviors that emerge from a system depends on the environmental conditions. By controlling the environment the scientist can initiate a transition. During the transition the system becomes harder to be described. Phenomena occur that only exist for a limited combination of parameters. The number of experimental parameters to describe the precise environmental circumstances increases, indicates that for the system state multiple solutions are possible [69]; some stable, some unstable. To describe a system during such a transition many variables are needed. Pushed enough by the scientist, the system can pass through the transition to a 'different' system, a relatively simple model with a limited number of variables. In this way a 'new' stable system model is discovered.

System Theory makes it possible to quantify the complexity of a system. Still, it assumes that a large set of measurements is available with which the possible models can be compared, and gives no clue what the predictive limits of a model are.

2.4 Modeling and Simulation

Central in the modeling and simulation methodology [98] is the search for a simple model that represents the complex reality well. To be able to create such model, one has to define what a valid simplification is: we need to have measures for model validity and model complexity. In principal the following measures for complexity can be distinguished in the context of modeling and simulation [98]:

- **Analytic complexity** is equivalent to the global state space of the model: the total number of states that have to be analyzed and explored.
- **Simulation complexity** is a measure of the computational resources to execute a model. The simulation complexity can be smaller than the analytic complexity, because the global state space is sampled via the local state space of the simulated components.

System Theory was focused on the analytic complexity. In the modeling and simulation research field one focuses on the simulation complexity and the methods to battle this complexity.

The global state space is related to the local state space of the components the couplings between those components. Interaction between subsystems is taken into account by defining couplings between component's local states. The complexity depends on the size/resolution product, or when the interactions are taken into account on the size/resolution/interaction product. The size of the model is determined from the number of components in the model, and the resolution from the number of states per component. Yet, only the dependency is given, no explicit measure for complexity is given. Such an explicit complexity measure should be equivalent to the descriptive complexity of System Theory.

Several simplification methods are reviewed. Aggregation and omission are the methods to reduce the size and resolution, while linearization and replacements of deterministic and stochastic descriptions primarily affect the interactions between the components. Mapping from one simulation formalism to another (i.e. differential equations into discrete event models [70], combining discrete and continuous signals [14], coercing input and output [61]) can also reduce the simulation complexity.

Aggregation is seen as the main method to battle complexity. Components in the system model are clustered, so that a new level of partitioning is introduced to the system. This cluster forms a new structured system, an isomorphism of the original system with the same complexity. The clustering is followed by the definition of a new abstraction level; an attempt to find a transformation that hides some of the entities of subsystems inside the components, to create a lumped model. In this lumping process information can be lost; the transformation is a homomorphism to a simpler model that (hopefully) describes the behavior of the system well enough.

As a black box, a system can be modeled by its input $\vec{\omega}$ and output \vec{y} , and the functional relation between those two. The functional relations can be a specified on several levels, from a simple observation to a multicomponent model. From observations one can deduce relations between certain inputs and outputs, and after enough observations a function can be postulated to describe such a relation. At the moment that a system is only partly observable, an observer may be able to restore predictability by taking the systems history into account, to assume a form of memory. This is the moment to introduce an internal state \vec{q} and make the black box a state machine. A system can then be formulated with the following discrete value sets and functions:



Figure 2.2: A state machine where the new state \vec{q}' depends on its current state \vec{q} and the applied stimulus $\vec{\omega}$.

Here, $q_t = \Delta(\omega_{t-1}, q_{t-1})$ and $y_t = \Lambda(q_t)$ are respectively the *state transition function* and the *output function*. Formally, the system can now described by the following sets:

$$S = (T, X, \Omega, Y, Q, \Delta, \Lambda)$$
(2.2)

X, *Y*, *Q* are respectively the sets of input values \vec{x} , output values \vec{y} , and internal states \vec{q} . *T* is the time base, or support set. Ω is a subset of (X, T) which represents all segments where the input is changed, and where the output is observed. From this description other functions and value sets can be constructed, as for instance the function that describes the behavior of the system $\beta_a(\vec{\omega})$. This function is a combination $\Lambda(\Delta(\vec{\omega}, \vec{q}))$.

In [69] it is proven via homomorphism that when there is a second (simpler) model of the system, the set of behavior functions of the simple model is a subset of the behavior functions of the complex model. Such a homomorphism from a 'large' system *S* to a 'small' system *S*' can be performed by a triple of transformations (g, h, k) (see 2.3).



Figure 2.3: The transformations for a homomorphism from 'complex' S to 'simple' S'.

Notice that the transformation g is an identification of input segments of the 'small' system S' to input segments of the 'large' system S. Not all possible input segments of the 'big' system have to be covered, so not all states \vec{q} have to be visited. Let call the subset of states \vec{q} visited due to $g(\vec{\omega}') Q$. The transformations h and k are defined from the 'large' system S to the 'small' system S', for this subset Q of the statespace Q and the corresponding subset Y of the output Y. To keep the two models consistent, these transformations have the following constraints for all $\vec{q} \in Q$, $\vec{\omega} \in \Omega$:

$$h(\Delta(g(\vec{\omega}'), \vec{q})) = \Delta'(\vec{\omega}', h(\vec{q})) \tag{2.3}$$

$$k(\Lambda(\vec{q})) = \Lambda'(h(\vec{q})) \tag{2.4}$$

This transformation is a relabeling of a part of the states of the 'big' system to the states of the 'small' system, while preserving the set of states reachable with the state transition function $\Delta(\vec{\omega}, \vec{q})$ as a closed set (the system remains a predictable machine). When there is a one-to-one relationship between the states of *S* and *S'* this is not a homomorphism, but an isomorphism. Only a homomorphism can simplify a model, with the corresponding effect on the predicted output of the model. A simplification is seen as *valid* when the modeling error stays in bounds. It is essential to confirm it not once against a fixed tolerance, but to check in multiple steps to see how the errors propagate through the system, to be sure that they do not accumulate unmodeled. Remember that the history of the system could be stored in the unobservable internal state \vec{q} . Because the internal state of the 'complex' system and the 'simple' system could be defined on two different timescales, the accumulation can be fundamentally different.

To get a balance between the 'small' and the 'large' system, not only the transformations (g, h, k) is important but also their inverse (g^{-1}, h^{-1}, k^{-1}) . For a homomorphisms these transformations will be stochastic, which leaves us to find the correct functions to sample from those distributions [72; 75].

2.5 Conclusion

Complexity is a fundamental concept when a system is to be modeled. We have seen that there are many facets to complexity. Zeigler, for instance, claims that model complexity is to be measured as the time and space required to simulate; Klir claims that model complexity is measured from its intrinsic parameters; Ashby claims that the model complexity is as large as the variance in its output. In addition, complexity has a somewhat subjective connotation since it is related to the ability to understand or cope with the system under consideration. Yet, we have seen that objective measures exist that can indicate the relative complexity (which system is a simplification of another). In this thesis we will make use of this historical background, but will focus our methodology on the effort required to validate the model.

Chapter 3

Application

3.1 Introduction

The complex system that is analyzed in this thesis, is the road-pricing system that has been on the political agenda of the Netherlands for a long time (in 1986 Minister Kroes foresaw a possible introduction in 1996, but in 2007 Minister Eurlings still has the assignment to start the introduction before the year 2012). During this long time, several technical solutions for automatic charging have been studied. It is not the intention of this chapter to participate in the political debate or to describe the perfect technical solution to battle the congestion on the Dutch road network. The sole purpose of this chapter is to sketch the context in which the evaluation study in this thesis was performed.

In the beginning of the nineties it became technically feasible to build a multilane Automatic Debiting System, while most Debiting Systems in the world were still manually operated. In the eighties Automated Debiting Systems were introduced, operated on a single lane as bypass for frequent users on a toll-plaza. Such a toll-plaza is a major infrastructure, which disturbs the traffic flow and creates congestion ahead of the plaza. Single lane Automated Debiting Systems could be used to reduce the disturbance for frequent users, but in general Multi-lane Automatic Debiting Systems are preferable. Multi-lane Automatic Debiting Systems do not require the creation of a plaza in the road network, all they need is a gantry over the road.

In 1994 a large experimental survey was conducted by the TUV Rheinland [12], where a variety of technical solutions for multi-lane Debiting Systems were compared in one location for real traffic. Ten different systems were installed, some based on gantries and some based on satellites. The output of those

ten installed systems was constantly collected and evaluated against a reference system. The reference system consisted of two inductive loop systems. Many discrepancies between the systems and the reference system occurred, and the errors were classified into three classes: evaluation errors, implementation errors, and fundamental errors. The distinction between implementation and fundamental errors was made on the basis of the rule that the error could have been prevented or recovered in a more mature system. It was very difficult to make this distinction in a fair way for 10 very different systems. Insight was needed about the fundamental limitations of the sensor-techniques used. Measurements over a longer period on real traffic give an overwhelming amount of statistics. Many strange things can occur in real traffic during such period, and the designers of the systems under study could claim 'bad-luck' quite often. The Dutch government learned from this survey that experiments are not as valuable as one expects, if the experiments are not backed up with insight in the internal processes of the systems under study.

At the 'Workshop on an Evaluation Methodology for Automatic Debiting Systems' [42] a different approach was introduced. Here it was proposed to evaluate the systems with a mix of analysis, inspections, experiments and modeling & simulation at multiple levels of detail. The modeling and simulation environment should on the one hand:

- Enable multilevel simulations
- Enable the use of stochastic and deterministic models
- Enable the use of experimental data

On the other hand, the simulation and modeling was to be used to enhance the value of the experiments. As the German experiments had shown, it is difficult to trace an error back to underlying failure, because there are often multiple chains of events that ultimately manifest themselves into the same overall error. Simulations can be used to balance the contribution from different sources into accordance with errors found in large scale experiments, and can predict the error rates for different circumstances. These circumstances can sometimes be recreated on a testtrack, and the simulation can be used to find the interesting scenarios to be tried. In this way, modeling and simulation increases both the traceability and efficiency of the evaluation process, which was explicitly confirmed by the Dutch government [64] at the end of the project.

The comparison of the Fee Collecting System in the Dutch context was in a way easier than the German survey, because an initial version of the requirements of the Dutch system was already specified (although it was not clear whether those requirements were feasible), which limited the variety of technical solutions to systems based on Dedicated Short Range Communication (DSRC), as described in the European Standard [76].

At the moment that the technical feasibility of this variant of road-pricing was proven, and the preparations were made to really introduce this system, the political decision was made that the acceptance for such payment scheme was too low. The introduction of a road-pricing system was postponed, and a study was started that asked for the benefits of future technologies. The result was the 'MobiMiles' report [73]. In this report it was foreseen that in the near future navigation systems would become widely available in vehicles. With vehicle navigation systems a nationwide road-pricing system could be introduced and the price of the introduction would decrease in the years to come. Road pricing was further seen as a government service that could initiatate many business services. If the government set the national standard for a communication channel to vehicles which known their location to enable road-pricing, businesses could use that same channel for innovative services based on the location of the vehicle. At the end of the internet-hype this location-based service concept received a wide international interest. Many companies started with a design for a nationwide road-pricing system, and the government worked hard to specify a set of feasible requirements for the road-pricing system. Unfortunately, this work was abandoned after the change of government in 2002.

Still, the problems that inspired the government to consider the introduction of a road-pricing system are still there. In a multidisciplinary report [30] it is shown that both theoretical as practical a road-pricing system can be an effective way to change the road usage in such a way that the welfare of the Netherlands as a whole increases. They also indicate that the acceptance of such system is highly dependable of the perception of the public. To be effective, some people will have to pay more for their mobility than they currently do. For these people it has to be clear why they have to pay that price, and that it is compensated with an increase of welfare. The report gives several examples of how road pricing could be implemented, without going into technical details. We will follow the archetypes introduced in this report, to show the context of the decision makers, before we go into the technical details of the system that was worked out in the largest extent.

3.2 Archetypes

Based on the analysis in [30], the main design options can be distinguished based on the following characteristics:

Scale indicates if it will be a national, regional or local system, if it covers all the roads or only points in the network, if it covers the whole road or only specific lanes.

Differentiation indicates the parameters used to calculate the price for the

ownership and usage of a vehicle. These parameters can for instance be distance, time, location, and vehicle type.

- **Welfare goal** indicates the control variable (revenue, travel time, emissions, noise, safety) translated in a common equivalent (money) that can used to calculate the price level with an optimal effect.
- **Appropriation of revenues** indicates how, and to what extent, the money is fed back into society. Available choices are tax reduction, new roads, or alternative transportation.

Scale and differentiation have the largest impact on the technical system. The welfare goal is a characteristic which has consequences that are difficult to grasp. On the one hand there seems to be a large overlap between the goals (less vehicles means less congestion, emissions and noise) and the same technical system can be used for multiple goals (with slightly different rules). On the other hand there can be subtle conflicts (earlier departures means less congestion but more noise at an inconvenient time), which can have large consequences for the acceptance. This means that the rules (and therefore the differentiation, and therefore the technical system) have to be carefully balanced on their effect for the different welfare goals.

The dominant welfare goal has changed over the years. For instance, "Rekening Rijden" (1996) was primarily meant as local congestion measure, while in the focus shifted due to the Kyoto treaty towards the environmental effect for "Kilometerheffing" (2001). A study that tried to estimate the welfare 'price' of different influences [6], indicated that congestion, safety and environmental effects on welfare are comparable. The total yearly costs for the Netherlands are respectively estimated as $2 - 3 \, 10^9 \in 4 - 8 \, 10^9 \in 3 - 8 \, 10^9 \in$. The "Hofstraheffing" tries to balance those three effects explicitly by calculating a fixed tariff per vehicle based on 1/3 environment, 1/3 safety and 1/3 congestion. Yet, this balance is not constant; per trip it can differ largely. Many trips are performed at time and locations where the congestion effect is zero. During rush-hours the congestion costs can be four times as large as the combined safety and environmental costs [30, p. 18]. So differentiation in time and location is crucial to design a system beneficial for the Dutch society.

The appropriation of revenue seems only to be important for the perception of the people at the introduction of a system, and out of scope of a technical discussion. Indirectly revenues is also important for the selection of a technical solution, because the costs of the system have to be in proportion, to allow any revenue at all.

In the next sections we will introduce three road-pricing schemes that are well distributed over scale and differentiation. For each road-pricing scheme there exist one or several techniques to implement this scheme. The choices about

scale and differentiation determine which technique is the appropriate to apply, but a small change in the scheme can make other technical solution suddenly far more attractive on characteristics such as costs in relation to the revenues.

The three road-pricing schemes discussed are equivalent to the variants 1, 6 and 8 of the evaluation study from the Nouwen board [103]. In this study the effects of 10 road-pricing variants were compared on effects like travel time, emissions, safety and economic consequences. The 10 variants were relatively straightforward in allowing a clear comparison; it was never the intention to actually introduce a variant in that form. An actual road-pricing scheme would combine elements from those 10 basic variants. Based on the evaluation, the commission made the advice to the Ministry to start as soon as possible an initial phase on a local scale, with as welfare goal the appropriation of revenue for a concrete infrastructure. In the mean time research could be initiated on a cost-effectiveness system on a national scale with as welfare goal travel time.

3.2.1 Example 1: distance-based charging i.e. "Kilometerheffing"

Scale national, all public roads

- **Differentiation** fee per driven kilometer, possible differentiation on vehicle and fuel type
- **Welfare goal** the goal is explicitly not revenue. The price per trip should be designed such that the total revenue would equal to the previous tax situation. For the average user the tax will not change, only for the light/heavy users.
- Appropriation of revenues "Kilometerheffing" would replace the Dutch purchase and yearly tax on vehicles

"Kilometerheffing" was introduced as 'MobiMiles' by Roel Pieper[73], what meant that every car in the Netherlands had to pay depending on its usage. The payment was performed by a small box in every vehicle on the basis of the distance driven. The small box consisted of a number of standard navigation and communication devices and a 'trusted wallet'. The hope was that the price of all those components would drop quickly because of their massive usage. Later the situation looked even more attractive, when it was realized what sort of mobility services where possible if every vehicle in the Netherlands had the same communication and navigation device.

The problem of this system is to keep it cost-effective. A simple and cheap implementation of this system is possible, if the calculation is based on a rough estimate of the traveled distance. An accurate and reliable system to measure



Figure 3.1: Illustration from the 'MobiMiles' report [74]

the traveled distance is preferred by the Dutch tax-office. Such a system is technically feasible, but not cheap.

Advantages:

- The more you use, the more you pay
- Acceptable appropriation of revenues

Disadvantages:

- Risk that the tax increase for heavy users is unacceptable large.
- Tax-payment can only be avoided by not driving, which reduces the mobility of the Dutch citizens.

3.2.2 Example 2: regional and time dependent charging i.e. "Rekening Rijden"

Scale regional networks (both national and local roads)

Differentiation primary location (region), additionally time, possible differentiation on vehicle and fuel type

- **Welfare goal** travel time, the price can be adjusted to improve the fluidity of the traffic in the region.
- Appropriation of revenues regional infrastructure



Figure 3.2: Artist impression of the "Rekening Rijden"-project.

The congestion in the Netherlands is concentrated around the four largest cities, although the problems around Eindhoven have become as bad. The Netherlands has historically many cities, which are in a global setting small and compact. This makes it possible to define a corridor around the cities, and to introduce a tariff for every vehicle entering the city during rush-hours. This discourages incoming traffic to some extent, but more importantly forces transit traffic to look for alternatives. In this way the flow of the traffic towards the city can be improved.

An equivalent measure is introduced around London and Stockholm, which has a proven positive effect on the traffic situation in the city. Yet, such a corridor is typically an expensive measure, which requires the installation of many systems on all roads crossing the corridor. It is difficult to predict the effect on the traffic situation in the Netherlands, because all four Dutch cities are all quite different in the ratio between incoming and transit traffic, and the alternatives that are available for transit traffic. Although difficult to predict, the effect is estimated to be positive for all four cities [13].

Advantages:

• Local government is a clear stakeholder with in depth knowledge of regional travel patterns

- The successful examples of road charging systems are all regional systems (London, Stockholm, Singapore).
- Differentiation on location is a natural feature of this system: it will only be applied in congestion areas.

Disadvantages:

- Regional interest is not necessarily the same as the national interest
- Technical and administrative complexity of several regional systems
- Using the revenues locally is acceptable, but does not have to be efficient on a national scale.
- Travelers living in different regions have to pay, but have no democratic influence on the introduction ¹.

3.2.3 Example 3: congestion tracks i.e. "Almere - Amstelveen"

Scale local: congestion tracks where the pressure can be relieved with new roads

Welfare goal revenue

Differentiation track (trajectory) and time

Appropriation of revenues financing the new infrastructure

There are points in the Dutch road network that have been overloaded for a long time, but where the construction of an alternative road is always postponed due of the unappealingly high costs. For the same budget several other problems in the Dutch road network can be tackled. One example of such point is the connection between the Almere and Amstelveen at the south-east of Amsterdam. Construction of the road along the shortest path in a conventional way would disturb a unique nature reservation, drilling a tunnel would cost 4 times as much. After a study the alternative through the city is chosen (indicated on the right of figure 3.3), although this alternative cost as much as drilling a tunnel. The alternative through the city is extended with several environmental measures, to reduce the effects for the citizens living along this track.

Such a tunnel could be financed and built before that date as a toll-road, but it would take decennia before the construction costs are recovered. Such an

¹In Stockholm, 50% of the citizens of the city voted on a referendum in favor of a congestion charge, while 60% of the inhabitants of the municipalities around the city were against the measure (more details at http://www.stockholmsforsoket.se)



Figure 3.3: Two possible solutions for a connection between Almere and Amstelveen. The dashed trajectory represents a tunnel. From [100].

environmentally-friendly tunnel is not an attractive investment for a commercial consortium, and there is a free alternative outside the rush-hours: the current road. Further, every driver at the south-east of Amsterdam would benefit during the rush-hours from people that would take the tunnel. So, it could not be unfair to introduce a congestion charge on this track, and use the revenue to finance the environmentally-friendly alternative.

For the perception of the drivers it could be beneficial when the payments are concentrated a few years before, during and after the construction period, so that the usage of the collected fee is clear. In the limited period of a few years the revenue will only cover a part of the construction costs, so the government still has to contribute more than the construction costs of a conventional road. By introducing locally a congestion charge for the regular users of that track, above the tax all drivers already pay for the maintenance of the national roadnetwork, a balance can be found the interest of the region and the country.

Advantages:

- Earlier construction of infrastructure
- Limited scale, good demonstration, less implementation problems
- Damping of the congestion peaks due to time differentiation.

Disadvantages:

- Extra road capacity will attract new traffic, which will have a negative effect on emissions
- · Low efficiency compared to full pricing of the network

3.3 Technical Solutions

² The previous sections gave a short overview of a number of archetypes of payment schemes that can be used to charge vehicles for the use of the road. Those archetypes correspond loosely with one or more possible technical solutions to implement the payment scheme. Technical solutions can be combined into working systems. All systems can be implemented with current technology, but the actual choice of technology has a severe impact on characteristics as costs, reliability, security, ease of use, etc. One of the most important design questions is the assignment of functionality to equipment in the vehicle and at the roadside. The choice is between the following technologies:

- Links to vehicles' odometers or an inertial navigation system.
- License plate reading.
- DSRC (Dedicated Short-Range Communication), equipment insides the vehicle that communicates with portals at the roadside over dedicated links (microwave or infrared).
- WLAN (Wireless Local Area Network), equipment inside the vehicle that communicate with access points at the roadside over public links (microwaves).
- Cellular phones (GSM/UMTS) with additional localization technology.
- Global Navigation Satellite Systems (GNSS), such as the American Global Positioning System (GPS) or the European Galileo.

Many systems currently in operation use a combination of these technologies, which improves the reliability of the system, but substantially adds to the costs. The first example, "Kilometerheffing", is a system that benefits from a combination of many technologies.

3.3.1 Odometry

Every vehicle is already equipped with an odometer that registers the distance driven. It would be logical to build a charging system on these measurements. Unfortunately, the standard odometer of a vehicle is not very accurate, often still mechanical and easy to tamper with. In heavy goods vehicles there is a long history of tachographs, which has a verified link to the odometer. For that reason, payment systems dedicated to heavy goods vehicles, as for instance in

²This overview is largely based on A. Visser, H.H. Yakali, L.O. Hertzberger, 'A next generation ETC system based on time, distance and position - Evaluation', Technical Report UvA PCC Holding B.V., June 2002, confidential [88]

Austria³ and Germany⁴, have an interface to the vehicle's odometer. Navigation systems in personal cars rely more on internal inertial systems. An odometer can give information about the distance, speed and time, but not about the location of those movements.

3.3.2 License plate reading

Every vehicle is already equipped with a license plate. In the recent years the software to automatically read those license plates has improved significantly, which makes it possible to process immense streams of images. Still, many images have to be checked manually to reduce the number of mistakes. The system in London⁵ is an example of the usage of this technique. The organizational costs of the system in London are high [102, p. 14] compared to the Austrian and German system. To reduce the costs, they study [101] alternatives which are based on for instance DSRC or GNSS. The road side systems can give information about the passage location, speed and time of the vehicles, but are not suitable to provide a full picture of the traveled distance.

3.3.3 Dedicated Short-Range Communication

Dedicated short-range communication (DSRC) is the international standard for electronic tolling. A simple tag is communicating with beacons at the road side. The communication is limited to a restricted area by design, for safety, security and interference purposes. At several places in the world systems based on the microwave part of the standard are installed. Unfortunately, all those systems differ slightly, in spite of all interoperability efforts. These systems are also point-based systems, which in principle can only have information about the passage of a vehicle. When all entries and exits of specific routes are equipped with beacons, the traveled distance can be calculated.

3.3.4 Wireless Local Area Networks

Wireless local area networks are based on the international standard for communication between computers based on microwaves. The popularity of this standard has lead to a number of versions of the standard, with ever increasing bandwidth and/or covered distance. The mass market makes these communication devices cheap and powerful compared to the dedicated devices of DSRC. The cells of these networks have a suitable granularity for localization based services that can cope with an accuracy of around 100 meters. The

³http://www.asfinag.at

⁴http://www.toll-collect.de

⁵http://www.cclondon.com

downside of this technique is the potential security and interference problem with other applications.

3.3.5 Cellular Phone Networks

A mobile phone has become standard equipment inside a vehicle. Inside the network the location of a phone can be roughly estimated, good enough to map trajectories of mobile phones to routes [79]. For this rough localization estimate not only the cell-id is used, but also additional measures as angle-of-arrival, time-of-arrival and signal-strength level. Local circumstances can influence the signal significantly (e.g. reflections), with major effects on the accuracy of the localization (and the quality of the connections). The quality of localization can be improved by installing extra antennas in the network, creating micro- and picocells. Microcells provide radio coverage over distances of around 500 meters. Another technical aspect that is to be addressed concerns the substantial amount of bandwidth to the transfer the location, as known by the network, back to all the mobile phones. For the infrequent transfer of administrative data the cellular phone network is an excellent choice.

3.3.6 Global Navigation Satellite Systems

Global Positioning System (GPS), provided by the United States of America, is the Global Navigation Satellite System used in the navigation systems that are currently installed in cars. For this technology the time differences between the signals of four satellites are used to calculate the current position. The quality of this localization can be of the order of a few meters, but is also dependent of the local circumstances. High buildings can for instance obscure, reflect or distorted the signals. Navigation systems based on GPS will often deploy supplementary systems to increase reliability and accuracy, for example by using additional location inertia measurements and electronic mapping. In the London Trails [101] seven different GPS devices were tested, all equipped with supplementary systems. The average location error of these GPS devices was 9.7 meters, but the maximum error more than a kilometer. To be 99% confident that a vehicle was inside the charging zone, they had to extend the boundary around the inner city by a buffer zone with an average width of 57 meters, but on some places the width had to be extended to 250 meters. The buffer zone around the inner city of London is illustrated in figure 3.4.

3.3.7 Correspondence of archetypes and technical solutions

One of the archetypes previously presented is the "Kilometerheffing" system as specified in 2001. The Dutch "Kilometerheffing" system was seen as a sys-


Figure 3.4: The boundary around the charging zone of London City, with in purple (dark-grey) the buffer zone required to accommodate the unreliability of GPS-measurements (from [101]).

tem with most of its functionality on board the vehicle. The project never reached the phase where the technology was fixed, but it would probably be a combination of odometry, GPS and GSM, like the system introduced for the heavy goods vehicles in Germany. In Germany the devices are also equipped with DSRC antennas for control purposes, and even BlueTooth for short range communication with other smart devices in the cabin.

At the other end of the spectrum is 'CongestionTracks', which could be implemented fully from the road side. As long as the scale of the system is limited, one can base the charging on reading the license plates of all passing vehicles at the entry and exit points. Yet, this would be a world première, and an expensive one. License plate reading systems on tracks are already in use at several locations in the Netherlands, but only for speed enforcement (a few percent of the passages). Scaling these systems up to bill all passing vehicles would require a large administrative organization. In London the license plate reading is only used for enforcement, in principle every vehicle heading to London has paid their fee in advance (via internet, SMS, phone or office). In Stockholm payment was possible with distributed transponders or afterwards via retail stores. License plate reading was used to check if no payments are forgotten.

3.3.8 Technical solution for the "Rekening Rijden" system

The "Rekening Rijden" system, as specified in 1996, has both components onboard and off-board. This makes this system attractive to work out in more detail. The Dutch "Rekening Rijden" system was designed to work in a cordon around 4 large cities in the Netherlands (see figure 3.5). Each highway into a city was equipped with a system consisting of a number of gantries above the road. Traffic along other roads into the cordon was discouraged. Vehicles could pay directly and anonymously with an On Board Unit (OBU), or receive a bill afterwards on their home address based on license plate readings.



Figure 3.5: The cordons around the four largest cities in the "Rekening Rijden"-project.

Already in 1994 free-flowing charging systems were demonstrated, but all systems used a simple payment scheme with tokens, while in the Netherlands one would like to pay with real money in an electronic purse. The money on an electronic purse has to be protected with a verification and authentication protocol, which includes some encryption [91]. Encryption requires some heavy calculations which prevent direct responses, which means, given the average speed of a vehicle, that a payment requires more than one gantry, in contrast to the systems already deployed elsewhere in the world.

A typical passage of such a "Rekening Rijden" system with multiple gantries by a vehicle can be illustrated with events listed in table 3.1. As one can see, there are many decision points in the systems logic, and those decisions depend heavily on the timing of the different events. When multiple vehicles are under the gantries, the situation becomes more complicated, because the detections and payments have to be correlated, and multiple candidates are available. Further, the road-side system can only communicate with a limited number of vehicles at a given time instance, so the timing of the events is not the same for each passage, but depends on the state of the communication protocol.

From this description, three separate functionalities can be distinguished, which correspond to decomposing the system into three subsystems.

- **communication with OBU, and localization of OBU:** When the vehicle has an OBU (On Board Unit with the driver's electronic purse), a communication link with the road-side system can be established. A number of messages exchanged according to a protocol make it possible to pay the required fee electronically. During communication, the approximate location of the OBU is determined using the characteristics of the microwave signal. The localization information can be used to indicate to the road-side system (RSS) which vehicle is the one that has paid. Without communication or localization information, the fee is collected via the registration of the license plates.
- detection, tracking and classification of the car body: The road-side system needs to be able to sense and track vehicles. The tracks can be matched with successful communications, to know which On Board Unit (OBU) is located in which vehicle. Vehicles that pass without electronic payment need to be tracked, and registered. Automatic Debiting Systems that have a considerable spatial extent have to maintain long tracks, and the tracking is in that case an important and critical part of the total functioning. A system developer can choose between many different media (visible or infrared light, natural or artificial light sources, etc.) and type of sensing devices (laser curtain, camera, etc.) for this task, but in the end, they will always sense the outer shape of the car.
- **registration of license plates:** The system needs to take pictures of the license plates of all vehicles that do not pay electronically. This is called registration. The fee is collected afterwards via back-office processing by sending an invoice to the owner of the vehicle, who is identified by the number on the license plate. When there is a considerable spatial extent between the registration points and/or the communication point, proper tracking is important.

3.4 Sensitive Zones

To evaluate the interaction between those three subsystems and the vehicle it is important to realize that this interaction is not momentarily; the vehicle is meanwhile moving through the system. This means that the interactions take place at a number of sequential locations in space and time. The result of an interaction is typically a function of the precise location of that interaction.

Table 3.1: The sequence of events during a typical passage of a "Rekening Rijden" system. Figures courtesy of [57]

At the moment that a vehicle comes in reach of the short-range communication system, the On Board Unit with the driver's electronic purse (the OBU in the upper corner) wakes up.

The OBU presents itself to the microwave antennas on the first gantry of the road-side system (RSS) with a message containing the services it can handle (in this case paying a fee).

The road-side system (RSS) requests a payment. This is an operation that requires decryption, authentication and encryption, which takes so long that at full speed the first gantry is already passed before the payment is done.

At the next gantry the front of the vehicle is detected. When the vehicle travels at low speed, the payment can already be received and no further activity is required. When the payment is not (yet) received, the front license plate will be registered.

When the rear of the vehicle is detected and no payment was received, a picture of the rear license plate is taken. Those pictures are stored temporarily, because the vehicle has still the opportunity to communicate its payment via the short range communication at the third gantry. When at the third gantry no payment is received for this vehicle, the license plates are automatically read and a passage is added to the monthly bill.









Geometrical computations are very important to estimate the behavior of the subsystems, and will be heavily used for the modeling & simulation of the subsystems. This aspect will be worked out further in chapter 4.

The "Rekening Rijden" system is specified as a corridor around a city, where at the major roads gantries are placed with Automatic Debiting Systems installed on them. The systems are designed such that only vehicles passing under the gantries are debited; all subsystems have a limited range. This limited range means that there is a great commonality for all systems: for each subsystem a sensitive zone can be defined in which interaction between car and subsystem takes place. Outside the sensitive zone no interaction takes place. With this assumption the interaction results only have to be modeled for a finite, although continuous, space. For the communication the sensitive zone can be characterized with the outer envelope of the so called footprint, for cameras the sensitive zone is equivalent with the field of view.

An interesting aspect of the "Rekening Rijden" system, and other gantry based systems, is that the sensitive zones of the subsystems have volumes that are of the same scale as the passing vehicles. This means that the vehicles cannot be seen as points, but that the vehicles should be modeled as physical bodies with a volume. A distinction has to be made on interactions that take place on different surfaces of the vehicle. For instance the detection of the rear of a vehicle is something complete different than the detection of the front. The origin of this difference is not so much the difference in shape (although the shape of the rear can be quite complex due to antennas, carried bicycles, trailers, etc), but due to the timing. When the rear is detected there is already a history of measurements. There is already a running estimate of the dimensions of the vehicle, an estimate that can be wrapped up because the rear is detected.

Each subsystem interacts with a certain region of the car: the windscreen for the communication system, the top and sides for the detection, tracking and classification system, the front and rear for the registration. To have an accurate estimate of the location of an interaction in time and space the intersections of a surface of the vehicle and the sensitive zone should be calculated. This adds a considerable level of geometrical detail to the behavior models of the subsystems, but this detail is needed to create a realistic input for the algorithms that track and localize the vehicle. When the information of the tracking and localization is combined, the relative order of interactions of the subsystems can influence the quality of the match.



Figure 3.6: The intersection of the windscreen and a typical sensitive zone of a communication subsystem.

3.5 Statistical analysis

The Dutch "Rekening Rijden" project was in its domain quite unique with regard to the stringent requirements on the reliability of the system. Several failure events were defined, and the probability of some of those failure events should be as low as 10^{-6} . This meant that especially for this project a statistical analysis was made [81], to construct a statistical test to determine if the simulation results prove that the requirements are met. In this statistical analysis the simulation of every passing vehicle was seen as a Bernoulli experiment with only possible outcomes: success or failure. When the probability of failure is equal for all passing vehicles, a Binomial probability distribution can be expected.

Because of the formula of the Binomial distribution contains the factorial function of the number of passing vehicles; the distribution has to be approximated for large numbers of passing vehicles with a Poisson distribution. To test the hypothesis that a system meets its requirements, it has to be checked if the number of failure events exceeds a critical value. For such an one-tail test, the relationship between a Poisson distribution and a central χ^2 distribution can be used. For the χ^2 distribution the critical values can be calculated. For four million passages, the critical values for requirements of 10^{-6} , 10^{-5} , 10^{-4} are respectively 8, 49, and 427 failures (with a significance level $\alpha = 0.1$). Several vendors proposed a system for the "Rekening Rijden" system. Five vendors were allowed to work their system in more detail, to build a corresponding simulation model, and to evaluate the performance. For each proposed system for the "Rekening Rijden" project 4 million passages were simulated, and 123 thousand real passages analyzed. The results of this evaluation are described in the confidential final report of the first phase [87].

Although during the first phase of the "Rekening Rijden" evaluation it was

already clear that the requirements were met under normal circumstances, it was still an open question how to characterize the circumstances under which failures were made [89]. In the second phase of the evaluation study a probable cause could be identified for 90% of the failures. For 10% of the failures the occurrence seemed completely random. This work was partly the verification and validation effort that is requested in the next chapter.

3.6 Conclusion

Central to this thesis is the evaluation of a road-pricing application; an evaluation that had to be done very carefully because of the political delicateness. In this chapter we have given a short overview of the possible techniques that can be used to introduce such a road-pricing scheme in the Netherlands and of the political decisions surrounding this complex system under study in this thesis over the years. Three archetypes of road-pricing schemes are introduced. One system is the archetype of a road-pricing system that can implement the scheme exclusively by equipment on board of the vehicle. Another system is the archetype of a road-pricing system that can implement the scheme exclusively by equipment along the roadside. Finally, there is an archetype that implements the road-pricing partly on equipment in the vehicle and partly on equipment at the roadside. This latest archetype, the "Rekening Rijden" system, is the system studied in detail in this thesis. The behavior of this system is not trivial to describe, because it consists of several interactions between subsystems and vehicle parts. These different interactions are not at the same moment and location, but can be distributed over several meters and several seconds. The details about the order of these interactions (and their results), are important to estimate the quality of the overall system.

In the next chapter, the "Rekening Rijden" system will be described using a Discrete Event Modeling methodology. This methodology is able to model both the control logic as the non-deterministic order of events. At the end of that chapter we raise a number of questions, related to the challenges introduced in chapter 1. These questions will be answered in the chapters thereafter.

Chapter 4

Modeling Methodology

4.1 Introduction

¹ As part of the evaluation study of "Rekening Rijden" a modeling and simulation approach is developed and incorporated in a software environment [44]. The environment is in use by the industry for technical evaluation studies of Electronic Fee Collection. As a result of this specific work we have formulated a general simulation concept for Intelligent Transport Systems (ITS) that include sensor subsystems for detection, communication and registration. We will introduce the virtual sensor concept for the modeling of such sensor subsystems. As a result of the virtual sensor concept we can limit the model to what happens during a closed time interval. In this interval it is not necessary to model time continuously or to sample time equidistantly; the level of abstraction allows one to model only the moments that the state of the sensor changes significantly. As a result we can model the total system as a discrete-event model.

This chapter describes the general simulation concept and aims at initiating a discussion on the possibilities and limitations of the modeling and simulation techniques applied.

¹Based on: A. Visser, A.J. van der Wees, and L.O. Hertzberger, "Discrete Event Modelling Methodology for Intelligent Transport Systems", paper 2016 of the proceedings of the Wor ld congress on Intelligent Transport Systems, Torino, Italy, November 2000. This article is extended with contributions from [43],[28]

4.2 Modeling methodology

4.2.1 Characteristics of the System

Systems can be modeled on several levels. Important for the level of detail of the model, is the question to be answered. In this case of the "Rekening Rijden" evaluation, it is to estimate the reliability of the system. The Dutch government had very strict demands on performance of the Electronic Fee Collection system, with error rates of the order 10^{-5} . With these error-rates of the system it was considered that the number of incorrect administrative actions of the government, such as sending bills to people that already paid electronically, was on an acceptable level. Other cases are free rides, when the driver did not pay electronically, but was not charged. This is problem-driven research, where the responsibilities are clear. It is the responsibility of the system builders to provide evidence for the performance of their system, while it is the responsibility of the scientist to prove that this sort of performance demands are feasible and to come up with solid methods to provide this evidence.

The methodology proposed here is a combination of modeling & simulation and measurement analysis. To evaluate the performance of an Automatic Debiting System, the systems have to be tested under a rich set of weather conditions, traffic flows and vehicle types. Modeling & Simulation can be used to explore the dependency of the reliability on these parameters, and to find the measurements and tests that are optimal to get a good estimate of the reliability. There are so many scenarios, the internal structure of the system so complicated, and the demands so severe, that an evaluation in an actual setup is not practicable, even if prototypes were available. For these reasons, it was decided to perform the evaluation in *simulation*, validated with available test results.

To be able to validate the simulation models, the accuracy and precision of the system should be defined high above the conceptual level of the physical sensors and their dataflows. The concept of virtual sensors [28] is applicable here, a concept that is a focus of research at the University of Amsterdam for a long time [41]. This concept uses an uncertainty model of the sensor in conjunction with the influence of one or multiple physical sensors, from which the accuracy and precision can be estimated as a function of environment variables, such as weather conditions, traffic flows and vehicle types.

Because of this dependency, the performance of an ADS can only be evaluated when a test environment exists that is able to generate enough and realistic variance in the stimuli of the ADS. For this reason a simulator framework was created, able to generate a wide variety of scenarios.

4.2.2 The Virtual Sensor model

In this section, we introduce the Virtual Sensor concept as a reference for sensor system modeling. A *Virtual Sensor* is a (real or imagined) combination of a physical sensor with data processing and interpretation software, capable of measuring a feature of an object in the world. A *Virtual Sensor* is a (conceptual) device of which the output can be modeled in terms of the relevant characterizing parameters, and in the outputs of other physical or Virtual Sensors, see figure 4.1. The Virtual Sensors should be modeled at an appropriate level of abstraction. It should measure features that are easy to verify on the objects, independent of how difficult it is to actually measure that feature. The set of Virtual Sensors should be complete: with the set it should be possible to generate a sufficiently accurate characterization of the total system behavior. At the same time the abstraction level must keep the interactions between the various Virtual Sensors (relatively) simple, both in statistical and in causal relationships.

For an ADS a *Virtual Sensor* is a component that measures *one* of the features of a passing vehicle, for instance the length. The 'measurement' may be a single measurement or a composite result of a number of physical measurements. The length of a vehicle may, for instance, be determined by combining the time-stamps of the passage of the front and back of the vehicle (measured by laser curtains) with a measurement of its velocity. The length of a vehicle may also be determined from a single measurement, e.g., by processing a blob in an image. Both measurements can be represented with a Virtual Sensor that returns an estimate of the length of a vehicle, yet this measurement will be available at a different moment, with a different spread of the values, and a different sensitivity to the circumstances (such as the lighting conditions).



Figure 4.1: The Virtual Sensor Concept

The number of parameters one can use to characterize the performance is lim-

ited to the parameters of the scenarios that make up the simulation and validation tests. No assumption may be made on *any* lower level measurement data, otherwise the abstraction is lost, and validation results of lower level measurement data have to be extrapolated to the higher level. The dynamics of the parameters in the scenarios can be quite different. Some of the parameters, as for instance the viewing angle of a camera, are fixed for the design under study. Others, as for instance the distances between vehicles, can change constantly.

The parameters that characterize the passage of vehicles through an ADS can be divided in four categories:

- **Configuration parameters** These describe the layout of the ADS, fixed during operation. For example the installation height, camera angles, etc. This data follows from the design of the system.
- **Fixed vehicle parameters** These describe unchangeable properties of a single vehicle. Those properties are different for every passing vehicle. Examples of these properties are the shape and the color of the vehicle. The distributions of most of these properties in Dutch traffic are well known.
- **Dynamic vehicle parameters** These are parameters of the actual passage of the vehicle in the context of the traffic at that moment. This can be for instance the speed, track or the distance to other vehicles.
- **Scenario parameters** These describe the circumstances of the experiment, such as weather conditions, the intensity of the traffic, the fraction of heavy goods vehicles, etc. The distribution and correlation of most of these parameters in Dutch traffic are well known.

Since this set of parameters is the input for the simulation experiments, it makes sense to perform the modeling consistent at this level, and to use the same set for validation experiments.

The behavior of a Virtual Sensor has to be modeled on three different aspects in order to estimate the output of the sensor at different conditions:

- Accuracy and precision A Virtual Sensor which measures a vehicle parameter h (height) will do so with a limited accuracy and precision. These aspects are reflected in the mean and variance of the distribution of the measured h under similar circumstances, in a large number of trials. There are two sources that tend to blur the distribution:
 - Sensor noise, and
 - *Variation* of the measured parameters in the ensemble of vehicles during the trials

The *variation* is caused by the fact that the available parameters can only characterize certain *ensembles* of vehicles and conditions that are vali-

dated. This ensemble will exhibit an inherent spread, which manifests itself as a variance in the distribution (error contribution) in the Virtual Sensor output. This distribution can be larger or smaller for the different ensembles. The ensembles combined will contribute to multiple maxima. The actual distribution can only be estimated as a function of the characterizing parameters for each ensemble.

- **Timing** It is important to know *when* and *where* the data can be known. When is the measured parameter available and at what temporal rate does it changes significantly? If a virtual sensor for instance measures the height, the returned value will be much lower for an early request (the bonnet) than for a later request (the roof). Yet, one should not model the measurement as a snapshot of that moment. The algorithm could have a memory, and for instance remembers the maximum height encountered as the estimate of the roof height.
- (In)dependence Sometimes the output of a Virtual Sensor is not directly related to the passage of a single vehicle. It is also possible that a sensor gives multiple outputs for a single value (for instance when a shadow of a vehicle is identified as a separate vehicle). Further, it is possible that the Virtual Sensor gives only one result for two passages (for instance when two vehicles are visible as one, the merged result is a single measurement of double the length). To model the behavior of the Virtual Sensor in the latter case means that one has to model the correlation between the measurement of a parameter of one vehicle and the measurement of another vehicle.

Although not strictly necessary, it is desirable to choose the level of abstraction such that the parameters at that level are fairly independent, so that their interactions can be modeled in a simple way. This is important for the validation of the models and also for the number of cases that have to be evaluated separately in the simulator. Virtual sensors will thus preferably be chosen in a way that makes their interdependencies, *both in statistics and in timing*, simple and measurable.

4.2.3 Simulation Concept

A simulator is a powerful tool to evaluate a system. For an evaluation a tool not only has to be able to simulate the behavior of the system under study, but is also responsible for the creation of stimuli for the system (environment simulation) and to measure the system performance evaluation. The latter two are general for any system; only the behavior is specific for the system under study. The three aspects of the simulation concept are depicted as three vertical layers in figure 4.2. Horizontally the relation between description, model and

Environment Environment Environment Specification Model Simulation Monitoring Stimuli System (sub)System (sub)System Functional Model Simulation Specification (virtual sensors) (discrete event) System Settings Simulated System Performance System **Optimization** Performance. Specification System Analysis

implementation are given. Details of each aspect of the simulation concept are given in the description below the figure.

Figure 4.2: Simulation concept for Intelligent Transport Systems

Environment Modeling and Simulation

- **Environment Description** Here a description is given of what aspects of the environment of the system will be taken into account, in the sense that the environment is described as the source of external stimuli to the system. Based on the environment description, initial hypotheses can be raised regarding what environmental aspects are important for the performance of the system. These hypotheses can be refined during later system analysis. It is also possible to include control parameters in the environment description through the modeling of Monitoring Stimuli. For example, the system manager can change the speed of the flow through the system by prescribing a speed limit.
- **Environment Model** Here high-level modeling is applied to the Environment Description, in order to obtain a formal description of the environment which can be implemented in a simulation. At this level it is decided what aspects of the Environment Description are modeled or neglected.
- **Environment Simulation** This is a software model of the Environment Model. In our case, the environment simulation produces a sequence of stimuli for the (sub)System Simulation: the passages of vehicles. The Systems under study are sensitive systems compared to the systems typically used to measure the passages of vehicles. Mostly inductive loops are used, which typically only register that a vehicle has passed. The performance of the evaluated systems can be influenced by the individual properties of the vehicles, which means that the level of detail in En-



vironment Simulation is high compared with other Traffic Simulations [86].

System modeling and simulation

- **System Functional Description** Here a description is given of the functioning of the system, including systems settings (e.g. parameters calibrated for an expected work load or priority parameters for a communication protocol) that influence its behavior. The system settings can possibly be influenced by the system itself, for example the priority parameters may be changed by the system when the intensity of the flow through the system changes. The functional description also includes a geometric description of the system, the static "hardware" configuration of a system, or a network layout description, specifying the nodes and links in a communication network. In any case, the location of sensors and communication subsystems must be defined by using so-called sensitive volumes of the sensors, so that the discrete-event methodology applied in the simulation is able to compute the moment when the sensors start and stop receiving stimuli from the environment.
- (sub)System Model High-level modeling is now applied to the System Functional Description, in order to obtain a stochastic or deterministic description of the behavior of the system, which can be implemented in a simulation. The Virtual Sensor concept, discussed previously in this chapter is an essential means to establish this at a level that can be *validated*. At this level it is decided what aspects of the System Functional Description can be implemented, possibly applying an incremental development of the simulation. Since we consider systems where the software is an integral part of the system, the modeling of algorithmic errors (due to the realtime character of the system) is an important aspect of the (sub)System Model.
- (sub)System Simulation (discrete event) This gives the simulated system behavior, as implemented in the computer simulation. The Virtual Sensor concept encourages this choice of *discrete-event* simulation. In fact the framework of the simulator is only a scheduler of events, and all the system behavior is located in the event-handlers. The event-handlers define the response of the system on this event taking into account the circumstances at that moment. This means that the simulation environment needs to provide a rich library of spatio-temporal functions to calculate the circumstances during a passage. By describing the behaviors of the subsystems at specific places and times during the passage, the relation and interaction between subsystems-responses are made explicit.

System Performance Evaluation

- **System Performance Description** These descriptions are formulated to quantify the correct system response in a form of limits on the system performance. When these limits are explicitly known, the system can make use of this information and tune its behavior accordingly. For example, a balance can be created between the assured revenues of the system and system reliability. The performance of the system can be optimized against this balance by shifting the decision of a free ride in case of doubt.
- **Simulated System Performance** The simulation of (sub)Systems under stimuli of the environment will result in a System Response (logs and reports), which can be used to analyze the system.
- **System Analysis** This is a process that compares the Simulated System Performance with the System Performance data. A technical analysis (for example failure mode analysis) can be performed comparing the System Response with the requirements, giving SQFs (System Quality Factors) for the system. The system analysis may result in adaptations of the process, in the form of Monitoring Stimuli to the environment, change of System Settings in the System Functional Description and optimization of System Performance Description, as is shown in figure 4.2.

The above concept can be applied to many types of Intelligent Transport Systems.

4.3 Simulation environment

In the Dutch "Rekening Rijden" project [95] several prototypes were evaluated. For each prototype of a consortium a corresponding ADS model was implemented. As described in the previous section, only a part of the simulator is specific for the system under study. The Environment Modeling and Simulation and the System Performance Evaluation are generic. For this reason a simulator should have the structure of a framework and a kernel. The framework represents the generic part, where several kernels can be plugged in. Each kernel represents a different ADS model. The simulation environment provides a mechanism for this flexibility. The user provides a configuration file, containing the geometry of the ADS, and the event handling routines. The geometry of an ADS is defined by one or more gantries, with on each gantry a number of sensors. Each sensor is defined by a sensitive volume, pointed towards the road surface. Each sensory subsystem has a differently shaped sensitive volume, which can be modeled for the simulation environment by parameterizing a so called VolumeTriangles. A laser scanner can for instance be modeled with a very narrow VolumeTriangle, while for a camera system the

lateral and longitudinal opening angle are in the same order of magnitude (see figure 4.3).



Figure 4.3: The sensitive volumes of an ADS model consisting of two laser scanners (two red curtains at the foreground) and four cameras (blue cones in the background)

The event handling routines define the behavior of the subsystems. These routines, which are written in C, are linked with the ADS simulator through the model interface. They contain the parameterizations of the subsystems and all the logic to connect subsystems with each other (see figure 4.4).

The Weather and Traffic modules [85] in the evaluation framework represent the Environment Simulation mentioned previously in this chapter; the modules of the ADS simulator represent the (discrete-event) (sub)System Simulation and the System Analysis, while the sets described as "Unjustified Registration" ... "Free Ride" indicate the System Quality Factors (SQFs) that result from the System Analysis.

The facilities of the ADSSIM simulator are described in [64]. In this chapter we only describe how the simulation of the system behavior is carried out. The evaluation framework calculates trajectories of each vehicle and launches the vehicles along this trajectory passing the ADS. Subsequently the simulator calculates at which time the vehicle enters and leaves all sensitive zones of the subsystems of the ADS. At each of these times an event is scheduled, called InZone and OutZone events. These events, which are always scheduled by the simulator, represent the beginning and end of the time-interval where behavior of the subsystem can be expected. No state changes are allowed outside this time interval. The InZone event forms the basis to implement the actual behavior of the subsystem.

Basically, a particular virtual sensor defines the functionality of small set of



Figure 4.4: Software design of ADSSIM simulator

event handlers. A measurement of a vehicle parameter is distributed over at least three events: InDetectionZone, VehicleMeasured and DetectionCompleted, as indicated in figure 4.5. Each vendor that participated in the "Rekening Rijden" project defined a number of events that typically occurs during a passage of a vehicle through their system. The details of their system were encoded in the event handlers, provided in the detailed user models indicated with the files detection.c, classification.c and registration.c in figure 4.4.



Figure 4.5: Distribution of the Virtual Sensor Model over event handlers

In general the behavior of the virtual sensor is distributed in the following way over these events:

InDetectionZone

• The virtual sensor becomes active when a vehicle enters an appropriate sensitive zone, as marked by a vehicle event InDetectionZone. The actual

moment is based on calculating the first overlap between the 3D shape of the sensitive zone with 3D shape of the vehicle on a certain trajectory.

• In the event handler of InDetectionZone the modeler has to provide the code that calculates the appropriate time to schedule the sensor event that models the actual 'measurement' of a vehicle-parameter. The InDetectionZone only indicates the earliest time that a sensor can 'measure' or interact with a vehicle. The actual arrival time depends on the property to be measured and the sensor. Typically more of the vehicle should be visible before the a estimate of a vehicle-parameter can be given. The moment that typically an estimate can be given is called a VehicleMeasured event. Each vehicle-property has its own time window where it can be measured. For instance the width can often be earlier estimated than the height. When other processes are waiting for the output of a virtual sensor one should schedule the actual measurement at the earliest possible time. When the output is not time-critical, and the precision is improved by long sampling, one should schedule the actual measurement at the latest possible moment.

VehicleMeasured

- In the event handler of the VehicleMeasured events a reality check first has to be made. The virtual sensor is designed under certain assumptions of independence of parameters and vehicles. This event was scheduled at a time when a measurement for an undisturbed passage is typically available, but now a check has to be made if this passage is undisturbed. When there are vehicles nearby, it is possible that a part of the vehicle is obscured. It depends on the geometrical configuration how long this obstruction occurs. Typically this obstruction is only temporary, because each vehicle has to pass under the gantry, so that an undisturbed view from the top is possible. When the vehicle is not completely visible, and the required feature can not be accurately measured, the sensor event has to be rescheduled at a time that the vehicle becomes completely visible.
- After checking if a vehicle can be measured, an estimate has to be made how accurate a typical result of a measurement of the vehicle parameter can be made. The output of the virtual sensor would be a transformation of the true value as reported by the simulation environment augmented with the error contribution as described in Figure 4.1. The amount of noise will be a function of environment and dynamical vehicle parameters.

DetectionCompleted

• The final reality checking has to be performed when no more information can be expected from the virtual sensor (a DetectionCompleted event). This prepares all the information used by the coordination algorithm in a separate event handler. This DetectionCompleted event has to occur before the OutDetectionZone event, which indicates that the vehicle has left the sensitive zone of the sensor. In the simulation this means that there is no longer an overlap between the 3D-shape of the vehicle with the 3D-shape of the sensitive zone.

Formally, virtual sensor models are defined per vehicle parameter, each validated separately. The natural implementation of these validated models in the discrete-event simulator ADSSIM is done by grouping virtual sensors that have similar characteristics in timing. For instance, consider a sensor where the position measurements x and y become available simultaneously (due to the way the physical sensor plus the modeler's interpretation software actually works), and are handled simultaneously, then one would define a virtual position sensor for (x, y), triggered by entry into the physical sensor's detection zone in a single InDetectionZone event handler. In turn this would trigger the simultaneous treatment of the measurement in the same VehicleMeasured event handler. On the other hand, if x and y are determined independently or handled in different ways, then a modeler should treat the measurements of xand y in different event handlers, since in this case they do not share their timing. Thus the strict independence of virtual sensors in their statistical aspects (true by definition) does not necessarily imply independence of their timing and this is reflected in their implementation in the discrete-event simulator.

The microwave equipment can be modeled as a set of virtual sensors as well, especially the OBU-localization component. The InCommunicationZone event is then used as a counterpart of the InDetectionZone event. A modeler can use it to schedule the communication-related events OBUMeasured and CommunicationCompleted, which are equivalent with detection-related events VehicleMeasured and DetectionCompleted.

4.4 Formal description

Now that we have the functional and temporal description of the "Rekening Rijden" system, we want to extend this to a formal description. So let RR be representing the "Rekening Rijden" system. During a certain time-interval T the system RR has to correspond correctly on the passage of a number of vehicles X. During validation tests along the Dutch highway A12 measurements were collected for time-intervals T of half an hour, which corresponds

for typical Dutch traffic to a set of 5000 passages. Measurements of multiple time-intervals have to be combined to estimate the performance of the *RR* system with error rates in the order of 10^{-5} . Each passage, or input, is slightly different, so \vec{x} represents a vector with the actual values of the vehicle and its passage, as for instance its length and its speed. In the simulator this vector is implemented as an object with multiple attributes. At the end, the system *RR* responds with a number of payment descriptions *Y*, which is a set of outputs \vec{y} . If the system *RR* responds correctly this will be a set of 5000 outputs \vec{y} , but it can also occur that there are missed or extra outputs. The response of the system *RR* is a function of its internal state \vec{q} , which represents all the factors why the system *RR* would respond differently to two passages. Neither this internal state \vec{q} , nor the state transition function $\vec{q} = \Delta(\vec{\omega}, \vec{q})$ and output function $\vec{y} = \Lambda(\vec{q})$, as illustrated in figure 2.2, are known for the system as a whole. The two functions together describe the overall behavior of the system $\vec{y} = \beta_q(\vec{\omega})$, a behavior that can be compared with measurements along the road.

$$RR = (T, X, \Omega, Y, \beta_q(\vec{\omega})) \tag{4.1}$$

We hope to construct the behavior of the RR system on the basis of the behaviors of the subsystems. Based on the description 3.3.8, three different subsystems can be distinguished: the Communication, Detection & Tracking and Coordination & Registration subsystem. For each subsystem we will extend the formal description.

4.4.1 Communication subsystem

The microwave antennas on the portals are scheduled to send a welcome message regularly to start a transaction with OBUs on board of the passing vehicles. During the transaction administrative information is exchanged and the fee paid. The transaction consists of several messages, which can be all be used to approximate the location of the OBU based on the characteristics of the microwave signal, which results in a track.

$$COM = (T, X, \Omega_C, Y_C, \beta_{qC}(\vec{\omega}_C))$$
(4.2)

The output \vec{y}_C should be (TransactionCompleted, CommunicationTrack) when the passing vehicle \vec{x} has an OBU, and should be empty \emptyset when the passing vehicle \vec{x}' has no OBU. To reach TransactionCompleted, a protocol between two state-machines (one at the OBU, one at the Road Side System) has to be followed. The state-space at the roadside is more complex, to take care of multiple OBU's communicating at the same time.

4.4.2 Detection and tracking subsystem

The system needs to sense and track all vehicles passing. Traditionally this is done via loops in the road that respond to magnetic disturbances, but those systems are too crude for charging applications. Optical systems can sense vehicle passages more reliable and accurately. Optical systems can be based on cameras or on laser curtains, which results respectively in detailed track or a track consisting of a single point.

$$DET = (T, X, \Omega_D, Y_D, \beta_{q_D}(\vec{\omega}_D))$$
(4.3)

The output \vec{y}_D should be (DetectionCompleted, DetectionTrack) for every passing vehicle \vec{x} . To reach DetectionCompleted, the vehicle has to be out of sight. This seems to be a simple decision, but trailers and temporary occlusions require that this decision is made on the basis of multiple measurements, which also introduces a state space in the detection subsystem.

4.4.3 Coordination and Registration subsystem

The system needs to take photographic pictures of the front and the back of all vehicles passing without paying electronically. The fee can than be collected afterwards by sending an invoice to the owner of the vehicle, who is identified by the number of the license plate. Temporarily images can be made for the purpose of a coordination decision.

$$REG = (T, X_R, \Omega_R, Y_R, \beta_{q_R}(\vec{\omega}_R))$$
(4.4)

The input X_R of the *REG* system are a combination of the passing vehicles X and the output Y_C and Y_D of the *COM* and *DET* subsystems. Unfortunately, this combination is not sorted, it is not clear which \vec{x} , \vec{y}_C and \vec{y}_D belong to each other. This match has to be made in the coordination and registration subsystem. The input of these matching routine are all DetectionTracks and CommunicationTracks available at that moment. The output \vec{y}_R should be a TransferRegistrationResult for every passing vehicle \vec{x}' without OBU and a TransferCommunicationResult for every vehicle \vec{x} with an OBU, and no result \emptyset when no vehicle \emptyset passes. The correct TransferRegistration result should be that two pictures (from the front and rear license plate) were captured which contain a good view on the license plate of the correct vehicle. Yet, many other situations are possible: no license plate is visible, the license plate could only be partly in view (due to occlusion or inaccurate localization), or the license plate of another vehicle is in view (very inaccurate localization or a mismatch between the tracks).

4.4.4 Coupling of the subsystems

The RR system can be seen as a combination of those three subsystems, where the output of the DET and COM subsystem is fed to the REG subsystem. The output of the REGsystem is evaluated as the output of the whole RR system. The result is:

$$RR = (T, X_R, \Omega_C, \beta_{q_C}(\vec{\omega}_C), Y_C, \Omega_D, \beta_{q_D}(\vec{\omega}_D), Y_D, \Omega_R, \beta_{q_R}(\vec{\omega}_R), Y_R)$$
(4.5)

Several international industrial consortia, which could build the "Rekening Rijden" system for the Dutch government, have made models of the behavior of their subsystems based on this subdivision. They used the programming language C for these models. Inspection of their code revealed that an internal state could be found for every 10 lines of code. In total one could estimate that between 500 and 1800 states could be distinguished inside their code.

The output Y_R can be characterized by a set of 48 input-response combinations, from which 3 represent correct behavior. The systems under study were mature, and correct behavior can be expected for more than 99.98% of the cases. This means that we are near the lower bound of the Shannon Entropy, introduced in chapter 2, where the all outcomes, except three, are near zero. It seems nice that the uncertainty in the system behavior is low, and that we can predict nearly at all times the response (namely, when it behaves nominally). Unfortunately, this also means that experiments will not be very informative, because the information about the non-nominal responses is sparse. The Dutch government was precisely interested in modeling the occurrence of those sparse non-nominal events, with probabilities below 10^{-4} , 10^{-5} or even 10^{-6} .

Most of the code provided by the consortia is to model the non-nominal events. As comparison, a model of a "Rekening Rijden" system is made that always make the correct decision; all events are nominal events. Such a 'perfect' system can be made with 600 lines of code and a total of 12 states. The consortia provided between 5000 and 18000 lines of code to accurately describe their system. A lot of effort of the consortia is put into recreation of the control logic of the real systems. Different error sources are included, based on measurements, simulation, statistics or first principles. Special care has been taken to indicate problems with the timings in the different subsystems, because details in the timing can influence the accuracy of the match when the information of the detection and communication systems is combined.

4.5 Discussion

The modeling methodology presented in this chapter has been used within the Dutch "Rekening Rijden" project by five vendors in order to demonstrate the feasibility of their ADS system. The industrial end-user community has a growing awareness of the usefulness of simulation as a tool for demonstration and promotion of their ADS concepts. Full-scale evaluations (containing in the order of a million vehicles) can be done in a number of days, depending of course on the complexity of the implemented simulation model and the size of the available computer network.

The next discussion points exist with respect to the presented discrete-event methodology:

- The discrete-event property is valid as long as the number of hierarchical decompositions remains limited. In other words, the modeling is done on a high level and the granularity of the models must remain comparatively coarse. In case fine granularity is needed in the modeling, it is preferred that such detailed computations are computed without the use of extra events to avoid an explosion of the required number of events.
- A case study is performed on the level of granularity for one of the subsystems: the communications part [90], as described in chapter 6. The conclusion of this case study was that an increasing level of modeling detail could change the estimated performance both in the positive as in the negative direction. An increasing level of detail was equivalent with an increase of the used computer resources, but by applying a high-level model as worst-case assumption, the usage of the detailed model could be reserved for the passages that had a high chance of failure, hereby limiting the used computer resources to an acceptable level.
- Discrete event simulation is almost by definition a tool with promising design facilities, since only 'relevant' events have to be taken into account in the simulation, and details are only taken into account where needed.

Last, but not least, we have seen that modeling at the abstraction level of virtual sensors means that the behavior of the subsystem will be defined as function of the dynamic vehicle parameters, which means that there has to be an adequate generation of this vehicle dynamics in the simulation environment. In the next chapter validation of the traffic generation is worked out in more detail.

Chapter 5

Calibration of the Traffic Model

5.1 Introduction

¹ To model traffic, many models are available [37]. These traffic models can be divided into micro-, macro- and meso-models. This division can be made both on time and space scales. When space is taken into account, the distinction is between large-scale models, which cover a large area or country, medium-scale models, which cover (a part of) the network of a city or motorway network, and small-scale models, which can be used for small networks or single intersections. In general, large-scale models are used to study macro-economic or land-use aspects, like fuel price or developing rural areas. Medium-scale models are used to study effects like building new motorways, but also for traffic management measures like route guidance. Finally, small-scale models are used to study the effects of measures like ramp metering, traffic signal control, lane control, autonomous intelligent cruise control, etc. The size of the network strongly influences how time is modeled in the simulation process. For large-scale networks it is normal to model traffic streams as a whole, based on the flow theory of fluids. Meso-scale networks are typically modeled with cells occuppied with vehicles. Small-scale networks can be handled by simulating every individual vehicle.

¹Based on: A. Visser, H. Kok, M. Bergman, H.H. Yakali, L.O. Hertzberger, "Calibration of a traffic generator for high-density traffic, using the data collected during a road pricing project", paper 4052 of the 9th World congress on Intelligent Transport Systems, Chicago, Illinois, October 2002.

5.1.1 Why should a microscopic Random Traffic Generator be built?

The modeling of the system under study requires an efficient run-time model derived from the microscopic road traffic characteristics. The system characteristics are not only determined by macroscopic parameters such as the traffic flow intensity but also by microscopic properties such as head-ways between vehicles. For instance for the communication subsystem short headways can influence the available communication time and the probability of disturbances through data collisions and shadowing effects [15]. For the registration subsystem the head-way is an important parameter to determine the chance of occlusion.

It is important that the surrounding traffic is realistically represented. Initially for the evaluation a model was proposed [77], which showed a good correspondence between model and reality for both German and Dutch motorways on the aggregates of the important parameters. Unfortunately, not only the mean and variance of the parameters are important, but also the form of the distributions and the interrelationships between the parameters. Central in the initial traffic model [15] was the safe distance which is maintained when a vehicle is following another. This was modeled with a Gaussian mean and variance, which is a crude approximation of the actual asymmetric Pearson distribution (see figure 5.5).

In rush hour traffic, the vehicles follow each other at close distances. The drivers adjust their driving behavior on the basis of the perceived distance to the preceding vehicle. The modeling of this dynamic driving behavior is a task ion itself [7]. The realism of the dynamic driving behavior is highly dependent of the correct initialization of the distances between the vehicles, i.e. the arrival times of the vehicles at the start of the simulated track need to have the correct variation [46].

The initial solution was to feed the trajectory simulation with real measurements of arrival times, because a large set of measurements was available, and under the assumption that a model could never beat measurements in richness of variation in distributions. Unfortunately, traffic measurements are normally available in large quantities, but these raw measurements can contain artifacts from the measurement system. These artifacts can have a devastating effect on the simulation, when for instance one vehicle driving twice as fast as the rest of the traffic induces breaking and acceleration behavior in many other vehicles before and after him. So, we learned to inspect all traffic measurements for errors in the measurement system (often related with motorcycles or really heavy lorries), because one wrongly detected passage can disturb the simulation of a traffic stream for a long time. Analyzing and correcting artifacts in the measurements is only possible with good knowledge about the distribution of parameters, to be able to distinguish outliers from natural variance. The development of a Random Traffic Generator based on the Mixic model [7] was accompanied with a constant comparison of the generated data with the available data set. This validation effort not only improved the quality of the implementation, but also initiated multiple corrections in the data set. This indicates the benefit if one combines measurement driven simulation with simulation driving measurement analysis.

5.1.2 What can be learned from the development and calibration of the RTG?

Although the form of traffic distributions is known in principle [40], the parameters of those distributions have to be derived from measurements. The distributions for vehicle types, interarrival times and vehicle speeds have been studied and will also be presented in this chapter. A complicating factor is that there exists a correlation between the distributions, and correlation with values of the preceding vehicle [82]. This means that the traffic generation cannot be modeled by a simple random process. This is illustrated in figure 5.1. At the right side of the figure two times a speed distribution is shown, which seems to be a normal distribution which an average of 120 km/h and a standard deviation of 8 km/h for moderate traffic (intensities around 60% of the capacity of the road). At the left side the dynamics is shown for measured speeds, random draws from the normal distribution. If one compares the actual measurements with independent draws, one sees that the deviation from the average has the right amplitude, but the variance occurs at too high a frequency, indicating that the process has some memory, that correlation between passages.



Figure 5.1: The measured speed distribution compared with data from a random generator, without taking into account the correlation with preceding vehicle's speed. Example from [82]

To be able to model this correlation, the characteristics of two passages at time i and j are needed for a complete description a vehicles dynamics. By combining the characteristics of two passages in a single sampling variable set, the process regains the Markov property. In system theory [54] this would mean that we could define a behavior system by filtering the data with a mask of depth 2 on the five relevant vehicle variables: type, length, speed, arrival time and interarrival time IAT.

As described in chapter 2, a mask of depth two for five variables can lead to ten sampling variables S_{1-10} . In our case the influence of the previous IAT is not modelled, so nine sampling variables S_1 to S_9 can be distinguished, as illustrated in table 5.1. The sampling variables $S_5 - S_9$ at passage j are not only dependent on the sampling variables $S_1 - S_4$ of the previous passage i, but also cross-correlation with parameters of the current passage can be indicated. At the right side of table 5.1 the major correlations found in [7] are indicated in four different colors.



Table 5.1: The sampling variables for a behavior system that defines the RTG, and their major correlations.

A natural ordering of the generation process can be derived from these correlations. The variables of vehicle j which only depend on the previous passage i are S_5 and S_6 (the black correlation). When the black correlation is solved in step 1 by creating the generative functions $S_5 = f_{1a}(S_1)$ and $S_6 = f_{1b}(S_1)$, the type and length of $vehicle_j$ are known in principle. With this information other variables of current vehicle can be generated; the arrival time S_8 and the speed S_7 (the blue and green correlation). The generative function $S_8 = f_{2a}(S_2, S_3, S_4, S_5)$ for the blue correlation can be combined with a generative function $S_9 = f_{2b}(S_4, S_8)$ for the red correlation in a single step, because the red correlation only depends on the blue correlation. The arrival time and IAT of $vehicle_j$ are known. The last step is finding a generative function $S_7 = f_3(S_2, S_3, S_5, S_6)$ for the speed to solve the green correlation. The details of the generative functions are worked out in section 5.3, here the order of appliance of the generative functions is determined, as needed in section 5.2.

So, determination of the type S_5 and length S_6 of the vehicle is the first step in the generation process. Errors in the vehicle type distribution have severe consequences for calibration of the next steps in the generation process, because both interarrival time and speed are a function of the vehicle type in our model. Yet, many gas-kinetic and macroscopic traffic models assume homogenous traffic types [40]. In this study we combined the measurements of two independent systems which resulted e.g. in a much better vehicle type classification. As a consequence, the interarrival time and speed generators could be calibrated more accurately.

5.1.3 What is the scope of the results?

The road pricing system was meant to operate during the morning rush hours (week-days between 7:00 and 9:00), so the traffic generator was calibrated with passages from this period and validated with another set from the period. The traffic situations during this period can be classified as flowing at high speeds at very high densities (volumes around 86% of the road capacity). Although this situation is near instability according to classical traffic flow models [60], it is quite common during rush hours in the Netherlands (the other common traffic situation is congestion).

This combination of high speed and high density is a good condition to stress test the system under evaluation. High speeds reduce the time to correct errors in measurements of the subsystems. Further, at high densities the distances are relatively short. As indicated earlier, these short distances can influence the performance of the subsystems. This means that the interarrival time (IAT) has to be carefully validated, especially around large surfaces such as present on lorries.

The calibration and validation set were acquired on the Dutch highway A12 for a wide variety of circumstances in the years 1999-2001². The location was especially selected for the absence of disturbing effects like nearby curvature, entrances or exits. The Dutch highway system has a high density of this type of disturbances, so the location is atypical for Dutch traffic. The benefit of this location is the characteristic that, although the flow of traffic is high, traffic jams are very sporadically seen. This allows to study the combination of high flows and high speeds. In this sense it is a unique measurement point to gather experimental facts to understand the microscopic structures in synchronized traffic states [68], which was not reported outside of Germany yet.

In free-flowing and congested traffic the speed is constantly high or constantly low, respectively, which means that the traffic flow is mainly a function of the density. In synchronized traffic the density is nearly constant, which means that the traffic flow is dominated by the speed. The variations in the speed are small and slow, but overall all vehicles drive slightly below their intended speed. At the microscopic level this leads to following picture. In free-flowing

²A. Visser, Calibration and validation data for the simulation model ADSSIM, online overview, 2001. http://www.science.uva.nl/~arnoud/projects/TrafficLab/

traffic, platoons can be found of a few vehicles following a leader that is driving slightly slower than the intended speed of the followers. In synchronized traffic much larger platoons are formed (in the order of ten vehicles) of vehicles driving at the same speed, with as leader a vehicle which regulates its speed by looking slightly further ahead. Previous research [52] has shown that three types of synchronized traffic can be distinguished:

- i stationary and homogeneous states, where the average speed and flow are nearly constant for several minutes
- ii stationary, non-homogeneous states, where the average speed is constant for a lane, but the flow noticeably changes.
- iii non-stationary and non-homogeneous states, where both average speed and flow change abruptly.

In addition, it is found that transitions from these synchronized states to freeflowing traffic are rare, but transitions between these three synchronized states are frequent.

This means that for understanding the microscopic structures in synchronized traffic states the relations between several aggregates of single vehicle measurements have to be made. Important aggregate measurements are for instance average speed, average flow, average density, headway distribution and speed difference distribution. The dynamics of these one-minute aggregates over 5-10 minutes periods are important for a correct identification of the state.

5.2 Traffic generation

Generating traffic for a microscopic traffic simulation model comes down to simulating the registrations that would be made by measurement systems along the road: inductive loop detectors in the road pavement or road pricing systems installed on gantries above the road. New vehicles should be generated from a random distribution per lane, with the instantiation of the following features:

- the vehicle type: car, lorry, van, motorcycle. This will implicitely define the vehicle's length;
- the interarrival time (IAT) between the newly entering vehicle and the preceding vehicle: this interarrival time implicitly defines the exact moment of arrival at the entrance, since the arrival time of the preceding vehicle is known, as well as its length and speed (determining the clear-ance time);
- the speed of the entering vehicle.

We have assumed that these three features are highly coupled, and are also dependent on the characteristics of the preceding vehicle, as described in section 5.1.2. That is why the order of instantiation is important. Our model is based on the model used in the Mixic-simulator [7].

5.2.1 Basic structure of the Traffic Generator

During execution the random Traffic Generator consequently generates vehicles for a single lane and assigns type (including vehicle length), interarrival time and speed for the newly generated vehicle in a pipeline of subroutines (as is shown in the flow chart in figure 5.2). This serial process implies that a generator subroutine further down in the chain can use vehicle attributes that were generated by a subroutine that was executed earlier in the chain but not vice versa. For example the Speed Generator can build on vehicle type and interarrival time of the new vehicle j in determining the new speed, while the Inter Arrival Time Generator can only use the type and the length of the new vehicle.



Figure 5.2: Basic structure of the Random Traffic Generator for Free Flowing traffic. The new vehicle is indicated by the index *j*; the preceding vehicle indicated by *i*.

The subroutine 'Vehicle Type Generator' implements the generative function $S_5 = f_{1a}(S_1)$; $S_6 = f_{1b}(S_1)$ of section 5.1.2. The subroutine 'InterArrival Time Generator' implements both $S_8 = f_{2a}(S_2, S_3, S_4, S_5)$ and $S_9 = f_{2b}(S_4, S_8)$. Finally the subroutine 'Speed Generator' implements $S_7 = f_3(S_2, S_3, S_5, S_6)$. At the end of the series of subroutines the new vehicle j is launched into the trajectory simulation and its attributes can be used as input for the generation of the next vehicle. The details of the subroutines are given in section 5.3.

The trajectory simulation is responsible for the calculation of the vehicle's trajectory over a stretch of road, as indicated in figure 5.3. The trajectory simulator, in contrast with the traffic generator, is inherently multi-lane. Both lateral and longitudinal dynamic behaviors are modeled. This dynamic behavior can be switched off at the beginning and end of the road to prevent effects such as acceleration near the exit because no vehicles are left in front. The dynamic behavior is out of scope of this chapter, the trajectory simulation is just seen as consumer of the vehicles produced by the Traffic Generator. The Traffic Generator is able to produce vehicles on multiple lanes, but the outputs for the different lanes are produced by independent processes. Each lane (e.g. fast lane, slow lane) has its own characteristics, and should be separately calibrated and validated.



Figure 5.3: The simulated stretch of road. At the middle three sections, full dynamic behavior is simulated.

5.3 Calibration and Validation of the subroutines

In this section the subroutines of the Traffic Generator will be explained in more detail and the calibration and validation results will be presented.

5.3.1 Vehicle Type Generator

The determination of the vehicle type is done using a 4 x 4 one step transition matrix for each lane. This matrix contains the conditional probabilities P(type(i)|type(j)), where *i* is the index of the preceding vehicle, and *j* the index of the current vehicle. Normally the transition matrix is calibrated using only inductive loop detector measurements under various conditions and the measured length is then used to distinguish vehicle types. In our case we had access to both inductive loop detector measurements and laser curtain measurements. With a laser curtain the width and the height of the passing vehicles can be measured. We have combined the measurements of both detection systems in several combinations and found out that classification purely on the inductive loop length is not always correct [89]. ³ For instance, by setting a threshold of 6 meters on the length 4% of the trucks are missed and a considerable amount of false positives are generated: cars and vans are classified as trucks (19%). When the classification is based on the laser curtain height and width measurements no cars or vans are classified as trucks (for a 5 minutes

³Based on [89]'s appendix G "Number of Lories in a session" by H.H. Yakali.

validation interval) and 10% of the trucks are classified as vans. Not for all vendors laser curtain measurements were available.

Hence, for this study the transition matrix is calibrated with the height and width from the laser curtain measurements. Only for the sporadic case (< 1‰) that no laser curtain measurements are available, the length from the inductive loop measurements is used. The inductive loop system was also quite reliable (missed passages < 1%) and the chance that both systems missed a passage is very small ($2 \cdot 10^{-5}$) [89]. The combined chance is slightly larger than the product of both independent chances (< $1 \cdot 10^{-5}$), which indicates that the chances are correlated. The most likely source of the correlation is a motorcycle, from which it is known that they are poorly detected by inductive loop systems and also difficult to detect for laser curtains.

When laser curtain measurements were available we used the following rules:

motorcycle:	$\mathtt{width} < \mathtt{1.3} \pmod{\mathtt{meter}}$	
truck:	height > 3.0 AND width > 2.5	(meter)
van:	height > 1.8 AND width > 1.8	(meter)

When only inductive loop measurements were available we used the following rules:

 $\begin{array}{ll} \mbox{truck:} & \mbox{length} > 9.0 & (\mbox{meter}) \\ \mbox{van:} & \mbox{length} > 6.0 & (\mbox{meter}) \end{array}$

In all other cases we assumed that the vehicle is a passenger car.

Figure 5.4 shows the results of the classification. The distribution of the length, width and height are given. The measurements of the validation set are classified using the previous rules. Each vehicle type has its own color and one can distinguish the different height, width and length characteristics of the vehicle types. As one can see in figure 5.3.1, it is difficult to distinguish vans and cars in width; in that case the height is best indication. I expect that recent popularity of Special Utility Vehicles has made it even harder to distinguish van and cars based on their outher dimensions.

In table 5.2 the conditional probabilities are given that were obtained by applying the classification rules. The calibration set and validation set are derived from the measurements on the Dutch highway A12. The generated set is the result of simulation run of the traffic generator, with the parameters derived from the calibration set.

As can be seen in table 5.2, mainly passenger cars can be found in the fast lane (respectively 97.5%, 95.5%, 96.8%). In the slow lane only half of the vehicles are passenger cars (respectively 57.5%, 58.9%, 60.4%). It should also be noted that there is a small variation between the calibration and validation set, in the order of a few percent. This is the natural variation of measured data sets.



Figure 5.4: Distribution of the dimensions of the 4 vehicle types.

5.3.2 InterArrival Time generator

The interarrival time (IAT) of the new vehicle with respect to the preceding vehicle is defined as the time elapsed between the passing of the rear bumper of the preceding vehicle and the arrival of the front bumper of the new vehicle.

Two different IAT distributions can be distinguished:

- free or unconstrained vehicles: vehicles which do not have to modify their time-space trajectory when approaching their preceding vehicle;
- followers or constrained vehicles: their time-space behavior is influenced by the presence of their preceding vehicle.

For the 7:00-9:00 traffic we have assumed that all vehicles are constrained, which is true for 99.9% of the passages. The influence on the driving behavior can be seen up to 15 seconds interarrival time between vehicles [37]. The peak of the interarrival time is typically less than 2 seconds (see figure 5.5). The distribution function of the interarrival time of vehicles at a detection point can be described by probability density function (pdf) of the Pearson type III distribution PT3(x) [7]:

$$PT3(x) = \begin{cases} 0, & \text{for } x < d;\\ \frac{1}{\Gamma(\beta)} \alpha^{\beta} (x-d)^{\beta-1} e^{-\alpha(x-d)}, & \text{for } x \ge d. \end{cases}$$
(5.1)

where *x* is the variable (in this case the interarrival time IAT), *d* is an offset, $\Gamma(\beta)$ the Gamma-function, β the shape parameter and α the scale parameter.

Examples of these distributions are given in figure 5.5. As one can seen in this figure, the offset is small, but really exists (otherwise the vehicles collide). All

	Slow lane	Middle lane	Fast Lane
Calibration set	$\begin{bmatrix} 0.376 & 0.069 & 0.004 & 0.128 \\ 0.065 & 0.019 & 0 & 0.045 \\ 0.002 & 0 & 0 & 0.001 \\ 0.132 & 0.041 & 0 & 0.117 \end{bmatrix}$	$\begin{bmatrix} 0.827 & 0.064 & 0.006 & 0.011 \\ 0.064 & 0.009 & 0 & 0.001 \\ 0.004 & 0.001 & 0 & 0.001 \\ 0.012 & 0 & 0 & 0.001 \end{bmatrix}$	$\begin{bmatrix} 0.949 & 0.022 & 0.004 & 0 \\ 0.022 & 0.001 & 0 & 0 \\ 0.004 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
Generated set	$\begin{bmatrix} 0.384 & 0.067 & 0.009 & 0.129 \\ 0.059 & 0.016 & 0.001 & 0.039 \\ 0.012 & 0 & 0 & 0.001 \\ 0.134 & 0.032 & 0.003 & 0.113 \end{bmatrix}$	$\begin{bmatrix} 0.802 & 0.060 & 0.012 & 0.016 \\ 0.060 & 0.010 & 0.002 & 0.002 \\ 0.013 & 0.002 & 0 & 0.001 \\ 0.015 & 0.003 & 0.001 & 0.002 \end{bmatrix}$	$\begin{bmatrix} 0.912 & 0.027 & 0.015 & 0 \\ 0.028 & 0.001 & 0 & 0 \\ 0.015 & 0.001 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
Validation set	$\begin{bmatrix} 0.417 & 0.056 & 0.002 & 0.129 \\ 0.057 & 0.011 & 0 & 0.035 \\ 0.002 & 0.001 & 0 & 0 \\ 0.128 & 0.035 & 0.001 & 0.125 \end{bmatrix}$	$\begin{bmatrix} 0.854 & 0.054 & 0.006 & 0.011 \\ 0.055 & 0.004 & 0 & 0 \\ 0.005 & 0.001 & 0 & 0 \\ 0.010 & 0.001 & 0 & 0.002 \end{bmatrix}$	$\begin{bmatrix} 0.936 & 0.028 & 0.004 & 0 \\ 0.028 & 0.002 & 0 & 0 \\ 0.004 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Table 5.2: Transition matrices showing the probability whether a vehicle is of a certain type when the preceding vehicle was a car, van, motorcycle, or lorry respectively.

vehicles have an interarrival time of at least several hunderd microseconds, but already a substantial fraction drives at an interarrival time of half a second. In the fast lane the Pearson type III distribution peaks slightly above one second, followed by a fast exponential decay. For the slow lane, the peak is less sharp and the decay much slower. This effect is even more pronounced for vehicles classified as trucks and lorries. In the slow lane this indicates the existence of relatively large gaps between heavy vehicles. Still, the gaps between the heavy vehicles are less than 10 seconds, which means that the trucks follow each other. Passenger cars are reluctant to fill those gaps due to the speed difference between the slow lane and the other lanes. The percentage of heavy vehicles has direct influence on the capacity of the road, as studied in chapter 7.



Figure 5.5: IAT-distributions both measured and generated. Measurements are from the A2 3-lane motorway between Woerden and Utrecht.

5.3.3 Drawing Speeds with the Speed Generator

The speed of newly entering vehicles is generated using an empirical procedure. This procedure is based on speed observations and the assumption that the speed of the following vehicle depends on that of the preceding vehicle [7]. There will be a variation in the measured speeds, which we model in principle with a uniform distribution U. There will be a little asymmetry in the distribution: the skewness is a function of the lane, current type and previous type. Furthermore, the variation ΔV will be less for vehicles close by (IAT < 3 seconds) than for vehicles at larger distances, as depicted in figure 5.6.

Figure 5.6 is a plot of the average speed difference $(|\Delta V|/V)$ between a vehicle and the preceding vehicle. The average differences can be as large as nearly 20%. In the slow lane the speed differences are more dependent on the IAT than in the fast lane.


Figure 5.6: The average speed difference $(|\Delta V|/V)$ between a vehicle and the preceding vehicle.

To summarize, the following model is used to estimate the speed of vehicle *j* in our model:

$$V(j) = V(i) + \Delta V(lane, type, IAT)U(skewness - 1, skewness)$$
(5.2)

where *U* is the uniform distribution and ΔV a look-up table. An example of the output of the Speed Generator and the correspondence with measurements is shown in figure 5.7. The Speed Generator can generate both symmetric (fast lane) and asymmetric speed distributions (slow lane).

Also note that for the Calibration and Validation Sets the speed distribution is intermittent; some measurement values do not occur. This is an artifact from the induction loop used, which refused to report some specific speeds. No information was available on the nature of this artifact. Unfortunately, the road pricing system was not designed for speed measurements, so no additional information could be gained from this system.

5.4 Hierarchical Aggregation

In this chapter we have explained the details of validation and calibration of the three subroutines of the Random Traffic Generator. A reoccurring statistical aspect in this validation and calibration effort was the asymmetric distribution of passage features over the traffic model. For instance, the passage of a heavy vehicle on the middle lane is already rare (probability < 2%), but not so rare



Figure 5.7: The speed distributions, both measurements and generated

that one cannot identify simple statistical estimates as an average speed or interarrival time. Yet, for a more detailed estimation of the shape of distribution one also need to estimate higher order moments, which in turn can only be derived from larger sets of measurements, which can only be acquired by longer acquisition times. When using longer acquisition times it is difficult to guarantee equal circumstances, such as traffic volume, fraction of heavy goods vehicles, weather conditions. This is a generic problem, which occurs when there are several layers of detail in a multi-model simulation. To solve this problem, the following solution has been found.

A hierarchy of distributions was created for estimating variables such as the interarrival times, the speed, and the speed as function of the interarrival time. For each variable, the passages with equivalent characteristics were grouped into a number of measurement sets. For the IAT and speed variables for instance 48 sets could be created (3 lanes x 4 preceding vehicle types x 4 vehicle types). This grouping could be done with highly, medium or modest selectivity. For each set, a minimum of elements was defined, empirically set to 32. When the set became too small, a set was created with a less selective characteristic, as indicated in the table 5.3. As a last resort, the set was used that

consisted of all vehicles on all lanes at all times during that period. This last resort was only needed for some additional measurements performed in the middle of the night. The result of this selection procedure for one example, the speed difference as function of the IAT is illustrated for four minimum set sizes in figure 5.8. Compare this figure with figure 5.6. The results for the larger minimum set size are smoother.

	Highly selective	Medium selective	Modest selective	No selection
IAT distribution	$lane, \ type(i)_{preceding, \ vehicle} \ type(j)_{vehicle}$	$lane, type(j)_{vehicle}$	$type(j)_{vehicle}$	all IAT measurements
speed distribution	$lane, \ type(i)_{preceding}, \ type(j)_{vehicle}$	$lane, type(j)_{vehicle}$	$type(j)_{vehicle}$	all speed measurements
speed difference distribution	lane, floor(IAT)	lane, IAT > floor(IAT)	lane, IAT > last(IAT)	all speed differences

Table 5.3: the hierarchy of selection of a set of equivalent passages for different distributions.

This is one detail of a rich set of analysis tools which was implemented in MAT-LAB. This set made it possible to analyze the measurements, to aggregate those measurements into sets with different characteristics and over a range of periods, and fit those measurements with the appropriate distribution functions. An interface to MATISSE databases was created, which allowed to store and retrieve analysis results in a structured way. The results of this analysis can be visualized, with a number of convenient plotting functions. Further a coupling with the ADSSIM environment was made, which allowed the generation of configuration files for the Random Traffic Generator from the current analysis results. Simulations with these configurations produced traffic pattern files, which could be read, analyzed and visualized in the same way as the measurement results. The coupling with the ADSSIM environment makes it possible to generate large quantities of passages under specific circumstances by using a certain measurement set as example. The selected measurement set is analyzed, a traffic model is estimated and stored in configuration files, which can be directly used by the Random Traffic Generator.



Figure 5.8: The speed difference as function of the IAT aggregated with four different set sizes. The results of the minimum set size of 8 are equal to the minimum set size of 16.

5.5 Conclusions

The Random Traffic Generator presented in this chapter performed well for flowing traffic at high densities at high speed. The calibration of such a Random Traffic Generator process at a microscopic level is a labor-intensive job, which requires good models, rich datasets and correct initial estimates of parameter settings [80]. In this chapter we have tried to document our effort on a high level, so that other researchers could use our experience and dataset⁴. In our view further development of random traffic generators is constrained by the availability of well-documented traffic datasets and benchmarks. A rich dataset should be accompanied by a rich set of analysis routines. With this analysis routines a research can extract model data from parts of the dataset depending on his current focus of research.

Our analysis was unique in the sense that we had two independent measuringsystems, which classified the vehicle types using completely independent features. Because the distributions of both the speed and the distance from the preceding vehicle are highly dependent on the vehicle type, good estimates of the two distributions could be made. Heterogeneous simulation (combinations of passenger cars and trucks) is only possible when the vehicle type can be reliable estimated. The dynamics in the slow lane show a large natural variety, depending on the number of trucks in this lane. A good description of the process is essential to be able to predict what is happening on the road.

⁴A. Visser, Calibration and validation data for the simulation model ADSSIM, online available , 2001. http://www.science.uva.nl/~arnoud/projects/TrafficLab/

Chapter 6

Modeling of Dedicated Short Range Communication

6.1 Introduction

¹ In an Automatic Debiting System (ADS) on the road network, it is essential that the exchange of information between roadside system (RSS) and the onboard unit (OBU) in a moving vehicle is reliable and fast. This is the task of the dedicated short-range communication system (DSRC), one of the subsystems of the ADS. Other subsystems are used for vehicle detection, co-ordination and license plate registration.

In this chapter, a hierarchical approach is introduced to model the reliability of the physical layer of the communication subsystem. Step by step more detail is added to the model of the subsystem. We will show the trade-off between reliability of the results and computational effort. As a result of this analysis, insight is gained into the parts of the model that are critical for defining the reliability of the system.

6.2 Modeling

In order to simulate a complete Automatic Debiting System (ADS) [43], we have to model the different subsystems [28]. In this chapter we are only inter-

¹Based on: A. Visser, H.H. Yakali, A.J. van der Wees, M. Oud, G.A. van der Spek, L.O. Hertzberger, "An hierarchical view on modelling the reliability of a DSRC-link for ETC applications", IEEE Transactions on Intelligent Transportation Systems Vol. 3: No. 2, June 2002.

ested in the communication subsystem: the link between the microwave antennas at a gantry above the road (RSS) and the small patch antenna (OBU) in moving vehicles. Via this link a certain fee that has to be paid for the passage is collected. When the vehicles are not equipped with an OBU, their license plate will be registered, which is a task of the other subsystems of the ADS (detection and registration). The other subsystems are out of the scope of this chapter.

A communication link for electronic payments has to be reliable. To prove the reliability of such a system, a detailed analysis of the occasional errors is needed. Large scale simulations are well suited for this job, but most computational effort is spent on calculations of the details of communications without errors. The aim of this chapter is to find the right level of detail needed for such an analysis, and to simulate that level of detail when needed, and to use less detail otherwise. This is an example of multi-level modeling, where the execution of several models at different levels of abstraction is facilitated.

To study the appropriate level of detail, a hierarchical approach is used for modeling the physical layer of the communication. The physical layer is one of the OSI-layers of communication. Higher (OSI-)layers are as important for the reliability as the physical layer, and quite some modeling effort has been put into those higher layers, but in this chapter we concentrate on the physical layer. For this layer, five models will be described with their implementation. Each lower level model contains more details, but is also computationally more expensive. Still, the details can not recklessly be ignored, because a correct estimate of the performance of the system can depend on these details. Only after a careful analysis of the performance estimate at the different levels, can a heuristic be applied, that indicates the appropriate model for each passage, which leads to a reasonable estimate with a reasonable computational effort.

The following five models are distinguished:

- **Transmitter Geometry model** is the highest model with the least detail. It provides simply a volume description where communication is possible, not possible or perhaps possible.
- **Transmitter Field model** provides the spatial distribution of the strength of the electromagnetic (EM) field of the transmitting antennas.
- **Single-Receiver model** provides the EM vector-field (including amplitude, phase and polarization) at the receiver of an OBU, which could be direction, phase and polarization dependent.
- **Single-Vehicle model** is an extension of the Single-Receiver model with bonnet reflections and windscreen influences.
- **Multiple-Vehicle model** is an extension of the Single-Vehicle model with reflections and disturbances from other vehicles surfaces.

In the following section, the different implementations of the five models will be discussed. Notice that the Multiple-Vehicle model takes in account interdependencies between the passages of two vehicles. As discussed in Chapter 4, this has consequences for the validation, because this at least doubles the parameters to be controlled during experimentation. Yet, this study shows that the influence of reflections cannot be ignored. Overall reflections enlarge the area of communication, but also enlarge the uncertainty in the localization of the OBU in the field.

The example used in this chapter is a communication system designed by Philips/Kista system. From this system a large set of measurements from 1990 is available [18]. The Philips/Kista system is not completely comparable with the commercial systems currently available, but has the benefit that it is extensively studied. The results of this study can be extended to predictions about the current commercial systems; the systems whose performance was under study in the "Rekening Rijden" project.

Especially for this study the parameters of the model under study are tuned in such a way that the communication performance is critical; the microwave power is just enough to allow the exchange the information for the electronic payment in most cases. This deliberately modification of the system is pessimistic, but not unrealistic. For this critical system the effect of the different hierarchical models can be directly seen as a difference of the number of unsuccessful transactions.

6.3 Implementation

6.3.1 Transmitter Geometry model

The 'communication zone' is the area where the transaction takes place for most of the passages. In this area the signals on both downlink and uplink are of such level that a reliable communication link is guaranteed. A message is transmitted over the communication link by a sender by bitwise modulation of the microwave signal. The receiver should be able to detect if the microwave signal is modulated or not, which is hampered by the always present noise. The bit error rate is a function of the signal-to-noise ratio, functions that are known for the different modulation schemes. For strong signals the bit error rate is effectively zero. The area with strong signals is defined as the 'communication zone'. Around this zone is an area where the success of the transaction depends on many parameters.

Based on this, we define 'grey' zones (where a more detailed modeling of the communication is needed) next to 'white' zones (where a high level modeling

of communication is sufficient). The 'white zone' is the area where strong microwave signals can be guaranteed. The microwave signals as function of the position are measured in so called footprints.



Figure 6.1: Definition of a VolumeTriangle

The volumes in the ADS configuration where communication is possible are implemented as so called VolumeTriangles, which have roughly the shape of the field transmitted by the antennas. The Philips/Kista system [18] is modeled with the following parameters for the VolumeTriangle (see figure 6.1) where good communication is guaranteed: the 'white zone'.

LateralOpeningAngle:	50.00;	degree
YRotation:	-69.75;	degree
LongitudinalOpeningAngle :	32.25;	degree
TransceiverHeight:	5.30;	meter

In figure 6.2 we can see how this shape coincides with the actual volume where good communications is possible. Figure 6.2 gives two cross-sections of this volume: a side view and a top-view. The thick bars in the figure are the measurements performed on the system under study in 1990. A thick bar indicates the largest uninterrupted interval where the signal level exceeds a threshold, so that reliable communication is guaranteed. Before and after those thick bars the signals level can be large enough to exchange messages, but this not guaranteed error-free.

The topology of the VolumeTriangle is chosen in such way that uninterrupted



Figure 6.2: Footprints of the VolumeTriangle and measured uninterrupted interval of signal levels above a certain threshold at 1.25 m and 2.65 m height. The projection of the position of the transceiver is indicated.

signal levels inside the volume can be guaranteed. The only exception is that at the far edges at low height, the location of the communication zone is shifted, although the length of the communication zone is not overestimated. Still, the VolumeTriangle is an reasonable estimate of the 'white zone', the area where the signal is so strong that reliable communication is guaranteed. For the Transmitter Geometry model, we assume that no communication is possible outside this 'white' zone. This strict assumption is relaxed in the next model.

6.3.2 Transmitter Field model

In the Transmitter Field model, a rough estimation of the microwave signal based on a standard mathematical model for the main lobe of the antenna field pattern [48] is made. The geometrical shape of the lines of equal power is more complex than the previously used VolumeTriangle. In this way a 'grey' zone can be defined; the area where the signal is strong enough to allow communication, although not completely error-free.

In the Transmitter Field model, it is assumed that the actual power received at a certain location is independent of the orientation of the OBU and the shape of the vehicle carrying it. These assumptions are quite reasonable. For instance, the only significant difference in signal level between this model and the lower Single Receiver model, which takes the orientation into account, is in the area directly under the gantry (X > -5m, at the right side of the figures 6.3.a-d)

Also a difference in signal level can be seen in the lateral direction at the far edges of the communication zone (|Y| > 3.5m, not illustrated).

In the Transmitter Field model, we will use the following simple formula to calculate direct path loss that takes into account losses due to distance, azimuth angle and elevation angle:

 $lossDirectPath(r, \phi, \theta) = lossDistance(r) + lossAzimuth(\phi) + lossElevation(\theta)$ (6.1)

where

$$lossDistance(r) = -20log(r)[dB]$$
$$lossAzimuth(\phi) = \frac{-6}{log(cos(22.5^{\circ}))}log(cos(\phi))[dB]$$
$$lossElevation(\theta) = 20log(\frac{sinc(K_ssin(\theta - \theta_0)(1 + cos(\theta - \theta_0)))}{2K_ssin(\theta - \theta_0)})[dB]$$

and $\theta_0 = 35^{\circ}$ and $K_s = 8.87$.

The formulas used for calculating the azimuth and elevation losses are standard mathematical models [48, page 280-285] for the main lobe of moderate and narrow antenna patterns, respectively. The parameters in the azimuth and elevation loss equations are chosen in accordance with the specification of the system under study [18, page 13].

In figure 6.3 the field obtained from this model (a-b) and the actual measurements (c-d) for different values of the lateral position Y are shown and they match quite well. Since this is a crude model, there are some differences between model and measurements. Notice for instance that there are no sidelobes at x = -2 and x = 0 meter. The measurements are taken from [18, page A1, A5 & A9]), for an OBU with a 'standard' orientation (elevation angle $\theta_{OBU} = 45^{\circ}$, azimuth angle $\phi_{OBU} = 0^{\circ}$). For the calculations no OBU orientation is taken into account.

In the images at the top row, two curves can be found. Only the upper curves (for 2.45 GHz) should be compared with the measurements. Validation issues will be discussed thoroughly in section 6.4. Here, both calculations and measurements are used to illustrate the impact of the modeled effects. The lower curves (for 5.8 GHz) represent the powers used in the simulations that generated the results of section 6.5.

In the Transmitter Field model, it is assumed that the actual power received at a certain location is independent of the orientation of the OBU and the shape of the vehicle carrying it. The rationale behind these assumptions can be seen when the measurements in figure 6.3 are compared with the measurements



Figure 6.3: Calculations and measurements of the received power for two values of Y. The measurements are taken from [4, page A.1 & A.9].

in figure 6.6. The only significant difference in signal level between the two models, is in the area directly under the gantry (X > -5m), and at the far edges of the communication zone (|Y| > 3.5m).

6.3.3 Single Receiver model

On this level, both the transmitter and the receiver antennas are modeled. The antennas are modeled as arrays of patch elements. Although each patch antenna element that emits (or receives) signals has a wide field pattern, by combining the fields of all patches that make up an antenna, narrow antenna patterns can be obtained.

The antennas at the gantry of the Road Side System (the RSS) contain eight patch elements (array of 4x2). The antenna of the antenna in the vehicles (the OBU) contains a single or a double-patch element. The next figure shows the side view of both antenna positions with their schematic directional patterns.

The far field model (Carver and Mink [20]) of a linearly polarized rectangular microstrip patch antenna operating in the TM_{10} mode at location $P(R, \theta, \phi)$ is given by the following expressions:

$$E_{\theta,patch}(R,\theta,\phi) = E_{main}(R,\theta,\phi)cos\phi$$



Figure 6.4: The single-receiver model for the single patch version of the system under study.

$$E_{\phi,patch}(R,\theta,\phi) = E_{main}(R,\theta,\phi)cos\theta sin\phi$$

$$E_{main}(R,\theta,\phi) = e^{-j(k_0R-\frac{\pi}{2})} \frac{V_0 k_0 a}{\pi R} cos((ktcos\theta))$$
$$sinc(k_0 \frac{a}{2} sin\theta sin\phi) cos(k_0 \frac{b}{2} sin\theta cos\phi)$$

where $k = k_0 \sqrt{\epsilon_r}$, $k_0 = 2\pi/\lambda_0$, ϵ_r is the dielectric constant and V_0 the voltage applied to the patch.

The expressions are formulated in the local spherical coordinate system illustrated in figure 6.5. E_{θ} and E_{ϕ} denote the field components along the vectors \hat{u}_{θ} and \hat{u}_{ϕ} at point *P*. The field component in the radial direction E_R is zero (in the far field).

In order to generate elliptically polarized waves, we assume that there exists an equivalent second antenna emitting with a 90° phase difference at the same location, and with its coordinate system rotated 90° about the z-axis. At broad-side the polarization will be circular, the ellipticity ratio will gradually deviate from one when moving away from broadside ($\theta = 0^\circ$).

The contribution of all the (eight) patches in the gantry antenna are summed to compute its field pattern at a certain point:

$$E_{\theta}(R,\theta,\phi) = \sum_{n=1}^{N} \sqrt{P_n} E_{\theta,n}(R,\theta,\phi) e^{i\alpha_{\theta,n}}$$



Figure 6.5: Geometry for far-field pattern of rectangular microstrip patch.

$$E_{\phi}(R,\theta,\phi) = \sum_{n=1}^{N} \sqrt{P_n} E_{\phi,n}(R,\theta,\phi) e^{i\alpha_{\phi,n}}$$

where P_n denotes the power level and α_n the relative phase of the n^{th} patch element. The field components of the gantry array antenna are scaled by the directional sensitivity of the OBU antenna and projected onto the z = 0 (patch) plane of the OBU coordinate system. The OBU of the system under study is only responsive to the left-hand circular polarized component in that plane.

With the aid of this detailed and computationally intensive model of the transmitting and the receiving antennas, the amplitude and the phase of the received field can be accurately computed.

Figure 6.6 shows the calculated and the measured received power for different angles of the OBU with a single patch antenna. When these plots are compared using the locations of the maxima and minima, it can be seen that the patterns are nicely reproduced.

6.3.4 Single Vehicle model

In the previous model, the vehicle that carries the OBU antenna is not taken into account. The measurements were performed by driving an empty frame with the OBU antenna around. Yet, details of the shape and material of the vehicle are also important for the communication performance. The windscreen



Figure 6.6: Calculations and measurements of the received power for two values of OBU orientation θ_{OBU} . The measurements are taken from [18, page A.13, A.1 & A.14].

and the bonnet are the two parts of the vehicle which have the strongest influence on the communication link.



Figure 6.7: Specular reflection is used in ray tracing

The windscreen influence is currently modeled as a constant attenuation. From the bonnet, reflections can be expected. Circularly polarized fields change the sign (and ellipticity ratio) of their polarization when they reflect off a metallic surface. Since the receiver antennas are sensitive to only one type of polarization, the interfering effect of reflections from the bonnet is limited. However, the effect is not completely absent due to two reasons: the field is not perfectly circularly polarized in all directions and each reflection path and the direct path have a different length and loss. In the Single Vehicle model, only the bonnet reflection is taken into account. A simple form of ray tracing is performed, based on specular reflections only (see figure 6.7).



Figure 6.8: Calculations and measurements of the received power at Y = 0 and $\theta_{OBU} = 45^{\circ}$ for an vehicle with and without a bonnet. The measurements are taken from [18, page A.16 & A.17].

The effect of reflection from the bonnet can be seen as a peek around 7.5 meter from the gantry in figure 6.8.b. The same peak is visible in the measurements in figure 6.8.d. This peak in the measurements is a little broader, due to additional peaks already present in figure 6.8.c. An explanation of these additional peaks is that the windscreen has also an effect on the phase, which makes additional reflections (from the ground) visible.

6.3.5 Multiple Vehicle model

In the Multiple Vehicle model the vehicles that surround the communicating vehicle are also taken into account. The surrounding vehicles can have three effects:

- The communication-link can be blocked.
- The communication-link can be disturbed by (multiple) reflections.
- The communication-link can be disturbed by diffraction (edge effect).

In ADSSIM, the first two effects are modeled. The blocking of the communication can be modeled by checking for occlusion. Reflections can be taken in account with the ray tracing technique described in section 6.3.4. Singlereflections do not have a strong impact on the received signal because the direction of the circular polarizion is reversed during reflection. A double-reflection can have a strong impact on the received signal. A circularly polarized field that is reflected twice on a metal surface has a strong co-polar component, which will interfere with the field received via the direct path. Reflections on more than two surfaces are not taken into account in ADSSIM. A realistic scenario for a double reflection is a beam that is first reflected by the sideplane of a lorry driving next to a passenger car, and then, bounces via the bonnet of the passenger car to the OBU. With the same ray tracing technique as section 6.3.4 the reflection via a sideplane and a bonnet is studied. This yielded the field patterns as shown in figure 6.9:



Figure 6.9: Calculation and measurement of the received power at Y = 0 and $\theta_{OBU} = 45^{\circ}$ for a vehicle with a bonnet driving next to a large lorry. The measurement is taken from [18, page A.25].

The effect of the multiple reflections as modeled in ADSSIM can be seen if one compares figure 6.8.b with figure 6.9.a. Instead of a single broad disturbance of the main lobe multiple small peaks are present due to multiple reflections. If one compares the measurements, the difference between figure 6.8.d and figure 6.9.b is less significantly, and mainly present in the splitting of the main and side-lobes. This means that there is a small discrepancy between the simulation and measurement, which is probably due to small differences in the ma

terial, size and position of the reflecting surfaces. Note that, the measurements were performed at a frequency of 2.45 GHz, so that the reflecting surfaces were rather small in terms of wavelength. At the frequency of 5.8 GHz the reflecting surfaces can contain twice as many wavelengths, which results in interference with a comparable amplitude but a higher frequency. More simulation results will be presented for the wavelength of 5.8 GHz in the next section (see figure 6.12).

6.4 Validation

In the previous section, calculations based on five models at 2.45 and 5.8 GHz are already compared with the detailed measurements performed on the Philips/Kista system operating at 2.45 GHz. The comparison is presented in the figures 6.3, 6.6, 6.8 and 6.9.

In this section, our simulation results at 5.8 Ghz are further validated by comparison with other simulation tools. First the field pattern of a linearly polarized patch element described in the Single Receiver model is compared with field pattern of a patch element defined in Personal Computer Aided Antenna Design program (PCAAD version 2.1²). Figure 6.10 shows the plots generated for both programs. The field patterns fit exactly.



Figure 6.10: Field patterns of linearly polarized square patch antenna obtained with both PcAAD v2.1 and our algorithm. In both cases, the single square pat ch had a length of 2.65 cm, substrate thickness of 0.265 cm, operating frequency of 5.8 GHz and dielectric constant of 1. (Inner curve: E-plane, outer curve: H-plane.)

Further comparisons have been made using the antenna analysis program EN-SEMBLE [39], which is capable of analyzing the effect of patch antenna feeding. For this comparison, a 4 (rows) by 2 (columns) array antenna with circularly

²D. M. Pozar, PCAAD - Personal Computer Aided Antenna Design for Windows, version 3.0, Antenna Design Associates, Inc., 1996.

polarized square patch elements is used. The amplitude and phase calculated by both the Single Receiver model and ENSEMBLE are presented in figure 6.11. A close inspection gave the conclusion that the Single Receiver model can calculate correctly the amplitude of the field. However, ENSEMBLE produced a different phase pattern. ENSEMBLE models not only the path antennas, but also the feeding lines to these patches. The delays that the microwave signals obtain in the feeding lines explain the phase difference.



Figure 6.11: Array antennas field patterns obtained with our *Single Receiver model* (left column) and ENSEMBLE (right column) scanning the angle θ from the z-axis. Both programs used a 4 by 2 array antenna with circularly polarized square patches with length of 2.65cm, substrate thickness of 0.265cm, operating frequency of 5.8 GHz and dielectric constant of 1. (Upper row: E(0°)-plane, lower row: H(90°)-plane. Solid line: amplitude, dashed line: phase.)

The ray-tracing algorithm used in the Single and Multiple Vehicle models was validated with another simulation program called RAPPORT (Radar signature Analysis and Prediction by Physical Optics and Ray Tracing) [34]. This program combines ray tracing with Physical Optics. In the Physical Optics approach, a ray is not reflected in a single direction, but has a directivity pattern



due to scattering. Also the OBU patch antenna is modeled with directional sensitivity. The results obtained by both models are shown in figure 6.12.

Figure 6.12: The received power at y = 0, $\theta_{OBU} = 35^{\circ}$ and $\phi_{OBU} = 0^{\circ}$ for a vehicle with a bonnet and an OBU driving next to a large lorry calculated by both packages. In order to obtain the results for the worst case scenario, an OBU antenna with a uniform sensitivity in front of the patch plane is used. Plots show the combined effect of the direct path, the path reflected at both the sideplane of the lorry and the bonnet, and the path with a single reflection at the bonnet.

As can be seen from the figure they compare nicely. The difference is quite subtle and can for instance be seen around the top at X = -5m. Our model shows a sharp end of the interference pattern, because the specular reflection point falls off the bonnet. In contrast, the change in the interference pattern in RAPPORT is smooth, due to the diffused reflection. This difference becomes more pronounced for smaller bonnets.

6.5 Results

To distinguish the answers of the different models, we compare the outcome of the transactions. The outcome can be categorized in three classes: no transaction, an incomplete transaction and a complete transaction. Remember that the power in the simulated system was reduced to a level that no or incomplete transactions occurred. For each hierarchical model, we will compare the simulated transaction outcomes and the computational price paid for the increasing level of detail.

	No	Incomplete	Complete	Running
	transaction	transaction	transaction	time
Transmitter Geometry model	41.68%	0.00%	58.32%	0h03
Transmitter Field model	34.53%	0.01%	65.46%	0h08
Single-Receiver model	0.00%	2.46%	97.54%	3h21
Single Vehicle model	0.03%	8.73%	91.24%	4h45
Multiple Vehicle model	0.16%	9.71%	90.13%	5h33

Table 6.1: Communication results for the different hierarchical levels, without contribution of any other level.

In the table 6.1 the transaction results and computational costs of the different models are summarized. All calculations where performed on Sparc Ultra 10 workstations with a 300 MHz processor. The simulation results are based on 10.000 passages of vehicles, all equipped with an OBU.

One can see that the detailed Multiple Vehicle model needs more than hundred times the running time of the Transmitter Geometry model. The Single Receiver model has better results compared to both Transmitter models for two reasons. The first reason is related the RSS antenna pattern under the gantry. The Transmitter Field model contains only the main lobe. Yet, for many transactions, a message has to be exchanged when the OBU is between the main lobe and the first side lobe. For the Single Receiver model, the message is exchanged in the sidelobe, after a number of retries. In the Transmitter Field model there is no sidelobe, so the transaction is not completed. Detailed analysis showed that the side lobe is involved in 13% of the transactions. The second reason is that the Transmitter Field model rejects all communication when the Bit Error Rate is worse than 10^{-6} . In practice, messages can still be exchanged for rates behond 10^{-6} , although retries become likely. This effect contributes to successful transactions for 36% of the passages.

The percentage of completed transactions predicted by both Vehicle models is lower than that of the Single Receiver model. In this model, the receiver is a free-floating device in the air; windscreen and reflections are not taken into account. The influence of the reflections on the received signal can be both positive and negative. For instance, the variations in the signal level obtained using the Single Vehicle model are between -3.3 and +12.8 dB, and between -41.4 and +35.8 dB for the Multiple Vehicle model, compared to the Single Receiver model. On the average, the reflection increased the power level in the receiver. Notice that the reflections make it more difficult to accurately localize the OBU in the field, which has consequences on higher levels of the ADS.

The fact that the Transmitter Field model is more stringent makes it possible to use this model as a filter for a more detailed model. This filtering works as follows. When the Transmitter Field model predicts a successful transaction, this result is used. When the Transmitter Field model predicts an incomplete or unsuccessful transaction, a patch antenna model like the Multiple Vehicle model is used for a precise simulation of the transaction. With this filtering, not more than 20% of the running time is needed, without a significant loss of accuracy (see table 6.2).

Table 6.2: Communication results for the different hierarchical levels, with filtering by

 the *Transmitter Field model*.

	No	Incomplete	Complete	Running
	transaction	transaction	transaction	time
Single-Receiver model	0.01%	1.98%	98.01%	0h33
Single Vehicle model	0.01%	8.27%	91.72%	0h45
Multiple Vehicle model	0.01%	8.77%	91.22%	1h04

Although the multi-level approach, proposed in this study, requires more effort to create multiple models, the advantage is that simulations are performed in less time. Furthermore, insight is gained into which assumptions have the greatest influence on the performance of the communication link. Detailed calculations, such as for instance the ray tracing required to estimate the effects of reflections, are only performed when from the simple algebraic formula 6.1 it is not obvious that the transaction can be completed.

6.6 Conclusions

In this chapter, a hierarchical approach to model the reliability of the communication link is introduced and worked out. We have shown that the physical layer of the communication link can be modeled in five hierarchical levels. At each level, more detail is added to the model. More detail means more computational effort, and more effort to find (and calibrate) the required parameters. Yet, these details can have major impact on the performance of the communication link. In order to bring out the effect of our hierarchical model clearly, a system with a weak communication link was chosen.

Our study has shown that, although there are five logical hierarchical communication reliability models, there is no need to use all of them sequentially. We have shown that only two of them are sufficient to obtain a detailed analysis:

the Transmitter Field model (the spatial distribution of the strength of the electromagnetic field of the transmitting antennas) serves as a deciding factor whether more detail is necessary, or not

the Multiple-Vehicle model (the reflected and disturbed electromagnetic vector-field as received by the OBU) to perform this detailed analysis.

This reduced the computation time to 20% without a significant loss of accuracy. This approach makes it possible to simulate the performance of the short-range communication link for millions of vehicle passages, in a comparatively short time. This is needed to demonstrate that short-range communication is a reliable way of data exchange. When the reliability requirements are high, as for the "Rekening Rijden" project in the Netherlands [95], large number of passages are needed for accurate estimation of the occurrence of failure events with a low probability. The test along the A12 was not designed to provide evidence about the reliability of the communication subsystem, because only a few OBUs were installed. Simulation can give an estimate about the reliability, based on models validated on small-scale experiments.

This case study is a good example of what is possible with multi-level simulation. Essential for this approach was the possibility to convert the outcomes of the different models back and forth. The common ground was the field amplitude on a 3D volume. The multi-level approach was supported by the available measurement set, where systematically the effect of all influences was determined.

Chapter 7

An Architecture for a Virtual Traffic Laboratory

7.1 Introduction

¹ Our ability to regulate and manage the traffic on our road infrastructure, essential for the economic welfare of a country, relies on an accurate understanding of the dynamics of such system. Studies in Germany have shown very complex structures in the traffic flow [52; 68]. This state is called the synchronized state, which has to be distinguished from a free flowing and congested state.

The difficulty to understand the dynamics of a traffic system originates from the inherent difficulty to relate the observed time patterns in speed and density to the underlying drivers' behaviors, and the changes therein as function of the circumstances and driver motivation [49; 55].

Simulations play an essential role in evaluating different aspects of the dynamics of traffic systems. As in most application areas, the available computing power is the determining factor with respect to the level of detail that can be simulated [67] and, consequently, lack of it leads to more abstract models [85]. To be able to afford more detailed situations, we looked how we could use the resources provided by for instance Condor [84] or the Grid [35].

¹Based on: A. Visser, J.G. Zoetebier, H.H. Yakali, L.O. Hertzberger "The measurement architecture of the Virtual Traffic Laboratory", Proceedings of the International Conference on Computational Science (ICCS 2004) Part I, pp. 515-519, in series Lecture Notes in Computer Science, vol. 3036, Springer Verlag, Heidelberg, June 2004. ISBN 3-540-22114-X.

Simulation and real world experimentation both generate huge amounts of data. For a domain scientist the seamless integration of the information from simulated data and measurements would be very valuable, the so called datadriven approach (see for instance [29]). The computer scientists who are part of the Grid community have put a tremendous effort into giving domain scientists smooth access to computational, storage and visualization resources; the so called generic middle-ware on top of the grid technology. Yet, smooth access to experimental resources requires additional interaction, monitoring and control capabilities.

Our research group participated in the Grid-based Virtual Laboratory Amsterdam (VLAM-G) [3]. VLAM-G had as goal to hide resource access details from scientific users, and to allow scientific programmers to build scientific portals. These portals give access to the user interfaces for scientific studies: combinations of information gathering, processing, visualization, interpretation and documentation. Typical applications can be found in Astronomy, Earth Observation and High Energy Physics, Medical Diagnosis and Imaging, Food- and Bio-Informatics, as bundled in the 'Virtual Laboratory for e-science' [56].

In this chapter we show our experience with building our Virtual Traffic Laboratory as a data driven experimentation environment. This experience can be used as input for the future development of the Virtual Laboratory on other application domains.

VLAM-G Architecture

The domain scientist is the person that actually is performing the studies. The domain scientist could for instance be an astronomer or a biologist. In this chapter the domain scientist would be a traffic engineer. In a study often the same steps are repeated, as for instance testing a hypothesis on a certain dataset. Some steps can be quite time consuming, so the domain scientist can log out from this study, prepare another study, and come back to inspect the intermediate results and perform another step of the study.

So, when the domain scientist starts working with VLAM-G, there is support in the form of the information management system VIMCO [51] and the Run Time System RTS [16], as illustrated in figure 7.1. The information system VIMCO archives study, module and experiment descriptions, together with the application specific databases. The RTS takes care of scheduling, instantiating and monitoring the computational modules of an experiment. In doing so, it makes extensive use of the Globus toolkit, the actual process oriented standard in Grid computing.



Figure 7.1: The different systems for a study with VLAM-G

Front-End

The domain scientist can use the front end that is optimal to do an experiment in its domain. For complex system engineering, as traffic systems, we favor the MATLAB environment. So, we have coupled a prototype of the RTS [38] with the MATLAB environment. Here the RTS is used to control for the heavy computational tasks, because the RTS can distribute these tasks to a large number of computational nodes available via the Grid. The MATLAB environment is used for analysis and visualization of the results.

To be able to demonstrate the possibilities of MATLAB as front-end, we have implemented a gateway routine, which allows the user to load VLAM-G modules, couple them, configure them and start them. We have hidden our gateway routine inside a user-defined block of SIMULINK. SIMULINK is an interactive, graphical, tool for modeling, simulating, and analyzing dynamic systems.

Further we used the Condor system to start up a cluster of jobs on several resources, including the Globus-resources at our site. The SIMULINK system was used to monitor the progress of the jobs, by monitoring the database where the results of the analysis and simulation are stored.

SIMULINK is a part of the MATLAB suite. It has an extensive library of predefined blocks, which perform operations on their input and generate output. In that sense they are comparable with the modules of VLAM-G. The difference is that these operations are performed as threads of the current MATLAB process, on the current machine, while at VLAM-G the modules are processes on remote resources. The VLAM-G concept of generic re-usable modules is perfect for the initial processing, analyzing and cleaning of the data, while the library of functions that SIMULINK has to offer is perfect to perform some final filtering of the results before they are visualized to the domain scientist.

This method of initial processing of the experimental data on the Grid and the final filtering the results on the workstation of the scientist is more widely applicable than the Virtual Traffic Laboratory. Processing of experimental data on the Grid can benefit from the reuse of generic processing modules, in cooperation with application specific modules. The final filtering will not only be application specific, but will also be specific to current study of the domain scientist.

7.2 The laboratory

As described in chapter 5 several aggregates of vehicle measurements have to be made to characterize the traffic state. Important aggregate measurements are for instance average speed, average flow, average density, headway distribution and speed difference distribution. In practice the measurements were collected along a Dutch road and stored in a database. The raw data was further processed in several steps, partly manually, partly automated. The result of each processing step was stored in another database. To facilitate the processing of the measurements over time the architecture given in figure 7.2 was designed. In this figure the information flow between the first, second and third databases is illustrated. The first database is called 'DCP', the second 'A12' and the third 'a12meta'. The processes between the databases are called respectively *Readin* and *CreateA12meta*. In the 'a12meta' aggregated measurements of measured traffic are collected. The results of a simulation can be analyzed with the same methods as real measurements, and stored in the database via a process AnalyzeSimulation. The details of these databases and processes are worked out in the remainder of this section.

Along the Dutch multilane road A12 there was a relational database from SYBASE that collected the measurements from two independent measurement systems. One system was based on inductive loops in the road, the other on an optical system on a gantry above the road. Although both were quality systems, some discrepancies occur between measurements due to different physical principles. Aside from the two automated systems video recordings were also available. These video recordings were used manually to inspect the situation when discrepancies between the measurements occurred.

After this validation process, the measurements were converted to an object



Figure 7.2: The measurement analysis architecture

oriented database from MATISSE. This database was used to verify the quality of the measurement systems themselves. While the manual validation process was used to get the overall statistics of errors in the measurements, the object oriented database was used to analyze the reasons behind the measurement errors. Several hypotheses of underlying failure processes were raised, such as for instance the characteristics of trailers that had higher chances to be characterized as an independent passage. After the manual validation, in total 154,748 passages were available in the 'A12' database.

The validated measurements were not collected continuously, but at selected periods. In principal these periods were concentrated during the morning rush. When rare circumstances occurred (for instance snow) additional measurements were collected. Because the measurements were manually validated, one was very selective in the number of periods. Those periods could be combined into sets with equivalent circumstances. Different criteria could be used to create such set, for instance 'Flowing' versus 'Congestion', 'High-Density' versus 'Low-Density', etc, etc. When the selection criteria were applied more stringently, the actual shape of the probability distributions of the aggregate measurements is revealed. The aggregate measurements can not longer be characterized by the average and standard deviation of the Gaussian distribution, but additional shape parameters have to be taken into account. This requires that the measurements are fitted to non-Gaussian probability density functions. The estimation of all parameters of all distributions for all lanes and periods takes 20 minutes on an UltraSPARC-II workstation, which is too long to wait. This makes it worthwhile to store the results in a database. We decided to use an 'a12meta' database, separate from the 'A12' database. In this way the 'A12' database can be used in read-only fashion, which is positive from both a security and transparency point of view. The 'a12meta' database contains a completely different information model, based on aggregates over periods, and not longer information about separate passages, which is the information model of the 'A12' database.

7.3 Analysis results

The architecture described in section 7.2 is used for several studies. One of the research questions was how realistically the simulator would simulate the traffic for larger fractions of lorries. Lorries occupy more space than passenger cars, and a large fraction of lorries decreases the road capacity. The parameters of the traffic simulator are scaled relative to a reference volume (intensity over capacity rate) I/C^* . A decrease of the capacity is interpreted as an increase of the relative volume, with the corresponding effect in the parameters of the traffic simulator. This analysis investigated if this scaling was appropriate for large fractions of lorries. The results of one analysis is given in figure 7.3, where the average speed is given as a function of the flow (as percentage of the maximum flow) and the fraction of lorries (as percentage of the number of passages).



Figure 7.3: The average speed as function of the flow and the fraction of lorries (left measured, right simulated)

The average speed is indicated with a color code, red (top of the bar) indicates high speeds, blue (bottom of the bar) indicates low speeds. Each point in the left graph indicates an aggregate over longer period (30-60 minutes), which are typically equivalent with a few thousand measured passages. As can be seen, the speeds are high as long as the volume is below 50% and the percentage of heavy traffic is below 20%. Higher percentages of heavy traffic can be seen for lower volumes. Further the measurements above volumes of 70% show a variety of colors, indicating that for these high volumes several types of traffic exist, each with their own average speed. At these high volumes the transitions between traffic types as 'Flowing', 'Congested' and synchronized traffic occur.

As can be seen at the left side of figure 7.3, the measurement periods are not homogeneously distributed over the spectrum. The majority of measurements

have for instance a fraction of lorries between 10% and 30%. For this study we like to initiate simulations with higher volume / fraction combination than actual measured. To do this, we select a certain volume (say 30%) and look how many measurements periods are available for increasing fractions of lorries. A number of measurement periods with approximately the same volume, fraction and average speed are combined into a small subset. For these subsets the probability functions of the combined measurements could be translated into the parameters of a microscopic traffic simulator, ADSSIM [85], which could be used to generate simulated data.

The characteristics of the simulated data were aggregated in the same way as the real data, and the resulting dynamics were compared to the original dynamics, to see if the model was complete (see figure 7.3). As one can see, the simulated points (each representing 5000 passages) are more homogeneously spread over the spectrum because one can ask for certain combinations of parameters. Yet, the results are less trustworthy when one has to extrapolate far (more than a few percent) from actual measured parameter-combinations space. For instance, this is illustrated by the fact that the average speed is unexpectedly high for heavy goods vehicles percentages above 30%. This is due to the fact that high percentage of heavy goods vehicles was only seen for low volumes, before 6:00 AM. At these low volumes most traffic can be found in the slow lane. For this slow lane reliable statistics can be gathered, and these lane statistics dominate the statistics for the road. For these low volumes the fast lane is only sporadically used (with high speeds). The statistics for the fast lane are completely different from the statistics for the road as a whole. When the flow increases, the fast lane is used more often; this gives this lane more influence on the statistics of the road. The high average speed, recorded for a sporadic passage at low volume, is now extrapolated to all passages on the fast lane for higher volumes. When the volume is high enough, the fast lane gets 1/3 of all passages and the speed-characteristics of this lane are directly visible in the speed-characteristics of whole road.

7.4 Implementation

To perform this analysis an interface was build between MATLAB and the databases. Above this interface several MATLAB-functions were written that performed specific operations on the databases. A full analysis of a measurement period required a combination of many operations on the databases. For instance, the *CreateA12meta* and *CreateStochasticModel* were MATLAB-functions with more than 1000 lines of code. We converted those complex MATLAB-functions to standalone programs, which made it possible to run those functions in the background with the aid of the Condor software [84]. The latest

versions of this software even make it possible to add Grid-resources to the pool with the glide-in technique [36]. With the glide-in technique Condor daemons are spawned on remote machines, accessible as Grid resources. As long as the Condor daemons continue to run on those machines, the remote machines appear to be part of our Condor pool. This makes it possible to extend our Condor pool outside the domain of our local organization. Normally the usage of machines outside the local domain would require an additional effort to move data to and from the other domain (facilitated by the software of the Grid), but in our approach this effort was not needed. In our approach the measurements and the analysis results are both stored in a database, which makes them easily accessible from outside the local domain. This made it possible to separate the number crunching of the analysis from the visualization of the analysis results. The visualization is performed on the workstation of the scientist, by starting at the MATLAB command line a number of daemons, implemented in SIMULINK. Each daemon constantly monitors a number of variables in the 'a12meta' database and updates a diagram of those variables when new analysis results are ready.

7.5 Discussion

This traffic application is a good benchmark for the virtual laboratory, because of the complexity of both the measurement analysis and the traffic flow model. It is difficult to compare models from different domains, but we take as reference the 150 variables in the standard application in molecular dynamics [5] in the UNICORE environment [33]. UNICORE is a virtual laboratory environment, which could be compared with VLAM-G. The traffic flow model of ADSSIM has 585 variables that specify a specific traffic condition. Only looking at the number of variables, the traffic flow model is not less complex than a molecular dynamics model.

Another issue is the calibration. With so many variables model calibration requires extensive work. To be able to calibrate such a traffic model for a certain condition, the domain scientist needs to be able to select characteristic subsets from the bulk of measurements, and a flexible way of analyzing the combination of those subsets. To do this, the domain scientist should be able visualize the dynamics of the aggregates of those subsets in different ways. It is no problem that it takes some time to generate aggregates, because there are always aspects that need the domain scientist's attention. In an ideal analysis environment the domain scientist should be able to interactively analyses previous results with the possibility of inspecting the results of automated analysis of previous results, without redoing the work, is possible by storing the results in a database.

While processes in the background fill in the missing data points, the domain scientist can start the visualization of other dependencies, till an unexpected pattern appears. This unexpected pattern will be manifest for a number of data points, and it is the task of the domain scientist to identify what the common dependency of those data points is. Often multiple hypotheses about a common dependency are possible. To distinguish these hypotheses, the selection of measurements and simulations has to be done again. At that moment other subsets in the measurements can be distinguished. Each subset corresponds with a hypothesis and an expected pattern that would become visible when the measurements are analyzed. This means that new analysis should be started up in the background.

To prevent an unmanageable number of background process (and corresponding visualizations), the domain scientist could decide to stop previous analysis processes when new analysis processes are started up. In practice this is not the decision, because the analysis of the previous characteristic dependency is often quite far developed before a unexpected pattern appears. The analysis of the remaining measurements will cost relatively a small amount of time, and gives the possibility of verifying if the unexpected pattern is still visible for the complete subset.

The key of our approach is that we do not see the domain scientist as a user that repeats the same analysis repeatedly on the different datasets, but is an expert analyzing the problem from different viewpoints, trying to find new dependencies in the data. These viewpoints are not known beforehand, and slowly shifting. We like to decouple the user interaction from the large scale computation, by moving the computations to the background. The MATLAB environment allows full control of a dataset, and facilitates different ways of searching, fitting and displaying dependencies. The interactive MATLAB environment allows the domain scientist to discover, create and follow hypotheses, while the evidence for those hypotheses is collected in the background. At the moment an interesting dependency is found, additional data points can be generated with an interface to a high throughput system like Condor. The results are automatically displayed by monitoring the databases with meta-data via SIMULINK.

This approach differs from the approach of for instance [29], where MATLAB is seen as inappropriate due to license policies and speed issues. By using high throughput systems in the background speed is no longer an issue. The MATLAB license is used on the domain scientist's workstation, where a rich set of graphics and statistical tools is made directly available for this money. This allows the domain scientist to concentrate on the interactive analysis of the measurements, without the need to construct graphical or statistical func-

tionality from scratch. Yet, the usage of MATLAB at the front-end does not exclude the usage of other (open-source) packages in the background. In our view the programming effort should concentrate on open source software of often used measurement analysis algorithms, and optimize these algorithms into modules that can be executed in the background, while MATLAB is used for direct analysis and visualization.

This experience of an interactive domain scientist was gained for the traffic application, but is probably far more generic. This methodology of constantly spawning of new pattern analysis processes in the background will become more established when additional computational powers turn out to be as common as the Grid community promises.

7.6 Conclusions

In this chapter we have introduced an architecture for an environment to perform an experiment for the Virtual Traffic Laboratory. To aid the domain scientist, analysis results are stored in databases with aggregated data. This aggregated data can be seen as intermediate results, ready for a final analysis. This aggregated data allows to repeatedly display the results from different viewpoints, where the domain scientist does not have to worry that too rigorous filtering will force him to do the aggregation again. This gives the possibility to interactively analyze results where each point needed 20 minutes to be generated.

New analyses can be started on different parameter-combinations in the background on remote machines. Thes analyses can be performed seamlessly on both real data and simulated data. New data can be automatically displayed by adding monitors to the databases with aggregated data.

Chapter 8

Discussion

8.1 Introduction

When we combine computer and real-world experiments intimately on multiple levels, several challenges have to be faced. These challenges were introduced in chapter 2. In this chapter we compare our progress with initiatives at other research institutes.

8.2 Data fusion and uncertainty propagation

By stressing a system away from a stable situation, more and more information is needed to describe the precise status of the system. Stressed sufficiently, the system can go through a phase transition to another stable system. To describe the system during the phase transition not only the information needed to describe the situation can explode, but also the computational costs explode when the number of clauses and variables are balanced for NP-complete problems [66]. The solution can be found in the concept of backdoors [96], as introduced by Ryan Williams, which separates the modeling of a system in two steps: first finding the set of variables that are really independent, followed by solving the instantiation of auxiliary variables that are needed for the complete system encodings. This is equivalent to finding the variables with maximal descriptive power in System Theory [54]. Nonetheless, at the moment that a system is stressed to a phase transition, traditional statistics can no longer be used, because the transformations from the underlying dynamics to the macroscopic probabilities become time and space-dependent. Also the opposite is true [69]; the transformation from macroscopic variables to microscopic dynamics also becomes time and space-dependent. In terms of System Theory, space and time are no longer supports. The system temporarily breaks up into regions with their own characteristics, heavily interacting with neighboring regions in space and time. This description is comparable with the description of synchronized traffic in chapter 5.

8.3 Integration approach

The goal of a Modeling and Simulation effort is to gain deep knowledge; to be able to predict the behavior of the system under a variety of circumstances. To gain this knowledge, a flexible environment is needed, where hypotheses about dependencies between characteristics and system behavior can be checked. The Virtual Traffic Laboratory is such a flexible environment where dependencies could be checked. The Virtual Traffic Laboratory can be called a Problem Solving Environment [92]. Our environment has a tightly-integrated visualization [99], computation and analysis system. With our automatic update mechanism, the graphs in MATLAB always show the latest result. A slight modification of the graph drawing procedure, for instance the usage of a different filtering algorithm, and a new view on the data is generated. The procedures inside MATLAB are stored as scripts, which makes sharing of algorithms (on source-code level) very easy and transparent. The transparency gives full power to the scientist, and no further assistance is needed.

The integration approach proposed in this thesis, uses as a basis for its user interface MATLAB, equal to the approaches in Geodise [32] and Netsolve [21]. Both projects have made the connection to Globus and Condor [8; 97]. Yet, we were one of the first to use the module description language of Simulink as interface to Globus or Condor. Actor-oriented programs like Ptolemy [17] and MILAN [58] use the same dataflow model as Simulink, but use their own language and had originally no interface to a cluster or grid. The Ptolemy environment not only works in the dataflow domain, but also in many others, as for instance continuous time and discrete events [17]. The Geodise and Netsolve projects made the connection to both the execution management part and the data management part of the Grid. In our approach, only a connection to the execution management part [50] with generic data communication channels (partly build on GridFTP [4]).

Nimrod/G [1] is a tool above the middle layer of the grid specialized in parameter sweeps, which has an overlap with our interests. Nimrod/G is fed with plan files, which declare in the first place the parameters and their ranges, and in the second place the commands which have to be executed iteratively.

A comparable file can be created in Matlab, where parameters can easily be initiated as arrays. Nimrod/G stores the results of the jobs in a database, but this database is mainly used to maintain the relation between the parameter instantiations and the location of the output files. The database can be queried about the generated results, but these queries will simple in comparison with the approach advocated here.

8.4 Measurement representation

The ultimate measurement representation would be inside a living document [83], a publication which contains experimental results that could be reconstructed by a single user click. The experimental results are coupled to the workflow that created them, and this instantiation of the workflow could be used as an outline for other scientist to recreate the results with slightly different parameters. To recreate an experiment, one needs to archive the context of the experiment; this can be performed by storing metadata at various levels. A part of this metadata is information needed for the computational context, a part of this information is about the information consumed and produced. This vision is worked out in several provenance initiatives in e-Science (see [65] for an overview). From the grid point of view archiving the relations between the processes is important [65], from the database point of view the relations between the 'source' data and the 'derived' data are very important [19].

In e-Science, several derived databases can exist with meta-data originating from the same experimental data set. The added value of the multiple derived databases is in the corrections and annotations made by experts. The chain of databases described in Chapter 7 is a good example of multiple derived databases. The expert interaction can result in removal of artifacts from the databases. For this approach to be successful, it has to be transparent who made which modification where. Transparency is important in the vision of Ruth [78], who proposes to automatically register as much information from the measurement devices as possible with an experimental data-set. This data set is signed by the scientist who created the set, and if other scientists use, filter, enrich or aggregate the data set they should also sign. Such a history accompanying a data set would help the scientific process tremendously because artifacts in the data could be traced back to their origin, and would not propagate and contaminate the whole data set. In [78] it was suggested to couple a reputation system to the authentication of collaborative data sets, to prevent fraud and mistakes. In my opinion, such a reputation system would work counterproductively, because many artifacts only reflect the focus of the responsible scientist at the moment of the measurement or the post-analysis. That the dataset would have been far more valuable with a different choice in

samples or parameters in the eyes of later scientist, should have no effect on the reputation of earlier scientists. If they do not agree with the work of earlier scientist, they can always go back to the original data and start the process of correction and annotation from the beginning. Validation and correction in an experimental environment is a constant process, and with the availability of public databases with experimental data researchers can share their results and can make fast progress as a community.

8.5 Multi-level modeling

For multi-level modeling, we have seen that execution of models at different levels of abstraction can be facilitated. This can be in the form of simulation of individual subsystems in isolation as well as replacing detailed behavior with computational more efficient high level behavior, as demonstrated in chapter 6 and for instance also in the Milan project [58]. Yet, abstraction into a hierarchy of modules is for most engineering problems not enough. Different modules can be modeled with different formalisms, such as continuous-time, discreteevent or finite-state machines. Subsystems tend to have their natural and historical way of modeling, and although transformations between formalisms are possible, a way to combine modules representing different formalisms is to be preferred. Inside the Ptolomy project [59] the actor model is developed, which include so called directors which define the formalism at that hierarchical level. When a module is transparent, the same formalism as defined for the current level applies to the module. When a module is opaque, a different director with a different mechanism is active inside this module. This allows one to make models of systems on many levels and with many formalisms. For the work of this thesis we made five different models with one single formalism: the event-based approach.

The simulation of one such multi-level model can easily consume many resources, and it would be advisable to allow also looser couplings of modules, as demonstrated in many Grid-projects (for instance [93]) and distributed Modeling and Simulation environments [22]. A loose coupling of modules is possible with well-defined, meaningful and understandable interfaces between the components. Each module should demonstrate a consistent behavior, which allows users to build a cognitive model of the module and to predict its behavior. This module should also demonstrate consistent behavior when one ascends or descends a conceptual level. Each level should have its own resolution, and one should be able to understand the relationships with higher and lower-resolution variables. In [62], Reynolds advocates an approach to couple models at multiple levels to entities that provide multiple interfaces (both to high and low resolution models) and maintain the consistency explicitly in the
module. This module should incorporate the transitions from high to low and their inverse, and the logic when to apply them. Otherwise, it is extremely difficult to achieve consistency over cross-level interactions, when the models have not been designed together [26]. For the work of this thesis we have not split up the models in modules that could be separately simulated. The simulation of the system was a single program with one of the models. A program with another model could be easily made, when different event handlers are linked that define a different response of the system to stimuli.

8.6 Synopsis

With growing Modeling and Simulation power, it is very tempting to make the simulation as 'real' as possible, even when those details are not relevant for the questions at hand. The additional details are used to gain confidence in the results. In practice this effect was also visible inside the "Rekening Rijden" project. For instance, at the moment that it was showed that details such as multiple reflections could be simulated quite accurately in our environment, most people were satisfied with the results, and did not have the need to include the detailed calculations in their simulation. From the scientist's point of view this is a perfect paradigm, to simplify the system after validating whether the simplifications affect the study's conclusion. And also from the computational scientist's point of view this is a perfect paradigm, not because a simplified model takes less time to run. Finally a simplified model is easier to comprehend, analyze and validate.

Chapter 9

Conclusion

9.1 Overview

This dissertation describes the measurement driven simulation approach used to evaluate the Dutch "Rekening Rijden" project. This Dutch project had as goal to know the limits of the technique used in such a complex engineering system as early as possible in its design phase. The difficulty was to find the appropriate level to model the subsystems so that their behavior could be validated.

On the one hand a high level model of the subsystems is needed with a limited number of parameters which can easily be validated (a low descriptive complexity), on the other hand the inherent uncertainty about the response of a physical subsystem has to be incorporated (a small information divergence), which can lead to very low level models.

A solution to this dilemma is to create multiple-level models, and to compare the output of the response of model level relative to each other. For the communication subsystem, a critical part of the design, simulations at five different levels of detail were demonstrated, and the resulting behaviors compared. Although the generation of models at multiple levels is an extra effort of the designer, the benefit is that insight is gained on the appropriate level of modeling of the subsystem.

To be able to perform such a multi-level comparison, an environment was created where modeling and simulation were used to validate the descriptive power of the models, while a comparative analysis of measurement and simulation results are used to verify the difference in behavior between reality and the model.

9.2 Lessons learnt

9.2.1 Applicability of Complex System theory

The experience with the application of Complex System theory was two folded. Firstly, it was to my surprise how many concepts of Cybernetics were applied regularly in science & technology. Cybernetics is currently not longer seen as a main stream of science, but its former contribution is unfortunately disregarded. Secondly, it was surprisingly that the concepts of System Theory are not applied more in science & technology. The formal way of simplification of system models is an objective way to look at the description of a system.

Both Cybernetics and System Theory are focused on finding correlations between time series of variable measurements. While this is perfect to indicate that there exists a correlation between variables, the nature of this relation is hard to grab. Sometimes the relation can be described with a number of logic rules, sometimes the relation can be described with a number of differential equations. At the moment that engineers are given the full access to a programming language like C, as we did in the "Rekening Rijden" project, the relations are described with more elaborate than a number of rules and equations. A part of the complexity to describe the relation was not only in the variety in the values of the variables, but also to model the variety in the time that the values became available. System Theory should also include mechanisms to describe more complex time-dynamics of the system.

9.2.2 Value of the Virtual Sensor concept

When different systems that detect the same sort of objects are compared, it is difficult to do this at a detailed level, because it is difficult to come up with the properties that are applicable to a wide variety of systems. But the systems have per definition something in common; the objects they are detecting. By describing their behavior on the basis of the variety in the properties of the objects they detect, the Virtual Sensor concept, the different systems can be compared on an equal level of detail.

The Virtual Sensor concept is a probabilistic way of looking at the system, where engineers tend to look in a deterministic way to their system. In the probabilistic approach it is the task to separate the natural spread of the properties of the observed objects from the noise in the system. Here, noise is not classical random noise of the detector, but represents all the effects that are not included in the model. Although in theory everything could be modeled, in practice there should be a limit. Not represented aspects are therefore called noise, and have to be estimated for its effect. Establishing a good estimate by studying the statistical aspects involves a combination of engineering insights, parameter estimation techniques and non-linear modeling. Once these statistical aspects are resolved, valuable insight is gained and validation can be performed by looking at the variety in the output of the virtual sensor.

9.2.3 Benefit of the multi-level modeling and simulation

The benefits of multi-level modeling and simulation are numerous. An important benefit is from pedagogic nature. With multiple models, one can introduce the systems to outsiders describing the behavior of the system with the highest level (i.e. the simplest model). When this model is understood and the limits of the model become clear, one can descend to a more detailed model. This pedagogic aspect is important, to be able to explain the result of the modeling and simulation to rest of the world. Further, with multiple models it is easier to map validation measurements done on the system directly to an appropriate model. Another benefit is that not only the system is described on multiple levels, but also that the relationship between the levels are investigated. In this way the consistency between the levels can be maintained. Further, with multiple levels of detail for one subsystem it is easier to match a comparable level of detail when another subsystem is simulated at a certain level of detail. Last but not least, with multi-levels modeling & simulation a trade-off can be made between the accuracy of the model and the usage of computational resources.

9.2.4 Role of the Problem Solving Environment in the validation process

To compare the measured behavior of the system with the simulated behavior of the system, both results have to be analyzed in the same way on the variety of certain variables. When the variety is correctly modeled, the probabilistic distribution functions need to have the same shape. When the probabilistic distribution functions have a different shape, this can be due to a wrong parameterization of the simulated behavior model, or a difference between the measured circumstances and the simulated circumstances.

To be able to see this difference, a domain scientist should be able to observe the behavior of the system under a number of circumstances. This means that the person performing this study needs to have full control over the circumstances.

To get full control over the circumstances, knowledge has to be generated about what the spread is of certain parameters under 'normal' circumstances, and

what the difference is with the current circumstances selected from the measurements of the system under study. To generate this knowledge everytime information about the measured and simulated circumstances have to be aggregated. The person performing this study has to keep track of the circumstances, parameters and variables. A human only has a limited capacity to manage such a information set. We solved this by extending the environment with a database that stored intermediate results, ready for a final analysis. The human could select parts of the intermediate results by querying that database on certain aspects, and perform a final analysis interactively with a library of analysis functions.

9.2.5 Benefit of the measured driven simulation

The Dutch government asked for modeling and simulation during the first phase of the "Rekening Rijden" project. Their aim was to combine existing knowledge into a high level model of the system and to get insight about the performance by running simulations with this model. Running the simulations didn't give this insight directly, but the Dutch government got something more valuable in return.

It appeared that the creation of a high level model of the complete system required the combination of measurements, experiences and literature models for each subsystem, an effort that was not done before. Implementing such model in software means that no issues could be left open, because for each parameter a reasonable estimate was needed. At the end of phase 1, there was a good insight about how good the different estimates of the system parameters were, and where additional study was needed.

In phase 2 of the "Rekening Rijden" project four prototypes of the system were installed along the Dutch highway A12. At the beginning of phase 1 this sort of tests along the public road were not advocated, due to amount of effort and resources required for a nearly binary answer on the question of the performance of the system. Yet, with the insight gained in phase 1, detailed research questions could be formulated. Also many parameters of the model had already a good estimate, which made it possible to get a estimate of parameters that could not be measured before due to their dependence of other parameters. This allowed to upgrade the models of the systems to a mature level.

The behavior of the systems for normal traffic could now be accurately estimated. The difference in performance of the real system and the simulated system was mainly due to the response of the system to exceptional passages (think of the passage of a trailer with two motorcycles on it). The difficulty was not only in modeling the response of the system, but to include the details of the exceptional passage in the model of the environment. To solve this issue, in phase 3 of the project experiments on a closed testtrack were performed. The analysis tools for the measurements generated in the previous phase were still valuable. Also simulation played an important role in this phase, but no longer on a high level, but this time on a lower level. Simulation was used to estimate the effect of reflections of microwaves from large surfaces, and the effects of the usage of another payment protocol were investigated.

9.2.6 Scientific objectivity and a political setting

A project as the "Rekening Rijden" project has large social consequences, and is political issue for a long time. It is part of the public responsibility that the academic community takes part in this type of projects. In this light the academic tradition of critical evaluation of the underlying assumptions and investigation of the open issues can be very valuable. The academic contribution was well appreciated.

Yet, at some moments the political process accelerates, and fundamental decisions are made in a few days' time. The results of long term research are interpreted, reinterpreted and simplified. As researcher, you should be able to live with this process. Only arguable mistakes with far going consequences should be corrected.

Another aspect of the academic contribution is less known. It plays a crucial part in recording the scientific results for a period longer than the live time of a project. The government has a long tradition of creating internal reports, but is very reluctant to make the knowledge gained publicly available. This means that there are not that many entries to the set of internal reports once produced. With the project organization favored by the government for this type of political issues this means that once the project members are gone, it becomes extremely simple to lose the results of the project. Results that are only available electronically, as software and databases with measurements and analysis, the situation will be even worse. The academic community knows the necessity to publish their experiences in international media and the value of this sort of libraries and databases. The participation of universities is essential to saveguard the research performed inside such project for the moment that this knowledge is needed again.

Epilogue

So, the computer is still not the magic wand that solves the general problems, but now a collection of tools and filters is created that can serve as dowsing rod that focus the scientist attention on the appropriate places. As said before, an integrated environment that combines statistical analysis algorithms, multi-level modeling & simulation, personalized views and visualization of measurements and simulation results is needed to facilitate the research in more and more extreme elements. Progress towards such environment evolves in slow steps, with different type of steps for each application domain. Yet, each step means already a small improvement for that domain, and by learning from the experiences with measurement driven simulations and simulation driven measurements this evolution can be stimulated.

I like to continue on this road, and create environments that unite reality with a simulated world, to be able to invent and test new algorithms that combine information from multiple sources.

Summary

This thesis describes the study of the methods and algorithms necessary to evaluate the performance of a complex system as "Rekening Rijden". In the first chapter the concepts and challenges of this study are introduced. An important aspect of modeling a complex system lies in the different level of abstractions needed, including the transformation of the state-variables of those levels. For a balanced view on the system two-way transformations are needed; transformations from the high-level to the low-level and vice-versa. In this thesis the benefits are shown that could be gained from iteratively analyzing, modeling, extrapolation and simulation of dependencies in a model at multiple levels.

In the second chapter an overview is made about the treatment of the concept 'complexity' in different scientific domains. The behavior of a system can be described by deducing correlations between variables. Some variables can be seen as means to change the behavior of to system (the input) and other as means to observe the behavior of a system (output). Yet, many descriptions are possible, and a scientist needs to be able say which descriptions are good, because no description is best. Cybernetics makes it possible to order system models from simple descriptions with limited expression power to complex descriptions are equivalent, and combine those descriptions into families of system models. System Theory allows to simplify families of models in a controlled way. For each simplification, it can calculate if the system behavior is still well enough represented. This gives a theoretical background on multilevel modeling of complex systems, and the measures to estimate the quality of the modeling.

In the third chapter of the thesis the context and the details of the application studied is introduced: road pricing. First the system is shown from the perspective of the decision makers. There are many payment schemes that can be invented for road pricing. Three archetypes of payment schemes are worked out in more detail, and mapped to technological solutions. One archetype can be mapped on a solution with all equipment on board of the vehicle. For a second archetype a solution exists purely based on equipment along the road. The last archetype consists of a combination of equipment on board and along the road. This archetype, the "Rekening Rijden" system, is further elaborated. The system is decomposed in subsystems, adding an abstraction layer to the model. The decomposition, including the directed input and output, was described formally. This high-level model is general enough to describe of the nominal behavior of a system, in a way that is comparable for all consortia of companies that can design and build a system for "Rekening Rijden". To allow the description of the non-nominal behavior, each model of a consortia design needs additional internal state descriptions. Precisely these descriptions, based on the measurements and analysis provided by the consortia, was the knowledge of interest for the evaluation study performed by the Dutch government.

The fourth chapter introduces the virtual sensor concept. This concept is used in this thesis to find a level to model the internal state of the subsystems which can be validated with the least effort. A virtual sensor measures a single property of an object, in this case a passing vehicle, with a certain precision and accuracy. Typically this property cannot be measured directly, but should be estimated by combining several measurements from several sensors. The precision and accuracy of these estimates can described with an uncertainty model of the sensor that takes into account the influence of one or multiple physical sensors. Every virtual sensor has a natural timing, depending on when the measurements of the physical sensors are available. In general the precision and accuracy will increase as more measurements become available. It is not necessary to simulate all measurements from the physical sensors; it is enough to create a statistical model of the precision and accuracy of the answer at the moment that the property represented by the virtual sensor is used. This can be easily modeled with a discrete event simulation. The advantage of the statistical approach is that the validation data is related one moment of a passage, and not longer related to the internal clock of the physical sensors.

Ideally the statistical model is only dependent on the characteristic of a single passage, and independent of other factors like other passing vehicles. This would make the model and validation simple. Unfortunately, this is not always the case. The accuracy and precision can be influenced by other passing vehicles. This influence can occur because resources have to be shared between processes, or this influence originates from physical processes as diffraction, reflection or occlusion. The traffic situation where this sort of influence occurs the most is when vehicles drive with high speeds close to each other. In the Netherlands, this combination of high density and high speed is not uncommon.

When the influence of the relative distances and speeds between vehicles has to be taken into account in the virtual sensor model, the simulation framework must be able to generate a realistic spread in the characteristics of these variables.

During the tests along a Dutch highway, measurements were collected for flowing traffic at high densities with unique accuracy. Those measurements were translated into rich datasets with a number of analysis tools, capable of finding periods with comparable characteristics. The rich datasets were used to calibration of the traffic generation process at a microscopic level. Our analysis was unique in the sense that we had two independent measuring-systems, which classified the vehicle types on completely independent features. Because the distributions of both the speed and the distance from the predecessor are highly dependent on the vehicle type, good estimates of the two distributions could be made. With this level of detail in the traffic simulation, it is possible to model the sensor performance on the basis of traffic characteristics.

The communication subsystem is a component of the "Rekening Rijden" system where the precision and accuracy can be influenced by other passages. A measurable effect on the communication and localization has been seen for reflections on large metal surfaces. This effect is validated by extrapolating existing measurements and by comparison with results from other simulators. The simulation of this effect is based on detailed computations of the signals at the physical sensors, and is not a statistical approach as recommended in chapter 4. In chapter 6 is demonstrated how those two approaches can be combined. Several models of the communication subsystem have been implemented, with an increasing level of detail. At the most detailed level, reflections from nearby surfaces are taken into account, creating correlation between passages. By comparing the results at each level, a heuristic has been designed, which can be used to decide what the appropriate level of simulation is for each passage. With this heuristic a much faster simulation is possible with nearly the same accuracy and precision as a simulation on the most detailed level.

In the last chapter, an analysis environment, the Virtual Traffic Laboratory, is introduced, which makes it possible integrate the simulation and measurements results on a natural way. As a case study, the relationship between speed distribution, the fraction of heavy traffic and the traffic volume was explored. The goal of this study was to extrapolate from typical circumstances to the uncommon combination of high volumes and large fractions of heavy traffic. Different measurement-periods were combined, analyzed and translated into parameters for a traffic simulator. To do this, a scientist needs a flexible analysis environment, that integrates analysis of simulation and measurements to find dependencies over large parameter-spaces in the background, while in the foreground patterns could be analyzed in more detail. The iterative transformations from measurements to simulation models made it possible to find the breakpoints of the model; the circumstances occurring in the real world that are not well represented in the model or the analysis methods. At the moment that the limits of the model are known, the effect of the not represented parts can be estimated, and incorporated in a probabilistic way. When we know the limits of the system and its model, we have gained a deep insight in the system, and know which questions we can answer.

The Dutch government asked how the reliability of a complex system as the "Rekening Rijden" system could be estimated, and received a detailed insight about the maturity of the proposed systems. With the models and analyzed datasets generated during the evaluation many other questions can be answered, that are of importance when the technique is taken into account in the discussion of future road pricing systems. A project like "Rekening Rijden" has large social consequences, and it is part of the public responsibility of the academic community to follow the developments critically.

Samenvatting

Dit proefschrift beschrijft het onderzoek naar de methoden en algoritmen die nodig zijn om de betrouwbaarheid van een complex systeem als het 'Rekening Rijden' te kunnen modelleren. In het eerste hoofdstuk worden de aspecten die bij de modellering een rol spelen geïntroduceerd. Een belangrijk aspect is dat een systeem op meerdere niveaus kan worden beschreven. Voor een gebalanceerde beschrijving van een systeem zijn er dan transformaties nodig die van zowel hoog niveau naar laag niveau als vice-versa kunnen worden toegepast. Met behulp van deze transformaties kunnen de analyse, modellering, extrapolatie en simulatie iteratief op verschillende niveaus plaatsvinden, waarbij op een efficiënte manier een goed inzicht in de betrouwbaarheid van het systeem kan worden verkregen.

In het tweede hoofdstuk wordt een overzicht gegeven over de uitwerking van het begrip 'complexiteit' in verschillende wetenschappelijke gebieden. Het gedrag van een systeem kan worden beschreven door het deduceren van verbanden tussen variabelen. Oorzaak en gevolg worden onderscheiden, en variabelen worden ingedeeld in variabelen die het gedrag van het systeem beschrijven (de uitvoer) en de variabelen die het gedrag van het systeem kunnen veranderen (de invoer). Echter, vele verbanden tussen de invoer en uitvoer zijn mogelijk die hetzelfde gedrag kunnen beschrijven. Een wetenschapper moet onderscheid kunnen maken tussen al deze beschrijvingen. Cybernetics maakt het mogelijk om dit onderscheid te maken door systemen te ordenen van simpele beschrijvingen met beperkte extrapolatie vermogen tot complexe beschrijvingen met ruime extrapolatie mogelijkheden. Hierbij kan worden aangegeven welke beschrijvingen equivalent zijn. Equivalente beschrijvingen kunnen worden gecombineerd in families van systeem modellen. Systeem Theorie maakt het mogelijk om families van modellen te vereenvoudigen op een beheersbare manier. Voor iedere vereenvoudiging kan worden aangegeven of gedrag van het systeem goed genoeg is gerepresenteerd. Hiermee is de theoretische achtergrond gegeven voor de modellering van een complex systeem in dit proefschrift.

Het derde hoofdstuk introduceert de context en de details van de toepassing die centraal staat in dit proefschrift: het betalen voor het werkelijke gebruik van de weg. Eerst is de toepassing beschreven vanuit het perspectief van de beleidsmakers. Vele manieren van betalen voor het gebruik zijn mogelijk, maar van al deze manieren zijn drie mogelijkheden uitgewerkt en vertaald in technische oplossingen. Bij eerste mogelijkheid kan er gekozen worden voor een technische oplossing met alle benodigde apparatuur in de auto, bij de tweede mogelijkheid kan er worden gekozen voor een technische oplossing met alle benodigde apparatuur langs de kant van de weg. Bij de derde mogelijkheid, bestaat een technische oplossing uit een combinatie van apparatuur in de auto en apparatuur aan de kant van de weg. Dit laatste archetype, het 'Rekening Rijden' systeem, is verder uitgewerkt in subsystemen, wat een modellering op meerdere niveaus mogelijk maakt. Elk van de subsystemen, inclusief de wisselwerkingen, is beschreven op een formele manier. Deze hoog-niveau beschrijving is voldoende om de gewenste werking van het systeem te beschrijven, op een manier die vergelijkbaar is voor elk consortium van bedrijven die zo een 'Rekening Rijden' systeem kunnen ontwerpen en bouwen. Om te kunnen beschrijven wanneer er een mogelijke afwijking van het gewenste gedrag optreedt, is er voor elk ontwerp een aantal aanvullende interne toestandbeschrijvingen nodig. Precies deze gedetailleerde inzichten in de mogelijke ontwerpen, gebaseerd op metingen en analyses van de bedrijven zelf, is de waardevolle kennis die verkregen is tijdens de evaluatiestudies van het 'Rekening Rijden' systeem.

Het vierde hoofdstuk introduceert het concept van de virtuele sensor. Dit concept wordt in dit proefschrift gebruikt om een niveau van interne toestandbeschrijving te vinden dat gevalideerd kan worden. Een virtuele sensor meet een eigenschap van een voorwerp (zoals een passerend voertuig) met een zekere waarschijnlijkheid van de nauwkeurigheid en betrouwbaarheid. De eigenschap zal over het algemeen niet rechtstreeks kunnen worden gemeten, maar zal tot stand komen door meerdere metingen, mogelijk van meerdere fysieke sensoren, te combineren tot een enkele schatting. Elke virtuele sensor heeft een natuurlijke timing, afhankelijk van het beschikbaar komen van metingen van de fysieke sensoren. Over het algemeen zullen de nauwkeurigheid en betrouwbaarheid toenemen naar mate er meer metingen beschikbaar komen. Het is echter niet nodig alle metingen van elke fysieke sensor te simuleren, het is voldoende om een statistisch model te maken van de nauwkeurigheid en betrouwbaarheid van de virtuele sensor als op een bepaald tijdstip een schatting van de gevraagde eigenschap gebruikt wordt. Dit betekent voor een passerend voertuig dat het statistische model is gerelateerd aan de passage, en niet aan de interne klok van de fysieke sensoren, hetgeen de validatie vereenvoudigt tot het verzamelen van statistiek per passage. Hiermee is de modellering, en daarmee de validatie, op een niveau boven de details van de fysieke sensoren uitgekomen.

In het ideale geval is de statistiek van elke passage dezelfde, onafhankelijk van eerdere of latere passages. In de praktijk is dit niet het geval. De betrouwbaarheid en nauwkeurigheid van de subsystemen kunnen worden beïnvloed door andere passages, gedeeltelijk in de afhandelingsprocessen omdat er componenten tussen processen gedeeld moeten worden, gedeeltelijk fysiek in de vorm van diffractie, reflectie en occlusie. De situatie waarbij deze vorm van beïnvloeding het sterkst is, is als voertuigen met hoge snelheden dicht op elkaar rijden. In Nederland is de combinatie van verkeersstromen met hoge dichtheden en hoge snelheden niet ongewoon, zeker niet in de spits (het tijdstip waarbij het 'Rekening Rijden' systeem actief is). Als de invloed van andere passages op de betrouwbaarheid en nauwkeurigheid wordt meegenomen in een simulatie, moeten de onderlinge afstanden en snelheden bij dit soort verkeersomstandigheden realistisch kunnen worden geschat.

Tijdens de proeven tijdens de evaluatiestudies van het 'Rekening Rijden' systeem zijn er veel metingen verzameld met dit soort verkeersomstandigheden. Met behulp van een rijke set aan analysesoftware, konden de omstandigheden heel gedetailleerd geclassificeerd worden, waardoor het mogelijk was om perioden met precies dezelfde classificatie te combineren tot grotere datasets met een kleine spreiding in de omstandigheden. Deze datasets konden gebruikt worden om de simulatie van verkeersstromen op een microscopisch niveau te kalibreren. De analyse was uniek, omdat gebruik kon worden gemaakt van twee onafhankelijke meetsystemen, die gebruik maakten van twee fysisch geheel verschillende principes. Daarmee werd het mogelijk om voertuigtypen te schatten op twee compleet onafhankelijke manieren, waarmee het foutenpercentage in de voertuigtype schatting gereduceerd kon worden. Een juiste voertuigtype schatting is belangrijk omdat zowel de snelheid als de onderlinge afstand sterk afhankelijk zijn van het voertuigtype. Met behulp van een betrouwbare voertuigtype schatting kan de spreiding in snelheid en onderlinge afstand scherper geschat worden, wat realistische simulaties van verkeer bij hoge dichtheden en hoge snelheden mogelijk maakt.

Het communicatie subsysteem is een onderdeel van het 'Rekening Rijden' systeem, waarbij de betrouwbaarheid en nauwkeurigheid beïnvloed kunnen worden door de passages van andere voertuigen. Een meetbaar effect op de communicatie en de bijbehorende plaatsbepaling is waargenomen voor reflecties aan grote metalen oppervlakken. Dit effect is zowel gevalideerd door bestaande metingen te extrapoleren naar het 'Rekening Rijden' systeem en door de berekeningen van onze simulator te vergelijken met berekeningen van andere simulatoren. De simulatie van dit effect is gebaseerd op een gedetailleerde uitwerking van de fysische principes, en behoort daarmee niet tot de statistische modellering die in hoofdstuk 4 wordt aanbevolen. In hoofdstuk 6 wordt aangetoond dat er een brug kan worden geslagen tussen de gedetailleerde fysische modellen en meer generieke modellen. Het communicatie subsysteem is hiervoor op meerdere niveaus gemodelleerd, en de relatie tussen deze niveaus is bestudeerd. Aan de hand van deze studie kan voor elke passage worden beslist wat het juiste niveau van simulatie is, waardoor een veel snellere simulatie mogelijk is zonder een grote verschillen in betrouwbaarheid en nauwkeurigheid.

In het laatste hoofdstuk wordt het 'Virtual Traffic Laboratory' beschreven. Het 'Virtual Traffic Laboratory' is de analyseomgeving die het mogelijk heeft gemaakt om metingen en simulaties op een natuurlijke manier te koppelen. De eisen die aan zo een omgeving worden gesteld, worden geïllustreerd aan de hand van een onderzoek naar de relatie tussen de gemiddelde snelheid, de hoeveelheid verkeer en het percentage vrachtverkeer hierin. Gekeken is in hoeverre de snelheidsverdelingen geëxtrapoleerd kunnen worden van de gebruikelijke omstandigheden naar ongebruikelijke combinaties van grote hoeveelheden verkeer en hoge percentages vrachtverkeer. Voor dit onderzoek moesten perioden met meetgegevens aan de hand van bepaalde karakteristieken met elkaar worden gecombineerd, geanalyseerd en vertaald in parameters voor de simulator. De omgeving moet hiervoor twee verschillende werkzaamheden tegelijk ondersteunen. Grote parameterstudies kunnen op de achtergrond draaien, zolang de onderzoeker de mogelijkheid heeft om af en toe de voortgang te controleren. De aandacht van de onderzoeker kan zich dan concentreren op de gedetailleerde analyses van deelresultaten op de voorgrond. Een herhaalde vertaling van metingen naar simulaties en van simulaties naar modellen maakt het mogelijk om de limieten van het model te vinden: de omstandigheden waarbij de echte wereld niet langer meer nauwkeurig wordt gerepresenteerd door de simulaties of de analyse methoden. Wanneer de limieten bekend zijn, kunnen deze worden afgeschat en in de modellen worden verwerkt met behulp van statische methoden. Wanneer de limieten en het model van het systeem bekend zijn, is er diep inzicht verworven en kan er aangegeven worden welke vragen over het systeem betrouwbaar beantwoord kunnen worden.

De Nederlandse overheid heeft gevraagd om de betrouwbaarheid van een complex systeem als het 'Rekening Rijden' systeem te onderzoeken. Naar aanleiding van de uitgevoerde studie is goed inzicht gekregen in de mogelijkheden en beperkingen van de voorgestelde systemen. Met de modellen en geanalyseerde datasets die ontwikkeld zijn tijdens de studie kunnen ook vele vergelijkbare vragen beantwoord worden, vragen die van belang zijn wanneer de betrouwbaarheid van de techniek weer aan de orde komt in de discussie over toekomstige betaalsystemen voor het gebruik van de weg. Een project als "Rekening Rijden" heeft grote sociale consequenties, en het behoort dan ook tot de maatschappelijke verantwoordelijkheid van de academische gemeenschap om dit soort projecten kritisch te blijven volgen.

Bibliography

- [1] D. Abramson, J. Giddy and L. Kotler, "High Performance Modeling with Nimrod/G: Killer Application for the Global Grid", in "the International Parallel and Distributed Processing Symposium, (IPDPS)", pp. 520–528, May 2000.
- [2] H. Adams, *The Education of Henry Adams*, privately printed, February 1907.
- [3] H. Afsarmanesh, R. Belleman, A. S. Z. Belloum, A. Benabdelkader, J. F. J. van den Brand, T. M. Breit, H. Bussemaker, G. Eijkel, A. Frenkel, C. Garita, D. L. Groep, A. W. van Halderen, R. M. A. Heeren, Z. W. Hendrikse, L. O. Hertzberger, J. Kaandorp, E. C. Kaletaş, V. Klos, P. Sloot, R. D. Vis, A. Visser and H. H. Yakali, "VLAM: A Grid-Based virtual laboratory", *Scientific Programming*, volume 10(2):pp. 173–181, 2002.
- [4] W. Allcock, J. Bresnahan, R. Kettimuthu, M. Link, C. Dumitrescu, I. Raicu and I. Foster, "The Globus Striped GridFTP Framework and Server", in "Super Computing 2005 (SC05)", November 2005.
- [5] W. Andreoni and A. Curioni, "New Advances in Chemistry and Material Science with CPMD and Parallel Computing", *Parallel Computing*, volume 26:pp. 819–842, 2000.
- [6] J. A. Annema and B. van Wee, "Externe kosten van verkeer. Een vergelijking van mileu-, veiligheids- en congestiekosten", Arena, volume 10, May 2004.
- [7] B. van Arem, A. P. de Vos and M. J. W. A. Vanderschuren, "The microscopic traffic simulation model MIXIC 1.3", Technical Report INRO-VVG 1997-02b, TNO Inro, Delft, January 1997.
- [8] D. C. Arnold, H. Casanova and J. Dongarra, "Innovations of the NetSolve Grid Computing System", Concurrency and Computation: Practice and

Experience, volume 14(13-15):pp. 1457–1479, November/December 2002, special Issue: Grid Computing Environments.

- [9] W. R. Ashby, Design for a brain, John Wiley, New York, 1952.
- [10] W. R. Ashby, An Introduction to Cybernetics, Chapman Hall, London, 1956.
- [11] W. R. Ashby, "Some peculiarities of complex systems", Cybernetic Medicine, volume 9(2):pp. 1–7, 1973.
- [12] W. Asmuth, R. Demski, W. Melchers, M. Schmidt, K.-H. Stappert and K. Wolf, "Feldversuch 'Autobahntechnologien A 555"', Technical Report project FP-9.93906F1, TÜV Rheinland, November 1995.
- [13] D. Bakker, J. Gille, P. Mijjer and H. van Mourik, "Verkeerskundige effecten varianten 'Anders betalen voor Mobiliteit", Technical report, Rijkswaterstaat Adviesdienst Verkeer & Vervoer, March 2005.
- [14] F. J. Barros, "Dynamic Structure Multiparadigm Modeling and Simulation", ACM Transactions on Modeling and Simulation, volume 13(3):pp. 259– 275, July 2003.
- [15] C. Becker, *Leistungsbewertung CEN-DSRC-basierter Mautsysteme*, Ph.D. thesis, Aachen University, Lehrstuhl für Kommunikationsnetze, 2005.
- [16] A. S. Z. Belloum, Z. W. Hendrikse, D. L. Group, E. C. Kaletaş, A. W. van Halderen, H. Afsarmanesh and L. O. Hertzberger, "The VLAM-G Abstract Machine: A Data and Process Handling System on the Grid", in "High-Performance Computing and Networking: 9th International Conference, HPCN Europe 2001", *Lecture Notes on Computer Science*, volume 2110, pp. 81–93, Springer Verlag, 2001.
- [17] C. Brooks, E. A. Lee, X. Liu, S. Neuendorffer, Y. Zhao and H. Z. (eds.), "Heterogeneous Concurrent Modeling and Design in Java", Berkeley, CA USA 94720, July 2005.
- [18] A. J. A. Bruinsma, J. C. Henkus and F. J. van der Mark, "Experiments with prototype infrared and microwave datatransmission equipment for Rekening Rijden", Technical Report TPD-HAI-RPT-90-31, TNO-TPD, Delft, NL, June 1990.
- [19] P. Buneman, S. Khanna and W. C. Tan, "Why and Where: A characterization of data provenance", in "8th International Conference on Database Theory", *Lecture Notes on Computer Science*, volume 1973, pp. 316–330, Springer-Verlag, 2001.
- [20] K. R. Carver and J. W. Mink, "Microstrip antenna technology", *IEEE Transactions on Antennas Propagation*, volume 29:pp. 2–24, January 1981.

- [21] H. Casanova and J. Dongarra, "NetSolve: A network-enabled server for solving computational science problems", *The International Journal of Supercomputer Applications and High Performance Computing*, volume 11(3):pp. 212–223, 1997.
- [22] S. Cheon, C. Seo, S. Park and B. P. Zeigler, "Design and Implementation of Distributed DEVS Simulation in a Peer to Peer Network System", in "Military Government and Aerospace Simulation Symposium", 2004.
- [23] J.-F. Coeurjolly, R. Drouilhet and J.-F. Robineau, "Normalized information-based divergences", Cornell University Library electronic archive paper arXiv:math/0604246v2, November 2006.
- [24] J. S. Dahmann, R. M. Fujimoto and R. M. Weatherly, "The department of defense high level architecture", in "Proceedings of the 1997 Winter Simulation Conference", pp. 142–149, December 1997.
- [25] F. Darema, "Dynamic Data Driven Application Systems: A new paradigm for Application Simulation and Measurements", in "International Conference on Computer Science (ICCS 2004), Krakow, Poland", *Lecture Notes on Computer Science*, volume 3038, pp. 662–669, Springer Verlag, June 2004.
- [26] P. K. Davis and J. H. Bigelow, "Experiments in Multiresolution Modeling", Technical Report document MR-1004-DARPA, RAND institute, 1998.
- [27] P. K. Davis and J. H. Bigelow, "Motivated Metamodels", in "Measuring the Performance and Intelligence of Systems: Proceedings of the 2002 PerMIS Workshop", NIST Special Publication 990, August 2002.
- [28] L. Dorst, A. Hoekstra, J. M. van den Akker, J. Breeman, F. C. A. Groen, J. M. Lagerberg, A. Visser, H. H. Yakali and L. O. Hertzberger, "Evaluating Automatic Debiting Systems by modelling and simulation of virtual sensors", *IEEE Instrumentation and Measurement Magazine*, volume 1(2):pp. 18–25, June 1998.
- [29] C. C. Douglas, "Virtual Telemetry for Dynamic Data-Driven Application Simulations", in "International Conference on Computer Science (IC-CS 2003)", *Lecture Notes in Computer Science*, volume 2657, pp. 279–288, Springer-Verlag, 2003.
- [30] C. C. K. E. T. Verhoef *et al.*, "Vormgeving en effecten van Prijsbeleid op de weg", Technical report, Dutch Ministry of Transport, September 2004.
- [31] J.-P. Eckmann and D. Ruelle, "Ergodic theory of chaos and strange attractors", *Reviews of modern physics*, volume 57(3):p. 617, 1985.

- [32] M. H. Eres, G. E. Pound, Z. Jiao, J. L. Wason, F. Xu, A. J. Keane and S. J. Cox, "Implementation and utilisation of a Grid-enabled problem solving environment in Matlab", *Future Generation Computer Systems*, volume 21(6):pp. 920–929, June 2005.
- [33] D. W. Erwin and D. F. Snelling, "UNICORE: A Grid Computing Environment", in "7th International Conference Euro-Par, Manchester, UK", *Lecture Notes on Computer Science*, volume 2150, pp. 825–839, Springer Verlag, Berlin Heidelberg, 2001.
- [34] L. J. van Ewijk and G. A. van der Spek, "Simulation of reflection effects for a dedicated short range microwave communication system", *IEE Proceedings - Communications*, volume 147(3):pp. 195–199, June 2000.
- [35] I. Foster and C. Kesselman, *The Grid: Blueprint for a New Computing Infra*structure, Morgan Kaufmann, 1999.
- [36] J. Frey, T. Tannenbaum, I. Foster, M. Livny and S. Tuecke, "Condor-G: A Computation Management Agent for Multi-Institutional Grids", in "10th IEEE Symposium on High Performance Distributed Computing (HPDC10)", pp. 55–63, August 2001.
- [37] N. H. Gartner, C. J. Messer and A. K. Rathi (editors), *Revised Monograph* on Traffic Flow Theory, November 1997.
- [38] B. van Halderen, "Virtual Laboratory Abstract Machine Model for Module Writers", Technical report, Universiteit van Amsterdam, July 2000.
- [39] R. C. Hall and D. I. Wu, Design, Review & ID Array Synthesis, User's Guide, Boulder Microwave Technologogies, Inc, Boulder, Colorado, March 1997.
- [40] D. Helbing, "Traffic and related self-driven many-particle systems", *Reviews of Modern Physics*, volume 73:pp. 1067–1141, October 2001.
- [41] L. O. Hertzberger, G. A. Weller and F. C. A. Groen, "A sensor processing model incorporating error detection and recovery", in "Traditional and Non-Traditional Robotic Sensors", (edited by T. C. Henderson), volume F 63, pp. 351–363, Springer-Verlag, Berlin Heidelberg, 1990.
- [42] L. O. Hertzberger *et al.* (editors), Workshop on an Evaluation Methodology for ADS systems, October 1996.
- [43] A. G. Hoekstra, L. Dorst, M. Bergman, J. M. Lagerberg, A. Visser, H. H. Yakali, F. C. A. Groen and L. O. Hertzberger, "High Performance Discrete Event Simulations to evaluate Complex Industrial Systems. The case of Automatic Debiting System for Electronic Toll Collection on Motor Highways", in "High Performance Computing and Networking", *Lecture Notes in Computer Science*, volume 1225, pp. 41–50, Springer-Verlag, 1997.

- [44] A. G. Hoekstra, G. R. Meijer and L. O. Hertzberger, "ADS-SIM: A Generic Simulation Environment to Evaluate and Design Automatic Debitting Systems", in "4th World Congress on Intelligent Transport Systems", October 1997.
- [45] M. Hofmann, "On the Complexity of Parameter Calibration in Simulation Models", *The Journal of Defense Modeling and Simulation*, volume 2(4), October 2005.
- [46] S. P. Hoogendoorn, H. J. V. Zuylen, M. Schreuder, B. Gorte, and G. Vosselman, "Microscopic Traffic Data Collection by Remote Sensing", in "82th TRB Annual Meeting", Transportation Research Board, January 2003.
- [47] J. Horgan, "The End of Science Revisited", IEEE Computer, volume 37(1):pp. 37–43, January 2004.
- [48] R. C. Johnson, Designer Notes for Microwave Antennas, Artech House, Boston, 1991.
- [49] T. Jörgensohn, M. Irmscher and H.-P. Willumeit, "Modelling Human Behaviour, a Must for Human Centred Automation in Transport Systems?", in "4th IEEE/IFIP International Conference on Information Technology for Balanced Automation Systems (BASYS 2000)", September 2000.
- [50] E. C. Kaletaş, Scientific Information Management in Collaborative Experimentation Environments, Ph.D. thesis, Universiteit van Amsterdam, May 2004.
- [51] E. C. Kaletaş, H. Afsarmanesh and L. O. Hertzberger, "Modelling Multi-Disciplinary Scientific Experiments and Information", in "Computer and Information Sciences - ISCIS 2003, 18th International Symposium, Antalya, Turkey, November 3-5, 2003, Proceedings", (edited by A. Yazici and C. Sener), *Lecture Notes in Computer Science*, volume 2869, pp. 75–82, Springer-Verlag, 2003.
- [52] B. S. Kerner and H. Rehborn, "Experimental properties of complexity in traffic flow", *Physical Review E*, volume 53(5), May 1996.
- [53] G. J. Klir, *Architecture of Systems Problem Solving*, Plenum Press, New York, 1th edition, 1985.
- [54] G. J. Klir and D. Elias, *Architecture of Systems Problem Solving*, Kluwer Academic, New York, 2nd edition, 2003.
- [55] K. M. Kockelman, "Modeling Traffic's Flow-Density Relation: Accommodation of Multiple Flow Regimes and Traveler Types", *Transportation*, volume 24(4):pp. 363–374, 2001.
- [56] V. Korkhov, A. S. Z. Belloum and L. O. Hertzberger, "VL-e: Approach

to design a Grid-based Virtual Laboratory", in "5th Austrian-Hangarian yorkshop on Distributed and Parallel systems", September 2004.

- [57] P. Kranendonk, "Rekening Rijden", Universiteitsmodule, IBM IT Specialist Leergang, October 2002.
- [58] A. Ledeczi, J. Davis, S. Neema and A. Agrawal, "Modeling Methodology for Integrated Simulation of Embedded Systems", ACM Transactions on Modeling and Computer Simulation, volume 13(1):pp. 82–103, January 2003.
- [59] J. Liu and E. A. Lee, "A component-based approach to modeling and simulating mixed-signal and hybrid systems", ACM Transactions on Modeling and Computer Simulation, volume 12(4):pp. 343–368, October 2002.
- [60] S. Logghe, *Dynamic modeling of heterogeneous vehicular traffic*, Ph.D. thesis, Katholieke Universiteit Leuven, June 2003.
- [61] Y. Loitière, D. C. Brogan and P. Reynolds, "Simulation Coercion Applied to Multiagent DDDAS", in "International Conference on Computer Science (ICCS 2004), Krakow, Poland", pp. 789–796, June 2004.
- [62] P. F. Reynolds Jr., A. Natrajan and S. Srinivasan, "Consistency Maintenance in Multiresulation Simulations", ACM Transactions on Modeling and Computer Simulation, volume 7(3):pp. 368–392, July 1997.
- [63] R. M. McGraw and R. A. MacDonald, "Abstract Modeling for engineering and engagement level simulations", in "Proceedings of the 2000 Winter Simulation Conference", Orlando, Florida, USA, December 2000.
- [64] G. R. Meijer, L. O. Hertzberger and D. P. van Wijk, "Design and Operation of ERP by Simulation Support", in "International Conference On Transportation in the Next Millenium", September 1998.
- [65] S. Miles, P. Groth, M. Branco and L. Moreau, "The requirements of recording and using provenance in e-Science experiments", *Journal of Grid Computing*, December 2006, ISSN 1572-9814 (Online), published online, in press.
- [66] R. Monasson, R. Zecchina, S. Kirkpatrick, B. Selman and L. Troyansky, "Determining computational complexity for characteristic 'phase transitions", *Nature*, volume 400:pp. 133–137, July 1999.
- [67] K. Nagel and M. Rickert, "Dynamic traffic assignment on parallel computers in TRANSIMS", *Future Generation Computer Systems*, volume 17(5):pp. 637–648, March 2001.
- [68] L. Neubert, L. Santen, A. Schadschneider and M. Schreckenberg, "Single-

vehicle data of highway traffic: A statistical analysis", *Physical Review E*, volume 60(6), December 1999.

- [69] G. Nicolis and I. Prigogine, *Exploring complexity: an introduction*, Freeman, New York, 1989.
- [70] M. Otter, H. Elmqvist and S. E. Mattsson, "Hybrid Modeling in Modelica based on the Synchronous Data Flow Principle", in "IEEE international symposium on computer-aided control system design (CACSD'99)", Hawaii, August 1999.
- [71] E. H. Page and J. M. Opper, "Observations on the complexity of composable simulation", in "WSC '99: Proceedings of the 31st conference on Winter simulation", pp. 553–560, ACM Press, New York, NY, USA, 1999.
- [72] K. Pearson, "Mathematical Contributions to the Theory of Evolution.— XIX. Second Supplement to a Memoir on Skew Variation", *Philosophical Transactions of the Royal Society of London A*, volume 216:pp. 529–557, 1916.
- [73] R. Pieper, "Mobimiles: Bewust op weg", Dutch Ministry of Transport, April 2001.
- [74] R. Pieper and J. Prins, "Mobimiles: Introducing an integrated concept for variable road pricing based on time, location and vehicle", InterimIC, July 2001.
- [75] D. W. Pierce, "Beyond the means: validating climate models with highorder statistics", *Computing in Science & Engineering*, pp. 22–29, Sept/Oct 2004.
- [76] C.-H. Rokitansky, A. Feld, C. Becker and C. Wietfeld, "Mobile Communication Systems for Road Traffic Telematics: Validation of the European DSRC Standard", in "Proceedings of The Second European Personal Mobile Communications Conference (EPMCC'97)", pp. 577–584, Bonn, German, Sep 1997.
- [77] C.-H. Rokitansky, C. Wietfeld and C. Plenge, "SIMCO3++: SImulation of Mobile COmmunications based on Realistic Mobility Models and Road Traffic Scenarios", in "MATHMOD '94", volume 5, pp. 795–801, February 1994.
- [78] P. Ruth, D. Xu, B. K. Bhargava and F.Regnier, "E-notebook Middleware for Accountability and Reputation Based Trust in Distributed Data Sharing Communities", in "2nd International Conference on Trust Management", *Lecture Notes on Computer Science*, volume 2995, pp. 161–175, Springer-Verlag, 2004.
- [79] B. Rutten, B. Bartels and P. de Wolff, "Resultaten validatiestudie MTS", Technical report, LogicaCMG, 2004.

- [80] A. E. K. Sahraoui, "Issues about Traffic Simulation Models: Calibration, Validation and Context Based System Engineering Approach", in "7th World Congress on Intelligent Transport Systems", November 2000.
- [81] L. Speijker, "Statistical analysis of Rekening Rijden", Technical Report NLR CR 97168 L, National Aerospace Laboratory, April 1997.
- [82] C. Tampère and M. van der Vlist, "A Random Traffic Generator for Microscopic Simulation", in "78th TRB Annual Meeting", January 1999.
- [83] I. Taylor, S. Majithia, M. Shields and I. Wang, "Triana Metadata Specification", Technical report, the GridLab project, May 2002.
- [84] D. Thain, T. Tannenbaum and M. Livny, *Condor and the Grid*, chapter 11, pp. 299–336, John Wiley, 2003, ISBN 0-470-85319-0.
- [85] A. Visser, H. Kok, M. Bergman, H. H. Yakali and L. O. Hertzberger, "Calibration of a traffic generator for high-density traffic, using the data collected during a road pricing project", in "9th World congress on Intelligent Transport Systems", Chicago, Illinois, October 2002.
- [86] A. Visser and K. Nagel, "Traffic Simulation, special section", Future Generation Computer Systems, volume 17(5):pp. 625–665, March 2001.
- [87] A. Visser, P. de Waal, J. van den Akker and J. M. Lagerberg, "Contraexpertise Rekeningrijden Phase 1, Simulation, Final Report", Technical report, Universiteit van Amsterdam and National Aerospace Laboratory NLR, October 1999, confidential.
- [88] A. Visser, H. H. Yakali and L. O. Hertzberger, "A next generation ETC system based on time, distance and position - Evaluation", Technical report, UvA PCC Holding B.V., June 2002, confidential.
- [89] A. Visser, H. H. Yakali, J. M. Lagerberg and L. O. Hertzberger, "Contraexpertise Rekeningrijden Phase 2, Measurement Analysis, Final Report", Technical report, Universiteit van Amsterdam, June 2000, confidential.
- [90] A. Visser, H. H. Yakali, A. J. van der Wees, M. Oud, G. A. van der Spek and L. O. Hertzberger, "An hierarchical view on modelling the reliability of a DSRC-link for ETC applications", *IEEE Transactions on Intelligent Transportation Systems*, volume 3(2), June 2002.
- [91] P. R. de Waal, A. Visser, J. Lagerberg and H. J. Kalk, "Application of Chipknip Cards in Rekeningrijden", Technical report, National Aerospace Laboratory, August 1998.
- [92] D. W. Walker, M. Li, O. F. Rana, M. S. Shields and Y. Huang, "The Software Architecture of a Distributed Problem-Solving Environment", Con-

currency: Practice and Experience, volume 12(15):pp. 1455–1480, December 2000.

- [93] F. Wei and K. W. Ng, "A loosely coupled application model for Grids", in "Advances in Grid Computing - EGC 2005", *Lecture Notes on Computer Science*, volume 3470, pp. 1056–1075, 2005.
- [94] N. Wiener, *Cybernetics*, The Technology Press, John Wiley & Sons, New York;Hermann et Cie, Paris, 1948.
- [95] D. P. van Wijk and W. J. Jägers, "Evaluating the Performance of EFC Systems", in "Conference On Transportation in the Next Millennium", September 1998.
- [96] R. Williams, C. Gomes and B. Selman, "Backdoors to Typical Case Complexity", in "18th International Joint Conference on Artificial Intelligence (IJCAI 2003)", August 2003.
- [97] G. Xue, M. J. Fairman, G. E. Pound and S. J. Cox, "Implementation of a Grid Computation Toolkit for Design Optimisation with Matlab and Condor", in "Euro-Par 2003 Parallel Processing", Number 2790 in Lecture Notes in Computer Science, pp. 357–365, 2003.
- [98] B. P. Zeigler, H. Praehofer and T. G. Kim, *Theory of Modeling and Simulation*, Academic Press, San Diego, California, 2nd edition, 2000.
- [99] Z. Zhao, *An Agent Based Architecture for Constructing Interactive Simulation Systems*, Ph.D. thesis, Universiteit van Amsterdam, December 2004.
- [100] "Integrale MIT-verkenning van de verkeers- en vervoersproblematiek Haarlemmermeer-Almere", Dutch Ministry of Transport, February 2004.
- [101] "Congestion Charging Technology Trials Stage 1", Transport for London organization, February 2005.
- [102] "Het Kan! Techniek, organisatie, handhaving en kosten van varianten van Anders Betalen voor Mobiliteit", LogicaCMG, Capgemini and GetID for the Dutch Ministry of Transport, March 2005.
- [103] "Advies van het Nationaal Platform Anders Betalen voor Mobiliteit", Nationaal Platform ABvM for the Dutch Ministry of Transport, May 2005.

Author's publications

Journals

- [Afs02] H. Afsarmanesh, R. Belleman, A. S. Z. Belloum, A. Benabdelkader, J. F. J. van den Brand, T. M. Breit, H. Bussemaker, G. Eijkel, A. Frenkel, C. Garita, D. L. Groep, A. W. van Halderen, R. M. A. Heeren, Z. W. Hendrikse, L. O. Hertzberger, J. Kaandorp, E. C. Kaletaş, V. Klos, P. Sloot, R. D. Vis, A. Visser and H. H. Yakali, "VLAM: A Grid-Based virtual laboratory", *Scientific Programming*, volume 10(2):pp. 173–181, 2002.
- [Alb94] G. D. van Albada, A. Visser and J. M. Lagerberg, "Eye in hand robot calibration", *Industrial Robot*, volume 21(6):pp. 14–17, December 1994.
- [Alb95] G. D. van Albada, J. M. Lagerberg, A. Visser and L. O. Hertzberger, "A low-cost pose-measuring system for robot calibration", *Robotics and Autonomous Systems*, volume 15:pp. 207–227, 1995.
- [Dor98] L. Dorst, A. G. Hoekstra, J. M. van den Akker, J. Breeman, F. C. A. Groen, J. M. Lagerberg, A. Visser, H. H. Yakali and L. O. Hertzberger, "Evaluating Automatic Debiting Systems by modelling and simulation of virtual sensors", *IEEE Instrumentation and Measurement Magazine*, volume 1(2):pp. 18–25, June 1998.
- [Hom01] M. Hommes, A. Visser and L. O. Hertzberger, "VriendenOV, de gebruiker als bron van dynamische reisinformatie", Verkeerskunde, volume 3/2001, March 2001.
 - [Vis95] A. Visser, "An exception-handling framework", International Journal of Computer Integrated Manufacturing, volume 8(3), 1995.
 - [Vis01] A. Visser and K. Nagel, "Traffic Simulation, special section", *Future Generation Computer Systems*, volume 17(5):pp. 625–665, March 2001.

- [Vis02a] A. Visser, "Rekeningrijden", Nederlands Tijdschrift voor Natuurkunde, volume 68(9):p. 311, September 2002.
- [Vis02b] A. Visser, H. H. Yakali, A. J. van der Wees, M. Oud, G. A. van der Spek and L. O. Hertzberger, "An hierarchical view on modelling the reliability of a DSRC-link for ETC applications", *IEEE Transactions on Intelligent Transportation Systems*, volume 3(2), June 2002.

Conference papers

- [Alb95] G. D. van Albada, J. M. Lagerberg, A. Visser and L. O. Hertzberger, "A portable measuring system for robot calibration", in "28th International Symposium on Automative Technology and Automation ISATA, Dedicated Conference on Mechatronics", (edited by J. Soliman and D. Roller), pp. 441–448, Automotive Automation Limited, 42 Lloyd Park Avenue, Croydon CR0 5SB, England, September 1995.
- [Dor97] L. Dorst, F. C. A. Groen, J. M. van den Akker, J. Breeman, A. G. Hoekstra, J. M. Lagerberg, A. Visser, H. H. Yakali and L. O. Hertzberger, "The virtual sensor concept applied to simulation: the evaluation of Automatic Debiting Systems", in "1997 Workshop on Emergent Technologies & Virtual Systems for Instrumentation and Measurements", pp. 118–124, Ontario, Canada, May 15-17 1997.
- [Fas03] M. L. Fassaert, S. B. M. Post and A. Visser, "The common knowledge model of a team of rescue agents", in "1th International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster", July 2003.
- [Hoe97a] A. G. Hoekstra, L. Dorst, M. Bergman, J. M. Lagerberg, A. Visser, H. H. Yakali, F. C. A. Groen and L. O. Hertzberger, "High Performance Discrete Event Simulations to evaluate Complex Industrial Systems. The case of Automatic Debiting System for Electronic Toll Collection on Motor Highways", in "High Performance Computing and Networking", (edited by B. Hertzberger and P. Sloot), *Lecture Notes in Computer Science*, volume 1225, pp. 41–50, Springer-Verlag, 1997.
- [Hoe97b] A. G. Hoekstra, L. Dorst, M. Bergman, J. M. Lagerberg, A. Visser, H. H. Yakali, F. C. A. Groen and L. O. Hertzberger, "Modelling and simulation of automatic debiting systems for electronic collection on motor highways", in "IASTED International Conference Applied Modelling and Simulation", (edited by M. Hamzam), pp. 104–108, 1997.

- [Oli06] F. Oliehoek and A. Visser, "A hierarchical model for decentralized fighting of large scale urban fires", in "Hierarchical Autonomous Agents and Multi-Agent Systems (H-AAMAS'06)", May 2006.
- [Oom04] S. Oomes, P. Jonker, M. Poel, A. Visser and M. Wiering, "The Dutch AIBO Team 2004", in "8th International RoboCup Competition CD", July 2004.
 - [Pfi06] M. Pfingsthorn, B. Slamet, A. Visser and N. Vlassis, "UvA Rescue Team 2006; RoboCup Rescue - Simulation League", in "Proceedings CD RoboCup 2006, Bremen, Germany", June 2006.
 - [Pfi07] M. Pfingsthorn, B. Slamet and A. Visser, "A Scalable Hybrid Multi-Robot SLAM Method for Highly Detailed Maps", in "Proceedings of the 11th RoboCup International Symposium", July 2007, proceedings CD. To be published in the Lecture Notes on Artificial Intelligence series.
 - [Pos03] S. B. M. Post, M. L. Fassaert and A. Visser, "Reducing the communication for multiagent coordination in the RoboCupRescue Simulator", in "7th RoboCup International Symposium CD", July 2003.
 - [Pos04] S. B. M. Post, M. L. Fassaert and A. Visser, "The high-level communication model for multiagent coordination in the RoboCupRescue Simulator", in "7th RoboCup International Symposium", (edited by D. Polani, B. Browning, A. Bonarini and K. Yoshida), *Lecture Notes* on Artificial Intelligence, volume 3020, pp. 503–509, Springer-Verlag, 2004.
 - [Sch94] K. Schröer, S. L. Albright, M. Grethlein, A. Lisounkin, G. D. van Albada, J. M. Lagerberg and A. Visser, "Advances in robot calibration: Modelling, Parameter Identification, Measurement", in "Proceedings of the 25th International Symposium on Industrial Robots (25th ISIR)", pp. 733–740, Mechanical Engineering Publications, England, 1994.
 - [Sla05] B. Slamet and A. Visser, "Purposeful perception by attention-steered robots", in "Proc. 17th Dutch-Belgian Artificial Intelligence Conference, BNAIC'05", Brussels, October 2005.
 - [Soe06] D. van Soest, M. de Greef, J. Sturm and A. Visser, "Autonomous Color Learning in an Artificial Environment", in "Proc. 18th Dutch-Belgian Artificial Intelligence Conference, BNAIC'06", pp. 299–306, Namur, Belgium, October 2006.
 - [Stu07] J. Sturm, P. van Rossum and A. Visser, "Panoramic Localization in the 4-Legged League", in "RoboCup 2006: Robot Soccer World Cup X", (edited by G. Lakemeyer, E. Sklar, D. Sorrenti and T. Takahashi),

Lecture Notes on Artificial Intelligence, volume 4434, Springer-Verlag, Bremen, October 2007.

- [Ter01] F. Terpstra, A. Visser and L. O. Hertzberger, "An on-line planner for Marie", in "12th Irish Conference on Artificial Intelligence and Cognitive Science", pp. 199–209, September 2001.
- [Vis95a] A. Visser, G. D. van Albada, L. O. Hertzberger and G. den Boer, "Data and Event Handling for the MARIE Vehicle", in "28th International Symposium on Automative Technology and Automation ISA-TA", pp. 403–410, Automotive Automation Limited, Croydon, 1995.
- [Vis95b] A. Visser, G. D. van Albada and J. M. Lagerberg, "Photogrammetry Applied to Robot Calibration", in "14-th Benelux Meeting on Systems and Control — Final Program, Abstracts, Course Note", p. 142, Katholieke Universiteit Leuven, Departement Elektrotechniek, 1995.
- [Vis00] A. Visser, A. J. van der Wees and L. O. Hertzberger, "Discrete Event Modelling Methodology for Intelligent Transport Systems", in "Proceedings of the World Congress on Intelligent Transport Systems", November 2000.
- [Vis02a] A. Visser, I. E. M. Breuker and S. J. Huiberts, "Advanced Logistics Information Exchange - Hoe verbeteren we de dynamiek bij het vergelijken van logistieke diensten?", in "Vervoerslogistieke Werkdagen", November 2002.
- [Vis02b] A. Visser, H. Kok, M. Bergman, H. H. Yakali and L. O. Hertzberger, "Calibration of a traffic generator for high-density traffic, using the data collected during a road pricing project", in "9th World congress on Intelligent Transport Systems", Chicago, Illinois, October 2002.
- [Vis02c] A. Visser, H. H. Yakali, A. J. van der Wees, M. Oud, G. A. van der Spek and L. O. Hertzberger, "The reliability of a DSRC-link for ETC applications modelled on 5 levels of detail", in "IEEE 5th International Conference on Intelligent Transportation Systems", September 2002.
- [Vis03a] A. Visser, A. van Inge and F. C. A. Groen, "Experiences with the innovation of an Autonomous Systems Course", in "9th International Conference of European University Information Systems", July 2003.
- [Vis03b] A. Visser, S. B. M. Post and M. L. Fassaert, "The communication approach of the 'UvA Rescue C2003'-team", in "7th RoboCup International Robot Competition CD", July 2003.
- [Vis04a] A. Visser, G. Pavlin, S. P. van Gosliga and M. Maris, "Self-

organization of multi-agent systems", in "Proc. of the International workshop Military Applications of Agent Technology in ICT and Robotics", The Hague, The Netherlands, November 2004.

- [Vis04c] A. Visser, J. G. Zoetebier, H. H. Yakali and L. O. Hertzberger, "The measurement architecture of the Virtual Traffic Laboratory", in "International Conference on Computer Science (ICCS 2004)", (edited by M. Bubak, G. D. van Albada, P. Sloot and J. Dongarras), *Lecture Notes on Computer Science*, volume 3036, pp. 515–518, Springer Verlag, June 2004, ISBN 3-540-22114-X.
- [Vis06a] A. Visser, P. van Rossum, J. Westra, J. Sturm, D. van Soest and M. de Greef, "Dutch AIBO Team at RoboCup 2006", in "Proceedings CD RoboCup 2006, Bremen, Germany", June 2006.
- [Vis06b] A. Visser, J. Sturm and F. Groen, "Robot companion localization at home and in the office", in "Proc. 18th Dutch-Belgian Artificial Intelligence Conference, BNAIC'06", pp. 347–354, Namur, Belgium, October 2006.
- [Vis07a] A. Visser, B. Slamet, T. Schmits, L. A. González Jaime and A. Ethembabaoglu, "Design decisions of the UvA Rescue 2007 Team on the Challenges of the Virtual Robot competition", in "Proc. 4th International Workshop on Synthetic Simulation and Robotics to Mitigate Earthquake Disaster", pp. 20–26, July 2007.
- [Vis07b] A. Visser, Xingrui-Ji, M. van Ittersum, L. A. González Jaime and L. A. Stancu, "Beyond frontier exploration", in "Proceedings of the 11th RoboCup International Symposium", July 2007, proceedings CD. To be published in the Lecture Notes on Artificial Intelligence series.
- [Wel93] B. G. Welz, L. Camerinha, T. C. Lueth, S. Munch, L. Stochiero, J. Tramu and A. Visser, "A toolbox of integrated planning tools - a Case Study", in "Proceedings of the IFIP TC/5WG5.10 Working Conference on Interfaces in Industrial Systems for Production and Engineering", Elsevier, Amsterdam, 1993.
- [Wij05] N. Wijngaards, F. Dignum, P. Jonker, T. de Ridder, A. Visser, S. Leijnen, J. Sturm and S. van Weers, "Dutch AIBO Team at RoboCup 2005", in "Proceedings CD RoboCup 2005, Osaka, Japan", July 2005.

Technical Reports

[Oom05] S. Oomes, P. Jonker, M. Poel, A. Visser, M. Wiering, W. Caarls, S. Leijnen, S. van Weers, N. Wijngaards and F. Dignum, "The Dutch AIBO Team Report on RoboCup 2004", Technical report, Dutch Aibo Team, January 2005.

- [Stu05] J. Sturm, A. Visser and N. Wijngaards, "Dutch Aibo Team: Technical Report RoboCup 2005", Technical report, Dutch Aibo Team, October 2005.
- [Vis99] A. Visser, P. de Waal, J. van den Akker and J. M. Lagerberg, "Contraexpertise Rekeningrijden Phase 1, Simulation, Final Report", Technical report, Universiteit van Amsterdam and National Aerospace Laboratory NLR, October 1999, confidential.
- [Vis00] A. Visser, H. H. Yakali, J. M. Lagerberg and L. O. Hertzberger, "Contra-expertise Rekeningrijden Phase 2, Measurement Analysis, Final Report", Technical report, Universiteit van Amsterdam, June 2000, confidential.
- [Vis02] A. Visser, H. H. Yakali and L. O. Hertzberger, "A next generation ETC system based on time, distance and position Evaluation", Technical report, UvA PCC Holding B.V., June 2002, confidential.
- [Vis06] A. Visser, J. Sturm, P. van Rossum, J. Westra and T. Bink, "Dutch Aibo Team: Technical Report RoboCup 2006", Technical report, Dutch Aibo Team, December 2006.
- [Yak00] H. H. Yakali and A. Visser, "OBU localisation TCL assessment test for Q-Free", Technical report, Universiteit van Amsterdam, 2000, confidential.

Project's deliverables

Phase1

[RR-UvA-96-01]	A. Visser, "Systeem voor Rekening Rijden", Technical report, Universiteit van Amsterdam, January 1996.
[RR-UvA-96-02]	A. Visser, "Critical path analysis for an Automatic Debiting System", Technical report, Universiteit van Amsterdam, Fe- bruary 1996.
[RR-UvA-96-03]	A. G. Hoekstra, "Traffic Generator and Tracking for the Inte- grated System Simulator", Technical report, Universiteit van Amsterdam, February 1996.
[RR-UvA-96-04]	J. Lagerberg and A. Visser, "High-level events", Technical report, Universiteit van Amsterdam, February 1996.
[RR-UvA-96-05]	A. Visser, "High-level event-handling", Technical report, Universiteit van Amsterdam, May 1996.
[RR-UvA-96-06]	A. Visser, "Integration of the communication modules", Technical report, Universiteit van Amsterdam, June 1996.
[RR-UvA-96-07]	J. Lagerberg, "A fictitious ADS; from vendor proposal to si- mulation", Technical report, Universiteit van Amsterdam, June 1996.
[RR-UvA-96-08]	A. Visser, "Communication event-handling, version 1.0", Technical report, Universiteit van Amsterdam, June 1996.
[RR-UvA-96-09]	J. Lagerberg and A. Visser, "Generic events and functions for implementation of an ADS", Technical report, Universiteit van Amsterdam, July 1996.

- [RR-UvA-96-10] A. Visser, "System Quality Factors for the evaluation of 'Rekening Rijden' systems, version 1.0", Technical report, Universiteit van Amsterdam, August 1996.
- [RR-UvA-96-11] L. Dorst and H. H. Yakali, "The geometry engine", Technical report, Universiteit van Amsterdam, August 1996.
- [RR-UvA-96-12] J. Lagerberg and A. Visser, "Specification of the generic event list", Technical report, Universiteit van Amsterdam, August 1996.
- [RR-UvA-96-13] A. Visser, "Communication status", Technical report, Universiteit van Amsterdam, August 1996.
- [RR-UvA-96-14] A. Visser, "System Quality Factors for the evaluation of 'Rekening Rijden' systems, version 2.0", Technical report, Universiteit van Amsterdam, September 1996.
- [RR-UvA-96-15] A. Visser, "Communication event-handling, version 2.0", Technical report, Universiteit van Amsterdam, October 1996.
- [RR-UvA-96-16] L. Dorst, "Generic sensor modelling for Rekening Rijden", Technical report, Universiteit van Amsterdam, November 1996.
- [RR-UvA-96-17] A. Visser, "System Quality Factors for the evaluation of 'Rekening Rijden' systems, version 3.0", Technical report, Universiteit van Amsterdam, November 1996.
- [RR-UvA-96-18] J. Lagerberg, "A fictitious ADS from 2D to 3D", Technical report, Universiteit van Amsterdam, November 1996.
- [RR-UvA-97-01] H. H. Yakali and J. Lagerberg, "The traffic (disaster) file", Technical report, Universiteit van Amsterdam, January 1997.
- [RR-CMG-97-01] T. van der Post, "Output of Registration Subsystem", Technical report, Computer Management Group, January 1997.
- [RR-UvA-97-02] H. H. Yakali, "Level of abstraction for occlusion computation", Technical report, Universiteit van Amsterdam, February 1997.
- [RR-UvA-97-03] A. Visser, "An environment for modelling the protocol for DSRC-communication according to the European Pre-Standard", Technical report, Universiteit van Amsterdam, February 1997.
- [RR-UvA-97-04] H. H. Yakali and J. H. Breeman, "The benchmarking of some of the Utility Functions and different platforms", Technical report, Universiteit van Amsterdam, February 1997.
- [RR-UvA-97-05] A. Visser, "Gebruik van specificaties verkeersmodellen", Technical report, Universiteit van Amsterdam, March 1997.
- [RR-UvA-97-06] A. Visser, "An environment for modelling DSRcommunication with microwaves", Technical report, Universiteit van Amsterdam, March 1997.
- [RR-UvA-97-07] A. Visser and H. C. Kok, "An initial outline of the functionalities needed for scheduling events above the OSI-layer 7", Technical report, Universiteit van Amsterdam, March 1997.
- [RR-UvA-97-08] H. C. Kok and A. Visser, "Proposal for protocol modelling in ADSsim 2.0", Technical report, Universiteit van Amsterdam, April 1997.
- [RR-NLR-97-01] L. Speijker, "Statistical analysis of Rekening Rijden", Technical Report NLR CR 97168 L, National Aerospace Laboratory, April 1997.
- [RR-CMG-97-02] A. J. van der Wees and A. Visser, "Functional description of Channel model", Technical report, Computer Management Group, April 1997.
- [RR-UvA-97-09] A. Visser, "Hierarchical modelling applied to the FEL experiments", Technical report, Universiteit van Amsterdam, May 1997.
- [RR-UvA-97-10] J. Lagerberg, "Proposal for more realistic fADS for version 1.exp", Technical report, Universiteit van Amsterdam, June 1997.
- [RR-UvA-97-11] J. Lagerberg, "Proposal of gADS and gADSsim for version 1.exp", Technical report, Universiteit van Amsterdam, June 1997.
- [RR-UvA-97-12] H. H. Yakali, "Simplifications of the EM-field Equations", Technical report, Universiteit van Amsterdam, June 1997.
- [RR-UvA-97-13] H. H. Yakali, "Kinematic model based vehicle motion", Technical report, Universiteit van Amsterdam, June 1997.
- [RR-UvA-97-14] A. Visser, "Proposal for more generic ADS code", Technical report, Universiteit van Amsterdam, July 1997.
- [RR-UvA-97-15] A. Visser, "Communication Modelling Interface", Technical report, Universiteit van Amsterdam, July 1997.
- [RR-UvA-97-16] H. H. Yakali, "EM-field Computation for the communication simulation", Technical report, Universiteit van Amsterdam, July 1997.

- [RR-NLR-97-02] J. H. Breeman, "EM Field Patch Antenna", Technical report, National Aerospace Laboratory, July 1997.
- [RR-UvA-97-17] A. Visser, "Towards the experimental version ADSSIM 1.exp", Technical report, Universiteit van Amsterdam, August 1997.
- [RR-UvA-97-18] H. H. Yakali, A. J. van der Wees and A. Visser, "Functional description of Channel model update", Technical report, Universiteit van Amsterdam, August 1997.
- [RR-UvA-97-19] A. Visser, "Proposal for a realistic communication protocol for version 1.exp", Technical report, Universiteit van Amsterdam, August 1997.
- [RR-NLR-97-03] P. R. de Waal, J. M. van den Akker and J. H. Breeman, "A note on conditional scheduling", Technical report, National Aerospace Laboratory, August 1997.
- [RR-UvA-97-20] A. Visser, "A realistic communication protocol for fADS", Technical report, Universiteit van Amsterdam, September 1997.
- [RR-UvA-97-21] H. H. Yakali, "Sinc function test on the Software", Technical report, Universiteit van Amsterdam, October 1997.
- [RR-UvA-98-01] M. Oud, "Development and Validation of the EM-field Computation for the Communication Simulation", Technical report, Universiteit van Amsterdam, February 1998.
- [RR-UvA-98-02] A. Visser, "Proposal SQF implementation ADSSIM 2.1", Technical report, Universiteit van Amsterdam, February 1998.
- [RR-TNO-98-01] B. van Arem, A. Vieveen, M. van der Vlist, M. Vanderrschuren and C. Tampere, "Model Description RTS Module, – Concept –", Technical report, the Netherlands Organisation for Applied Scientific Research, 1998.
- [RR-UvA-98-03] A. Visser, P. R. de Waal and H. J. Kalk, "A Chipper Protocol for fADS", Technical report, Universiteit van Amsterdam, June 1998.
- [RR-NLR-98-01] P. R. de Waal, A. Visser, J. Lagerberg and H. J. Kalk, "Application of Chipknip Cards in Rekeningrijden", Technical report, National Aerospace Laboratory, August 1998.
- [RR-NLR-98-02] P. R. de Waal, A. Visser and J. Lagerberg, "Simulation Studies of Chipknip and Chipper Applications in Rekeningrijden",

Technical report, National Aerospace Laboratory, September 1998.

- [RR-UvA-98-04] A. Visser, "The realistic linkbudget", Technical report, Universiteit van Amsterdam, October 1998.
- [RR-UvA-99-01] J. Lagerberg, "Communicatie fouten als functie van de PSAM tijden - Simulatie studie voor InterPay", Technical report, Universiteit van Amsterdam, February 1999.
- [RR-NLR-99-01] P. R. de Waal, A. Visser, J. Lagerberg and M. van den Akker, "Simulation Study of Chipknip and Chipper purses with relation to the SSSL/Combitech system", Technical report, National Aerospace Laboratory, March 1999.
- [RR-UvA-99-02] H. H. Yakali, "Topology Tool Specifications", Technical report, Universiteit van Amsterdam, March 1999.
- [RR-UvA-99-03] A. Visser, "The algorithm of the coordination subsystem of ANH", Technical report, Universiteit van Amsterdam, June 1999.

Phase2

- [RR-UvA-99-04] A. Visser, "Fraude mogelijkheden op de secondaire weg", Technical report, Universiteit van Amsterdam, October 1999.
- [RR-UvA-99-05] A. Visser, "Gevoeligheidsanalyse CT", Technical report, Universiteit van Amsterdam, October 1999.
- [RR-UvA-99-06] H. H. Yakali, "Effects of reflections on monopulse-based localisation errors", Technical report, Universiteit van Amsterdam, November 1999.
- [RR-UvA-00-01] A. Visser, "Database security", Technical report, Universiteit van Amsterdam, March 2000.
- [RR-UvA-00-02] A. Visser, H. H. Yakali, J. Lagerberg and L. O. Hertzberger, "Contra-expertise Rekeningrijden Phase 2- Measurements Analysis", Technical report, Universiteit van Amsterdam, July 2000.
- [RR-UvA-00-03] A. Visser, "Gebruik meetgegevens Lelystad", Technical report, Universiteit van Amsterdam, December 2000.
- [RR-UvA-00-04] J. Lagerberg, "Gebruik meetgegevens A12", Technical report, Universiteit van Amsterdam, December 2000.

Phase3

[RR-UvA-01-01] A. Visser and H. H. Yakali, "Readability of Foreign License Plates", Technical report, Universiteit van Amsterdam, Februari 2001.

Acknowledgments

First I have to thank my mother and father who encouraged me and my sister to study at the university. At the Universiteit Leiden I was introduced in the experimental work at a physics laboratory by Rob van den Berg, Hans van der Laan and Pieter-Jan van der Zaag. At the Universiteit van Amsterdam I got the job of Geleyn Meijer, who was able to manage multiple projects at the same time. I worked directly on many robotics related research projects with many colleagues. In the beginning I closely cooperated for instance with George de Boer, Michiel Wiedijk, Toto van Inge, José Lagerberg and Hamideh Afsarmanesh. I also remember from those early years the methodological excellence of Dick van Albada and my introduction into international project management by Frank Tuijnman.

The "Rekening Rijden" project was compared to other projects the largest and the most prolonged. I cooperated with many people inside and outside the institute. To name a few: Alfons Hoekstra, Leo Dorst, Henk Kok, Martin Bergman, Mireille Oud, Gerard van der Spek, Ignas van Megen, Bart van Arem, Martie van der Vlist, Chris Tampère, Aad-Jan van der Wees, Peter de Waal, Marjan van den Akker and Jan Breeman. I also was present at many interesting technical discussions in the Ministry of Transport, with for instance Anne Tip, Rinus van Pelt, Jesper Engdahl, Stefan Eisses, Wim van Nifterick, Ben Kleiss, Wim Jägers, Piet Lagerweij, Arie Lodder and Dik van Wijk. It is unbelievable how much work is performed by these people to guarantee a technical feasible solution for "Rekening Rijden". This thesis only touches a small part of this effort. Inside the institute, my closest cooperation was with my roommates Hakan Yakali and José Lagerberg.

For this thesis I have to thank many. First I thank my family, who had to miss me at Sunday afternoon for so long. Also thanks to my promoter, who showed so much patience. I was fortunate with members of the Intelligent Autonomous Systems group, who included me long before my title was there. I am very thankful to Breanndán Ó Nualláin, who checked the complete thesis on consistency. My wife, Anniek Elias, did the same for the "Samenvatting". In the mean time, my students cheered me up with scientific discussions, daily progress and the latest trends.

Further, I like to thank all the other members of the institute, such as the secretaries and the people of the computer support group, who are so important for the functioning of the institute. Special thanks to Jeroen Roodhart and Derk Bouhuijs, who installed in short notice the software needed to make this thesis publisher ready, and Jan Wortelboer, who was able to recover a five years old document from my mail. At the laboratory, there were students that were always present, such as Frank Terpstra, Mattieu Hommes, Joost Zoetebier, Erik de Ruiter, Ilja Breuker, Koen Boinck, Jürgen Sturm, David Knibbe and Martijn Liem. Outside the institute, I met a lot of interesting people, who make it worthwhile to work in this field. Most intense were the encounters at the RoboCup competition, where I travelled with for instance Maurits Fassaert, Stef Post, Stijn Oomes, Stefan Leijnen, Silvain Weers, Joost Westra, Niek Wijngaards, Paul van Rossum, Jürgen Sturm, Max Pfingsthorn, Bayu Slamet and Tijn Schmits. At home, I have a family, friends and neighbors who actively remind me that there is more in life than science. This thesis was a long lasting project, but with the combined support of you all I managed it.