

Vehicle-class Specific Control of Freeway Traffic

Thomas Schreiter

Delft University of Technology, 2013

This thesis is the result of a project funded by
the Port of Rotterdam Authority (Havenbedrijf Rotterdam),
the Traffic Management Company Rotterdam (De Verkeersonderneming) and
the Netherlands Research School for Transport, Infrastructure and Logistics TRAIL.



Cover illustration: Peter van Dorst

Vehicle-class Specific Control of Freeway Traffic

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op maandag 18 maart 2013 om 12.30 uur
door

Thomas SCHREITER

Diplom-Informatiker, Universität Karlsruhe (TH), Duitsland,
geboren te Zwickau, Duitsland.

Dit proefschrift is goedgekeurd door de promotor:

Prof. dr. ir. S.P. Hoogendoorn

Copromotor: Dr. ir. J.W.C. van Lint

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. ir. S.P. Hoogendoorn,	Technische Universiteit Delft, promotor
Dr. ir. J.W.C. van Lint,	Technische Universiteit Delft, copromotor
Prof. dr. ir. B. De Schutter,	Technische Universiteit Delft
Prof. dr. R.L. Bertini,	Portland State University
Prof. Dr.-Ing. M. Papageorgiou,	Technical University of Crete
Prof. dr. P.B. Mirchandani,	Arizona State University
Prof. Dr.-Ing. K. Bogenberger,	Universität der Bundeswehr München
Prof. dr. ir. B. van Arem,	Technische Universiteit Delft, reservelid

TRAIL Thesis Series no. T2013/4, the Netherlands Research School TRAIL

TRAIL

P.O. Box 5017

2600 GA Delft

The Netherlands

Phone: +31 (0) 15 278 6046

E-mail: info@rsTRAIL.nl

ISBN: 978-90-5584-163-9

Copyright © 2013 by Thomas Schreiter

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

Printed in the Netherlands

Preface

After I had finished my diploma thesis in computer science in 2008, I had absolutely no intent to stay in academia and spend days sitting in front of a computer to solve abstract problems. My advisors then pushed me to turn the thesis into a paper and submit it to a conference. After a few months of thinking and a visit to an interesting far-away place to meet researchers from around the world, I found that academia looked much brighter. I then searched for interesting groups outside of Germany that could educate a computer scientist in transportation, which led me to the Transport and Planning department in the beautiful city of Delft.

Serge Hoogendoorn and Hans van Lint invited me to an interview to discuss possible research projects. What was intended as a two-hour talk turned into a six-hour visit, where I received a warm welcome by the members of the group and had a glimpse into the social activities between the members. This eventually led me to accept the offer and move to Delft. I want to thank Serge and Hans for that welcome and for the discussions and education in Dynamic Traffic Management in the following four years. You have taught me how to become a scientific researcher in a very interesting field. I also want to thank Ernst Scheerder from De Verkeersonderneming and Zlatan Muhurdarevic from the Havenbedrijf Rotterdam for sponsoring and supervising the practical part of my PhD project. Becoming a PhD student was one of the best decisions I have made. I have met many interesting people, learned a lot about traffic, been educated as a researcher and traveled to many places. I have lived in a country that is culturally similar to Germany, with the differences being that there are canals in the cities but absolutely no mountains, the people are somehow much more relaxed, the weather is milder, there is an incredible bike infrastructure, and the best food is a raw fish pickled in saltwater.

I also want to thank my roommates Tony, Victor, Femke, Chris, Maaïke, Nina, Yufei, Olga, Tamara and Gerdien for the challenging (and nerdy) discussions and the intercultural experience. Furthermore, I enjoyed playing ping-pong after lunch and in the evenings and getting to know my colleagues from a different side. I believe that this table supports socializing between the members, which especially helps newcomers integrate into the group. Another big thanks goes to the secretaries, particularly Priscilla, who manage the paperwork and magically keep track of the schedule of every person so that the department and the travels run smoothly. Also a big thank you goes to Peter, Kees and Edwin, who manage the databases and the hardware, which includes my

bike that you helped fix many times. A general thank you goes to all the Dutch people who remained patient while I was speaking Dutch. In general, the atmosphere in our group is exceptionally friendly and supportive. In fact, I did not even bother to make friends outside of our group, since there are so many fantastic people at the department whom I spent a lot of time with after work; thank you Pavle, Daniel, Riccardo, Egidio, Francesco, Gijs, Erik-Sander, Kakpo, Shiomi, Bernat, Nikola, Meng, Olga, Giselle, Victor, Mo, Adam, Tamara, Yufei and Ramon.

There are a couple of colleagues whom I especially want to thank. Pavle, we had some interesting and deep discussions about politics and culture. Ramon, we did not just spend a substantial amount of time in a window-less room equipped with a whiteboard and a coffee machine, we also had inspiring discussions about life. I am happy that the two of you accepted to be my paranymphs. Yufei and Femke, we started roughly at the same time and worked with the same traffic flow model. Naturally, we had many discussions about modeling and collaborated in many ways. The finale comes in March, when we will defend our theses in the same week. I am looking forward to celebrating together, although I am nervous already.

I also want to thank my family for their support and I know that I can always rely on you folks, although I probably do not call home as often as I should. Finally, my biggest thanks goes to my girlfriend Lisa, who supports me unconditionally both in research and in other aspects of life. Sharing my life with you makes me very happy. I am looking forward to experiencing our next step in Berkeley.

Thomas Schreiter, February 2013

Contents

Preface	v
List of Figures	xvi
List of Tables	xvii
Notation	xix
1 Introduction	1
1.1 Problem Definition	2
1.2 Scope of the Thesis	4
1.3 Scientific Contributions	4
1.4 Practical Contributions	5
1.5 Structure of the Thesis	6
2 Design and Model Choices for the Control Loop	9
2.1 Design Choices for the Control Component	10
2.1.1 Requirements of the Control Component	10
2.1.2 Control Methods	11
2.1.3 Conclusions of the Control Component	15
2.2 Design Choices for the Prediction Component	17
2.2.1 Requirements of the Prediction Components	17
2.2.2 Prediction Models	21
2.2.3 Conclusion of the Prediction Component	25
2.3 Design Choices for the Estimation Component	27

2.3.1	Requirements of the Estimation Component	27
2.3.2	Estimation Models	28
2.3.3	Conclusion of the Estimation Component	30
2.4	DTM Measures Considered in This Thesis	31
2.5	Traffic Sensors Considered in This Thesis	33
2.6	Conclusion	34
3	Estimation of Spatiotemporal Traffic Characteristics	37
3.1	Background	38
3.2	Literature Overview	40
3.3	Methodology	41
3.3.1	Preprocessing	44
3.3.2	Edge Detector	44
3.3.3	Line Detector	45
3.3.4	Postprocessing	47
3.4	Extracting Shock Waves from Empirical Data	48
3.4.1	Extracting Shock Waves Occurring at Stop-and-go Waves	49
3.4.2	Extracting Shock Waves Occurring in Free Flow	50
3.4.3	Extracting Shock Waves Occurring at Fixed Bottlenecks	50
3.4.4	Sensitivity Analysis	51
3.5	Extracting Shock Wave Speeds from Synthetic Data	52
3.5.1	Experiment 1	53
3.5.2	Experiment 2	54
3.5.3	Experiment 3	54
3.5.4	Discussion	55
3.6	Conclusion	56
4	Fast Freeway Traffic State Estimation	57
4.1	Background and Approaches of Traffic State Estimation	58
4.2	Optimizing the Adaptive Smoothing Method	59
4.2.1	Methodology of the Adaptive Smoothing Method	59

4.2.2	The Adaptive Smoothing Method Solved with the Cross-correlation	61
4.2.3	The Adaptive Smoothing Method Solved with the Fast Fourier Transform	65
4.2.4	Experimental Setup to Compare the Three Implementations	67
4.2.5	Results and Discussion	68
4.2.6	Conclusion of the Optimization of the Adaptive Smoothing Method	69
4.3	Optimizing Traffic State Estimation Based on the Extended Kalman Filter	70
4.3.1	Methodology of Extended Kalman Filtering	71
4.3.2	The Localized Extended Kalman Filter	74
4.3.3	Experimental Setup to Compare the Localized with the Global Extended Kalman Filter	79
4.3.4	Results and Discussions	80
4.3.5	Application of the Localized Extended Kalman Filter to a Real-size Network	82
4.3.6	Conclusion of the Optimization Based on the Extended Kalman Filter	83
4.4	Comparison of the Characteristics of the Adaptive Smoothing Method and Localized Extended Kalman Filtering	83
4.5	Conclusion	86
5	Multi-class Flow Analysis and Modeling	89
5.1	Modeling Multi-class Traffic with Fastlane	90
5.1.1	Basics of Macroscopic Traffic Flow Modeling	90
5.1.2	Multi-class Macroscopic Traffic Flow Modeling	91
5.1.3	Dynamic PCE Value	93
5.2	Discretization of Fastlane	93
5.2.1	Link Model	94
5.2.2	Node Model	94
5.3	Performance Indicators of Network Traffic	96
5.3.1	Definition of Total Time Spent and Total Cost	96

5.3.2	Factors Contributing to the Total Cost	96
5.4	Analysis of Class-specific Properties on Network Traffic Flow	97
5.4.1	Effects of Class-specific Properties on Throughput at an Active Bottleneck	98
5.4.2	Effects of the Class-specific Properties on Spillback	99
5.4.3	Effects of the Class-specific Properties on the Total Cost in Free Flow	102
5.5	Conclusion	107
6	Multi-Class Traffic Control Concepts	111
6.1	Conceptual Framework of Multi-class Traffic Control	112
6.2	Vehicle-class Specific Lanes	113
6.2.1	Layout	113
6.2.2	Experimental Setup	114
6.2.3	Results	115
6.2.4	Conclusion	118
6.3	Multi-class Ramp Metering	118
6.3.1	Layout and Implementation	118
6.3.2	Experimental Setup	121
6.3.3	Results	122
6.3.4	Conclusion	125
6.4	Multi-class Route Guidance	125
6.4.1	Layout and Implementation	125
6.4.2	Experimental Setup	126
6.4.3	Results	130
6.4.4	Conclusion	138
6.5	Conclusion	139

7	Case Study: Multi-class Control of the Dutch Freeway A15	141
7.1	A15 Site Description	142
7.1.1	Network of the Site	142
7.1.2	Recurring and Nonrecurring Traffic Problems on the Site . . .	143
7.1.3	Data Sources for Estimation and Calibration	143
7.1.4	Multi-class DTM Measures Used in the Case Study	144
7.1.5	The System BOS-HbR Applied to the A15	145
7.2	Setup of Estimation Component	145
7.3	Setup and Calibration of the Prediction Component	147
7.3.1	The Network Model	148
7.3.2	Model of the User Classes and Their Flow Through the Network	149
7.3.3	Calibration of Inflows and Turnfractions	149
7.3.4	Calibration of the PCE Function	154
7.3.5	Calibration of the Fundamental Diagram at Non-bottleneck Locations	154
7.3.6	Calibration of the Fundamental Diagram at Active Bottlenecks	155
7.4	Validation of the Prediction Component	157
7.4.1	Validation During Regular Congestion	157
7.4.2	Validation During Incidental Conditions	158
7.4.3	Discussion of Validation	159
7.5	Setup of the Control Component	160
7.6	Setup of the Case Study	162
7.7	Results of the Case Study	162
7.7.1	Results of Multi-class Control Under Regular Conditions . . .	162
7.7.2	Results of Multi-class Control Under Incidental Conditions . .	163
7.8	Discussion of the Performance	166
7.8.1	Analysis of the Performance	166
7.8.2	Potential Approaches for Improving the Performance	169
7.9	Discussion of the Computation Time	171
7.9.1	Analysis of the Computation Time	171
7.9.2	Potential Approaches for Improving the Computation Time . .	171
7.10	Conclusion	175

8	Conclusions and Recommendations	177
8.1	Main Findings and Conclusions	178
8.2	Recommendations for Further Research	180
8.2.1	Traffic State Estimation	180
8.2.2	Traffic State Prediction	181
8.2.3	Traffic Control	181
8.3	Recommendations for Practical Use	183
8.3.1	Traffic State Estimation	184
8.3.2	Traffic State Prediction	184
8.3.3	Traffic Control	185
8.4	Workflow Towards Implementation of Multi-class Dynamic Traffic Management in Practice	185
8.5	Application of BOS-HbR in Practice	187
	Bibliography	189
	Appendices	199
A	Calculation of the Class-specific Turnfraction on the Main Route	201
B	Conversion of Traffic Data to Initialize Fastlane	203
B.1	Problem Description	203
B.2	Solution Procedure	203
B.3	Proof	204
	Summary	207
	Samenvatting	209
	Zusammenfassung	211
	TRAIL Thesis Series	213
	About the Author	215

List of Figures

1.1	Operating room of the Traffic Management Center Rhoon	2
1.2	The control loop for Dynamic Traffic Management (DTM)	3
1.3	Structure of the thesis	7
2.1	Optimal control	13
2.2	Overview of traffic control approaches	16
2.3	Effects of congestion spillback	18
2.4	Dissolution of congestion	19
2.5	Comparison of spacings of vehicle classes	20
2.6	Spacing of vehicle classes – dynamic pce value	20
2.7	Overview of traffic flow predictions models	25
2.8	Overview of traffic estimation models	30
2.9	Structure of the Estimation component	31
2.10	DTM measures considered in this thesis	32
2.11	Control loop for multi-class Dynamic Traffic Management	34
3.1	Shock waves observed in spatiotemporal freeway traffic data	38
3.2	Relation between characteristic wave speeds and the fundamental diagram	40
3.3	Structure of the Wave Speed Estimator	42
3.4	Intermediate results of the Wave Speed Estimator for the case of shock wave speeds at stop-and-go waves	43
3.5	Example of edge detection in one dimension	44
3.6	Relation between Cartesian $x-t$ image and its Hough transform to polar coordinates	46

3.7	Frequency of wave speeds of lines detected	48
3.8	The freeway A13 near Delft, The Netherlands	49
3.9	Results of the Wave Speed Estimator applied to empirical data	50
3.10	Results of extracting shock waves occurring in free flow	51
3.11	Results of extracting shock waves at fixed bottlenecks	52
3.12	Results of Experiment 1	53
3.13	Results of Experiment 2	54
3.14	Results of Experiment 3	55
4.1	Input and output speed data of the ASM	68
4.2	Computation time of the ASM against the number of filter points	69
4.3	Computation time of the ASM against the varied variables	70
4.4	Structure of the Extended Kalman Filter	72
4.5	Error covariances under different conditions	76
4.6	Principle of the Localized-EKF	77
4.7	Experimental network on which the Localized-EKF was verified	79
4.8	Results of the Localized EKF: density patterns	80
4.9	Comparison of the different filters	81
4.10	Freeway network around the city of Rotterdam	82
4.11	Computation times of the Localized EKF	82
5.1	Elements of a traffic network modeled in Fastlane	90
5.2	Example of a fundamental diagram in Fastlane	92
5.3	Typical pce function in Fastlane	93
5.4	Node types in Fastlane	95
5.5	Setup for the analysis of spillback	100
5.6	Example network for the analysis of the total cost in free flow	102
5.7	Total cost of the base case	104
5.8	Total cost when increasing the length of the longer route	105
5.9	Total cost when lowering the inflow	106
5.10	Total cost when decreasing the free speed cars	107

5.11	Total cost when ignoring trucks dynamics: $\pi_{\text{truck}} \equiv 1$	108
5.12	Total cost when increasing truck's value of time	109
5.13	Overview of the contributing factors to the total cost and the optimal prioritization of the classes	109
6.1	Conceptual framework of multi-class traffic control	112
6.2	Vehicle-class specific lanes	114
6.3	Results of the trucklane experiments: total monetary flow	117
6.4	Physical layout of ramp meters	119
6.5	Implementation of multi-class ramp meter in Fastlane	120
6.6	Experimental setup of multi-class ramp meter	121
6.7	Simulation results of multi-class ramp metering	123
6.8	Simulation results of multi-class ramp metering (cont.)	124
6.9	Layout of a multi-class route guidance controller	126
6.10	Experimental setup of multi-class route guidance	127
6.11	The model-predictive control loop	129
6.12	Results of multi-class route guidance	131
6.13	Difference of total vehicle count at northern exit between MC MPC and SC MCP	132
6.14	Performance of the four route-guidance controllers for varying incident strengths	133
6.15	Control signals of the multi-class MPC for different values of time and different incident strengths	135
6.16	Results of multi-class MPC for different incident strengths	136
6.17	Results of multi-class MPC for different incident strengths (cont.)	137
7.1	The network of the A15 site for the case study	142
7.2	Truck percentage of the A15	143
7.3	Results of the Estimation component	146
7.4	Network model of the site	148
7.5	Modeling of destination classes for rerouting	149
7.6	Detector configuration for calibrating inflow and turnfraction	150

7.7	Example of calibrated total inflow and total turnfraction for the case study	152
7.8	PCE function used in case study	154
7.9	Example of a successful capacity calibration	156
7.10	Example of a failed capacity calibration	157
7.11	Validation of the Prediction component during regular conditions . . .	158
7.12	Performance of the validation	158
7.13	Validation of the Prediction component during incidental conditions .	159
7.14	Performance and computation time of the controller dependent on the lengths of the prediction horizon and the control horizon	161
7.15	Results of mixed-class control during regular conditions	164
7.16	Results of multi-class control under regular conditions	164
7.17	Results of no control under incidental conditions	165
7.18	Results of mixed-class control at an incident	166
7.19	Results of multi-class control at an incident	167
7.20	Comparison of performance during incidental conditions	168
7.21	Average computation time of the Control component	171
8.1	Screenshot of BOS-HbR Website	187

List of Tables

2.1	Overview of prediction models and their capabilities to reproduce traffic flow phenomena	26
4.1	Simulation parameter settings for comparing the ASM implementations	67
4.2	Speedup factor of the FFT implementation of the ASM	71
4.3	Error measurements of the ASM implementations	72
4.4	Qualitative comparison of characteristics between Adaptive Smoothing Method and Localized Extended Kalman Filter	84
6.1	Ramp meter policies applied in both experiments	122
6.2	Rerouting of the vehicle class by the multi-class MPC dependent on the incident strength	138
7.1	Sensors providing traffic data of the site	144
7.2	Calibration parameters of the Predictions component of the case study	147
7.3	User classes in the case study	150
7.4	Calibration of the user-class specific inflows and turnfractions the case study	153
7.5	Results of the case study	162

Notation

x	space
t	time
u	class
v_u	speed of class u (in $\frac{\text{km}}{\text{h}}$)
π_u	passenger-car equivalent (pce) function of class u (in $\frac{\text{pce}}{\text{veh}}$)
$\pi_u(v_u)$	passenger-car equivalent (pce) value of class u at given speed (in $\frac{\text{pce}}{\text{veh}}$)
h	rear-bumper-to-rear-bumper time headway between two vehicles (in s)
r	rear-bumper-to-rear-bumper spacing between two vehicles (in m)
q_u	(vehicular) flow of class u (in $\frac{\text{veh}}{\text{h}}$)
Q_u	effective flow of class u (in $\frac{\text{pce}}{\text{h}}$)
Q_{tot}	total effective flow (in $\frac{\text{pce}}{\text{h}}$)
k_u	(vehicular) density of class u (in $\frac{\text{veh}}{\text{km}}$)
K_u	effective density of class u (in $\frac{\text{pce}}{\text{km}}$)
K_{tot}	total effective density (in $\frac{\text{pce}}{\text{km}}$)
η_u	share of class u in traffic (in %)
ζ_u	value of time of class u (in $\frac{\text{€}}{\text{vehh}}$)
T	total time spent (in vehh)
$T^{\text{€}}$	total cost (in €)
v_u^{free}	fundamental diagram: free-flow speed of class u (in $\frac{\text{veh}}{\text{km}}$)
C	fundamental diagram: (effective) capacity (in $\frac{\text{pce}}{\text{h}}$)
K_{crit}	fundamental diagram: critical (effective) density (in $\frac{\text{pce}}{\text{km}}$)
K_{jam}	fundamental diagram: (effective) jam density (in $\frac{\text{pce}}{\text{km}}$)
c	propagation speed of a shock wave (in $\frac{\text{km}}{\text{h}}$)

Chapter 1

Introduction

Many freeways are highly congested on a daily basis. One reason is that traffic demand is simply too high for the current infrastructure so that congestion always arises at the same bottlenecks during rush hours. A second reason is incidents, where the freeway is partially or totally blocked due to accidents.

Congestion caused by recurrent bottlenecks can be solved by reconstructing the infrastructure to increase the number of lanes or to create new freeways. Those measures, however, take years to come into effect and are above all very expensive. Furthermore, long-term impacts include induced traffic demand, which damp the expected reduction of congestion. Alternatively, travelers can be incentivized to travel by another mode of transportation or to travel outside of the rush hour, for instance by road pricing. These mobility measures have a more immediate effect than reconstructing the infrastructure.

A solution to reduce congestion caused by either recurrent bottlenecks or incidents with a faster effect is to intervene by controlling the traffic flow. Traffic management centers like the one in Figure 1.1 have been established to monitor the traffic conditions of a freeway network. If traffic conditions deteriorate, the traffic management center can influence traffic by activating control measures. Such approaches are known as Dynamic Traffic Management (DTM).

Multiple DTM measures have been developed and applied successfully over the past decades. A common example is route guidance, which informs travelers about incidents on their way and suggests alternative routes around the incident. Another example are ramp meters, which are installed on on-ramps to the freeway; they restrict the inflow into the freeway to prevent the freeway traffic from breaking down into congestion. A further example is the peak-hour lane, which is an extra lane that is opened for traffic during rush hour.

The current practice of DTM is that it operates on the traffic flow as a whole, disregarding the different vehicle classes of which it is composed. Vehicles can be categorized along different properties, such as the vehicle length, the maximum speed, the acceleration capabilities, the number of passengers, the value of time, the emissions, the



Figure 1.1: Operating room of the Traffic Management Center Rhoon in the Netherlands; source: Inter Visual Systems (2012)

destination and many more. Some of these class-specific properties have a direct effect on the traffic flow of the network: given the same speed, short vehicles can travel with a shorter (rear-bumper to rear-bumper) distance headway than long vehicles, leading to a higher throughput. Other properties do not directly affect the flow of traffic, but have an effect on the performance of it. The value of time, for example, is not directly related to traffic flow, but it is an important factor when assessing the network performance in terms of the costs congestion induces on delayed vehicles.

This thesis expands DTM to take vehicle-class specific properties into account. The effects of different vehicle classes on traffic flow are analyzed and vehicle-class specific control strategies are developed and tested in simulations to improve the traffic both in regular and incidental conditions.

1.1 Problem Definition

Dynamic Traffic Management applies the classical control loop as illustrated by Figure 1.2. The traffic system under consideration is observed by sensors that provide traffic data to the traffic management center. The traffic management center uses these data to determine appropriate signals for the DTM measures like ramp meters, route guidance measures or similar controllable devices that influence the traffic system. The traffic management center performs this task in two parts. First, the current traffic is estimated based on the sensor data. Second, suitable control signals are computed based on the current traffic state. In current practice, these signals are determined by control scenarios, which are usually expressed as simple rules or switching schemes. In scientific work, more advanced control concepts have been developed that determine the control signals proactively by employing a traffic flow model which predicts the traffic state ahead.

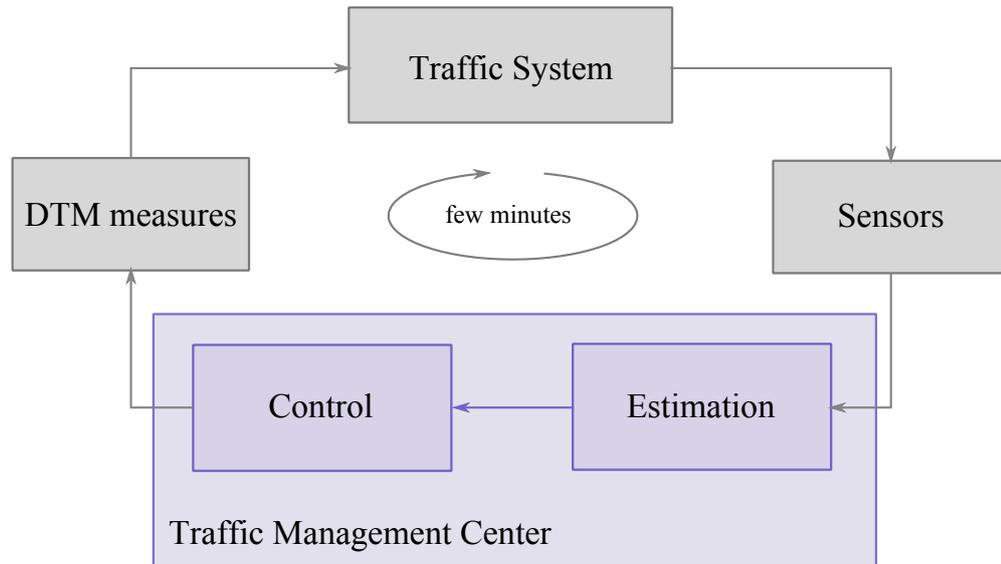


Figure 1.2: The control loop for Dynamic Traffic Management (DTM)

The aim of this thesis is to expand the control loop of Figure 1.2 to a DTM framework that takes multiple vehicle classes into account in order to improve the performance of the traffic system. Besides being able to exploit the specific properties of each vehicle class, the computation of the control signals has to be fast in order to react quickly to the current traffic conditions. For practical applications, a cycle of the control loop therefore has to be performed within a few minutes.

To achieve this objective, the following research questions are addressed:

1. What methods are suitable in the components of the control loop in order to perform multi-class DTM?
2. How can the current traffic state of a realistically-sized freeway network be estimated quickly and accurately for each vehicle class?
3. How can features of spatio-temporal traffic data be extracted in order to calibrate traffic state estimators and traffic flow models?
4. How can multi-class traffic flow be modeled mathematically so that it satisfies key requirements from traffic flow theory, and what are the effects of the class-specific properties on traffic flow?
5. How can existing DTM measures be expanded to multiple vehicle classes and what are their effects on traffic flow?
6. What is the benefit of multi-class DTM compared to conventional, mixed-class DTM for a realistic freeway network?

1.2 Scope of the Thesis

This thesis deals with freeway traffic, i.e. traffic flows in only one direction, traffic is not hindered by crossings on the same level, and vehicles can easily overtake each other in light traffic conditions. The networks discussed in this thesis are assumed to be of realistic size, i.e. in the order of tens of kilometers; the computation times stated in this thesis therefore refer to practical applications. Furthermore, the time frame of DTM is short-term, i.e. the actions of the DTM measures affect the traffic system immediately and last a few hours at most. This thesis regards only traffic itself and does therefore not address mobility or demand management. The focus is furthermore on the effects of multi-class DTM on the traffic flow; topics like implementation, hardware, maintenance or enforcement of the DTM measures are beyond the scope of this thesis.

In this thesis, the class-specific properties of vehicle length, maximum speed and value of time are used. Many vehicle classes can be modeled by these means; in this thesis, most experiments are performed regarding the two classes cars and trucks. Cars are short, have a high maximum speed, and have a low value of time; whereas trucks are long, have a low maximum speed, and have a high value of time. The exact values will be named later in the experiments.

1.3 Scientific Contributions

This thesis contributes to science in the following ways:

- *A Framework for multi-class Dynamic Traffic Management.* Currently, DTM focuses on controlling traffic as a whole, disregarding the vehicle classes of which it is composed. The framework that is developed shows how the components of the current control loop of Figure 1.2 have to be expanded in order to take multiple vehicle classes into account. (Chapter 2)
- *A novel method to automatically extract shock waves and their propagation speeds from spatiotemporal traffic data.* Shock waves characterize the transition between two traffic states. Furthermore, both traffic regimes are characterized by a shock wave speed, which is a key characteristic of the fundamental diagram. A method is developed that automatically extracts shock wave speeds from spatiotemporal data. It supports the calibration of traffic models that employ a part of the fundamental diagram, or support the evaluation of the performance of traffic flow models that aim at reproducing shock waves propagating with correct speeds. (Chapter 3)
- *The reformulation of two traffic state estimators so that they compute the traffic state of a realistically sized freeway within a few seconds.* Firstly, it is shown that

the Adaptive Smoothing Method by Treiber et al. (2011) can be reformulated to apply the Fast Fourier Transform which solves the model significantly faster. Secondly, the Extended Kalman Filter can be reformulated to apply measurements only locally, which omits expensive calculations that have only negligible effects on the result. (Chapter 4)

- *Insights into the effects of vehicle-class specific properties on the total cost of the network traffic.* It is shown that the network performance in the form of the total cost, which is the sum of the individual travel times weighted by the value of time, is mainly determined by the throughput at bottlenecks, the spillback of congestion, and the free-flow travel time. All of them are affected by the vehicle length and the maximum speed of each vehicle class. (Chapter 5)
- *The generalization of conventional, mixed-class DTM measures to multiple vehicle classes.* Ramp meters and route guidance measures are typical examples of DTM measures that are currently applied mainly as mixed-class measures. The effects of their multi-class counterparts are analyzed. Furthermore, the effects of a trucklane is discussed. (Chapter 6)
- *Insights into the potential of the multi-class DTM framework by applying it to simulations of the Dutch freeway A15.* In a case study, the traffic state of the freeway is estimated by the methods developed in Chapters 3 and 4, and traffic is controlled by a multi-class ramp meter and two multi-class route-guidance measures developed in Chapter 6. The performance improvements of multi-class DTM with respect to mixed-class DTM is shown. (Chapter 7)

1.4 Practical Contributions

This thesis contributes to practice in the following ways:

- *A method to automatically identify shock wave speeds from spatiotemporal traffic data plots.* This tool can be used to support traffic state estimators that are parameterized by characteristic shock wave speeds, to estimate parts of the fundamental diagram, and to evaluate the predictive performance of traffic flow models. (Chapter 3)
- *Development of traffic state estimators that run efficiently within a few seconds for real-sized freeways.* Two existing traffic state estimators are reformulated. The validation shows that both compute the current traffic state within a few seconds and are therefore applicable for DTM. (Chapter 4)
- *Generalization of current state-of-the-practice DTM measures to multiple vehicle classes.* Ramp meters and route-guidance measures are generalized to distinguish between multiple vehicle classes, and their positive effects on the traffic performance are shown. (Chapter 6)

- *Development of the prototype software BOS-HbR, which implements the multi-class DTM framework to the Dutch A15 eastbound near the harbor of Rotterdam.* All components of the control loop are implemented in Matlab and applied to traffic data from the freeway, which are gathered in real-time. This prototype is called “Beslissingsondersteunend Systeem voor het Havenbedrijf Rotterdam” (BOS-HbR, in English.: “Decision Support System for the Port of Rotterdam Authority”). The following three points name its details.
- *A traffic state estimator for the Dutch A15 in real-time.* The traffic state estimators developed in Chapters 3 and 4 are combined and applied to the traffic data gathered from the A15; they estimate the traffic state within one minute. (Chapter 7)
- *A validated multi-class traffic flow model for the Dutch A15.* The multi-class traffic flow model is validated so that is capable of predicting the traffic state one hour ahead, given the current traffic state of the freeway. (Chapter 7)
- *An optimal controller for multi-class DTM of the Dutch A15 in order to improve the traffic performance.* Based on the current traffic state and the prediction, control signals are computed by applying the multi-class DTM framework developed in Chapter 2. (Chapter 7)
- *The outline of a workflow plan to move towards practical implementation of the prototype BOS-HbR.* The prototype is currently implemented in Matlab with the goal to show a proof of concept of multi-class DTM. In order to apply BOS-HbR in practice, a path towards practical implementation for robust multi-class control is set up. (Chapter 8)

1.5 Structure of the Thesis

The remainder of this thesis is structured as shown in Figure 1.3.

Chapter 2 refines the control loop outlined in Figure 1.2 in order to be applicable in multi-class DTM. For each component of the control loop, applicable methods are reviewed and a suitable method is chosen.

The two chapters thereafter are concerned with the Estimation component. Chapter 3 develops a tool that automatically extracts features from spatiotemporal traffic data. Specifically, the tool estimates how fast the transitions between traffic states, so-called shock waves, propagate.

Chapter 4 analyzes two existing traffic state estimation methods and reformulates them to estimate the traffic state within a few seconds. The first method is the Adaptive Smoothing Method, which will be solved by the Fast Fourier Transform. The second

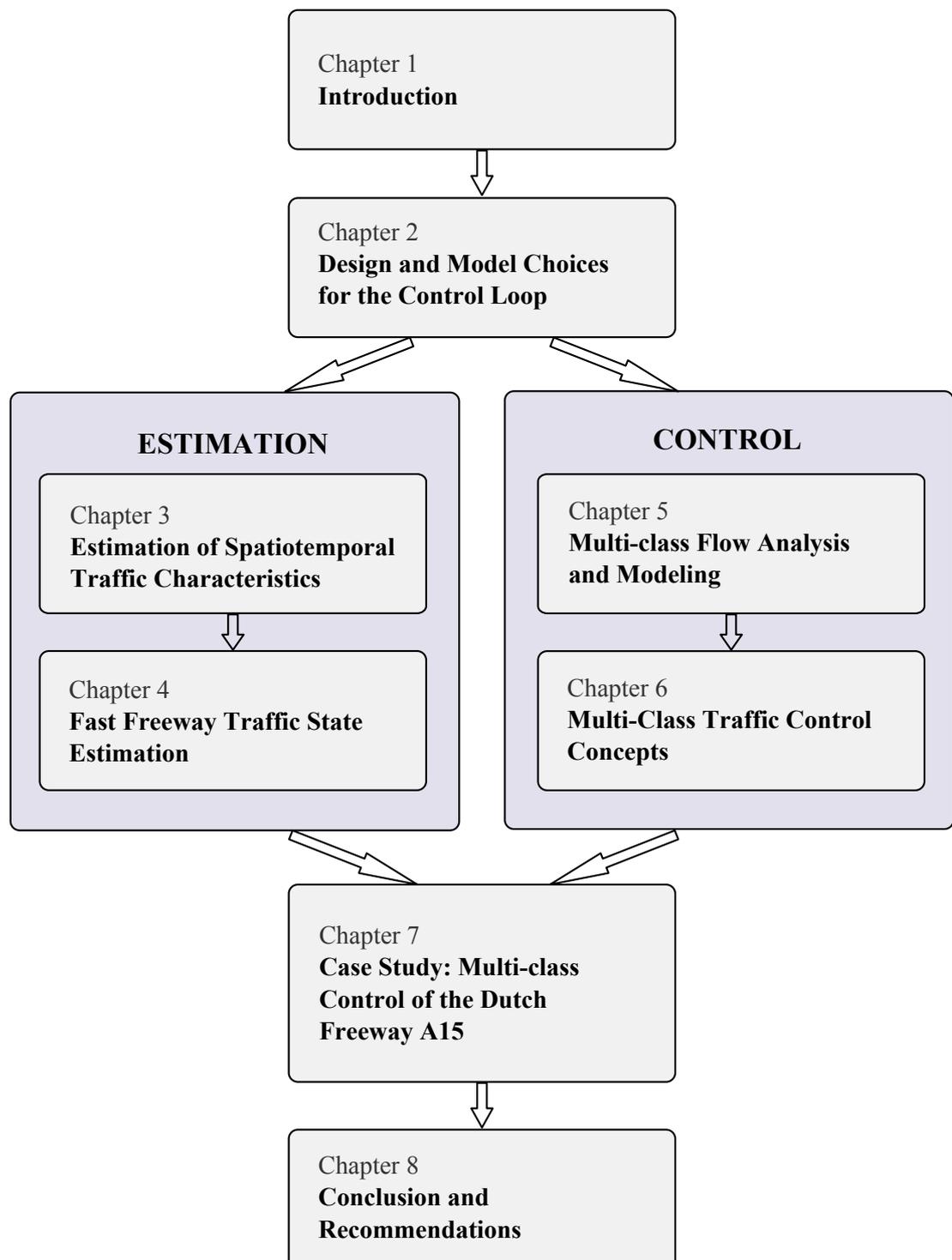


Figure 1.3: Structure of the thesis

method is based on the Extended Kalman Filter, which is reformulated to apply the sensor data only locally.

The following two chapters then deal with the Control component. Chapter 5 presents a traffic flow model that takes the properties of different vehicle classes into account. Based on this model, called Fastlane, the effects of the vehicle properties on the network performance in the form of the total cost of traffic are analyzed.

Chapter 6 generalizes existing DTM measures to multi-class DTM measures. The effects of a trucklane, of multi-class ramp metering and of multi-class route-guidance on the traffic performance are shown by using the multi-class traffic flow model.

Chapter 7 combines the estimation and control concepts developed in the preceding chapters to the multi-class DTM prototype BOS-HbR. In a case study, the Dutch freeway A15 eastbound near the harbor of Rotterdam, the Netherlands, is simulated and multi-class DTM measures are coordinated to improve the traffic conditions in both regular and incidental conditions.

Chapter 8 presents conclusions and recommendations for further research and practice. In addition, a workflow towards practical implementation and the online version of BOS-HbR that can serve as the basis of multi-class DTM are presented.

Chapter 2

Design and Model Choices for the Control Loop

In this chapter, we motivate the major design and model choices for the components of the control loop introduced in Figure 1.2. This chapter is not intended as a state-of-the-art review for all components, but to underpin the scope of this research and to motivate the choices made. These choices are made on the basis of scientific curiosity, theoretical and application-specific requirements and in some cases practical arguments. For many components, details are provided in the ensuing chapters. This implies that in this chapter, we make forward references to the ensuing chapters in this thesis.

Section 2.1 discusses suitable control methods for the analysis of multi-class Dynamic Traffic Management. In order to be able to predict the effects of multi-class DTM measures, the control method will contain a predictive element. Section 2.2 therefore discusses suitable prediction models that reproduce both essential traffic phenomena occurring on freeways and the effects of multi-class DTM measures. In Section 2.3, estimation models are discussed to determine the current traffic state on the freeway. An overview of the DTM measures and traffic sensors used in this thesis are given in Sections 2.4 and 2.5, respectively. Conclusions including a refined version of the control loop for multi-class Dynamic Traffic Management are presented in Section 2.6.

2.1 Design Choices for the Control Component

This section reviews control methods in order to analyze the effects of multi-class traffic control on the network performance. First, we define the requirements of the Control component of the control loop of Figure 1.2. Two of the basic requirements are that the controller is both effective, i.e. it improves the performance of the network, and fast, i.e. it the control signals are computed quickly so that the controller is applicable in real time. Subsequently, we discuss how common control approaches meet these requirements. We then conclude that a model-predictive control is the most suitable approach to analyze multi-class Dynamic Traffic Management, since it is able to optimize the network performance given the new multi-class DTM measures that will be developed in Chapter 5.

2.1.1 Requirements of the Control Component

Several control methods have been developed in the past. In order to choose a suitable one for this thesis, we first discuss the requirements of the control component. There is a long range of requirements which can be considered, among others that the controller should be effective, fast, easy to develop, easy to deploy, easy to maintain, easy to use, inexpensive, flexible, intuitive and sustainable (Munroe, 2012). In the following, we will select the most important criteria.

The scope of this thesis is to develop and analyze multi-class control of freeways. The basic requirements of any control method is therefore to be able to handle multi-class traffic and multi-class DTM measures. Furthermore, the scope of the controller is a freeway network in the order of tens of kilometers over a horizon of a few hours.

In order to apply Dynamic Traffic Management, the controller has to compute the control signals fast. For practical DTM applications in real-time, this leads to a computation time in the order of a few minutes.

Many freeways are subject to two types of traffic conditions. Under regular conditions, traffic congestion arises practically at the same location every day. Irregular conditions, however, can cause very different congestion patterns. For example, an incident can temporarily create a new bottleneck which results in a unique congestion pattern. Another example is a drastic change in the traffic demand pattern. The controller must be flexible to work under both regular and incidental traffic conditions.

The final requirement is that the controller be flexible to adapt to new infrastructure. For instance, freeways can be reconstructed to increase the number of lanes. During construction, the network layout will change often; among others the location of the bottlenecks can change multiple times. Another example is that new DTM measures can be installed. The controller thus must be able to adapt to a new network layout or to changing DTM measures.

In summary, since the focus of this thesis is to analyze the effect of multi-class control on traffic flow and the network performance, we require a multi-class network-wide freeway control method that both computes the control signals quickly and is flexible so that it works under regular and irregular traffic conditions and under changing infrastructure.

2.1.2 Control Methods

This section discusses control methods for multi-class DTM that are fast and flexible to various extents. The following methods are not entirely disjoint, so some of them can be combined together. For each method, we define how the approach works, name examples, and discuss its computational speed and its flexibility.

2.1.2.1 Rule-based Reactive Control

Approach Rule-based controllers set the control signals by applying predefined rules, usually in the form of simple “if-then” statements. In the reactive version, the rules are applied to the current traffic state.

Examples An example of a reactive controller is the coordinated ramp meter algorithm HERO by Papamichail & Papageorgiou (2008), which sets the outflow of the ramp dependent on the current queue length of the other ramps. The rule thereby is to keep the queues of each ramp at equal length. Another coordinated ramp meter controller is ACCEZZ by Bogenberger et al. (2001). Its rules were derived from extensive ex ante simulations with a traffic flow model; the rules were computed by applying a neural network and genetic algorithms.

Wahle et al. (2000) simulated a route guidance controller that advised the fastest route based on measured travel times. They showed that such a controller leads to oscillations in the travel times and to an underutilization of traffic.

Computational Speed By only applying predefined rules, the computational speed of a rule-based controller is very high.

Flexibility A drawback is, however, that the rules have to be defined before the controller is deployed. Moreover, rules have to be defined for every traffic situation that can possibly arise. Incidents can occur at many different locations and times with different strengths, so that their effects on the network can vary widely. The number of rules that have to be developed and validated can become very large. A further drawback for some of the methods is that when the infrastructure changes, a new set of rules has to be developed, which makes a rule-based approach difficult to maintain over a longer period.

2.1.2.2 Case-based Reactive Control

Approach A case-based reactive controller employs set of cases, which describe which control actions to take give a specified traffic state. The current traffic state is then compared to the case base, and the case that matches the current traffic state best is selected.

Examples A case-based system is the BOSS Scenario Evaluation System (BSES) for the ringroad of Amsterdam developed by Hoogendoorn et al. (2003) and De Schutter et al. (2003). The network is split into subnetworks, whereby each subnetwork is an agent. Hodge et al. (2011) developed a case-based decision support system where the cases are compared by a pattern matching algorithm. Among the cases which closely match the current conditions, the one that leads to the highest performance is advised. Almejalli et al. (2007) developed a decision support system based on fuzzy inference. The fuzzy membership functions are trained from previous cases and adapted with each new case.

Computational Speed Since this approach has to match the current traffic state to an appropriate case in the case base, the computation time is higher than rule-based controllers.

Flexibility Case-based controllers can interpolate between existing cases to match the current traffic state to the closest case. They are therefore more flexible than rule-based systems. However, for very different traffic situations or new infrastructure, new cases have to be developed

2.1.2.3 Rule-based and Case-based Predictive Control

Approach The rule-based and case-based controllers of the two previous sections react only to the current traffic state. These approaches can be extended so that the rules and cases apply to the future traffic conditions. By this means, negative effects can be anticipated beforehand and appropriate DTM measures can be activated to counter these effects before they arise. In a network with multiple routes, for example, a route guidance measure can advise the route with the lowest expected travel time. To predict the future traffic conditions, a traffic flow model is applied.

Examples Wang et al. (2003) applied predictive control to guide vehicles via the fastest route, based on predicted arrival times.

Computational Speed The predictive rule-based and case-based controllers compute the control signals slower than their reactive counterparts, since the prediction model has to be applied in addition.

Flexibility Predictive methods are more flexible than reactive ones, as they predict the effects of the current traffic state. For example, they predict the congestion that is caused by an accident and can counter its negative effects. However, like their reactive counterparts, if the network infrastructure changes, the rules and cases have to be revised.

2.1.2.4 Optimal Control

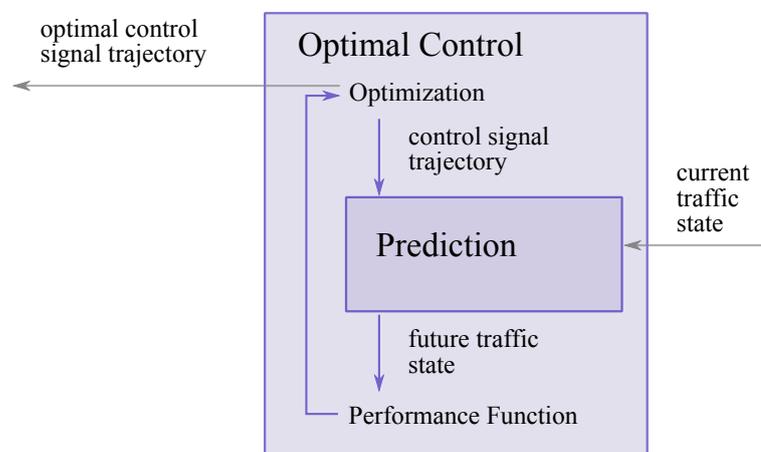


Figure 2.1: Optimal control

Approach A very different approach is optimal control. It predicts the effects of the DTM measures on future traffic state and optimizes their control signals in order to minimize a specified performance function (Figure 2.1). The road authority thereby can choose the performance function it want to minimize. Common functions are the total time spent, the travel time on route or the emissions.

Since optimal control analyzes the predicted future traffic state, it requires a prediction model that reproduces the effects of the DTM measures. Furthermore, the control signals not just for the current moment, but also for the near future, namely during the prediction horizon, are optimized; in other words, the result of optimal control is a control signal trajectory. The optimization procedure finds the global minimum of the objective function, which maps the control signals to a real number, the performance value. This objective function contains the traffic prediction model, and evaluates the performance of its prediction. Since the traffic prediction model is usually elaborate, a closed-form expression to determine the minimum of the objective function does not exist. Instead, the minimum is found by an iterative approach. Due to these iterations,

the prediction model will be executed multiple times, until the minimum is found. The point of the minimum is the optimal control signal trajectory.

The whole trajectory of the optimal control signals is applied to the DTM measures, i.e. the signals are not re-evaluated at a later point. Since there is no feedback, optimal control is mainly used for simulation purposes. If the feedback loop is closed, then it is part of model-predictive control, which will be discussed below.

Examples Kotsialos et al. (2002) and Papamichail et al. (2010) developed the Advanced Motorway Optimal Control (AMOC) system and applied it to the ramp meters of the ringroad of Amsterdam. They show that congestion can be completely prevented if all on-ramps are metered and coordinated. Carlson et al. (2010) applied optimal control to meter the mainstream of a freeway in order to prevent the capacity drop and a lane drop.

Computational Speed The computational speed is relatively low, since the prediction model is executed multiple times during the optimization procedure. Dependent on the number of DTM measures and the length of the control signal trajectory, the computation can take between several seconds and multiple hours. For practical applications, the computational speed has to be traded off against the number of DTM measures.

Flexibility A big advantage of optimal control is that it is flexible to changing conditions. If an incident occurs, its effects are directly accounted for in the optimization by the prediction model. Developing rules or cases for different incidents is therefore not necessary, which reduces the development time of an optimal controller compared to a rule-based approach. Furthermore, if the network changes, the prediction model has to be revised and re-calibrated.

A drawback of pure optimal control is that the control signals are not re-evaluated. If traffic conditions change, e.g. a congestion dissolves earlier than expected, the previously computed control signals might no longer be optimal.

2.1.2.5 Model-predictive Control

Approach Pure optimal control is applied only once and the control signals are not re-evaluated. This drawback is remedied by computing new control signal trajectories after a while by feeding the current traffic state back to the controller. This approach is called model-predictive control (MPC) or receding-horizon control. In each iteration of the control loop, an optimal controller optimizes the control signals of the DTM measures to achieve the optimal traffic performance.

Examples Hegyi et al. (2005) applied model-predictive control to coordinate ramp meters and variable speed limits to minimize the total time spent of a freeway.

Computational Speed Since optimal control is applied in every feedback iteration, the computation times of optimal control and MPC are the same.

Flexibility MCP exhibits the same advantages as optimal control, as irregular conditions can directly be taken into account and a changing infrastructure can be modeled in the prediction model. In addition, the feedback of the current traffic state leads to a frequent recalculation of the control signal trajectory, which enables a more robust performance than optimal control. Even if the real traffic behaves differently than expected by the prediction model, the new traffic state is used as the basis for the new signals so that prediction errors can be corrected.

2.1.2.6 Anticipatory Control

Approach A different approach is anticipatory control. Like the two previous approaches, it predicts the effects of the DTM measures. However, it further anticipates the reactions of the travelers to the DTM measures. For example, if a ramp meter is activated and causes a waiting queue to emerge, travelers may react and choose a faster route via a different on-ramp. Anticipatory control predicts and takes these reaction of the travelers into account.

Examples Taale & Hoogendoorn (2012) developed a framework for the Amsterdam area to coordinate meter installations on on-ramps and freeway junction.

Computational Speed Since the control signals and the travelers' reactions have to be computed, this control approach is slow.

Flexibility Due to taking the drivers' reactions into account, this approach is very flexible. For incidents, this approach might not be necessary, since drivers cannot adapt within a few minutes, but usually change their behavior on a day-to-day basis. Anticipatory control is thus useful for developing DTM strategies for commuting traffic, i.e. for regular traffic conditions.

2.1.3 Conclusions of the Control Component

The diagram in Figure 2.2 summarizes the control methods in terms of computational speed and flexibility. The fastest, but least flexible approaches are rule-based and

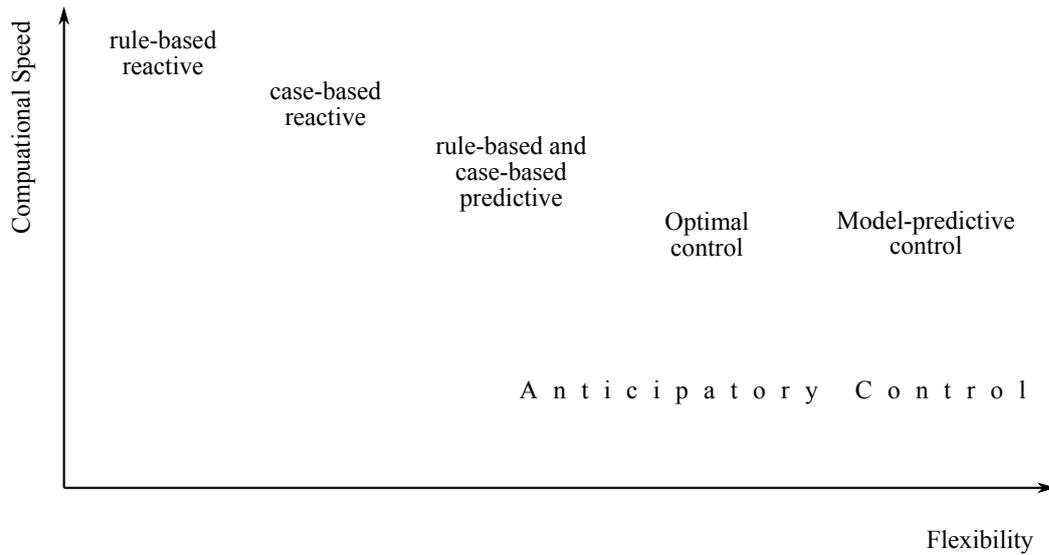


Figure 2.2: Overview of traffic control approaches

case-based controllers. Optimal control and model-predictive control both predict the effects of the traffic conditions and of the DTM measures. Due to the optimization procedure which executes a prediction model multiple times, optimal control and MPC are noticeably slower than the previously named approaches. However, their advantage is that they are flexible to incidents and changing infrastructure. Finally, anticipatory control also considers the expected reaction of the travelers, so its computational burden is even higher.

Some of the control approaches can be combined. The ramp meter controller AC-CEZZ (Bogenberger et al., 2001) is a rule-based system, whereby an optimal controller was used to develop and validate the rules offline. Similarly, anticipatory control can employ a model-predictive controller to optimize the control signals; the prediction model then also contains a model of the travelers' reactions.

Decision

The purpose of this thesis is to develop and analyze multi-class control for freeways that works both under regular and under incidental conditions, and can optimize the signals of the multi-class DTM measures that will be developed in Chapter 5. We therefore choose optimal control and model-predictive control as control methods, since they both predict the effects of the incidents and of the multi-class DTM measures and optimize the traffic performance. To keep the computational time low, especially for real-time applications, the number of DTM measures will be limited.

Model-predictive control is used in one of the experiments in Chapter 6. Its methodology is explained in Section 6.4.2.2. In the case study of Chapter 7, the prediction model is validated for a horizon of one hour. Then, model-predictive control based on the validated traffic flow model is used to optimize the traffic state of the Dutch A15

near the harbor of Rotterdam. A simpler version of the case study is currently running online (Section 8.5). Since the control signals are not applied to the actual traffic system, an optimal control approach computes the signals for this one hour and applies them to the simulated freeway. If operators of a traffic management center would apply the advised signals to real network, then the feedback of the whole control loop would be closed so that the approach would then be a full-fledged model-predictive controller.

As a remark, some of the DTM measures employ a local reactive controller. For instance, some ramp meter installations use an Alinea algorithm to determine the outflow of the ramp based on the traffic data gathered on the freeway. In fact, this resembles another control loop, namely within the DTM measures. The control approach in this thesis is therefore a hybrid one: the model-predictive control loop as global control, and the control loop within the DTM measures as local control. For the sake of brevity, the local control loop within the DTM is not shown in the figures.

Since optimal control and model-predictive control employ a Prediction component, a suitable prediction model is required. The next section reviews possible prediction models and chooses one.

2.2 Design Choices for the Prediction Component

This section compares traffic prediction models that can be used to forecast traffic for the Control component. First, the requirements for the Prediction Component are outlined. Then, existing traffic flow prediction models are discussed. Finally, we conclude that the macroscopic multi-class traffic flow model Fastlane is the most suitable one to predict the traffic conditions for the multi-class controller.

2.2.1 Requirements of the Prediction Components

The main goal of the Prediction component is to predict the expected traffic state for a short term, given the current traffic state and the control signal trajectory for the multi-class DTM measures. The prediction horizon in this thesis will be in the order of one hour.

In order to apply the controller in real-time, the prediction model has to compute the future traffic state fast. Since the controller optimizes the signals in an iterative approach, the prediction model is executed multiple times and therefore must run within a few seconds. Furthermore, in order to evaluate the effects of different control signals, the model outcome should only depend on the current traffic situation and the control signals, and should not exhibit a random behavior; in other words, it should be deterministic.

The traffic predictions of the model have to be valid in two ways. Firstly, the effects of the (multi-class) DTM measures have to be predicted correctly. Secondly, the model

must reproduce the important traffic flow phenomena that occur in practice. In the following, these phenomena are discussed.

Traffic Flow Phenomena

Congestion Probably the most important and also most commonly known phenomenon of traffic is congestion. It is characterized by a high traffic density and low speeds leading to low traffic performance such as high total time spent or low network throughput.

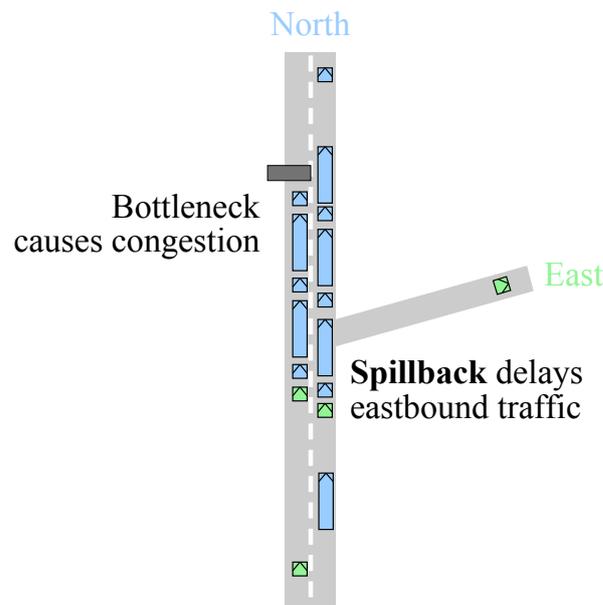


Figure 2.3: Effects of congestion spillback: spillback delays travelers (east) who do not want to pass the bottleneck (north) and therefore severely increases the total cost

When congestion emerges at a fixed bottleneck, a queue forms and spills back upstream. Figure 2.3 shows how the vehicles that head north (blue) and therefore want to pass the bottleneck have to queue up. However, since congestion spills back over the off-ramp that is located further upstream, vehicles that head east (green), and therefore do not want to pass the bottleneck, also have to enter the congestion. An immediate consequence is that the eastbound vehicles are now delayed, too, although they will never pass the actual bottleneck. A secondary consequence is that the congestion now grows even faster, since both the northbound and the eastbound vehicles have to queue. This effect increases the congestion spreading speed even more. Spillback is thus a self-perpetuating effect if it blocks upstream infrastructure.

Congestion dissolves in two different ways, dependent on the cause of its dissolution. If the traffic demand decreases, e.g. at the end of the rush hour, then the head of the congestion stays fixed at the bottleneck location, while the tail moves downstream until it hits the bottlenecks so that the congestion is dissolved (Figure 2.4(a)). Conversely, if

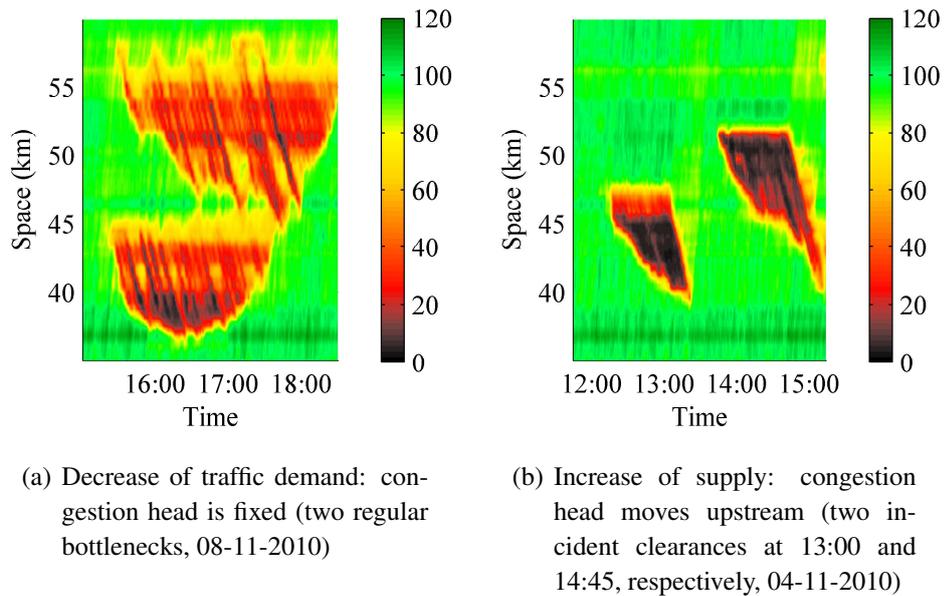


Figure 2.4: Dissolution of congestion, dependent on the cause (spatio-temporal speed plots of the Dutch A15R)

the traffic supply increases, e.g. when an incident clears, then the head of the congestion moves upstream until it hits the tail (Figure 2.4(b)). In the latter case, this effect can lead to congestion that is located far away from the original location of its cause.

When drivers leave congestion at its head, they slowly accelerate to their intended speed. This maneuver takes several seconds and can cover several hundred meters. Due to this slow transition of traffic states, the head of the congestion thus is not visible as a sharply defined line in the spatiotemporal speed plots. Furthermore, this acceleration out of congestion can leave gaps in the traffic, i.e. when drivers finally drive at their intended speed, the gap to their predecessor is unnecessarily large. In essence, the capacity of the road is not fully used. This phenomenon is called the capacity drop. Its strength varies, though its value is approximately ten percent (Chung et al., 2007).

Many drivers have experienced that they suddenly have to stop for no apparent reason and then can continue traveling after about a minute. They have just passed a so-called stop-and-go wave. This is a region of slow-traveling traffic which covers a few hundred meters and propagates upstream. Stop-and-go wave often occur in congestion (see Figure 2.4(a)), but they sometimes also emerge when the freeway is operating near capacity but is still in free flow.

Multiple vehicle classes Vehicles differ in a number of ways from each other. Firstly, they differ in their free speed capabilities. Cars can usually driver much faster large vehicle like trucks or buses. The travel time of cars is therefore lower than that of trucks.



Figure 2.5: Comparison of spacings of vehicle classes, (frame of a video taken at the Dutch A15R near on-ramp Charlois)

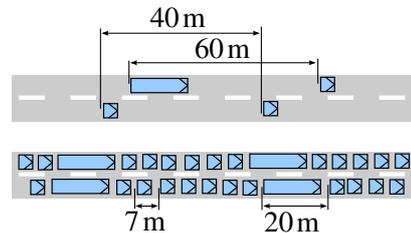


Figure 2.6: Spacing of vehicle classes: the spacing depends on the vehicle class and on the traffic state; top: in free flow, the pce value is low; bottom: in congestion, the pce value is high

Secondly, vehicles differ in their length. A truck is much longer than a car. This has a substantial effect on the spacing between cars, and thereby on the throughput of the road. Figure 2.5 shows the traffic of the Dutch A15 during the evening peak, which is mostly composed of trucks and cars. Since trucks are substantially longer than cars, they occupy much more space than a car. These differences in spacing led to the definition of the so-called passenger-car equivalent (pce) value

$$\pi_u = \frac{r_u}{r_{\text{car}}} \quad (2.1)$$

of a class u , which is the ratio between the spacing r_u of a vehicle of that class and the spacing r_{car} of a car in the same traffic conditions.

Furthermore, this pce value depends on the traffic state, as illustrated in Figure 2.6. The more congested the freeway is, the more relative space a truck occupies compared to a car. For comparison, in free-flow near capacity the spacing of a truck approximately equals the spacing of one and a half cars. In severely congested conditions, this ratio is one truck equals approximately three cars. Elefteriadou et al. (1997) provides an overview of typical pce values for truck under different traffic and road conditions.

A third important property of multi-class traffic is that vehicle-classes differ in their value of time. Although the value of time (VOT) does not directly affect the traffic flow, it is important for evaluating the performance of the traffic network if costs are an objective. For the Dutch road authority, for example, a truck is three times as valuable as a car (Rijkswaterstaat, 2011).

There are further categories to differentiate between vehicles, though these will only play a minor or no role in this thesis. The origin and the destination of the vehicles can differ. As we already showed previously, the destination can be exploited to prevent spillback (Figure 2.3). In the case study, will we differentiate vehicle also by their destination (Section 7.3.2). Another category is the exhaust gas or noise emission of a vehicle. Like the value of time, emissions do not directly affect traffic flow, but are an important measure if the goal to minimize the total emission of a network.

2.2.2 Prediction Models

This section discusses prediction models (or categories thereof) that can potentially be used in a model-predictive control approach for multi-class DTM. For each model, we presents its main principle, some examples its computational speed and the traffic phenomena it is capable of reproducing.

2.2.2.1 The Basic Macroscopic Traffic Flow Model: LWR

Model Macroscopic models represent traffic as a fluid which resembles the average behavior of traffic. The conservation of vehicles states that no vehicle can be destroyed or created. Mathematically, this is expressed by describing the traffic density k and traffic flow q over space x and time t by the partial differential equation

$$\frac{\partial k}{\partial t} + \frac{\partial q}{\partial x} = 0. \quad (2.2)$$

The LWR model by Lighthill & Whitham (1955) and Richards (1956) was the first macroscopic model developed. Although it is not a multi-class model, it is named here because of its importance for the models that are discussed thereafter.

The average traffic behavior is modeled by the fundamental diagram q^{FD} , which describes an equilibrium relation between the traffic flow q and the traffic density k :

$$q = q^{\text{FD}}(k) \quad (2.3)$$

Furthermore, the speed

$$v = \frac{q}{k} \quad (2.4)$$

in free-flow conditions is constant, and the flow linearly decreases with increasing density in congestion, i.e. the fundamental diagram is triangular. Acceleration and deceleration are infinite. With the assumption of flow maximization, the LWR model is uniquely solvable.

Computational Speed A discretized version of the LWR model exists in the form of the Cell Transmission Model by Daganzo (1994), which partitions the network into segments, commonly in the order of one hundred meters length. The computation is fast; usually much faster than microscopic models of similar size.

Phenomena The LWR model reproduces congestion correctly, including its emergence and dissolution. Acceleration is unrealistically high. The capacity drop is not reproduced. The propagation of stop-and-go waves is reproduced, though not their emergence.

2.2.2.2 Mixed-class Generalizations of the LWR Model

Model The LWR model has been expanded in several different ways. Many models generalize the fundamental diagram (2.3) to assume another shape. Higher-order models replace the flow maximization with a second partial-differential equation. Since vehicles do not appear or vanish inside of network, the conservation of vehicles (2.2) remains.

Examples Lebacque (2003) limits the acceleration of the traffic to a finite value. This enables the reproduction of the capacity drop. Payne (1971) replaces the flow maximization of the LWR model with a partial differential equation modeling the speed:

$$\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial x} = \frac{v^{\text{FD}}(k) - v}{\tau} + \frac{1}{2\tau} \frac{dv^{\text{FD}}}{dk} \frac{1}{k} \frac{\partial k}{\partial x}, \quad (2.5)$$

with a parameter τ . The Metanet model by Messmer & Papageorgiou (1990) is a discretization thereof and commonly used for simulations of traffic control.

Computational Speed Generalizations of the LWR model usually include more equations. This usually leads to a higher computational speed.

Phenomena Dependent on the generalization, different phenomena like the capacity drop or the emergence of stop-and-go waves in additions to the ones reproduced by the LWR model can be captured.

2.2.2.3 Multi-class Macroscopic Traffic Flow Models with Fixed PCE Value

Model Multi-class macroscopic models expand the LWR model to multiple vehicle classes. The conservation of vehicles (2.2) is generalized for each vehicle class u :

$$\frac{\partial k_u}{\partial t} + \frac{\partial q_u}{\partial x} = 0. \quad (2.6)$$

Furthermore, the speed can differ between classes so that the fundamental diagram is class-specific, too:

$$q_u = q_u^{\text{FD}}(K_{\text{tot}}), \quad (2.7)$$

whereby K_{tot} represents a traffic density aggregated over all user classes. Furthermore, the relative spacing between vehicle classes can differ. In this model, we assume that the ratio of the spacings between two vehicle classes, i.e the pce value π_u (2.1), is fixed. Later, we will discuss a model with a dynamic pce value.

Examples Multi-class models have been proposed for the case where only the speed of the vehicles differs (Wong & Wong, 2002; Logghe & Immers, 2008), and for the case of different vehicle lengths (Chanut & Buisson, 2003; Ngoduy & Liu, 2007). Hoogendoorn (1999) has proposed a lane-specific multi-class model.

Computational Speed The computational speed is lower than that of the LWR of comparable size, since the state of multiple classes and their interactions have to be calculated as well.

Phenomena Besides the phenomena of the LWR model, it also reproduces the traffic composition and different speeds per class. Macroscopic traffic flow models also have the potential to reproduce the capacity drop and the emergence of stop-and-go waves. Currently, however, no multi-class macroscopic model has been developed that is able to do so.

2.2.2.4 Multi-class Macroscopic Traffic Flow Models with Dynamic PCE Value

Model Macroscopic multi-class models can be extended to take the dynamics of the spacings between vehicle classes into account, i.e. the pce values $\pi_u(v_u)$ (2.1) are dynamic dependent on the traffic state. For example, a truck can equal three cars in congestion, but only one and a half cars in free flow.

Examples The model Fastlane by Van Lint et al. (2008b) contains a dynamic pce function which defines a pce value dependent on the traffic state.

Computational Speed The computational speed is slightly lower than other macroscopic multi-class models of same size, since the pce value has to be calculated as well.

Phenomena In addition to the phenomena reproduced by other macroscopic multi-class models, a model with a traffic-state dependent pce function also reproduces the vehicle spacings more realistically.

2.2.2.5 Microscopic Traffic Flow Models

Model In contrast to macroscopic models where traffic is modeled as a fluid, microscopic models represent each vehicle individually. A vehicle thereby reacts dependent on the vehicles in its vicinity – usually its predecessor for the longitudinal behavior and the those in the adjacent lanes for lane-changing behavior. Since every vehicle is modeled, microscopic models can easily represent multiple vehicle classes.

Many microscopic models are non-deterministic, i.e. they include some random behavior by employing random number generator. Consequently, the results of two different model runs with the same or similar input can lead to very different outputs. For example, congestion can emerge at different times. To achieve a representative result, a microscopic model is usually run multiple times and the results are then averaged.

Due to the detailed representation of traffic, the network has to be modeled in great detail. For example the shape and curvature of the on-ramps, or the slope of hills must be represented.

Examples Many microscopical models have been developed for the longitudinal behavior. In the model by Newell (2002), in congestion, the vehicle α simply determines its speed $v_\alpha(t)$ at time t dependent on the distance $x_{\alpha-1}(t) - x_\alpha(t)$ to its leader $\alpha - 1$:

$$v_\alpha\left(t + \frac{\tau}{2}\right) = \frac{1}{\tau} (x_{\alpha-1}(t) - x_\alpha(t)) - \frac{d}{\tau}, \quad (2.8)$$

whereby d and τ are a spatial and a temporal delay parameter, respectively. Essentially, the follower α copies the trajectory of its leader $\alpha - 1$ delayed by d in space and τ in time. In free-flow, the vehicle simply travels with its maximum speed. This model is a microscopic version of the macroscopic LWR model, which will be discussed soon.

The Optimal Velocity Model by Bando et al. (1998) determines the acceleration a_α of a vehicle according to a distance-speed relationship V , also called the optimal velocity function:

$$a_\alpha = \beta \cdot (V(x_{\alpha-1} - x_\alpha) - v_\alpha), \quad (2.9)$$

whereby β is a parameter expressing the sensitivity of the driver. The Intelligent Driver Model by Treiber et al. (2000) determines the acceleration based on the current speed, speed difference and distance to its leader:

$$a_\alpha = \beta \left[1 - \left(\frac{v_\alpha}{v^{\text{free}}}\right)^\delta - \left(\frac{r^*(v_\alpha, \Delta v_\alpha)}{x_{\alpha-1} - x_\alpha}\right)^2 \right], \quad (2.10)$$

with the desired minimum gap

$$r^*(v_\alpha, \Delta v_\alpha) = r^{\text{min}} \sqrt{\frac{v_\alpha}{v^{\text{free}}}} + T v_\alpha + \frac{v_\alpha \Delta v_\alpha}{2\sqrt{\beta\gamma}}, \quad (2.11)$$

with the parameters r^{min} , T , v^{free} , β , γ and δ . These models can easily be generalized to multiple vehicle classes by making the parameters vehicle-dependent.

Whereas the models named above are deterministic, the following ones include random behavior and are therefore non-deterministic. The model by Leutzbach & Wiedemann (1986) is a psycho-spacing model, whereby the driver begins or stops accelerating depending on the distance and speed difference to its leader.

Longitudinal microscopic models combined with lane-changing models are then typically used in practice for simulation. The software tool FOSIM (Dijker, 2002) is calibrated for Dutch freeways. Commercial packages for network-wide simulations include ptv Vissim, Quadstone Paramics and Aimsun.

Computational Speed The computational speed of a microscopic model can be low. One reason is that a lot of vehicles have to be simulated. Since the number of vehicles in the network is much higher in congestion than in free-flow, the model performs significantly slower when congestion is present. Another reason is that multiple simulations have to be performed in order to create a meaningful average result that can be used in the model-predictive controller, if it is a non-deterministic model.

Phenomena Microscopic model can potentially reproduce very detailed phenomena. Besides congestion, multiple vehicle classes and their effects on the spacing, emerging and dissolution of spillback, as well as bounded acceleration and the capacity drop can be reproduced. However, a microscopic model that reproduces all phenomena well does not yet exist.

2.2.3 Conclusion of the Prediction Component

The matrix in Table 2.1 summarizes the models' capabilities to reproduce traffic flow phenomena. The LWR model was the first macroscopic traffic flow model that reproduces the congestion well, including spillback and dissolution. It has been expanded in order to reproduce different vehicle classes. Though current multi-class models do not capture the capacity drop, several macroscopic mixed-class models such reproduce it to some degree. A combination of those models might be able to be both multi-class and reproduce the capacity drop. Furthermore, macroscopic models are divided into ones with a fixed pce value and ones with a dynamic pce value. The latter ones are more realistic macroscopic models. Microscopic models have the potential to reproduce all traffic flow phenomena; though currently, no microscopic model reproduces all phenomena. Furthermore, they are not deterministic in general.

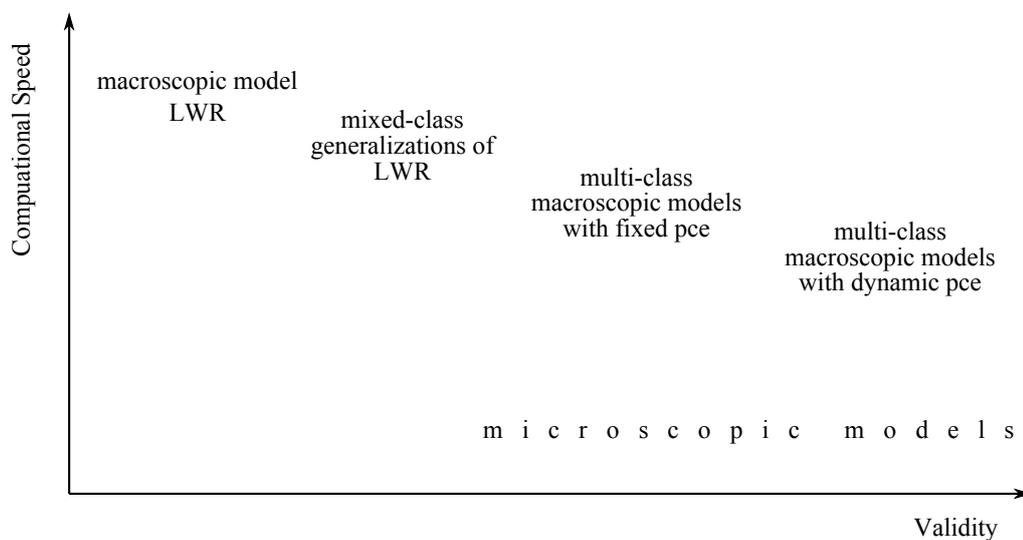


Figure 2.7: Overview of traffic flow predictions models

	macroscopic LWR model	mixed-class generalizations of LWR model	macroscopic multi-class model fixed pce	macroscopic multi-class model dynamic pce	microscopic models
congestion	✓	✓	✓	✓	✓
spillback	✓	✓	✓	✓	✓
congestion dissolution after increase of demand	✓	✓	✓	✓	✓
congestion dissolution after increase of supply	✓	✓	✓	✓	✓
capacity drop	–	✓	○	○	✓
emergence of stop-and-go waves	–	✓	○	○	✓
propagation of stop-and-go waves	✓	✓	✓	✓	✓
multi-class	–	–	✓	✓	✓
dynamic pce value	–	–	–	✓	✓
deterministic model	✓	✓	✓	✓	–

Table 2.1: Overview of prediction models and their capabilities to reproduce traffic flow phenomena; ✓: model reproduces phenomenon, –: model does not produce phenomenon, ○: model reproduces phenomenon only partially or has the potential to reproduce it if it is extended

The diagram in Figure 2.7 compares the prediction models in terms of computational speed and validity for our purposes of applying a model-predictive controller for multi-class freeway traffic. Microscopic models have the potential to model many traffic phenomena, though currently they require many executions in order to produce a meaningful average result for the model-predictive controller. The class of macroscopic models represent the average behavior of traffic so that one model execution is sufficient, which renders them much faster than microscopic models. The more realistic a macroscopic model is, the more equations it usually contains and therefore the slower it computes the result. Macroscopic models that take the dynamics of the spacings between vehicle classes in the form of a pce function into account are the most valid models for the purpose of this thesis.

Decision

Since a goal of this thesis is to develop a multi-class controller, the multi-class behavior of traffic should be represented sufficiently accurate. The two candidates for that purpose are microscopic models and multi-class macroscopic models with a dynamic pce function. Since microscopic models require a long time to produce a meaningful average result, they are unfeasible for the control purposes of this thesis. Other reasons against them are that the data on an individual level are not gathered, and that the network layout has to be known in detail. Macroscopic models, in contrast, are much faster and can run within a few seconds.

We therefore choose Fastlane as the prediction model used in the rest of this thesis. The model originates from Van Lint et al. (2008b); its continuous model is also presented in this thesis in Section 5.1 and discretized in Section 5.2.

In order to use Fastlane in a predictive control environment, it has to be calibrated to the traffic network under consideration. As part of the case study of this thesis, Fastlane is calibrated and validated to the Dutch A15 in Sections 7.3 and 7.4, respectively. The input of Fastlane is the current traffic density over space. The next section chooses an appropriate model to estimate the current traffic state.

2.3 Design Choices for the Estimation Component

This section presents an overview of traffic state estimation methods. First, the requirements for the Estimation Component are outlined. Then, estimation methods are discussed. Finally, we conclude that both the Adaptive Smoothing Method and the Extended Kalman Filter technique are suitable. These two methods will be analyzed further in Chapter 4, and we will decide there to ultimately use the Adaptive Smoothing Method.

2.3.1 Requirements of the Estimation Component

The task of the Estimation component is to map the traffic data from the sensors to the input of the Prediction component, which represents the current traffic state. In most real-time freeway applications, the spatiotemporal data in the form of speed, flow or density have to be fused to the density over space, preferably for each vehicle class.

In order to perform control in real-time, the current traffic state has to be computed fast, i.e. within a few seconds. Furthermore, the traffic state estimated must closely match the true traffic state. Since the sensors do not cover the whole freeway, but instead provide only partial information, the data have to be interpolated in a valid way.

2.3.2 Estimation Models

In the following, several traffic state estimation models are discussed. For each method, we present its main idea and examples, and discuss its computational speed and its validity.

2.3.2.1 Interpolation in Space

Method A simple method of estimating the traffic state is to interpolate the most recent data between the sensors. Simple interpolation can be done constantly (using the data of the downstream detector), linearly, (a weighted average between the two adjacent detectors), or by some other interpolation function.

Examples Examples of piecewise constant or piecewise linear interpolation are discussed in Van Lint & Van der Zijpp (2003).

Computational Speed Due to their simplicity, these interpolation methods compute the traffic state very fast.

Validity The drawback is, however, that the traffic state is not very accurate. The main reason is that the dynamics of traffic are not taken into account.

2.3.2.2 Interpolation in Space and Time: Adaptive Smoothing Method

Method The Adaptive Smoothing Method (ASM) (Treiber & Helbing, 2002) does take the dynamics of traffic into account. Some characteristics of traffic, e.g. speed and flow, travel with a certain speed along the freeway. This method interpolates the data along this propagation speed, i.e. it interpolates the data over space and time.

Examples The Adaptive Smoothing Method was developed originally by Treiber & Helbing (2002) for one data source. It was then expanded to multiple data sources by Treiber et al. (2011) and Van Lint & Hoogendoorn (2010).

Computational Speed Due to the more complicated model, current implementations compute the traffic state within a few minutes.

Validity Since the propagation of the traffic characteristics is taken into account, this model performs better than a simple spatial interpolation method.

2.3.2.3 Nudging

Method Newtonian relaxation, or nudging, combines the data of the sensors with the prediction of a traffic model. Traffic flow models like the ones discussed in Section 2.2 can serve as the prediction model. Depending on the traffic flow model, the propagation of the traffic characteristics are taken into account. The errors of the model can therefore be corrected with data of the sensors.

Examples Nudging is used by Herrera & Bayen (2010) for estimating the traffic state based on mobile phones by adding or removing vehicles in the model prediction.

Computational Speed Since nudging employs a prediction model, its computational burden is higher than those of the previous estimation models.

Validity Since nudging employs both a traffic flow model that takes the propagation direction into account and furthermore corrects the model errors to some extent, nudging can lead to better estimation results than the ASM.

2.3.2.4 Recursive Bayesian Estimation

Method Like nudging, recursive Bayesian estimators combine sensor data with a model prediction. In addition to the previous approach, the model and sensor uncertainties are modeled. These data are then fused by applying Bayes' Law.

Examples Commonly used recursive Bayesian estimators in traffic state estimation include the Extended Kalman Filter (Wang et al., 2007; Tampère & Immers, 2007; Van Lint et al., 2008a), the Unscented Kalman Filter (Ngoduy, 2008), the Ensemble Kalman Filter (Work et al., 2008), and the Particle Filter (Mihaylova et al., 2007).

Computational Speed Recursive Bayesian estimators consist of some demanding mathematical operations like matrix inversions; they can therefore be relatively slow.

Validity Since Bayesian estimators also take the model and sensor noise distributions into account, they can be the most valid. However, both the prediction model and the noise parameters have to be calibrated.

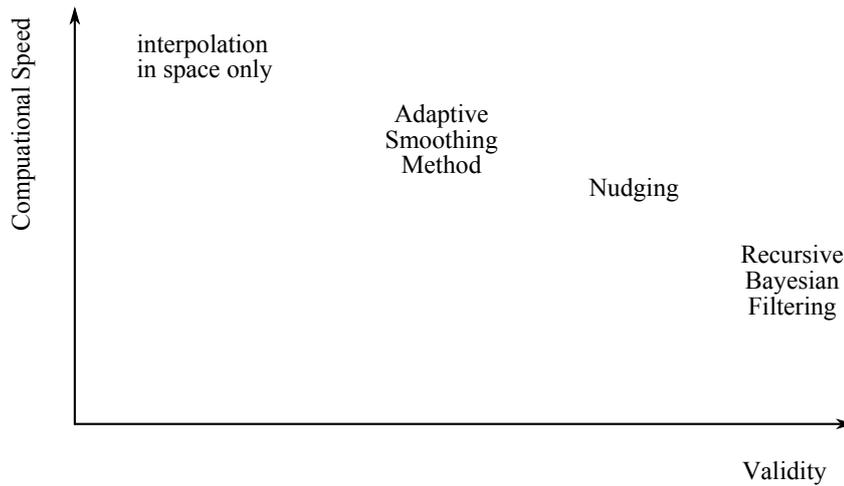


Figure 2.8: Overview of traffic estimation models

2.3.3 Conclusion of the Estimation Component

The diagram in Figure 2.8 compares the estimation models in terms of computational speed and validity. The more information is taken into account, the more valid a model can be, though also that also increases the computational complexity. Simple interpolation methods can be very fast, but they can exhibit large estimation errors. The Adaptive Smoothing Method takes the propagation direction of the characteristics into account, which produces better results. Since nudging and recursive Bayesian estimation combine the data with the prediction of a traffic flow model, they have the potential to be the most valid ones. Because they use a prediction model, they are also the slowest estimators.

Decision

Both the Adaptive Smoothing Method and recursive Bayesian estimators exhibit a good performance. In Chapter 4, we will further analyze both the ASM and the Extended Kalman Filter, which is a form of a recursive Bayesian estimator. They will be computationally improved so that they can estimate the current traffic within a few seconds and are therefore suitable for real-time applications. In that chapter, we will also decide on a traffic state estimator. We will choose the Adaptive Smoothing Method as a suitable traffic state estimator to be applied in the Estimation Component, since it is much easier to calibrate than the recursive approaches.

Furthermore, Chapter 3 will develop a method to automatically estimate the propagation of the characteristics, which is used to calibrate the state estimators. The structure of the Estimation Component of the control loop is illustrated in Figure 2.9.

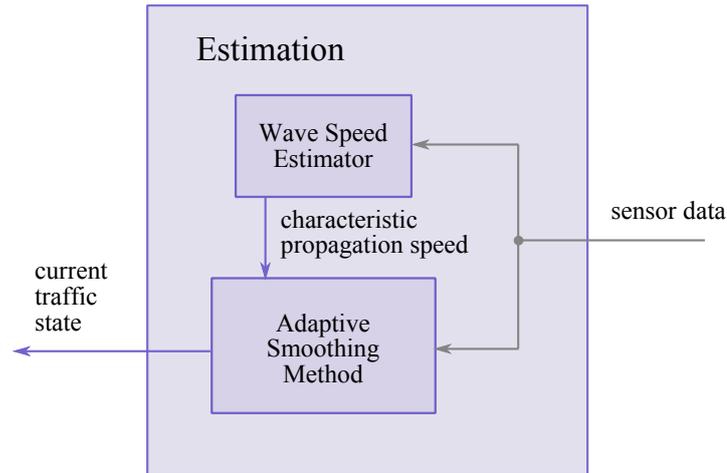


Figure 2.9: Structure of the Estimation component

2.4 DTM Measures Considered in This Thesis

In this thesis we will test and develop multi-class traffic control concept in which three types of freeway DTM measures are considered: ramp metering, route guidance and dynamic truck lanes.

Ramp Meters are installed at on-ramps to limit the inflow into the freeway. One aim thereby is to cap the traffic flow in order to prevent (or at least delay) the onset of congestion on the freeway. Furthermore, if traffic is kept in free flow, the capacity drop often caused by congestion (Chung et al., 2007) is avoided, thereby ensuring a larger capacity than without ramp metering. Another aim is to improve merging of traffic by granting vehicles access to the freeway when a sufficiently large gap is present, which homogenizes traffic and therefore decreases the chance of the onset of congestion.

Figure 2.10(a) shows the ramp meter installation at the on-ramp Delft Noord to the Dutch A13. The green time of the traffic light is fixed, allowing one vehicle to pass. By varying the red time, the outflow of the ramp and therefore the inflow into the freeway (in $\frac{\text{veh}}{\text{h}}$) is controlled. A simple demand-capacity algorithm can control the ramp flow in such a way that it equals the difference between the capacity of the freeway and its current flow (Masher et al., 1975). A similar approach is the algorithm used by the Dutch road authority (Taale & Middelham, 2000), though it smooths the freeway traffic data over time. The control algorithm Alinea (Papageorgiou et al., 1991) is a PI-controller that assigns the ramp flow in such a way that the traffic state downstream of the on-ramp is kept at a specified traffic density, which is usually set slightly below the critical density so that the freeway traffic stays in free-flowing conditions.

The second DTM measure used in this thesis is route-guidance. Travelers can usually choose between multiple routes from their origin to their destination. Route guidance measures are used to influence that route choice in order to distribute traffic streams across the network. This is a measure to route vehicles around a congested area.



(a) Ramp meter (at on-ramp Delft Noord of the Dutch A13, from Wikimedia Commons (2012))



(b) Route guidance (near ring freeway of Amsterdam, from Wikimedia Commons (2012))



(c) High-occupancy and toll lane (HOT, near Seattle, WA, SR 167, from Homepage Texas Transportation Institute (2012))

Figure 2.10: DTM measures considered in this thesis

Routes can be advised by several different actuators. Road-side signs can directly advise a route or, alternatively, inform the drivers about the travel times or traffic conditions of each possible route (Figure 2.10(b)). Since many travelers use personal systems like navigation devices or smart phones, the route can also be advised to each traveler individually. Route Guidance measures control the turnfraction of traffic streams at network bifurcations like off-ramps or junctions. Route guidance control has been implemented in several countries, among others in Scotland (Messmer et al., 1998) and Denmark (Mammar et al., 1996). Route guidance has been applied in simulations of the Dutch A10 ringroad around Amsterdam (Wang et al., 2006).

The third DTM measure discussed is the vehicle-class specific lane. Traffic is composed of multiple vehicles classes, whereby the values of the classes differ. Examples of valuable vehicles are trucks with time-critical goods, cars with multiple passengers, or cars with a single passenger who is willing to pay a toll to gain access to a faster lane. Congestion thus has a particularly large negative effect on the travel costs of these

vehicles. A vehicle-class specific lane is a measure to let the valuable vehicles bypass a congested area. A vehicle-class specific lane can be static, or it can be dynamically switched on or off dependent on the traffic state.

Figure 2.10(c) shows a variable sign of a high-occupancy and toll lane (HOT), where the price for cars with a single passenger can be adjusted dynamically. A vehicle-class specific lane separates the flows of traffic across lanes and controls the traffic composition of the lanes. High-occupancy vehicle lanes (HOV) and high-occupancy and toll lanes (HOT) are popular in the US. Buckeye (2012) presents an overview of vehicle-class specific lanes applied in the state of Minnesota. The optimal price setting for HOT is an issue of current research, see for example Lou et al. (2011). De Palma et al. (2008) analyzes the optimal toll rate of a truck lane in terms of travel costs, noise and safety.

Ramp metering and route guidance will be generalized to multi-class DTM measures in Chapter 6 and eventually applied in the case study of Chapter 7. Furthermore, since vehicle-class specific lanes like truck lanes already exist, we will include them in the analyses of Chapter 6.

2.5 Traffic Sensors Considered in This Thesis

The literature of road sensing technology is large (Tyburski, 1988). This section provides only a brief overview. As shown in the control loop of Figure 1.2, the traffic data are used by the Estimation component to compute the current traffic state. For further purposes like analyzing historic data or traffic model calibration, data are usually logged in a database.

Stationary sensors provide data at a fixed location. The most commonly used stationary sensor is the inductive loop, which is placed in the surface of the road. When a vehicle passes the loop, its metallic frame induces an electric current. The onset and the offset of this current mark the passing and leaving time of the vehicle. By counting the number of passing vehicles, the traffic flow can be calculated. There are several types of inductive loops. Dual-inductive loops are common in the Netherlands. Since the distance between the loops is known, the speed and the length of the vehicle can be deduced. Single-inductive loops are common in the United States. The vehicle speed is therefore not directly observed. Nevertheless, methods have been developed to infer the vehicle speed from the measured flow and occupancy (Coifman & Kim, 2009). There are also other sensors to gather stationary data like flow and speed; examples are infrared detectors and microwaves.

In-vehicle sensors like Global Positioning System devices (GPS) measure and disseminate the position and the speed of a vehicle. GPS sensors are common nowadays in navigation devices or smartphones. These floating-car data (FCD) do not measure the traffic flow. However, the flow can be deduced if the percentage of equipped vehicles is known.

Point-to-point sensors like Bluetooth or fixed cameras with automatic vehicle identification (AVI) provide information about the travel time and partially about origin-destination information. To calculate the complete origin-destination information, a similar requirement as for FCD applies, namely that a known percentage of the traffic of each origin-destination pair is equipped with Bluetooth devices.

In this thesis, data from various sensors are used. We mainly use data from induction loops, since these are easily available. The database of Regiolab Delft (2012) stores data from dual-inductive loops of the freeways in the Netherlands. Roportis (2012) stores loop data from the road network in the harbor area of Rotterdam. Furthermore, we use videos gathered from a helicopter perspective. Although these data are not used to estimate the current traffic state, they are valuable for the calibration of traffic flow models, as it will be performed in Section 7.3 as part of a case study.

2.6 Conclusion

This chapter designed the parts of the control loop that was outlined in its basic form in Figure 1.2. Since the scope of this thesis is to control freeway traffic vehicle-class specifically, most of the parts of the control loop are multi-class. The refined version is shown in Figure 2.11.

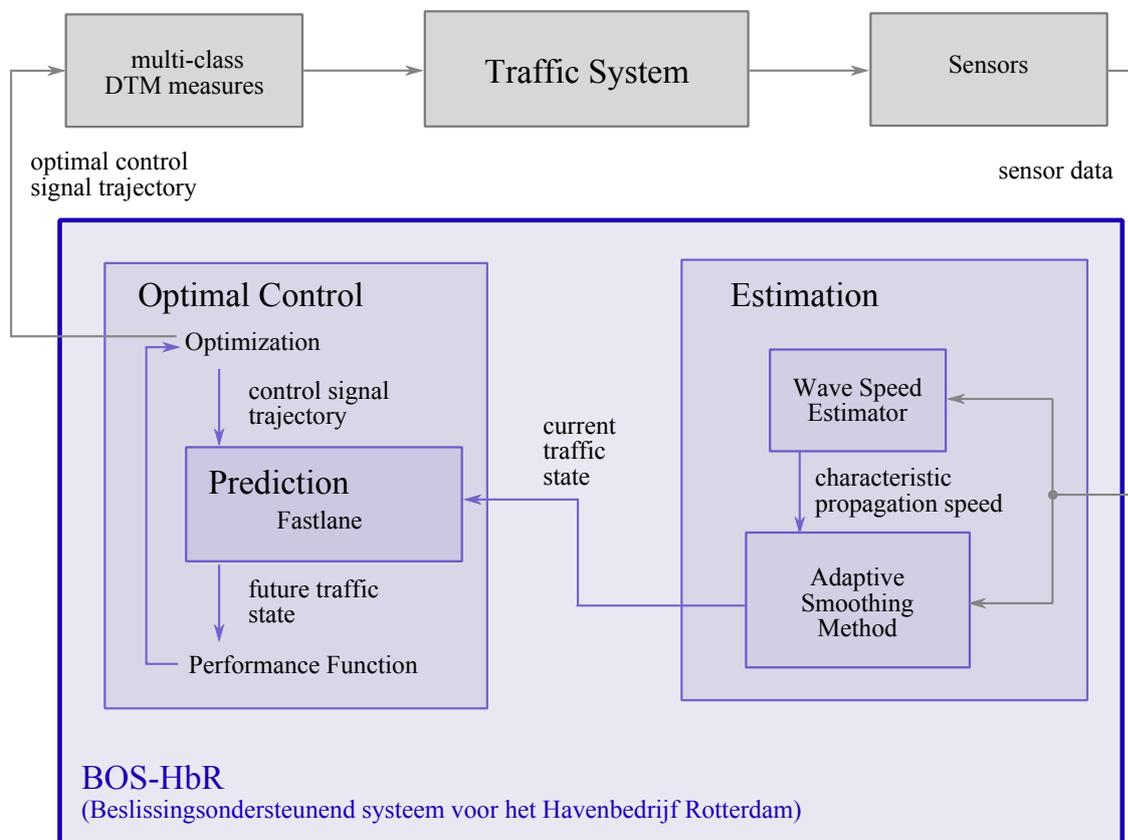


Figure 2.11: Control loop for multi-class Dynamic Traffic Management

As a control approach, optimal control and model-predictive control are chosen, since they can directly take the effects of irregular traffic conditions such as incidents into account, they are able to predict the effects of the multi-class DTM measures, and they optimize the future traffic conditions based on a defined objective such as minimizing the total cost.

To predict the future traffic conditions, the multi-class macroscopic traffic flow model Fastlane will be used. It reproduces the most important traffic flow phenomena, such as the onset and dissolution of congestion, spillback, and a dynamic pce value dependent on the current traffic state.

The current traffic state is estimated by the Adaptive Smoothing Method, which interpolates the data based on the propagation speeds of the traffic characteristics such as flow and speed.

The DTM measures are vehicle-class specific, i.e. they can control each vehicle class separately. In this thesis, a class-specific lane, a class-specific ramp meter and a class-specific route guidance measure will be used. To observe the traffic state in real-time, inductive loop detectors are used.

The following chapters will further develop the components of the control loop of Figure 2.11. Chapter 3 develops a method which automatically estimates the propagation speed of the characteristics. Chapter 4 computationally improves both the Adaptive Smoothing Method and an Extended Kalman Filter in order to estimate the traffic state within a few seconds. Chapter 5 analyzes the effects of different vehicle classes on traffic throughput, congestion spillback and total cost on the prediction model Fastlane. Chapter 6 develops multi-class DTM measures and test their performance and their effects on traffic.

In a case study in Chapter 7, the three components Estimation, Prediction and Control will be applied to the Dutch A15 near the harbor of Rotterdam and form the system BOS-HbR, which stands for “Beslissingsondersteunend Systeem voor het Havenbedrijf Rotterdam” (in English: “Decision Support System for the Port of Rotterdam Authority”.) An online version will be shown in Section 8.5, where the current traffic state is estimated in real-time, the future traffic conditions are predicted and control signals are optimized for the next hour.

Chapter 3

Estimation of Spatiotemporal Traffic Characteristics

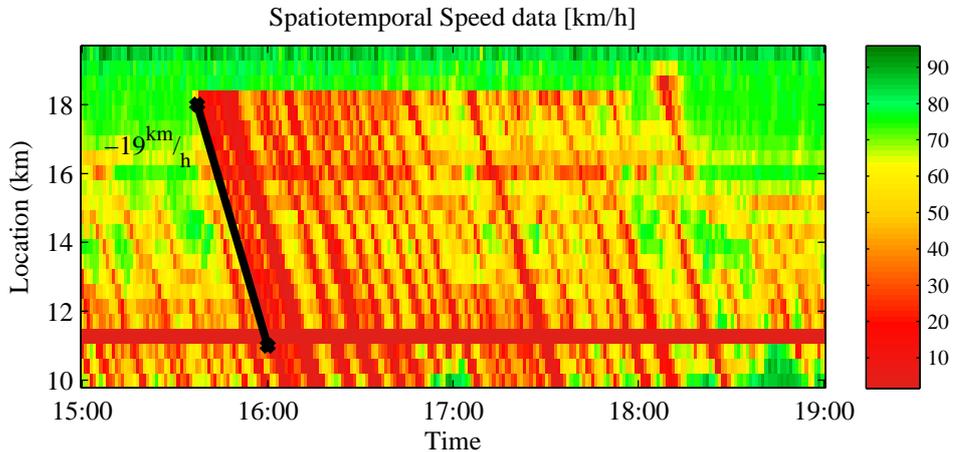
The goal of this first part of the thesis is to accurately estimate the traffic state of a freeway within a few seconds given available raw traffic data. This chapter develops a method that estimates the location of the transitions of traffic states. The propagation speed of shock waves is analyzed from spatiotemporal traffic data by applying image processing tools. The propagation speed of the shock waves are used to calibrate the traffic state estimators described in Chapter 4.

First, Section 3.1 presents a background of traffic shock waves and their relevance for traffic state estimation. Then, Section 3.2 reviews state-of-the-art methods of wave speed estimation. The new method of wave speed estimation is developed in Section 3.3. This method is applied to empirical data from Dutch freeways in Section 3.4. Another application is the performance evaluation of traffic flow models; in Section 3.5, this method is applied to synthetic data to evaluate the speed of shock waves generated by a traffic flow model. The conclusions are presented in Section 3.6.

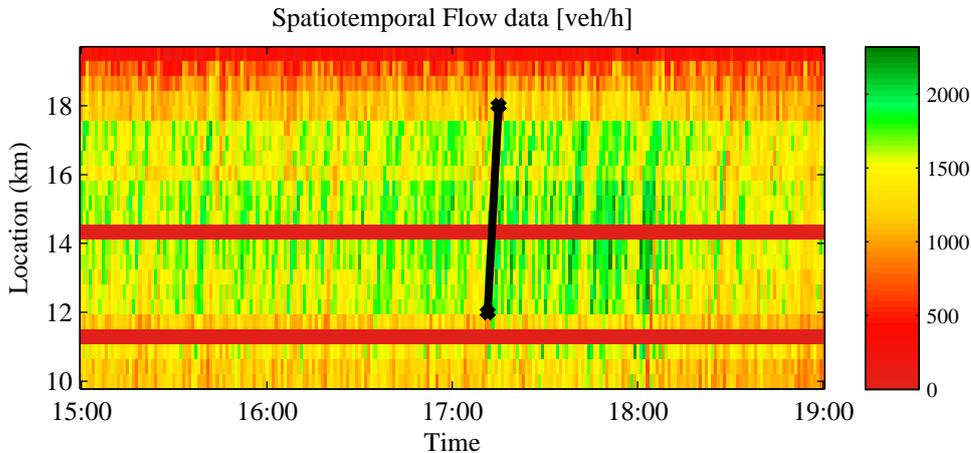
This chapter is an extended version of Schreiter, Van Lint, Yuan, & Hoogendoorn (2010b): “Propagation Wave Speed Estimation of Freeway Traffic with Image Processing Tools” presented at the 89th Annual Meeting of the Transportation Research Board.

3.1 Background

The estimation of spatiotemporal characteristics is an important part for estimating the current traffic state. This chapter focuses on the estimation of shock waves and their propagation speeds, since they will be used later to calibrate the traffic state estimators.



(a) Spatiotemporal speed data, a shock wave at the upstream front of a stop-and-go wave (24-04-2009)



(b) Spatiotemporal flow data, a shock wave as platoons of vehicles travel through light traffic (25-04-2009)

Figure 3.1: Shock waves (black) observed in spatiotemporal freeway traffic data (Dutch A13R)

A typical example of a shock wave occurs at stop-and-go waves. From a driver's perspective, traffic slows suddenly down so that they have to brake; after a few minutes, traffic resumes free-flow conditions so that drivers can continue their trip. When observing the traffic stream from atop, the stop-and-go wave is visible as a region of a few hundred meters length where traffic is standing still. Since vehicles leave that region at its head and new vehicles enter that region at its tail, the stop-and-go wave propagates upstream. Such a stop-and-go wave is thus characterized by two shock waves, one at

the head and one at the tail. This pattern is visible in the printouts of spatiotemporal traffic flow data. Figure 3.1(a) shows multiple stop-and-go waves in a speed contour plot. The upstream shock of a stop-and-go wave is marked by the thick black line. This shock wave emerges at Kilometer 18 at 15:35 and then propagates upstream with a constant speed of $c = -19 \frac{\text{km}}{\text{h}}$.

Shock waves also occur in free-flow traffic, as the spatiotemporal flow plot of Figure 3.1(b) shows. Vehicle platoons cause a regions of high flow, which are surrounded by shock waves. One of them is marked by the thick black line.

Empirical data show that the propagation speed of the characteristic congestion wave speed varies between $-25 \frac{\text{km}}{\text{h}}$ and $-15 \frac{\text{km}}{\text{h}}$ (Schönhof & Helbing, 2009; Kerner & Rehborn, 1997; Treiber et al., 2000; Kerner, 1998; Chiabaut & Leclercq, 2011; Bertini & Leal, 2005; Graves et al., 1998; Zhang & Rice, 1999; Cassidy & Mauch, 2001). This variation can be explained with the following simple car-following law (Pipes, 1967):

$$r = r^{\min} + h^{\min} \cdot v \quad \Leftrightarrow \quad v = \frac{r - r^{\min}}{h^{\min}}, \quad (3.1)$$

where r denotes the average gross distance headway between vehicles, r^{\min} the distance at standstill, v the speed and h^{\min} the minimum time headway. Note, that this relation only holds in sufficiently high densities k . Since $k = \frac{1}{r}$, from (3.1) follows

$$v(k) = \frac{\frac{1}{k} - r^{\min}}{h^{\min}} \quad \Rightarrow \quad q = kv(k) = \frac{1 - r^{\min}k}{h^{\min}}, \quad (3.2)$$

showing that this car-following law leads to a linear congestion branch of the fundamental diagram expressed in flow q and density k as illustrated in Figure 3.2. Then,

$$c_{\text{cong}} = \frac{dq}{dk} = -\frac{r^{\min}}{h^{\min}} \quad (3.3)$$

shows the relation between c_{cong} and the car-following parameters. This relation provides a tool to show how changes in composition, road condition and weather conditions affect the wave speed. For instance, a large share of trucks changes the average distance at standstill r^{\min} substantially, causing an increase in the speed c_{cong} . Reductions in the minimum headway h^{\min} , for example caused by changes in weather conditions, geometry or improved visibility, also lead to an increased wave speed (in absolute terms). The correct estimation of the site-specific c_{cong} is thus important.

In low densities, the flow grows approximately linear with the density, as the left branch of the fundamental diagram in Figure 3.2 shows. The characteristic waves speeds correspond therefore directly to the slope of the two branches of the fundamental diagram. This relationship is also used in traffic state estimation with the Adaptive Smoothing Method (Treiber et al., 2011), which will be analyzed later in Chapter 4. The characteristic wave speeds are therefore used to calibrate the fundamental diagram and traffic state estimators.

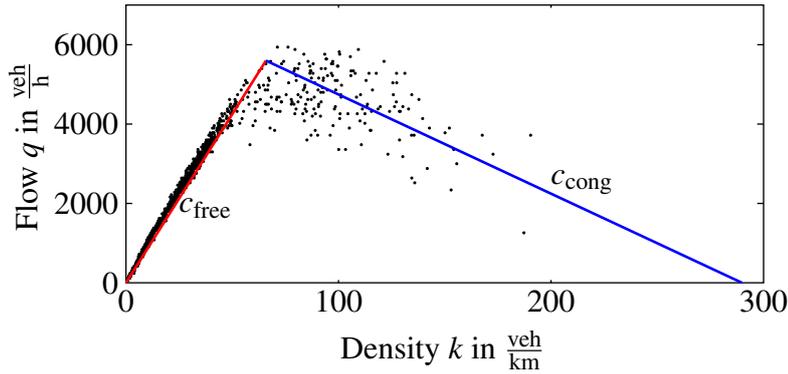


Figure 3.2: Relation between characteristic wave speeds and the flow-density fundamental diagram (data from Dutch freeway A15 eastbound by dual-inductive loops at Km 55.1 from 20-03-2012)

Conversely, the estimation of the characteristic shock wave speeds is possible from the flow-density fundamental diagram. However, the direct estimation of the fundamental diagram from a single stationary loop detector is difficult, since the data are very noisy. An example is shown Figure 3.2 by the scattered dots, which represent data gathered at a stationary detector. A reason for this noise is that loop detectors average the data over one minute, i.e. the data are aggregated over multiple traffic states, which leads to measurement errors (Laval, 2010). Moreover, on Dutch freeways, the sensors aggregate the speed arithmetically, which leads to a systematic overestimation of the speed (Knoop et al., 2009) and therefore to a bias in the fundamental diagram. Shock wave speeds can therefore hardly be estimated accurately from single stationary detector data.

An alternative approach is to analyze spatiotemporal data plots like the one in Figure 3.1 directly. There, the transitions of traffic states and the shock waves are directly visible, despite the imprecision and bias caused by the sensors. In this chapter, we develop a method that directly analyzes spatiotemporal data and scans such a plot for the transition of traffic states by applying image processing methods, namely an edge detector and a line detector. The method is able to localize shock waves that propagate with constant speed and it determines this propagation speed. The method can be applied for determining the characteristic shock wave speed of a traffic regime or to locate a bottleneck in the road stretch.

3.2 Literature Overview

The most intuitive method of identifying shock waves is to manually inspect spatiotemporal traffic data plots like the ones in Figure 3.1. A shock wave is visible as the border between two areas of different color. By measuring the spatial (Δx) and temporal (Δt) spread of that line, the propagation speed c of the corresponding shock wave speed is

the slope of that line:

$$c = \frac{\Delta x}{\Delta t}. \quad (3.4)$$

Nevertheless, methods have been developed to support this process.

Zheng et al. (2011a) analyze the speed signal of a stationary detector. A strong change of speed indicates a change of the traffic regime. If two of these changes are found within a few minutes, then a stop-and-go wave likely had passed. Such a strong change is detected by applying a wavelet transform, which is a common method in signal processing. More specifically, a peak in the energy distribution of the wavelet transform indicates a strong change of speeds over a longer period. In this way, the wavelet method is robust for narrow peaks in the signal, as it can be caused by detector errors. By applying this method to multiple adjacent detectors, the shock waves are located in space and time by tracing the peaks of each wavelet signal. A linear fit of these peaks then yields the propagation speed of the stop-and-go wave. Similarly, Zheng et al. (2011b) apply the same wavelet transform to vehicle trajectories to estimate the shock wave speeds.

Another method is to estimate the propagation wave speed by matching cumulative vehicle count curves of neighboring detectors. According to Newell's car-following model (Newell, 2002), in congestion, the trajectories of successive vehicles are similar; they are only shifted in space and time by the wave speed. This principle is used by Chiabaut et al. (2010) by comparing to successive vehicle trajectories. Furthermore, Chiabaut & Leclercq (2011) apply the same principle to macroscopic data: the shift between the speed and cumulative flow counts of neighboring detectors determines the shock wave speed.

Treiber & Kesting (2012) analyze the speed time-series of speed data of neighboring detectors by using the cross-correlation, which expresses how similar two signals are. Since in congestion the time-series are shifted by the characteristic shock wave speed c , the cross-correlation is maximized when the downstream time-series is shifted by $\Delta t = \Delta x \cdot c$ (with Δx as the distance between the detectors). However, different regimes are characterized by different wave speeds. A regime change thus leads to different signals between the detectors so that the cross-correlation approach may not work if applied to the whole signal. This method therefore requires a pre-selection of the data so that only data of the same traffic regime are used in the analysis. Similarly, Coifman & Wang (2005) use the cross-correlation of the cumulative vehicle count and speed signals between neighboring detectors to determine the shock waves speeds in congestion.

3.3 Methodology

This section develops the methodology of the Wave Speed Estimator (WSE). Based on spatiotemporal speed and flow data from freeways, the WSE computes the shock wave speeds. It consists of four components, as Figure 3.3 illustrates.

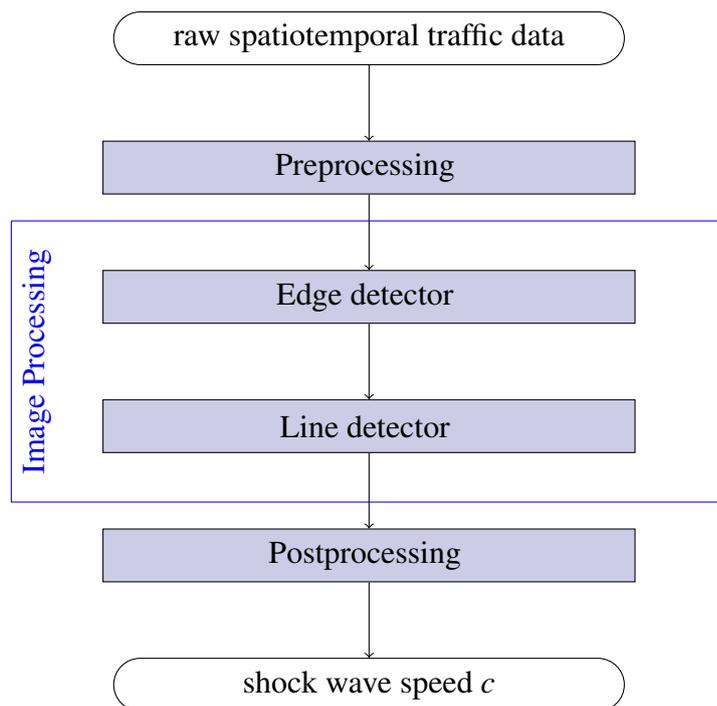


Figure 3.3: Structure of the Wave Speed Estimator

The two main components originate from the field of image processing. An edge detector localizes the border between two objects. A line detector is able to localize straight lines. The research field of image processing is well known and widely applied in traffic science as well as in many other domains. Applications are, for example, license plate recognition used in automatic vehicle identification systems (Shapiro et al., 2006; Anagnostopoulos et al., 2006; Abdullah et al., 2007), optical road estimation for autonomous vehicles (Dahlkamp et al., 2006), the surveillance of traffic by video sensors (Kastrinaki et al., 2003), the automatic detection of pedestrians and bicycles (Malinovsky et al., 2009) or the detection of vehicles (Chi & Caldas, 2011). In every-day life, image processing methods are used in face recognition applied in digital cameras.

Since the image processing methods operate on images, conversion from traffic data to image data and back are necessary. These are performed by the preprocessing and the postprocessing component of the WSE.

The remainder of this section explains each component of the WSE in detail. For illustration purposes, the intermediate results of each component are exemplified in Figure 3.4, where the characteristic shock wave speed of congested traffic is determined. In the experiments of Sections 3.4 and 3.5, we show other applications of the method such as the estimation of the characteristic free-flow speed and the localization of a bottleneck.

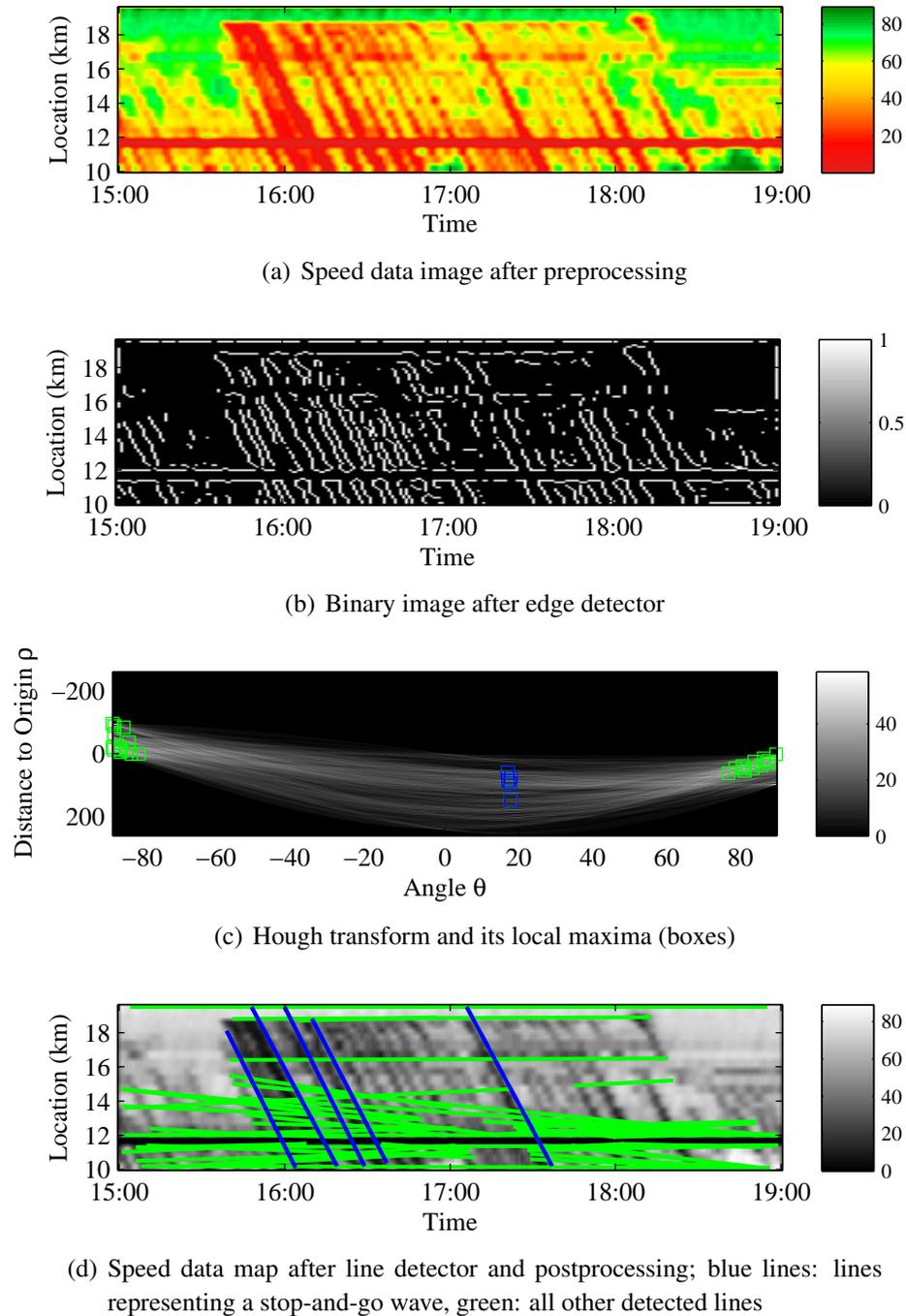


Figure 3.4: Intermediate results of the Wave Speed Estimator for the case of shock wave speeds at stop-and-go waves

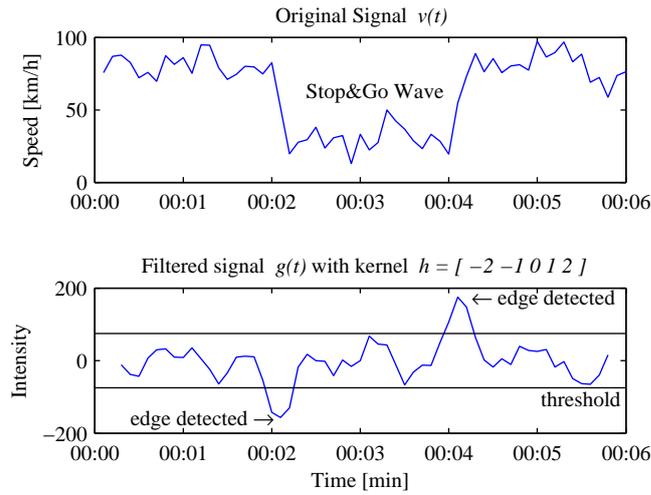


Figure 3.5: Example of edge detection in one dimension

3.3.1 Preprocessing

Before applying the image processing tools, the spatiotemporal traffic data is converted to an image. An image is a matrix where each element represents a so-called pixel, which is a rectangular area of fixed size and is assigned one value representing the brightness. In colored images, each pixel is assigned a vector of three values, representing the red, green and blue intensity, respectively. The image is created by interpolating the traffic data at equidistant sample points with a fixed resolution in space Δx_{discr} and time Δt_{discr} .

Furthermore, the raw traffic data may contain outliers and other high-frequency noise that obstruct the detection of the waves. A moving-average low-pass filter is therefore applied to reduce the high frequencies.

As an example, Figure 3.4(a) shows the result of applying the preprocessing steps to the raw data from Figure 3.1(a). Due to the low-pass filtering, the results look smoother than the original raw data. Still, the borders between high and low speeds are preserved so that shock waves are still visible.

3.3.2 Edge Detector

The next step is to detect the transitions of traffic states. An edge detector is able to detect the transitions between dark and light regions in an image.

Figure 3.5 exemplifies the application of an edge detector in one dimension. The top diagram shows a raw traffic signal $v(t)$ over time, for example from a stationary detector. Two traffic state transitions occur: at 00:02, traffic breaks down from $80 \frac{\text{km}}{\text{h}}$ to $30 \frac{\text{km}}{\text{h}}$; two minutes later, traffic recovers.

These transitions can be detected by a gradient-based method. Convolving the signal with a gradient-based kernel $h = [-2 \ -1 \ 0 \ 1 \ 2]$ results in the signal

$$g(t) = (v * h)(t) \quad (3.5)$$

shown at the bottom of Figure 3.5. The traffic state transitions are amplified and are visible by strong extrema.

An edge detector applies the same principle in two dimensions. The gradients g_x and g_t are determined by applying a gradient-based kernel (3.5) in each dimension. Then, these gradients are superimposed:

$$G(x, t) = \sqrt{g_x(x, t)^2 + g_t(x, t)^2} . \quad (3.6)$$

If the superimposed gradient G exceeds a specified threshold γ in a point (x, t) , then an edge is detected at the point:

$$E(x, t) = \begin{cases} 1 & \text{if } G(x, t) > \gamma \\ 0 & \text{else} \end{cases} . \quad (3.7)$$

The result of an edge detector is thus a binary image E , representing the edges of the traffic state conditions. An example is shown in Figure 3.4(b), where the diagonal lines represent the edges that are caused by the stop-and-go waves.

Note that the absolute values of the original signal are irrelevant, as long as the state transitions are visible. For example, the dual-loop detectors used in the Netherlands are inherently biased because they aggregate the speed arithmetically over one minute, which leads to an overestimation (Knoop et al., 2009). By applying an edge detector, however, the traffic state transitions can be located despite this bias.

3.3.3 Line Detector

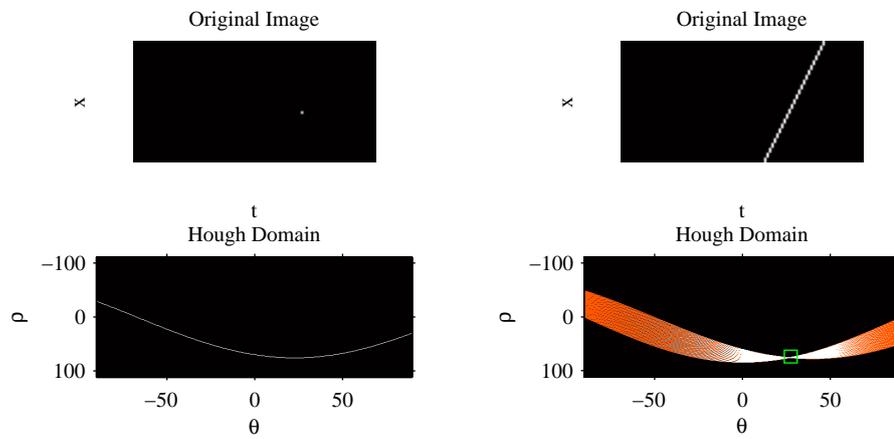
In the resulting image of the edge detector, the traffic state transitions are visible as white lines. To locate them, a line detector is applied to the edge image.

The Hough transform (Hough, 1962) is a widely-used method of detecting lines. It transforms an original image from the Cartesian x - t -plane into the so-called Hough domain or ρ - θ -plane. These parameters specify the polar coordinates of straight lines in the original picture by the relation

$$\rho = t \cos \theta + x \sin \theta , \quad (3.8)$$

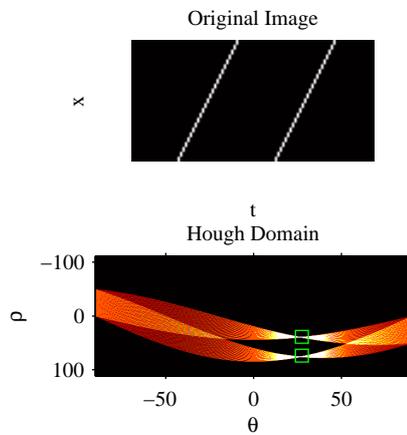
with the angle θ and the distance ρ to the origin of the image.

Figure 3.6 illustrates the important properties of the Hough transform. A point (x, t) in the original picture corresponds to a sine wave in the Hough domain, as Figure 3.6(a) shows. Moreover, a set of collinear points corresponds to a set of sine waves that all



(a) a point transforms into a sine wave

(b) a straight line transforms into a set of sine waves that intersect in a single point (green box) describing the angle and the distance to the origin of the straight line



(c) two straight lines transform into a set of sine waves that intersect in two points (green boxes) with the same angular coordinate

Figure 3.6: Relation between Cartesian $x-t$ image and its Hough transform to polar coordinates

intersect at exactly one point, as the green box in Figure 3.6(b) shows. This point (θ, ρ) defines the angle θ and the distance to the origin ρ of its corresponding straight line. Two parallel lines therefore result in a set of sine waves that intersect in two points with the same angular coordinate, as Figure 3.6(c) shows.

The propagation speed of the corresponding shock waves is therefore determined by the angle θ :

$$c = -\tan(90^\circ - \theta) \cdot \frac{\Delta x_{\text{discr}}}{\Delta t_{\text{discr}}} . \quad (3.9)$$

The distance ρ of the line to the origin of the Hough plane (the lower left corner) is used to locate the shock wave. In the case of standing waves, this can be used to locate the cause of the wave, such as a recurrent bottleneck or an accident.

The Hough transform is applied to the output image of the edge detector. This leads to a grayscale image in the Hough domain, as shown in Figure 3.4(c). The local maxima, which are the points with the highest intensity (boxes), are detected in the Hough domain. The results of the Hough transform and the line detector are thus a set of lines. An example of the result of the line detector is shown in Figure 3.4(d). (The background is a grayscale version of the preprocessed image from Figure 3.4(a).)

3.3.4 Postprocessing

The result of the line detector is thus a set of lines. If the goal is to determine the shock wave speed that characterizes a specific traffic flow pattern, these lines have to be analyzed further.

Each traffic pattern is characterized by a prior probability density function (pdf) f^p that describes which wave speeds it causes. The lines found by the line detector then represent a second pdf

$$f^L(c) = \frac{1}{\sum_{i=1}^n s_i} \cdot \sum_{i=1}^n s_i \cdot \delta(c - c_i) , \quad (3.10)$$

with their corresponding propagation speeds c_1, c_2, \dots, c_n weighted by their corresponding lengths s_1, s_2, \dots, s_n , and the functional δ as the Dirac delta function:

$$\delta(c) = \begin{cases} +\infty & \text{if } c = 0 \\ 0 & \text{else} \end{cases} \quad (3.11)$$

$$\int_{-\infty}^{+\infty} \delta(c) dc = 1 . \quad (3.12)$$

By applying Bayes' Law, the estimated posterior pdf

$$f^e(c) = \frac{f^L(c) \cdot f^p(c)}{\int_{-\infty}^{+\infty} f^L(c) \cdot f^p(c) dc} \quad (3.13)$$

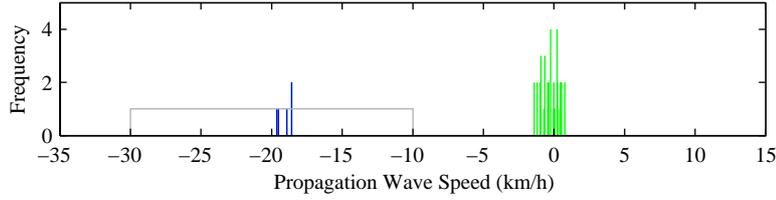


Figure 3.7: Frequency of wave speeds of lines detected; blue: lines within prior distribution (gray) of Figure 3.4, green: all other lines detected

describes the distribution of filtered shock wave speeds. The mean of that pdf then is the mean shock wave speed

$$c = \int_{-\infty}^{+\infty} f^e(c') \cdot c' dc', \quad (3.14)$$

which is the result of the WSE.

In the ongoing example of Figure 3.4(d), the shock waves of stop-and-go waves are of interest. Since stop-and-go waves propagate with a speed between $-25 \frac{\text{km}}{\text{h}}$ and $-15 \frac{\text{km}}{\text{h}}$ (Schönhof & Helbing, 2009; Kerner & Rehborn, 1997; Treiber et al., 2000; Kerner, 1998; Chiabaut & Leclercq, 2011; Bertini & Leal, 2005; Graves et al., 1998; Zhang & Rice, 1999; Cassidy & Mauch, 2001), we chose a uniform pdf

$$f^p(c) = \begin{cases} \frac{1}{b-a} & \text{if } c \in [a, b] \\ 0 & \text{else} \end{cases} \quad (3.15)$$

in the range of $[a, b] = [-30 \frac{\text{km}}{\text{h}}, -10 \frac{\text{km}}{\text{h}}]$ as prior distribution f^p . Figure 3.7 shows a histogram of wave speeds of all lines detected (f^L). The gray function is defined by the prior pdf f^p . The posterior distribution f^e (blue) therefore contains all the lines that fall within the prior function. The mean of the posterior distribution is $c = -19.9 \frac{\text{km}}{\text{h}}$.

3.4 Extracting Shock Waves from Empirical Data

In this section, the proposed method is validated for empirical data by applying it to spatiotemporal data gathered from Dutch freeways. Three experiments are performed in order to estimate the shock waves caused by stop-and-go waves, by vehicle platoons in free flow, and by a fixed bottleneck. To evaluate the performance of the WSE, its estimated shock wave speeds are compared to the ones found by manual inspection. We generated the latter ones by visualizing the raw data and estimating the shock waves speeds by hand. Finally, a sensitivity analysis is performed.

The WSE is applied to freeway data gathered from the A13 southbound between The Hague and Rotterdam (Kilometers 10 to 20, Figure 3.8). The dual-loop detectors are placed approximately 500m apart on average and aggregate the speed and flow data



Figure 3.8: The freeway A13 near Delft, The Netherlands; source: Google Maps (2012)

over one minute. The evening traffic data between 15:00 and 19:00 from 14 days (from 24-04-2009 to 07-05-2009) serve as raw data. Figure 3.1 shows two examples of the raw data. Since some of those days were weekends or holidays, not all of the traffic patterns occurred on all of the days.

3.4.1 Extracting Shock Waves Occurring at Stop-and-go Waves

Stop-and-go waves are a common phenomenon in congestion, as Figure 3.1 exemplified. Speed data are used as input, since stop-and-go waves are clearly visible there. The prior distribution f^p of the postprocessing component is set to a uniform distribution (3.15) with a range of $[a, b] = [-30 \frac{\text{km}}{\text{h}}, -10 \frac{\text{km}}{\text{h}}]$.

Figure 3.9 presents the results of the Wave Speed Estimator. For both wave speeds c_{cong} (blue, this section) and c_{free} (red, Section 3.4.2), the estimated wave speed (crosses) and the minimum and maximum speed values of all lines detected (triangles) are plotted. For comparison, the results of the manual inspection method are plotted as well (grey).

Congestion occurred on 9 out of the 14 days. On 8 out of these 9 days, the WSE estimated the congested wave speed c_{cong} within a range of $2 \frac{\text{km}}{\text{h}}$ to the manual method. On the twelfth day, no lines of congestion were detected. Since the values estimated by the WSE are close to those of the manual inspection method, we conclude that the WSE is capable of estimating the characteristic shock wave speed in congestion.

As the minimum and maximum values indicate, the lines detected are within a very small range; Figure 3.7 shows a histogram of all lines detected of the first data set. This narrow distribution of lines indicates that the shock waves occurring in congestion can be well summarized by one wave speed value c_{cong} for that day.

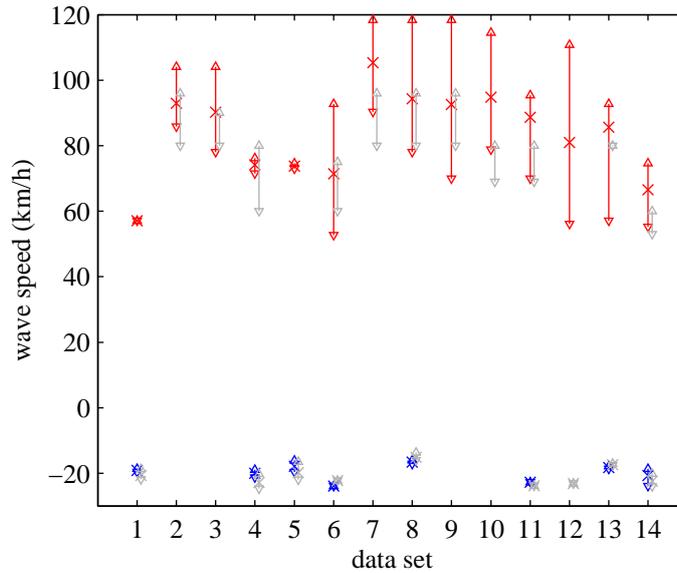


Figure 3.9: Results of the Wave Speed Estimator applied to empirical data to detect the wave speed of shock waves (in $\frac{\text{km}}{\text{h}}$); crosses: average wave speed estimated, triangles: the minimum and maximum values found; blue: congestion (Section 3.4.1); red: free-flow (Section 3.4.2); gray: results of the manual inspection method

3.4.2 Extracting Shock Waves Occurring in Free Flow

Vehicle platoons or other fluctuations in free flow cause shock waves that propagate downstream. Since they are well visible spatiotemporal flow plots (see Figure 3.1), flow data are used as input. The prior distribution f^p is set to a uniform function (3.15) with a range of $[a, b] = [50 \frac{\text{km}}{\text{h}}, 120 \frac{\text{km}}{\text{h}}]$.

An example result of the WSE for the second day is shown in Figure 3.10(a). Many shock waves are detected correctly. These shock waves propagate with different speeds, as the histogram in Figure 3.10(b) shows. Similarly, the other days show a large range of wave speeds as well (Figure 3.9). This result suggests that the free-flow branch of the fundamental diagram is bent.

In the case of free flow, the waves are difficult to detect manually. The results of the manual inspection (gray) are therefore very imprecise. Nevertheless, the results of Figure 3.9 show that the results of the WSE match the manual results in many cases. We conclude therefore that the WSE is also capable of estimating shock waves in free-flow.

3.4.3 Extracting Shock Waves Occurring at Fixed Bottlenecks

Fixed bottlenecks cause standing shock waves. In this experiment, the WSE is used to determine the location of a bottleneck. Figure 3.11(a) shows the speed contour plot of

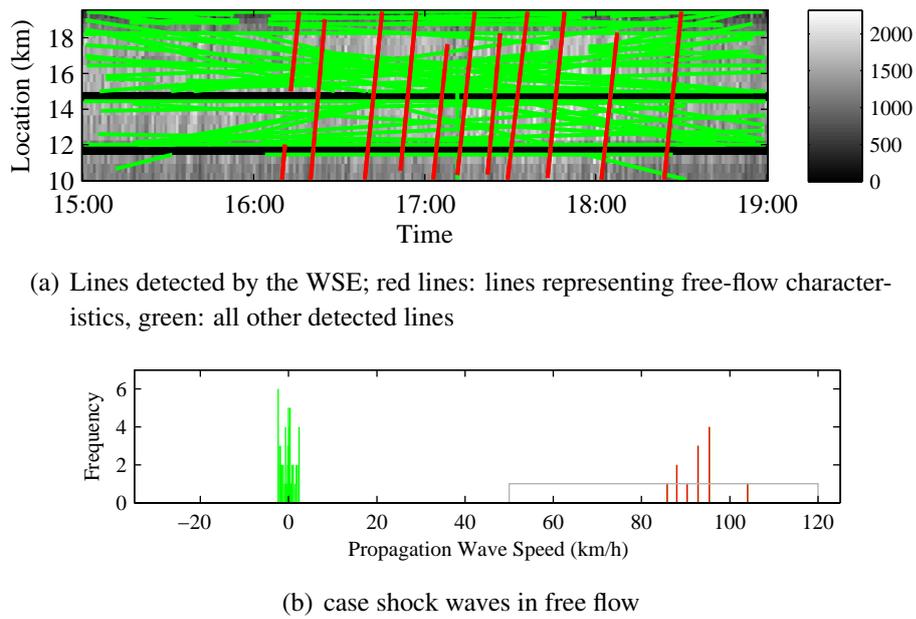


Figure 3.10: Results of extracting shock waves occurring in free flow

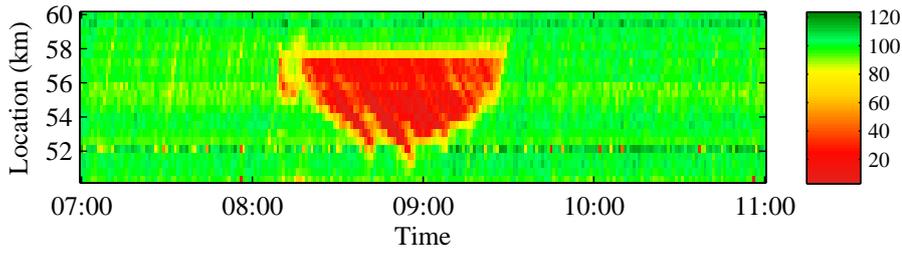
an accident on the A15 near Km 57.5 at 13-01-2010. Since the speed of a shock wave created by a fixed bottleneck is zero, the prior distribution (3.15) is set to a uniform function with the range $[a, b] = [-1 \frac{\text{km}}{\text{h}}, 1 \frac{\text{km}}{\text{h}}]$.

Figure 3.11(b) presents the results of the WSE for the case of standing waves. The downstream front of the bottleneck is clearly estimated. By applying (3.8), the distance coordinate ρ of the Hough image is used to determine the bottleneck location. The WSE reports a location of $x_{\text{bn}} = 57.6 \text{ km}$, which matches the true location. The WSE is therefore capable of localizing fixed bottlenecks. Since the shock between the congestion tail and free-flow propagates with drastically changing speeds, no straight line is detected there.

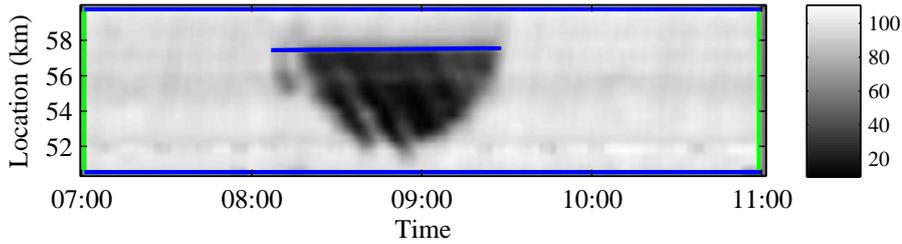
3.4.4 Sensitivity Analysis

The sensitivity of the estimated shock wave speeds is analyzed in order to determine how reliable the results of the WSE are and how difficult it is to calibrate. The parameters values of the pixel size Δt_{discr} and Δx_{discr} , the size of the raw data in space and time, the resolution of the Hough angle θ and further parameters of the image processing methods were varied. The raw data from Figure 3.1 were used as input with the goal of estimating the characteristic speed of shock waves in congestion.

The results show that the estimated shock wave speeds vary little within a range of $1 \frac{\text{km}}{\text{h}}$. However, if the amount of input data is changed drastically, e.g. to a data window of 24h by 30 km, then the WSE might not detect any shock waves. The reason is that the intermediate images are then much larger than the default case of 4h by 10 km, for



(a) Raw data (speed)



(b) Result of WSE; blue lines: lines representing a standing wave, green: all other detected lines

Figure 3.11: Results of extracting shock waves at fixed bottlenecks

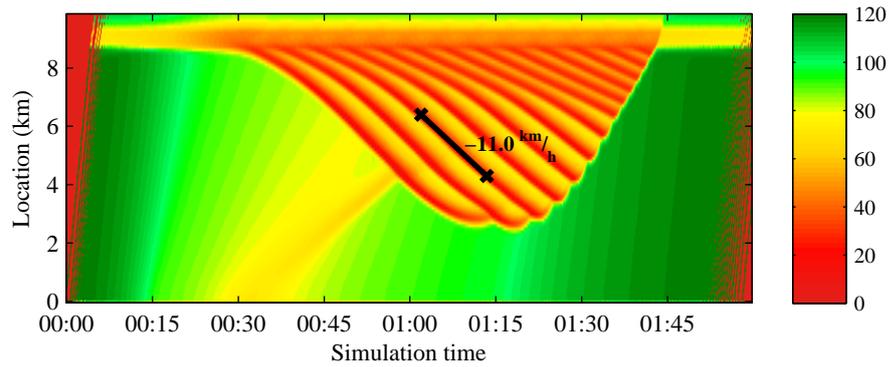
which the WSE was calibrated. Due to the different image size, the parameter values of the image processing tools have to be adjusted. For example, the pixel sizes Δx_{discr} and Δt_{discr} of the preprocessing component can be increased so that the size of the image matches that of the default settings.

To conclude, the sensitivity analysis shows that if the WSE estimates a shock wave speed, then the result is reliable. For some parameter values, however, no shock waves were detected. Especially for the case of changing data sizes, the WSE might have to be re-calibrated.

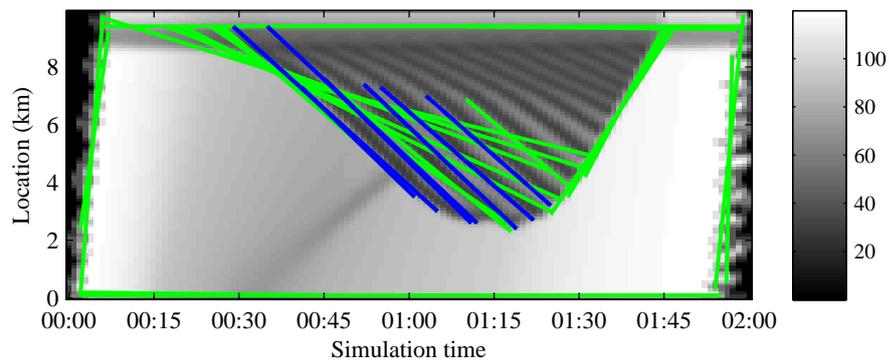
3.5 Extracting Shock Wave Speeds from Synthetic Data

This section applies the Wave Speed Estimator to spatiotemporal plots generated by a traffic flow model to validate it for synthetic data. Three experiments are performed on a road stretch; two of them reproduce congestion to assess the speed of stop-and-go waves, and one produces vehicle platoons in free-flow to assess the characteristic free-flow speed.

The synthetic data are created by the Intelligent Driver Model (Treiber et al., 2000), since it realistically reproduces traffic flow phenomena such as congestion or the emergence and propagation of stop-and-go waves. The output of this traffic flow model is vehicle trajectories. The trajectories are then aggregated to macroscopic quantities flow and speed according to Edie's definitions (Edie, 1965) (resolution 200m by second[10]). The WSE was then applied to these spatiotemporal quantities.



(a) Result of the traffic flow model (speed) and the manual inspection method



(b) Result of the Wave Speed Estimator

Figure 3.12: Results of Experiment 1

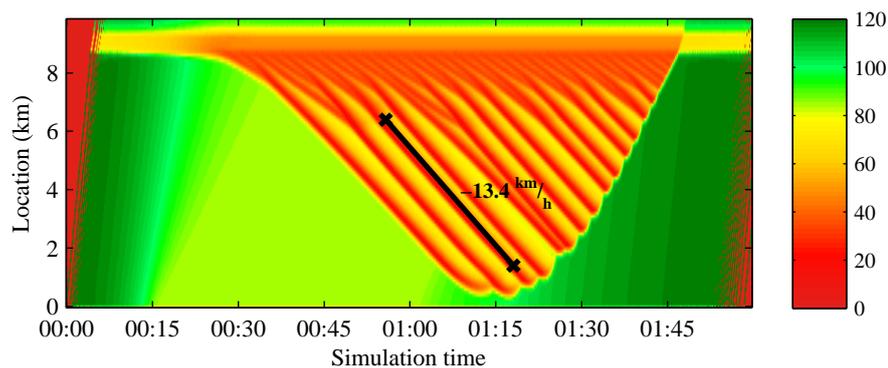
3.5.1 Experiment 1

A bottleneck of half a kilometer length is located near the end of the road stretch. In the bottleneck, the parameter values of the model are modified to reduce the capacity. The inflow into the road stretch is chosen in such a way that congestion emerges at the bottleneck.

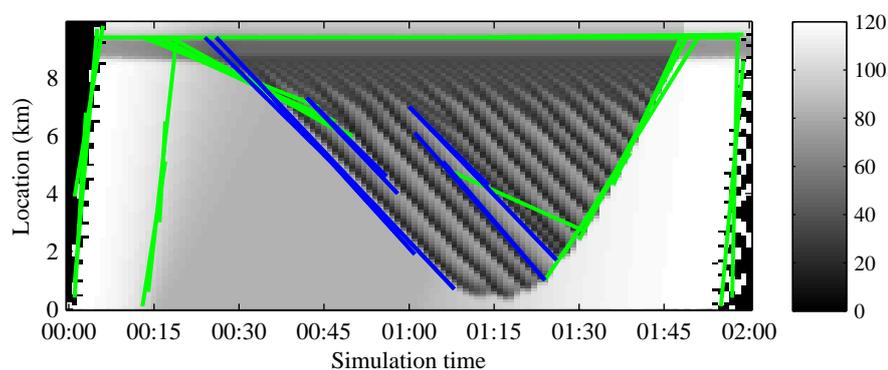
Figure 3.12(a) shows the output of the traffic flow model. Stop-and-go waves emerge upstream of the bottleneck. Near the bottleneck, the waves are not yet stabilized; further away from the bottleneck, the waves maintain a constant speed of $c = 11.0 \frac{\text{km}}{\text{h}}$, as can be verified by manual inspection (black line).

The output of the WSE is shown in Figure 3.12(b). As can be seen, the WSE detects many shock waves (blue) in the region a few kilometers upstream of the bottleneck, where the stop-and-waves have been stabilized. The average of the lines detected is $11.0 \frac{\text{km}}{\text{h}}$.

Some more shock waves are detected (green). Congestion grows with nearly constant speed between 00:40 and 01:05; similarly, it dissolves with nearly constant speed between 01:25 and 01:40. The WSE detects these boundaries, as the lines in the figure show. Furthermore, since the traffic state changes drastically at the bottleneck, the



(a) Result of the traffic flow model (speed) and the manual inspection method



(b) Result of the Wave Speed Estimator

Figure 3.13: Results of Experiment 2

WSE detects a shock wave there with a propagation speed of zero.

3.5.2 Experiment 2

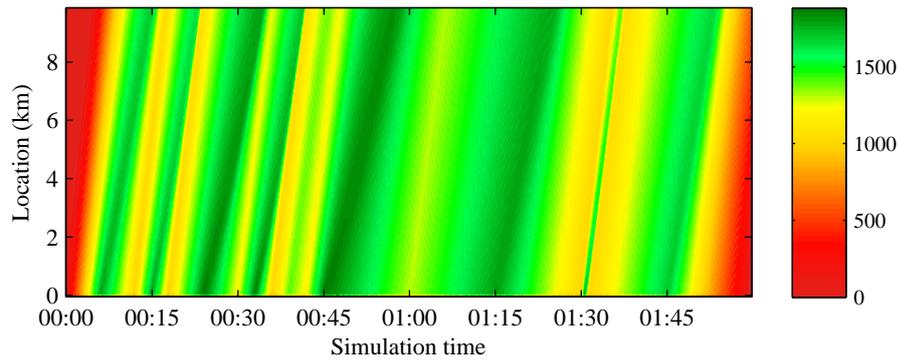
In the second experiment, some parameter values of the traffic flow model are changed, as it would be done in model calibration.

Figure 3.13(a) shows the output of the traffic flow model. Again, congestion and stop-and-go waves emerge at the bottleneck. They now propagate with a speed of $c = 13.4 \frac{\text{km}}{\text{h}}$.

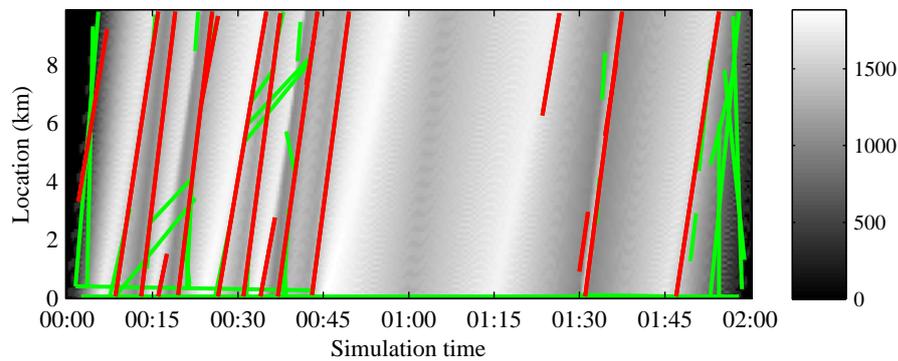
The results of the WSE are shown in Figure 3.13(b). As can be seen, the WSE detects many stop-and-go waves a few kilometers upstream of the bottleneck. The propagation speed detected now is $13.1 \frac{\text{km}}{\text{h}}$.

3.5.3 Experiment 3

In the third experiment, light traffic conditions with a fluctuating inflow rate are simulated to create vehicle platoons. The bottleneck from the previous experiments is



(a) Result of the traffic flow model (flow)



(b) Result of the Wave Speed Estimator

Figure 3.14: Results of Experiment 3

removed so that no congestion emerges.

Figure 3.14(a) shows the spatiotemporal flow output of the model. The vehicle platoons are accompanied by shock waves: at 00:15, for example, a long platoon of vehicles enters the freeway. Similarly, at 01:30, a short platoon is clearly visible in the flow.

Figure 3.14(b) shows the results of the WSE. It detects many shock waves that fall into the range for free-flow waves (red lines). For example, the shock waves created by the platoons mentioned above are correctly estimated. Similarly to the experiments with real data, the speeds of the shock waves in free flow differ largely, as can be seen in the figure. The calculation of an average shock wave speed is therefore omitted.

3.5.4 Discussion

The results show that the Wave Speed Estimator is capable of estimating the speeds of shock waves from synthetic data as well. The propagation speeds detected by the WSE are close to the ones found by manual inspection.

Furthermore, in the congested case, wave speeds were found only where the stop-and

go waves are stabilized. As can be seen in the output, in the proximity of the bottleneck the stop-and-go waves are still growing and merging. A few kilometers upstream of the bottleneck, the stop-and-go waves then stabilize and then propagate with a constant speed and a constant amplitude. Due to its line detector component, the WSE only finds straight lines, and therefore only the stabilized stop-and-go waves.

All components of the WSE are based on matrix operations that are computable in polynomial time. The method is thus scalable if the input size increases. The experiments presented here are calculated in less than 30s per data set on a usual laptop with a CPU clock frequency of 2.5 GHz and main memory of 3.5 GB in a Matlab implementation. The computation time is therefore lower than the manual inspection method.

3.6 Conclusion

In this chapter, we developed and validated a method that automatically detects shock waves and their propagation speeds from spatiotemporal traffic data. It uses image processing methods to detect the transitions between traffic states. Since only the transitions are detected but not the absolute values of the traffic state, this method is robust towards biased data as they occur in speed measurements of the freeways by inductive loops in the Netherlands. By averaging the propagation speed of the shock waves, the characteristic wave speed of traffic regimes can be estimated.

An application of the characteristic shock wave speeds is the estimation of parts of the fundamental diagram. These are important parameters of the traffic state estimators that will be introduced in the following chapter. In the case study of Chapter 7, the method of this chapter is used to automatically calibrate the Adaptive Smoothing Method (Treiber et al., 2011).

A further application is the evaluation of the performance of traffic flow models. This is a recurrent task in model calibration. The WSE can thus be applied in the calibration of traffic flow models that aim at reproducing shock waves with the correct propagation speed.

For further research, the number of stop-and-go waves occurring per time could be estimated, if the WSE is fine-tuned so that both borders of all stop-and-waves can be estimated. Furthermore, the composition of traffic could be estimated based on the shock wave speed according to (3.1). Research questions in this line are how strong a relationship between shock wave speeds and the traffic composition is observed in empirical data, and how precisely the WSE must be able to detect the shock wave speeds in order to determine the traffic composition accurately and precisely.

Chapter 4

Fast Freeway Traffic State Estimation

In this chapter, two fast methods for estimating the current traffic state of a real-size network in real-time are developed and validated. The Wave Speed Estimator developed in the previous chapter thereby forms part of the calibration of these state estimators. The current traffic state is then the basis for the Control component of the control loop of Figure 2.11.

First, a background of freeway traffic state estimation is given in Section 4.1. In Section 4.2, the Adaptive Smoothing Method is reformulated so that it filters road stretch data of realistic size within a few seconds. Its algorithm is reformulated so that it uses the Fast Fourier Transform. In Section 4.3, a fast recursive state estimator based on the Extended Kalman Filter technique is developed and validated. The sensor data are applied only locally where they carry information about the traffic state, instead of globally to the whole network. Both techniques are qualitatively compared in Section 4.4. Conclusions are presented in Section 4.5.

Section 4.2 is a revised version of Schreiter, Van Lint, Treiber, & Hoogendoorn (2010a): “Two Fast Implementations of the Adaptive Smoothing Method Used in Highway Traffic State Estimation” presented at the IEEE Conference on Intelligent Transport Systems. Section 4.3 is a brief version of Van Hinsbergen, Schreiter, Zuurbier, Van Lint, & Van Zuylen (2012): “Localized Extended Kalman Filter for Scalable Real-Time Traffic State Estimation” in IEEE Transactions on Intelligent Transport Systems. A revised version of that paper also appeared as a chapter in the dissertation of Van Hinsbergen (2010).

4.1 Background and Approaches of Traffic State Estimation

One component of the control loop in Figure 2.11 is the estimation of the current traffic state. The current traffic state describes, where the traffic is currently located; commonly, the traffic density for each location is used for that description. The current traffic state is estimated based on sensor data gathered at the road side in real-time. However, the sensor data are usually subjected to noise. Furthermore, the freeway is not completely covered by sensors so that the data contain “holes.” The data thus do not provide the traffic state directly. Therefore, the data have to be analyzed in order to estimate the traffic state.

A simplistic traffic state estimator removes the holes in the data by interpolating the existing data piecewise constantly or piecewise linearly (Van Lint & Van der Zijpp, 2003). These methods assume that traffic behaves equally under all conditions. As mentioned in Chapter 3, in reality, the direction in which information travels through the network depends on the traffic conditions: in free-flow conditions, information travels downstream, but in congested conditions, information travels upstream. Therefore, these simple methods exhibit significant bias (Van Hinsbergen et al., 2008).

One way to take the information direction into account is to smooth the data over space and time dependent on the prevailing traffic conditions. The Adaptive Smoothing Method (ASM) (Treiber et al., 2011) is able to interpolate traffic conditions between detectors over space and time, taking the information direction into account. Originally, the ASM was developed for loop detector data (Treiber & Helbing, 2002), and then generalized to other data sources like floating-car data (Van Lint, 2010; Treiber et al., 2011).

Another way of taking the information direction into account is to use a recursive state estimation technique of Kalman filtering (Kalman, 1960), which combines the sensor data with the traffic state predicted by a traffic flow model. This filter combines the data with the predicted state dependent on the error covariance of the state, the noise of the data, and the difference between the data and the expected observations. A common recursive filter is the Extended Kalman Filter (EKF) (Evensen, 2003), whereby several versions of it have been used in traffic state estimation (Wang & Papageorgiou, 2005; Tampère & Immers, 2007).

Another technique of estimating the traffic state by the combination of a model and data is the particle filter. Mihaylova et al. (2007) fuse speed and flow data from induction loops to estimate the density. A major drawback of particle filters is that they require a huge number of particles, which renders them computationally infeasible for real-time applications. Furthermore, Newtonian relaxation (also known as nudging) combines the prediction of a model with the observation. In contrast to Kalman filter approaches, the correction of state is dependent on the spatial and temporal distance of the state to the measurement location and the difference between data and expected measurement,

and not on the covariance of the state. Herrera & Bayen (2010) developed and implemented Newtonian relaxation for loop and floating-car data. Qing (2011) developed a method to improve loop detector data by combining them with measured travel times without using a traffic flow model. However, since measured travel times are required, the latter approach is only suitable for offline applications.

As decided in Section 2.3.3, in this thesis, we focus on two of the estimation techniques named above, namely the Adaptive Smoothing Method and the Extended Kalman Filter. However, current implementations of them work in real-time only for small-sized networks. In order to use them in real-time applications, this chapter analyzes both the ASM and the EKF for traffic state estimation and subsequently develops faster implementations. Both methods are then tested to prove that they are applicable in real-time for online applications in Dynamic Traffic Management. Finally, in Section 4.4, we reason why we apply the Adaptive Smoothing Method in the case study.

4.2 Optimizing the Adaptive Smoothing Method

The first traffic state estimator discussed is the Adaptive Smoothing Method (ASM) by Treiber & Helbing (2002). Since its initial conception, it has been generalized to multiple data sources (Van Lint & Hoogendoorn, 2010; Treiber et al., 2011) and used in various applications (Van Lint, 2010; Kesting & Treiber, 2008). The ASM estimates the traffic state based on spatio-temporal data. Its methodology is presented in Section 4.2.1. Conventional implementations require multiple minutes to estimate the traffic state. However, Section 4.2.2 reformulates the ASM and solves it by a cross-correlation operation that operates on the whole spatiotemporal data matrix at once. This methodology is even further improved in Section 4.2.3 by solving the cross-correlation with the Fast Fourier Transform, which originates from the field of signal processing. The three implementations of the ASM are tested with data from a Dutch freeway in Section 4.2.4. The results show improvements of computation time up to two orders of magnitude and run within a few seconds (Section 4.2.5). The proposed implementations can therefore replace the conventional implementation in practical applications (Section 4.2.6).

4.2.1 Methodology of the Adaptive Smoothing Method

The ASM takes speed data $v^{\text{raw}}(x, t)$ as input, observed at locations $x \in X^{\text{raw}}$ at times $t \in T^{\text{raw}}$. A second spatio-temporal traffic data variable is optional. For instance, the flow observed at the same points as the speed is used, but other macroscopic quantities such as traffic density can also be used. In the remainder of the paper, the symbol z refers to any macroscopic traffic quantity, whereas v specifically denotes the speed.

The output of the ASM is a continuous, spatio-temporal variable z^{out} . In order to solve the ASM numerically, however, the filtered map is discretized at locations X^{out} and

times T^{out} . Usually, this underlying space-time grid is chosen to be equidistant with resolution Δx^{out} and Δt^{out} , respectively.

The calculation of the output map is based on kinematic wave theory. Depending on the underlying traffic regime, the characteristics of traffic travel with a certain wave speed over space and time. Each of these regimes has one typical wave speed with which the characteristics travel. In congestion, this wave speed is approximately $c_{\text{cong}} = -20 \frac{\text{km}}{\text{h}}$, in free flow approximately $c_{\text{free}} = 80 \frac{\text{km}}{\text{h}}$. The exact values of these two calibration parameters can be determined by applying the Wave Speed Estimator presented in Chapter 3.

The data map z^{raw} is nonlinearly transformed into a smooth map z^{out} , whose elements are a weighted sum of smoothed elements of both traffic regimes:

$$z^{\text{out}}(x, t) = w(x, t) \cdot z^{\text{cong}}(x, t) + [1 - w(x, t)] \cdot z^{\text{free}}(x, t). \quad (4.1)$$

The intermediate functions z^{cong} and z^{free} represent the traffic in congested and in free-flow conditions, respectively. The weighting factor w depends on the underlying traffic regimes. The congested function z^{cong} is defined by

$$z^{\text{cong}}(x, t) = \frac{1}{n^{\text{cong}}(x, t)} \sum_{x_i} \sum_{t_j} \varphi^{\text{cong}}(x_i - x, t_j - t) \cdot z^{\text{raw}}(x_i, t_j) \quad (4.2)$$

with the normalization factor

$$n^{\text{cong}}(x, t) = \sum_{x_i} \sum_{t_j} \varphi^{\text{cong}}(x_i - x, t_j - t), \quad (4.3)$$

whereby the sums cover all data locations $x_i \in X^{\text{raw}}$ and data times $t_j \in T^{\text{raw}}$. The smoothing kernel

$$\varphi^{\text{cong}}(x, t) = \exp \left(-\frac{|x|}{\sigma} - \frac{\left| t - \frac{x}{c_{\text{cong}}} \right|}{\tau} \right) \quad (4.4)$$

is an exponential function with the spatial parameter σ and the temporal parameter τ . The characteristic congested wave speed c_{cong} influences the skew of the kernel.

The free-flow function z^{free} is similarly defined to z^{cong} (4.2)–(4.4) with a normalization factor n^{free} , the free-flow smoothing kernel φ^{free} and the free-flow wave speed c_{free} .

The weighting factor w in (4.1) depends on the intermediate speed functions v^{cong} and v^{free} :

$$w(x, t) = \frac{1}{2} \left[1 + \tanh \left(\frac{v_{\text{crit}} - v^*(x, t)}{\Delta v} \right) \right] \quad (4.5)$$

with critical speed v_{crit} , transition speed range Δv and

$$v^*(x, t) = \min \left(v^{\text{cong}}(x, t), v^{\text{free}}(x, t) \right). \quad (4.6)$$

For details about the ASM refer to Treiber et al. (2011).

In conventional algorithms, the double sum of z^{cong} (4.2) is coded as a double loop. In scientific simulation tools that are specialized in matrix operations such as Matlab, the execution of “for”-loops is particularly slow.

The Computation Time Complexity of the ASM

The computationally most complex function is the calculation of z^{cong} (4.2) (and similarly z^{free}). The conventional implementation loops over space and time, where the number of filter points in these dimensions is $\frac{X}{\Delta x^{\text{out}}}$ and $\frac{T}{\Delta t^{\text{out}}}$, respectively, with X and T denoting the length of the resulting rectangle in space and time, respectively.

For practical reasons, not all data are taken into account, because the values of the kernel φ^{cong} (4.4) quickly approach zero. Therefore, only a certain space-time rectangle around the filter point (x, t) is relevant for the computation. Usually, this rectangle is chosen to be of length $2a\sigma$ in space and $2a\tau$ in time, for a kernel width factor a . A value of $a = 5$ provides good results for an exponential kernel.

The number of these relevant data points therefore depends on a and the average data resolution Δx^{raw} and Δt^{raw} . To conclude, the computation time complexity of the conventional method is

$$ASM \in \mathcal{O} \left(\frac{X}{\Delta x^{\text{out}}} \cdot \frac{T}{\Delta t^{\text{out}}} \cdot \frac{\sigma \tau a^2}{\Delta x^{\text{raw}} \cdot \Delta t^{\text{raw}}} \right). \quad (4.7)$$

(The Big-Oh notation here denotes the growth rate of the computation with respect to the input variables for large values. For example, the computation time grows linear in the road stretch length X , meaning doubling the road stretch length leads to a doubling of the computation time.)

4.2.2 The Adaptive Smoothing Method Solved with the Cross-correlation

The crucial part of the ASM is the computation of the intermediate regime functions z^{cong} and z^{free} (4.2). This equation can be solved in a different way, relying on matrix arithmetics suitable for Matlab. For this purpose, the continuous equations of the ASM are discretized and solved with the two-dimensional cross-correlation \otimes , which is a fast operation. The cross-correlation is defined as

$$(A \otimes B)(m, n) = \sum_{\mu} \sum_{\nu} A(\mu, \nu) \cdot B(m + \mu, n + \nu), \quad (4.8)$$

where A and B are matrices. This section presents the algorithm of the cross-correlation implementation.

By applying a shift of indices with $\xi_i = x_i - x$ and $\tau_j = t_j - t$, the numerator of z^{cong} can be expressed by a cross-correlation operation:

$$\sum_{\xi} \sum_{\lambda} \varphi^{\text{cong}}(\xi, \lambda) \cdot z^{\text{in}}(x + \xi, t + \lambda) = (\varphi^{\text{cong}} \otimes z^{\text{in}})(x, t). \quad (4.9)$$

This is a continuous function; however, to solve the ASM numerically, a discretized version is used, where these functions are sampled at equidistant points in space and time. For this purpose, the data z^{raw} are discretized at equidistant points with the spatial resolution Δx^{out} and the temporal resolution Δt^{out} to the matrix Z^{in} . The congested matrix then reads

$$Z_{lk}^{\text{cong}} = \frac{\sum_i \sum_j \Phi_{ij}^{\text{cong}} \cdot Z_{l+i, k+j}^{\text{in}}}{\sum_i \sum_j \Phi_{ij}^{\text{cong}} \cdot M_{l+i, k+j}} = \frac{(\Phi^{\text{cong}} \otimes Z^{\text{in}})_{lk}}{(\Phi^{\text{cong}} \otimes M)_{lk}}, \quad (4.10)$$

with a discretized smoothing kernel Φ^{cong} and a binary indication matrix M . The division of the matrices in (4.10) is element-wise. The denominator originates from the normalization function n^{cong} (4.3), which is a weighted sum of the kernel at all data points. In the discretized version, only elements of Φ^{cong} that correspond to a data point should be summed up. Therefore, the indication matrix M is defined by

$$M_{lk} = \begin{cases} 1 & \text{if data at } Z_{lk}^{\text{in}} \text{ available} \\ 0 & \text{else} \end{cases}. \quad (4.11)$$

The following algorithm explains the steps of the cross-correlation in more detail.

The Algorithm of the ASM solved with Cross-correlation

The ASM is solved by the cross-correlation in the following four sequential steps.

1) Discretization of data The data maps in the equations of the ASM are discretized at equidistant points with the spatial resolution Δx^{out} and the temporal resolution Δt^{out} . The data point $z^{\text{raw}}(x, t)$ is mapped to the discretized point in Z^{in} which is closest to (x, t) . All remaining elements in Z^{in} are set to zero.

To clarify this discretization, consider the following example where the following data matrices are given:

$$\text{speed data } V^{\text{raw}} = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & f & 9 \end{bmatrix}, \quad (4.12)$$

$$\text{at locations } X^{\text{raw}} = [0 \quad 1000 \quad 2900], \quad (4.13)$$

$$\text{at times } T^{\text{raw}} = [0 \quad 60 \quad 120], \quad (4.14)$$

where f indicates missing data. Let the output resolution be $\Delta x^{\text{out}} = 500$ and $\Delta t^{\text{out}} = 30$. Then, the discretized variables read

$$X^{\text{out}} = [0 \ 500 \ 1000 \ \dots \ 3000] , \quad (4.15)$$

$$T^{\text{out}} = [0 \ 30 \ 60 \ 90 \ 120] , \quad (4.16)$$

$$V^{\text{in}} = \begin{bmatrix} 1 & 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 & 0 \\ 4 & 0 & 5 & 0 & 6 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 7 & 0 & 0 & 0 & 9 \end{bmatrix} , \quad M = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{bmatrix} . \quad (4.17)$$

Note that a discretization error occurs in this example: the data at location 2900 are relocated to the nearest sample point at 3000. In practice, these discretization errors are negligible, if the output resolution is chosen high, for example 100 m.

2) Discretization of the kernel Next, the kernel matrix φ^{cong} (4.4) is discretized with the same resolution Δx^{out} and Δt^{out} , where the maximum is in the center of the matrix:

$$X^{\text{kern}} = [-a \cdot \sigma, \dots, -\Delta x^{\text{out}}, 0, \Delta x^{\text{out}}, \dots, a \cdot \sigma] \quad (4.18)$$

The temporal points T^{kern} are defined in a similar way. The congestion kernel matrix is defined as

$$\Phi_{ij}^{\text{cong}} = \exp \left(-\frac{|X_i^{\text{kern}}|}{\sigma} - \frac{\left| T_j^{\text{kern}} - \frac{X_i^{\text{kern}}}{c_{\text{cong}}} \right|}{\tau} \right) \quad (4.19)$$

The free flow kernel matrix Φ^{free} is defined similarly.

3) Apply smoothing kernels by cross-correlation Apply the cross-correlation \otimes to determine the intermediate regime functions Z^{cong} and Z^{free} (4.10).

4) Weighting and summing The result Z^{out} is computed by weighting (4.5) and summing (4.1) the intermediate regime matrices (4.10):

$$W = \frac{1}{2} \cdot \left(1 + \tanh \left[\frac{v_{\text{crit}} - \min\{V^{\text{cong}}, V^{\text{free}}\}}{\Delta V} \right] \right) , \quad (4.20)$$

$$Z^{\text{out}} = W \cdot Z^{\text{cong}} + (1 - W) \cdot Z^{\text{free}} , \quad (4.21)$$

with element-wise minimum, element-wise tanh and element-wise multiplication. Z^{out} is the output of the cross-correlation implementation of the ASM.

The Computation Time Complexity of the ASM Solved using the Cross-correlation

The complexity is significantly defined by the cross-correlation operation (4.10). In the following, the runtime complexity of numerator $\Phi^{\text{cong}} \otimes Z^{\text{in}}$ is analyzed; similar results hold for the denominator.

The complexity of the cross-correlation of two matrices $A \in \mathbb{R}^{N_A \times M_A}$ and $B \in \mathbb{R}^{N_B \times M_B}$ is (Cormen et al., 2001)

$$A \otimes B \in \mathcal{O}(N_A M_A \cdot N_B M_B) . \quad (4.22)$$

The size of the input matrix Z^{in} is linearly dependent on the road stretch length X and the filter grid resolution Δx^{out} :

$$|Z^{\text{in}}| \in \mathcal{O}\left(X \cdot \frac{1}{\Delta x^{\text{out}}} \cdot T \cdot \frac{1}{\Delta t^{\text{out}}}\right) . \quad (4.23)$$

The size of the smoothing kernel matrix Φ^{cong} is linearly dependent on the kernel width factor a and the resolution Δx^{out} :

$$|\Phi^{\text{cong}}| \in \mathcal{O}\left(\sigma a \cdot \frac{1}{\Delta x^{\text{out}}} \cdot \tau a \cdot \frac{1}{\Delta t^{\text{out}}}\right) . \quad (4.24)$$

The cross-correlation operation is therefore the product (4.22) of both complexities (4.23) and (4.24):

$$\Phi^{\text{cong}} \otimes Z^{\text{in}} \in \mathcal{O}\left(X \cdot T \cdot \frac{\sigma \tau a^2}{(\Delta x^{\text{out}})^2 \cdot (\Delta t^{\text{out}})^2}\right) . \quad (4.25)$$

The remaining operations in the cross-correlation implementation are multiplications, additions and other element-wise operations, which are less complex than the cross-correlation. Since cross-correlation implementation is a sequential algorithm, its complexity equals the complexity of the most complex operation (4.25).

The algorithm is linearly complex in the road stretch length X , which makes it easily scalable in practical applications. The quadratic complexity in the resolution Δx^{out} might be problematic, because a high resolution is needed to minimize discretization errors caused by the equidistant input grid of Z^{in} . The argument in the temporal dimension is similar because of the symmetry of the algorithm. Nevertheless, the experiments will show that the algorithm runs very fast even for very high resolutions.

The complexity of this cross-correlation implementation (4.25) is nearly equal to the one of the conventional implementation (4.7). The only difference is an exchange of the variables of the data resolution with variables of the filter data resolution. However, although there seems to be no improvement of the complexity class, there is a speedup in practical runtime, as will be presented in Section 4.2.5.

4.2.3 The Adaptive Smoothing Method Solved with the Fast Fourier Transform

A fast implementation of the cross-correlation already exists. However, an even faster computation is possible with the Fast Fourier Transform (FFT). The FFT and the cross-correlation are both related to the convolution $*$, which is defined as

$$(A * B)_{mn} = \sum_{\mu} \sum_{\nu} A_{\mu\nu} \cdot B_{m-\mu, n-\nu} \quad (4.26)$$

of two matrices A and B . In the special case that one of the matrices is symmetric, i.e.

$$A_{mn} = A_{M-m+1, N-n+1}, \quad (4.27)$$

the cross-correlation \otimes equals the convolution $*$. In all cross-correlation operations of Section 4.2.2, the kernel matrices Φ^{cong} or Φ^{free} are involved. Since these are indeed symmetric, the cross-correlation operations of the previous section can be replaced by convolution operations.

The convolution law

$$A * B = \mathcal{F}^{-1}(\mathcal{F}(A) \cdot \mathcal{F}(B)) \quad (4.28)$$

connects the convolution $*$ and the Discrete Fourier Transform \mathcal{F} , where \cdot here denotes the element-wise multiplication of two (complex) matrices. The Discrete Fourier Transform (DFT)

$$(\mathcal{F}(A))_{mn} = \sum_{\nu=1}^N \left(e^{-\frac{2\pi i \cdot \nu}{N}} \cdot \sum_{\mu=1}^M e^{-\frac{2\pi i \cdot \mu}{M}} A_{\mu\nu} \right) \quad (4.29)$$

converts a two-dimensional discrete signal from its space-time domain into the frequency domain. This operation is inverted by the Inverse Discrete Fourier Transform (IDFT)

$$(\mathcal{F}^{-1}(A))_{mn} = \frac{1}{NM} \sum_{\nu=1}^N \left(e^{\frac{2\pi i \cdot \nu}{N}} \sum_{\mu=1}^M e^{\frac{2\pi i \cdot \mu}{M}} A_{\mu\nu} \right). \quad (4.30)$$

The convolution of two matrices can therefore be computed by converting these matrices into the frequency domain by the DFT, multiplying their results element-wise, and transforming this product back into the space-time domain by the IDFT (4.28).

The Fast Fourier Transform (FFT) is a fast implementation of the DFT, provided that the input matrices are of the same size and that their number of elements in each dimension is a power of two. The efficiency of the FFT is widely exploited in other scientific fields, for example in mathematics to speed up polynomial multiplications (Cormen et al., 2001), in the digital image format JPEG (Wallace, 1992) (instead of the FFT itself, the slightly different Discrete Cosine Transform (DCT) is used), but also in traffic engineering to efficiently determine the reliability of travel times (Ng & Waller, 2010).

The Algorithm of the ASM Solved with the FFT

The computation of the ASM solved with the FFT is based in the cross-correlation implementation of Section 4.2.2. Every cross-correlation operation (4.10) is replaced by the convolution law (4.28).

To make the matrices of same size and their length in each direction a power of two, they are padded with zeros. Furthermore, the size of the zero-padding area has to be sufficiently large to avoid overlapping of the circular FFT operations: the data map has to be zero-padded in each dimension with at least the size (4.18) of the kernel matrix (4.19). After the IFFT operation, the padded zeros are removed to restore the original size of the matrices. This zero-padding procedure ensures that the FFT implementation leads to the same results as the cross-correlation implementation.

The remaining operations of the FFT implementation are equal to the cross-correlation implementation.

The Computation Time Complexity of the ASM Solved with FFT

The complexity of the FFT implementation differs from the cross-correlation implementation only in the computation of the actual cross-correlation terms.

Let $A \in \mathbb{R}^{N \times M}$ and $B \in \mathbb{R}^{N \times M}$ be matrices of the same size. Then, the complexity of the FFT (4.29)

$$\mathcal{F}(A) \in \mathcal{O}(NM \log NM) \quad (4.31)$$

is nearly linear in the number of elements (Cormen et al., 2001). The IFFT (4.30) is in the same complexity class, due to the similar definition. The elementwise multiplication in (4.28) is linear in the number of elements. In conclusion, the cross-correlation can be solved with the FFT

$$A \otimes_{\text{FFT}} B \in \mathcal{O}(NM \log NM) \quad (4.32)$$

in less than quadratic time.

Note that the size of the kernel Φ^{cong} must be the same as of the data map Z^{in} . The computation time complexity of this equation is therefore

$$\Phi^{\text{cong}} \otimes_{\text{FFT}} Z^{\text{in}} \in \mathcal{O} \left(\frac{XT}{\Delta x^{\text{out}} \Delta t^{\text{out}}} \log \frac{XT}{\Delta x^{\text{out}} \Delta t^{\text{out}}} \right). \quad (4.33)$$

The comparison of the complexity of the FFT implementation (4.33) with the cross-correlation implementation (4.25) shows that the FFT implementation is less complex in the resolution; i.e., the higher the resolution, the faster the FFT implementation runs compared to the cross-correlation implementation. (In fact, the log part in (4.33) is barely noticeable in practice; the quadratic complexity of the resolution is therefore

Parameter	Value(s)	Description
X	$\{5, 10, 20, 30\} \frac{\text{km}}{\text{h}}$	road section length
T	$\{2, 4, 6, 8, 10\} \text{h}$	measurement time length
Δx^{out}	$\{10, 25, 50, 100\} \text{m}$	spatial resolution of results
Δt^{out}	30 s	temporal resolution of result
c_{cong}	$-18 \frac{\text{km}}{\text{h}}$	congested wave speed
c_{free}	$80 \frac{\text{km}}{\text{h}}$	free-flow wave speed
Δv	$10 \frac{\text{km}}{\text{h}}$	length of transition region
v_{crit}	$70 \frac{\text{km}}{\text{h}}$	critical speed
σ	500 m	spatial kernel length
τ	60 s	temporal kernel length
a	5	size of kernel matrix

Table 4.1: Simulation parameter settings for comparing the ASM implementations

reduced to nearly linear.) In contrast, the FFT method is more complex in the road length X and the measurement time T , favoring the cross-correlation implementation for very large data sets.

The complexity is insensitive to the resolution of the data points Δt^{raw} and Δx^{raw} . This allows for the combining of loop detector data with an arbitrary amount of floating-car data without losing runtime.

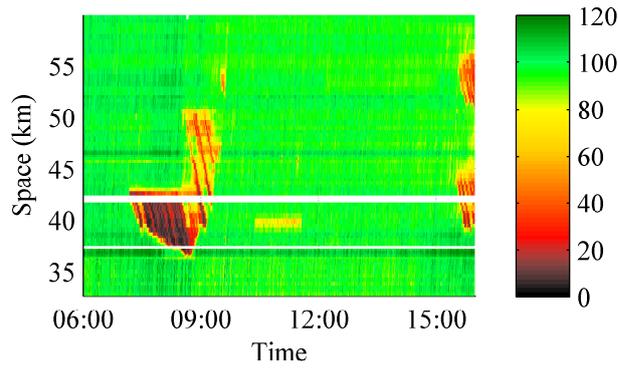
4.2.4 Experimental Setup to Compare the Three Implementations

To test the quality and computation time of the proposed implementations of the ASM against the conventional one, simulations are performed. In order to measure comparable computation times, all implementations are executed with the same parameter settings and run on an idle state-of-the-art laptop.

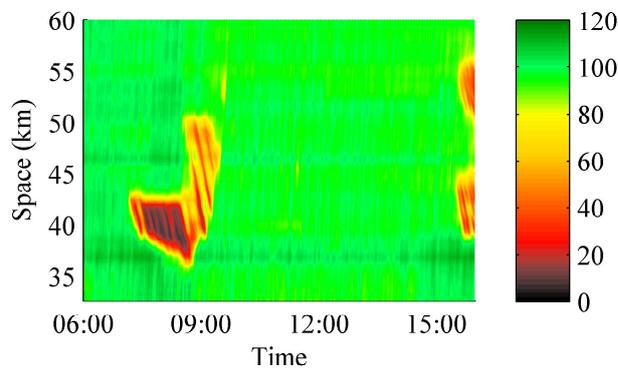
Three parameters are varied: the road length X , the measurement time T , and the resolution Δx^{out} . For every combination, simulation runs of data over 10 days are performed. The parameter settings are listed in Table 4.1.

Raw data were gathered from dual-loop detectors from the Dutch A15 East between Kilometers 30 and 60 between 06:00 and 16:00 from 01-10-2009 to 10-10-2009. These observed speed and flow aggregated over 1 min at every 500m on average.

The computation time is averaged arithmetically over these 10 observation days. The quality of the filter result is measured by the Root Mean Squared Error (RMSE) and the Mean Absolute Percentage Error (MAPE) of the speed and flow maps of the cross-correlation and the FFT implementation against the conventional ASM.



(a) Raw data



(b) Filtered data (by FFT implementation)

Figure 4.1: Input and output speed data of the ASM

4.2.5 Results and Discussion

An example of input and output data of the ASM is shown in Figure 4.1. The induction loops of the A15 eastbound near Rotterdam, the Netherlands, collected speed data over a length of $X = 30$ km and a measurement time of $T = 10$ h. Both traffic regimes, free flow (green) and congestion (yellow and red), are present. Note that some detectors do not provide data, leaving holes in the data map (Figure 4.1(a)). The ASM smooths these data, resulting in a complete speed map (Figure 4.1(b)). In this figure, the FFT implementation was used.

Figure 4.2 shows the relationship between computation time and the number of filter grid data points. Huge differences in computation time between the three implementations are apparent. The computation time of the conventional method increases sharply up to several minutes for data sets of practical size, whereas the cross-correlation and especially the FFT run much faster.

Figure 4.3 shows the computation times in more detail. In each subplot, the computation time is plotted against one of the three variables varied from Table 4.1, whereas the remaining two are fixed. (Note that the scale differs between the subplots.) In the largest data test with a road length of $X = 30$ km, a measurement time $T = 10$ h

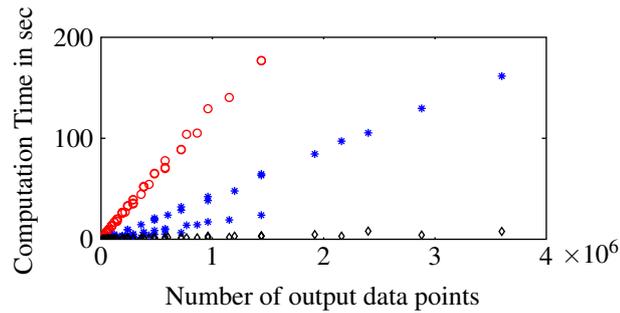


Figure 4.2: Computation time of the ASM against the number of filter points of the three implementations: conventional (red circles), proposed cross-correlation (blue stars), proposed FFT (black diamonds)

and fine-grained resolution of $\Delta x^{\text{out}} = 10\text{m}$, the computation time difference is drastic (Figure 4.3(c)). The conventional implementation takes almost five minutes, whereas the cross-correlation computes the result in less than two minutes. The FFT, however, performs even faster, with a computation time of less than ten seconds.

Table 4.2 lists the speedup of the FFT implementation against the conventional ASM for all varying variables, averaged over the ten days. This speedup factor is at least one order of magnitude. The highest speedup gained, however, was more than a factor of 100. In the cross-correlation implementation, speedup factors between 2.7 and 25 were observed.

In Table 4.3, the errors between the two implementations proposed and the conventional implementation were calculated. The mean absolute percentage error (MAPE) of the speed and the flow maps for both implementations is significantly less than half a percent. The root mean squared error (RMSE) of the flow map is approximately $10 \frac{\text{veh}}{\text{h}}$, and of the speed map at less than $0.2 \frac{\text{km}}{\text{h}}$. Further, both implementations proposed have the same error.

The filter of the two new presented implementations is therefore extremely close to the conventional implementation, leaving only very small errors due to the discretization of the data points. Other than the small discretization errors, the quality of the proposed implementations is equal to the conventional one.

4.2.6 Conclusion of the Optimization of the Adaptive Smoothing Method

The reformulation of the ASM treats the input data not as single points, but operates on the spatiotemporal data matrix as a whole. By interpreting the ASM as a cross-correlation and subsequently solving it by the Fast Fourier Transform, the computation time is drastically reduced while preserving the estimation quality. Experiments based on real data with realistic data sizes show a gain in computation time of a factor of up to one hundred. In practice, the ASM now runs within a few seconds, rendering it applicable for real-time purposes.

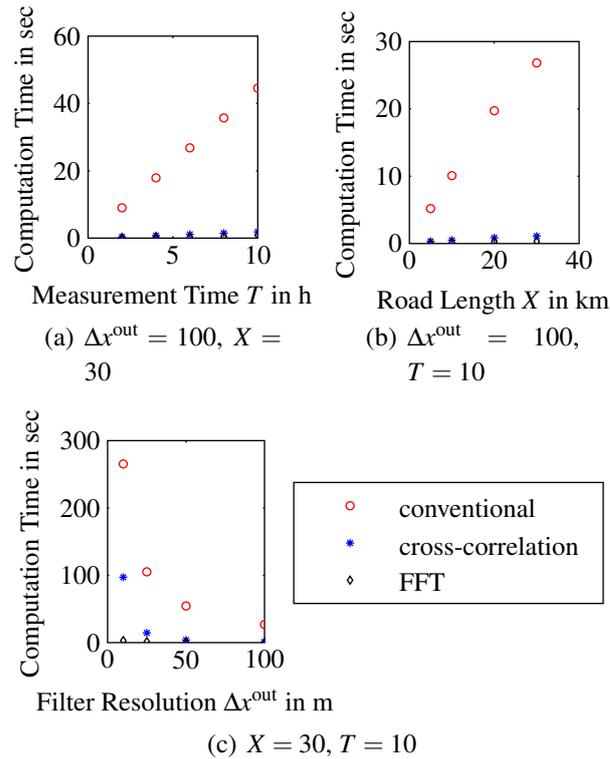


Figure 4.3: Computation time of the ASM against the varied variables (c.f. Table 4.1)

4.3 Optimizing Traffic State Estimation Based on the Extended Kalman Filter

In this section, the recursive filter technique of the Extended Kalman Filter (EKF) is analyzed and improved. In contrast to the ASM of the previous section, a recursive filter employs a predictive (traffic flow) model, which describes how the (traffic) system evolves over time. To correct for model inaccuracies, the system state is updated with sensor data. Section 4.3.1 explains the methodology of the EKF. In common EKF implementations, data are used to correct the traffic state of the whole network. However, Section 4.3.2 shows that a measurement provides information only in the physical vicinity of the sensor. The EKF is therefore localized to update the traffic state only in the vicinity of the sensor. Experiments are set up in Section 4.3.3; its results show that the local application of measurements leads to a significant speed up in computation time while preserving the estimation quality (Section 4.3.4). Experiments in a real-sized network prove that the Localized EKF (L-EKF) is applicable in real-time (Section 4.3.5). Conclusions are drawn in Section 4.3.6.

(a) $\Delta x^{\text{out}} = 10 \text{ m}$

	2h	4h	6h	8h	10h
5 km	43.5	35.5	62.4	58.4	43.4
10 km	59.4	43.5	72.7	69.5	53.5
20 km	53.9	42.4	59.1	56.3	40.1
30 km	69.2	53.1	87.9	85.6	57.3

(b) $\Delta x^{\text{out}} = 25 \text{ m}$

	2h	4h	6h	8h	10h
5 km	43.4	32.2	43.9	40.3	28.8
10 km	46.7	36.7	47.7	43.3	33.3
20 km	64.3	48.0	76.5	73.5	58.4
30 km	62.5	48.2	75.5	72.0	59.4

(c) $\Delta x^{\text{out}} = 50 \text{ m}$

	2h	4h	6h	8h	10h
5 km	47.2	33.0	43.6	39.8	28.3
10 km	71.6	48.7	68.9	61.9	42.4
20 km	58.1	44.4	57.8	53.3	39.6
30 km	87.6	60.3	106.0	97.9	73.4

(d) $\Delta x^{\text{out}} = 100 \text{ m}$

	2h	4h	6h	8h	10h
5 km	48.8	35.0	44.4	39.8	26.7
10 km	60.3	41.9	56.0	50.5	36.0
20 km	86.6	56.8	81.5	72.5	49.8
30 km	88.2	59.4	84.0	75.0	52.2

Table 4.2: Speedup factor of the FFT implementation of the ASM against the conventional implementation, averaged over all 10 days

4.3.1 Methodology of Extended Kalman Filtering

This section first presents an overview of the methodology and then shows the two parts of prediction and correction in detail.

4.3.1.1 Overview of Kalman Filtering

The Kalman filter (Kalman, 1960) is a recursive filter, i.e. it consists of a prediction and a correction component as illustrated in Figure 4.4. Its purpose is to estimate the true (traffic) state $\tilde{\mathbf{k}}_j$ at time step j , which is here the traffic density over space. Since the state is not directly visible, sensors observe the system and provide measurements \mathbf{z}_j .

Since both the traffic system and the sensors are not perfectly known, the true traffic

	Speed	Flow
MAPE cross-correlation	0.097 %	0.301 %
MAPE FFT	0.097 %	0.301 %
RMSE cross-correlation	0.130 $\frac{\text{km}}{\text{h}}$	9.198 $\frac{\text{veh}}{\text{h}}$
RMSE FFT	0.130 $\frac{\text{km}}{\text{h}}$	9.198 $\frac{\text{veh}}{\text{h}}$

Table 4.3: Error measurements of the ASM implementations proposed against the conventional ASM implementation

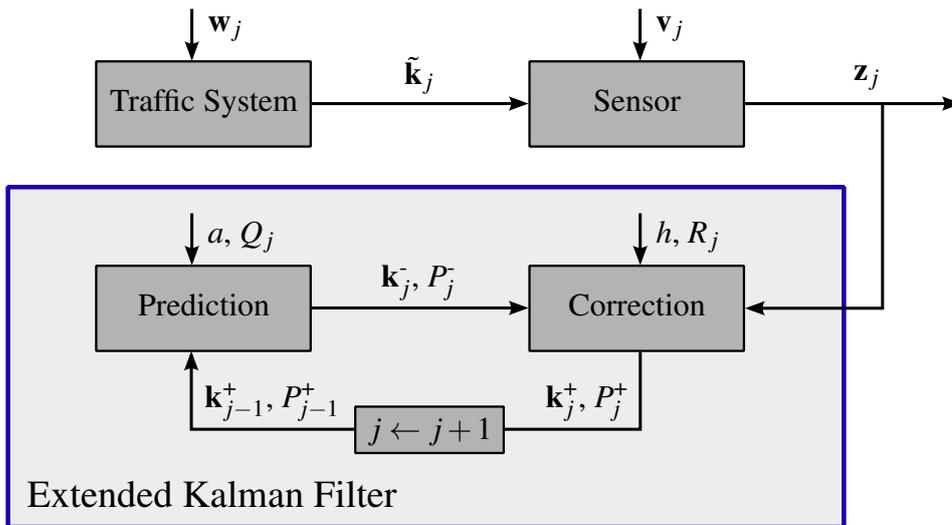


Figure 4.4: Structure of the Extended Kalman Filter

system cannot be estimated perfectly. However, it is possible to estimate a probability density function (pdf), which represents the distribution of possible states. In general, any pdf is possible to represent this belief state, and many filters have been proposed.

One of the most commonly recursive filter techniques applied in practice is the Kalman Filter. Since traffic is a non-linear process, extensions of the linear Kalman filter are used. The Extended Kalman Filter (EKF) (Sorenson, 1985) linearizes the model at the current state; examples of the EKF in traffic applications are Wang & Papageorgiou (2005) and Tampère & Immers (2007). The Unscented Kalman Filter (UKF) (Julier & Uhlmann, 1997) avoids such a linearization by using a small sample set of possible system states instead; the UKF has been applied in traffic state estimation by Ngoduy (2008). The Ensemble Kalman Filter (EnKF) (Evensen, 2003) is applicable for highly non-linear systems. It describes the belief state by a large non-deterministically determined sample of possible system states. The EnKF was applied in freeway traffic state estimation based on floating-car data (Work et al., 2008). For a discussion on Kalman filter techniques applicable for traffic state estimation, refer to Blandin et al. (2012).

Another recursive filter is the Hybrid Density Filter (HDF) (Huber & Hanebeck, 2007), which models the belief state by a Gaussian Mix density, which allows the description

of a multi-modal probability density. Since the representation of the pdf requires many parameters, the computation time of that filter is relatively high compared to Kalman-like filters. Up to now, the HDF has been applied to state estimation and optimization of an autonomous vehicle in computer simulations (Weissel et al., 2008), but an application to network-wide state estimation used for Dynamic Traffic Management is expected to be computationally unfeasible.

The Kalman filter and its extension thus are the commonly used recursive filters in traffic state estimation. In the remainder of this section, we use the Extended Kalman Filter, though the methodology is applicable to other recursive filters as well.

4.3.1.2 Prediction Step

The EKF models the probability function by a Gaussian distribution, which is characterized by the mean state vector \mathbf{k}_j^+ and the error covariance matrix P_j^+ at time step j . The real traffic system is modeled by applying a traffic flow model a . Since the system is usually not perfectly known, a noise term \mathbf{v}_j is taken into account. In the EKF, this noise is assumed to be additive and Gaussian, and is modeled by the system noise covariance matrix Q_j of appropriate size. The estimated mean state and covariance matrix are thus predicted over time by

$$\bar{\mathbf{k}}_j = a(\mathbf{k}_{j-1}^+) \quad (4.34)$$

$$P_j = A_j P_{j-1}^+ A_j^T + Q_j, \quad (4.35)$$

whereby the matrix A_j is the linearization of the system model a at the current estimated mean state \mathbf{k}_j^+ (Jazwinski, 1970):

$$A_j = \nabla a(\mathbf{k})|_{\mathbf{k}_{j-1}^+}. \quad (4.36)$$

4.3.1.3 Correction Step

Because of the system noise Q_j , the estimate of the state estimate becomes more uncertain over time, so that the error covariance matrix P_j grows over time. Therefore, the estimate is corrected by taking the measurements \mathbf{z}_j into account. The sensors are modeled by the function h . By using loop detectors, for example, a relation between density and flow or density and speed, commonly referred to as the fundamental diagram, is used. However, since the sensors are not perfectly known, too, a noise term \mathbf{v}_j is taken into account. In the EKF, this noise is represented by the measurement noise covariance matrix R_j of appropriate size. The estimated state is updated according to Bayes' law by

$$C_j = P_j H_j^T (H_j P_j H_j^T + R_j)^{-1} \quad (4.37)$$

$$\mathbf{k}_j^+ = \bar{\mathbf{k}}_j + C_j (\mathbf{z}_j - h(\bar{\mathbf{k}}_j)) \quad (4.38)$$

$$P_j^+ = P_j - C_j H_j P_j. \quad (4.39)$$

The Kalman gain C_j (4.37) indicates how much the state estimate is corrected based on the relative values of the uncertainties of the a priori state estimate (through P_j^-) and of the measurements (through R_j). The matrix H_j is the linearization of the sensor model h at the current estimated a priori mean state \mathbf{k}_j^- :

$$H_j = \nabla h(\mathbf{k}_j^-)|_{\mathbf{k}_j^-}. \quad (4.40)$$

The result of the EKF is an a posteriori state vector \mathbf{k}_j^+ (4.38) and an a posteriori estimate of error covariance matrix P_j^+ (4.39), which are a balanced estimate of the traffic state, given both the prediction of the traffic model and the sensor data.

Note that the EKF is generally applied with a Gaussian assumptions on both the distributions of the data and the model. Generally, Gaussian distributions are not found in practice in traffic. However, the EKF can still be applied when distributions are non-Gaussian. The value of the EKF has been shown in the many cases where it has been applied for traffic state estimation (Van Hinsbergen et al., 2008; Wang & Papageorgiou, 2005; Tampère & Immers, 2007; Wang et al., 2007; Zuurbier et al., 2006; Van Lint et al., 2008a).

4.3.1.4 Global Extended Kalman Filter

Usually, the EKF is applied at once to the entire network, so that state vector \mathbf{k}_j represents all cells in the entire network and P_j contains estimates of the covariance of the errors between all cells (Wang & Papageorgiou, 2005; Zuurbier et al., 2006). Each time, when measurements become available somewhere in the network, the densities in all cells are corrected at once. This process, which is termed Global EKF (G-EKF) here, uses the available data to its maximum potential, as all densities in all cells are corrected using the error covariance between all measured cells and all non-measured cells. However, this procedure has one major concern, namely the calculation times can become very high.

The EKF contains two expensive operations: the matrix inverse operation in (4.37) that scales in the number of measurements, and the matrix multiplications of (4.39) that scales in the number of cells in the network. Theoretically, both of these operations scale in the order of $\mathcal{O}(M^{2.8074})$ with the Strassen algorithm (Strassen, 1969). For larger networks (containing more than, e.g., a few hundred measured cells), the complexity of these operations will make real-time calculations impossible on a normal computer, rendering the G-EKF infeasible for large-scale online applications.

4.3.2 The Localized Extended Kalman Filter

In this section, a new EKF implementation that is much faster on larger networks is proposed because it simplifies the matrix inverse operation of the correction step (4.37). First, it is important to notice that error covariance matrix P_j contains many values close to zero.

4.3.2.1 Progression of the Covariance over the Network

As the traffic system progresses over time, a nonzero error covariance can exist between the errors of any two cell states. This section shows that the covariance under most conditions decrease as the distance between two cells increases.

Through (4.35) and (4.39), it can be seen that the covariance is influenced by the linearization H_j of the fundamental diagram, the linearization A_j of the model, and Kalman gain C_j . Because of the nonlinearity of the system and the stochasticity of the model and the data, it is very hard to analytically prove under which conditions the covariance will decrease with increasing distance. However, through extensive experimentation on different networks with different sizes, it has been observed that, under most conditions, the error covariance between two cells further away is smaller than between two cells close to each other. This is a very intuitive result: only a very small portion of traffic at a certain location will travel to another location, for example 10 km away; therefore, it can be expected that the error covariance between these two locations is practically zero.

Figure 4.5 shows the error covariance between the cells on a certain route. For this result, the small network as will be presented later (see Figure 4.7) was simulated for 600 time steps. The demand and supply of the origins and destination were varied to cause state transitions. In most cases, the covariance between two cells is smaller the further they are apart; only in the third case is the covariance between cell 1 and cell 9 larger than before. However, further downstream, the error covariance is again very close to zero. Similar results were obtained on different networks with other structures and other congestion patterns.

The fact that the covariance values are usually smaller further away from a certain cell means that, in the G-EKF, matrix P_j will contain many values close to zero for cells far apart in the network. Corrections to states based on these very small covariance values will be negligible.

It is important to note that the nonzero values will not always be close to the diagonal of the matrix, because cells that are spatially close in a network cannot be guaranteed to be close to each other in the matrix. In addition, experimentation has shown that P_j is not always diagonally dominant. These two issues prevent more efficient algorithms from being applied for the inverse operation, and an alternative is required.

This section therefore proposes to use the structure of the network topology in the corrections and to use a measurement to correct only the states of cells in the vicinity of the detector. The resulting scheme is named Localized EKF (L-EKF) to indicate the local nature of the corrections.

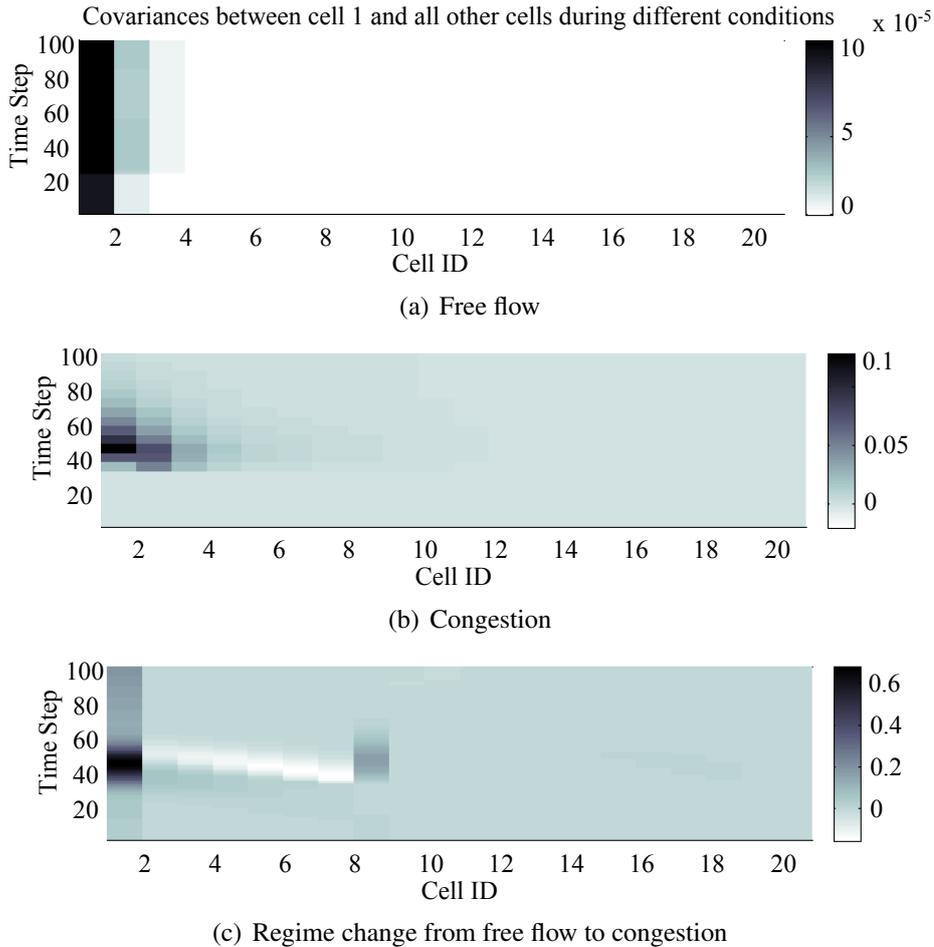


Figure 4.5: Error covariance values of the first cell of the 21-cell link and all other cells on the link under three different conditions: A dark color indicates high covariance. The gradient indicates that the covariance decreases the further the two cells are apart. Note that different scales apply to the different figures.

4.3.2.2 The Algorithm of the Localized Extended Kalman Filter

In the L-EKF, many local EKFs are sequentially applied for each cell that contains measurements, instead of constructing one large EKF for the entire network. Local measurements are no longer used to correct the errors of cells far downstream or upstream but are only used to correct the state of cells within a certain radius r of the measurement. Figure 4.6 shows the principle of the L-EKF with a radius of $r = 3$ cells. Note that r can be held constant throughout the simulations or can be varied under the prevailing traffic conditions. To remain focused, r is kept constant in the remainder; future work needs to validate if a dynamic r can improve the results.

In the local EKF scheme, first, a “global” estimate of state vector \mathbf{k}_j^G and of error covariance matrix P_j^G is made using full-sized A_j , P_j , and Q matrices. Global vectors and matrices are indicated by a superscript G and can be calculated quickly because

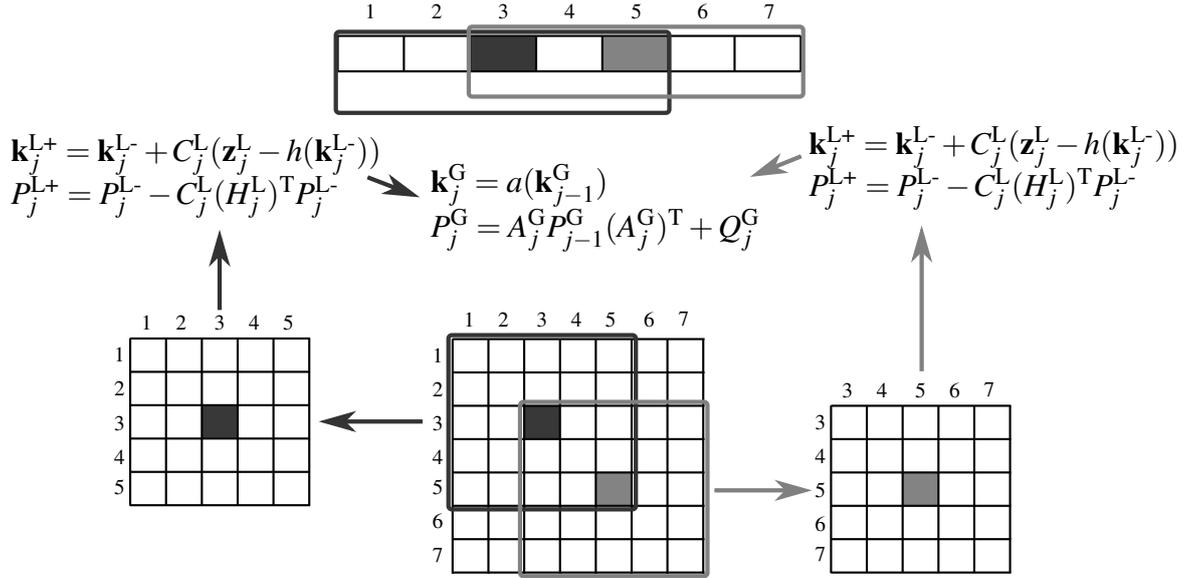


Figure 4.6: Principle of the L-EKF on a seven-cell link with measurements in cells 3 and 5. On top, a seven-cell link is shown. First, the global a priori state vector \mathbf{k}_j^G and a priori error covariance matrix P_j^G are computed using (4.41) and (4.42). The 7×7 matrix represents P_j^G . Then, an L-EKF is constructed for cell 3, which extracts a P_j^{L-} -matrix (the dark gray 5×5 matrix). This L-EKF corrects the states of cells 1–5. The resulting estimates of \mathbf{k}_j^{L+} and P_j^{L+} are copied back into the global matrices. Then, an EKF is constructed for cell 5 (light gray square), and the process is repeated for cells 3–7.

the required operations are relatively light, i.e.,

$$\mathbf{k}_j^G = a(\mathbf{k}_{j-1}^G) \quad (4.41)$$

$$P_j^G = A_j^G P_{j-1}^G (A_j^G)^T + Q_j^G . \quad (4.42)$$

Then, a local EKF is constructed for the first measured cell. A new local a priori density vector \mathbf{k}_j^{L-} is created by copying all elements within the filter radius r from \mathbf{k}_j^G , and a local a priori error covariance matrix P_j^{L-} is obtained by copying the relevant values from P_j^G . Finally, a new derivative matrix H_j^L is created by substituting \mathbf{k}_j^{L-} in (4.40). Then, new estimates of the densities and of the variances in the vicinity of the measurement are determined using

$$C_j^L = P_j^{L-} (H_j^L)^T (H_j^L P_j^{L-} (H_j^L)^T + R^L)^{-1} \quad (4.43)$$

$$\mathbf{k}_j^{L+} = \mathbf{k}_j^{L-} + C_j^L (\mathbf{z}_j^L - h(\mathbf{k}_j^{L-})) \quad (4.44)$$

$$P_j^{L+} = P_j^{L-} - C_j^L H_j^L P_j^{L-} . \quad (4.45)$$

The procedure continues by substituting state estimates \mathbf{k}_j^{L+} and error covariance estimates P_j^{L+} back into the global vector \mathbf{k}_j^G and global matrix P_j^G at the correct coordinates. The preceding process is then repeated for the next measurement.

Note that the order in which the local filters are applied is not important if the model is linear. The Kalman filter (not the EKF which is an approximation) is a Bayesian

optimal estimator that finds the maximum of the posterior of the state of a cell i at time j , given data vector \mathbf{z}_j , i.e.

$$p(k_{ij}|\mathbf{z}_j) = \frac{p(\mathbf{z}_j|k_{ij}) p(k_{ij})}{p(\mathbf{z}_j)}. \quad (4.46)$$

Consider the case where two sequential corrections are made, i.e., one with data point z_1 and one with data point z_2 , and where the posterior of the first correction is the prior of the second correction. Indexes i and j will be omitted to simplify notations. In the case where z_1 is first used to correct, the first correction step of the Kalman Filter can be written as

$$p(k|z_1, z_2) = \frac{p(z_1|k, z_2) p(k|z_2)}{p(z_1|z_2)}. \quad (4.47)$$

The second correction $p(k|z_2)$ is also found using the Kalman Filter, i.e.,

$$p(k|z_2) = \frac{p(z_2|k) p(k)}{p(z_2)}. \quad (4.48)$$

Substituting (4.48) into (4.47) and using the rule $p(a|b)p(b) = p(a, b)$ lead to

$$p(k|z_1, z_2) = \frac{p(z_1|k, z_2) p(z_2|k) p(k)}{p(z_1|z_2) p(z_2)} \quad (4.49)$$

$$= \frac{p(z_1, z_2|k) p(k)}{p(z_1, z_2)}. \quad (4.50)$$

It can now be seen that the same result would be obtained if z_2 was first used and then z_1 . Because the measurement model h is nonlinear, it can be expected that the order *does* influence the solution; however, no a priori knowledge is present on what order to follow. If the state of the model is close to the actual state, i.e. if the model is well calibrated and previous corrections have led to the state being approximately correct, then the linearization is more accurate; in that case, the order in which the corrections are applied will influence the solution less. In this thesis, the filters are applied in the order in which data arrive in the processing computer.

The L-EKF process has two major advantages compared with the G-EKF: First, the measurement error covariance matrix R in (4.43) is of size 1×1 . This means that the inverse operation becomes scalar and thus is very fast. Second, the matrix multiplications (4.45) are performed on much smaller matrices, which again results in gain in computation time. The L-EKF procedure linearly scales in the number of measurements; for each available measurement, (4.43)–(4.45) need to be carried out one more time, but each of these operations is very light. Compared with the G-EKF, the L-EKF is therefore suitable for large-scale and real-time applications.

As opposed to the G-EKF, in the L-EKF, the states of cells far away are not corrected. This leads to a potential loss of accuracy because not all covariance values are used for correction. However, as the error covariance between cells further apart is generally

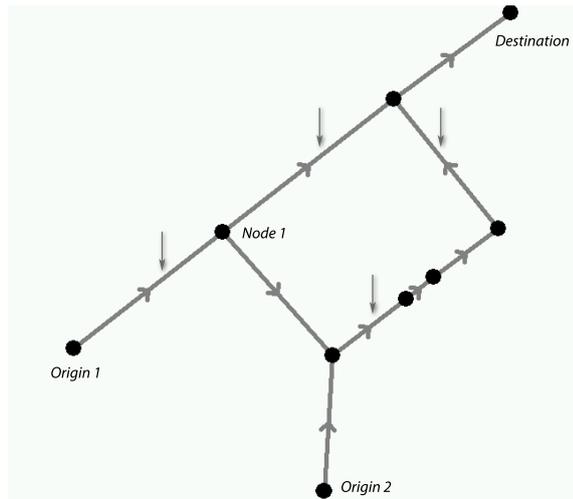


Figure 4.7: Experimental network on which the L-EKF was verified; the vertical arrows indicate the four measurement locations

small, the loss in accuracy is expected to be negligible if the L-EKFs have a large radius and if the measurement network is sufficiently dense.

To show the difference between the L-EKF and G-EKF in both accuracy and computation time, two separate experiments were conducted: one on a small scale with synthetic data and one on a large scale with real-world data.

4.3.3 Experimental Setup to Compare the Localized with the Global Extended Kalman Filter

To illustrate the accuracy of the L-EKF compared with the G-EKF, an experiment on a small-scale network is conducted. The L-EKF and G-EKF have been programmed in the software package JDSMART, which is a Java-based implementation of the LWR model solved by the Godunov scheme. For the matrix operations, the fast Universal Java Matrix Package Java library (UJMP, 2010) has been used. All computations are performed on a Windows XP machine with a 3.0-GHz dual-core processor and 2 GB of memory.

Figure 4.7 shows the network of this experiment; it is discretized into 59 cells. Arrows on the links indicate the driving direction. First, a “ground-truth” simulation is performed, with a certain demand pattern on the two origins and with fundamental diagram parameters for each link, which together caused a complex congestion pattern on the network. Each time step, the densities of all cells are stored as the ground truth. The speeds in four cells throughout the network indicated by the vertical arrows in Figure 4.7 are stored each time step, which are distorted with zero-mean Gaussian noise with a standard deviation of $5 \frac{\text{km}}{\text{h}}$.

The network is then simulated again, with the same fundamental diagrams but with zero-mean Gaussian noise added to the demands at the two origins (standard deviation

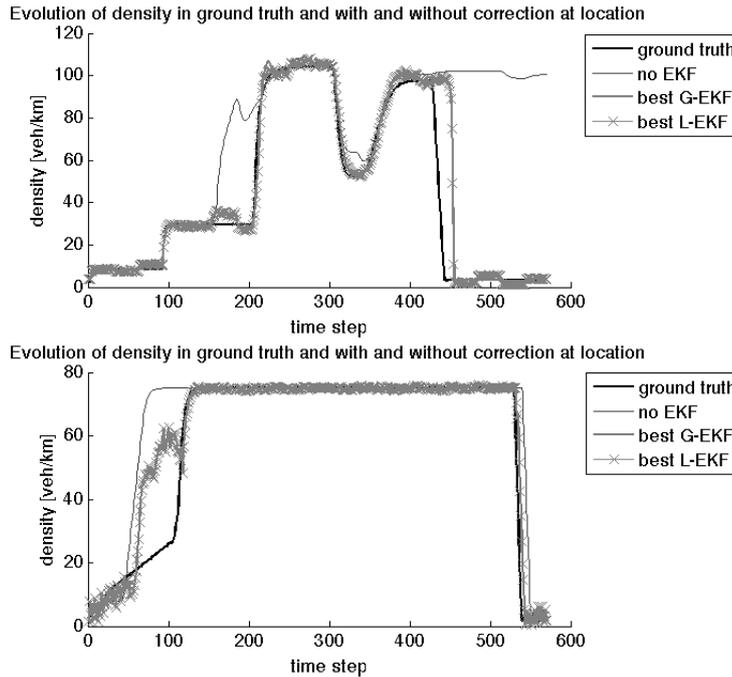


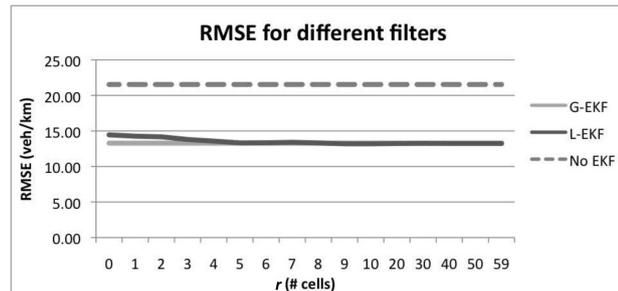
Figure 4.8: Results of the L-EKF: density patterns for two of the four locations: locations 1 (top) and 2 (bottom) indicated by vertical arrows with 1 and 2, respectively, in Figure 4.7. The figure illustrates that, for almost all time steps, the densities from G-EKF and L-EKF are equal.

of $200 \frac{\text{veh}}{\text{h}}$) and the turn fractions of node 1 (standard deviation of 20%). This causes the resulting congestion pattern to be significantly different from the ground-truth experiment. Using the (noisy) speed measurements from the ground-truth simulation, the states can be corrected (the intentionally added noise removed) using the L-EKF or the G-EKF. The process of adding noise to the speeds, demand, and turn fractions is repeated 25 times to be able to generalize the results.

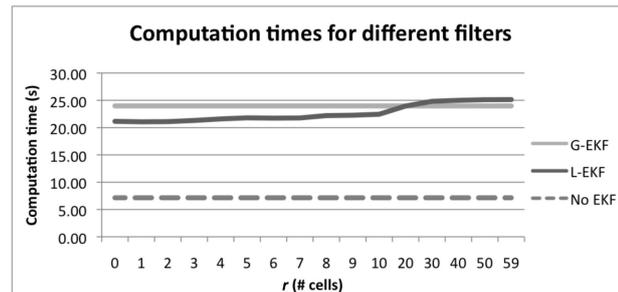
The parameters of the L-EKF and G-EKF (matrices R and Q , and L-EKF radius r) were set as follows: The values on the diagonal of R are set to $25 \frac{\text{km}^2}{\text{h}^2}$ for both the L-EKF and the G-EKF, since the measurement error has a standard deviation of $5 \frac{\text{km}}{\text{h}}$. For each of the 25 simulations, the EKFs are tested with different values on the diagonal of Q , and the best scoring values are chosen; for the L-EKF, radius r is also varied (but equal for all filters in one simulation) between 0 and 59, which is the network size.

4.3.4 Results and Discussions

Figure 4.8 shows an example of the ground truth and distorted and corrected cell densities for one of the 25 simulations at two of the four selected locations. For the corrected densities, the best performing G-EKF and L-EKF are plotted. The figure illustrates that the estimated densities are closer to the ground truth densities when compared with the simulation without EKF but, most importantly, that the L-EKF and G-EKF overlap for



(a) RMSE



(b) Computation times

Figure 4.9: Comparison of the different filters. (Dark solid line) L-EKF is compared for different horizons with the base simulation without (dashed line) EKF and (light solid line) G-EKF.

almost all time steps for all locations.

Figure 4.9(a) shows the average root mean square error (RMSE) between the corrected and ground-truth states for all time steps for all 25 simulations, along with the average computation times. As can be seen, both EKF's result in lower errors than when no correction is applied. The L-EKF with a small radius ($r < 5$) performs worse compared with the G-EKF, because not all data are used to their full potential; however, with sufficiently large radii ($r > 5$), the same level of accuracy is obtained. This result confirms that corrections made by the G-EKF to cells far away are indeed negligible. In addition, the results of the L-EKF with full radii (59 cells) confirm that the order in which the filters are used is, in this case, not important: The sequentially applied filters are as accurate as the G-EKF.

Figure 4.9(b) shows that, even for this small network, the L-EKF is faster than the G-EKF for $r < 20$. For larger r , the calculation times start to increase beyond the computation times for the G-EKF because of the overhead in copying the data back and forth and because of the other matrix operations (4.43)–(4.45).

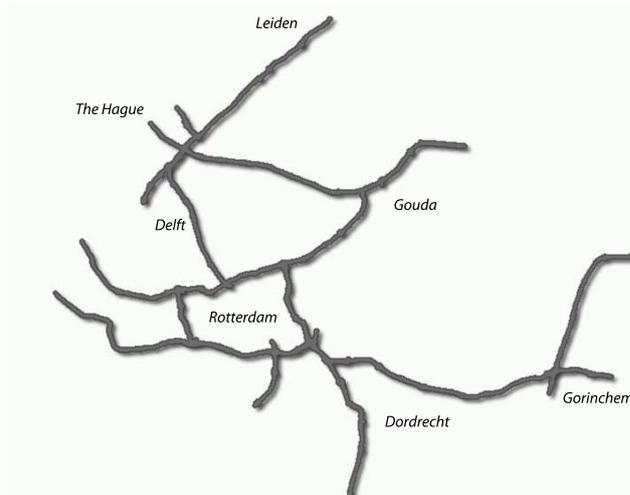


Figure 4.10: Freeway network around the city of Rotterdam with a total length of 272 km.

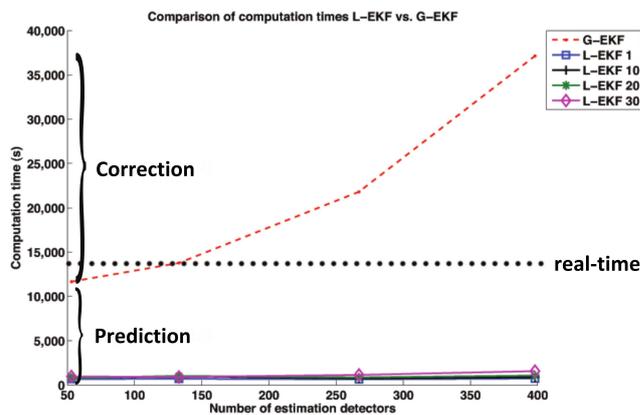


Figure 4.11: Computation times as a function of the number of detectors used for estimation (10-03-2008). The L-EKFs show hardly any increase in computation time, whereas the G-EKF shows a rapidly increasing computation time.

4.3.5 Application of the Localized Extended Kalman Filter to a Real-size Network

The L-EKF is also applied to the freeway network of Rotterdam, The Netherlands, as shown in Figure 4.10. Due to the size of this network, this experiment is a benchmark for realistic traffic state estimation applications. For brevity, this section presents the results of the computation time only. For detailed information about the setup and the results of this experiments, refer to Van Hinsbergen et al. (2012).

Figure 4.11 presents the computation time of the filters for different number of detectors. The conventional G-EKF performs in real-time only if a few loop-detectors are used. For realistic cases, however, the G-EKF requires much more time and thus is not capable of estimation the traffic state in real-time. In contrast, the L-EKF performs significantly better, and runs in real-time also for large radii. This difference is due to the

correction step of the EKF. Since the prediction step of the G-EKF and the L-EKF are the same, the vast improvement in computation time thus is due to the reformulation of the correction step.

4.3.6 Conclusion of the Optimization Based on the Extended Kalman Filter

In this section, the Extended Kalman Filtering technique was reformulated so that the data are used to correct only the traffic state in the physical vicinity of the sensor. By this localization, the computation time of the correction step in the EKF reduces from almost cubic to linear complexity in the number of measurements. For practical applications, the state estimation of real-sized networks is now possible in real-time.

Therefore, both the Adaptive Smoothing Method and the Extended Kalman Filter improved in this chapter are able to estimate the traffic state in real-time. The following section compares them qualitatively.

4.4 Comparison of the Characteristics of the Adaptive Smoothing Method and Localized Extended Kalman Filtering

Both the Adaptive Smoothing Method and the Localized Extended Kalman Filter technique are now capable of estimating the traffic state in real-time. This section qualitatively compares both techniques. The focus is on the three categories of traffic flow theoretic aspects, the application and implementation, and the computation time. Table 4.4 presents a qualitative comparison between the ASM and the EKF approach.

The first category compares whether the two estimation techniques are valid in terms of traffic flow theory related aspects. The ASM is based on shockwave theory; the characteristic shock wave speeds describe with what speed the traffic flow and traffic speed propagated over the freeway. Due to the smoothing, however, the conservation of vehicles is not ensured. Furthermore, the propagation of the traffic composition differs from that of speed or flow; for example, the composition propagates with a wide range of speed, namely approximately with the current speed of traffic, whereas the speed or flow propagate with one out of the two characteristic speeds. The ASM in its current version is thus not suitable for the estimation of multi-class traffic.

The L-EKF obeys shockwave theory, since the prediction step applies a traffic flow model that is based on it. Since the correction step does not describe any vehicle dynamics, it is not affected by shockwave theory. The L-EKF adheres partially to the conservation of vehicles; the underlying traffic flow model is based on the conservation of vehicles, therefore the prediction step adheres to it. The sensor model of the

Category	Criterion	Adaptive Method	Smoothing	Localized Filter	Extended Kalman
Traffic Flow Theory	<i>Shockwave theory</i>	yes		prediction step: yes; correction step: n/a	
	<i>Conservation of vehicles</i>	no		prediction step: yes; correction step: no	
	<i>Multi-class</i>	no		traffic model a and sensor model h can be expanded to multi-class	
	<i>Generalization to other models or theories</i>	fixed to shock wave theory		generic, any prediction and sensor model can be used	
Application and Implementation	<i>Network topology</i>	freeway road stretch		urban and freeway, any network	
	<i>Data estimated</i>	flow, density or speed		any, dependent on prediction model	
	<i>Calibration parameters</i>	characteristic speeds c_{cong} and c_{free} (Chapter 3), kernel sizes σ and τ		prediction step requires calibrated traffic flow model a , i.e. network layout, fundamental diagram, future network inflows and turn-fractions, etc.; correction step requires calibrated sensor model h ; noise covariance matrices Q and R ; influence radius r	
	<i>Robustness to missing data</i>	data required (in practice every few kilometers and few minutes)		works without data by skipping correction step, i.e. use only traffic model a	
	<i>Robustness to estimation near capacity</i>	smooth transition between free-flow and congestion by weighting function w (4.5)		strong non-linearity of fundamental diagram near capacity leads to difficult estimation	
Computation Time	<i>Complexity in road length</i>	slightly more than linear		slightly less than cubical (4.35); however, expensive matrix inversion of Kalman Gain C (4.37) reduced to linear	
	<i>Complexity in number of measurements</i>	constant		linear	
	<i>Real-time ability</i>	yes		yes	

Table 4.4: Qualitative comparison of characteristics between Adaptive Smoothing Method and Localized Extended Kalman Filter

correction step, however, is designed in such a way that the density is adjusted. In essence, vehicles are added to or removed from the network, which is a violation of the conservation of vehicles. Although the filter applied in the experiments is mixed-class, the L-EKF can be applied to multi-class traffic by using a suitable traffic model and a suitable sensor model. In fact, any prediction or sensor model can be used, which renders the L-EKF very generic.

The second category deals with the application and implementation of the two estimation techniques. Since the ASM assumes that the characteristics travel with either one of the two characteristic shock waves, the ASM cannot take traffic lights into account so that it is limited to freeways. Furthermore, only one spatial dimension is modeled, so the ASM is limited to road stretches, and it estimates only one data type, commonly the speed, flow or density. The ASM is calibrated by the two characteristic wave speeds and by the sizes of the kernels. The former can, in fact, be calibrated automatically by applying the Wave Speed Estimator developed in Chapter 3. The ASM is a solely data-driven method; so it works only if the freeway is densely measured both in space and time. Furthermore, the ASM is robust when traffic is flowing near capacity; a smooth transition between the two traffic regimes ensures that the estimated traffic state is close to the true traffic state.

The L-EKF can be applied to any road type and to any network layout. Furthermore, any data types can be estimated. The L-EKF requires many parameters to be calibrated. Since it uses a traffic flow model and a sensor model, both of them have to be calibrated, which entails the network topology, the fundamental diagram, the inflows at the network boundaries and the turnfractions at bifurcation points. Furthermore, in practice both the system and the sensor model are not perfect so that they exhibit errors. The L-EKF models this error in a simplified way as additive white Gaussian noise, which causes an estimation error. Moreover, the noise distributions are usually unknown, which makes the noise covariance parameters difficult to calibrate. The final calibration parameter is the influence radius, which in our experiments shows good results when set to a few kilometers. Since the L-EKF entails a prediction component, it provides results even if data are temporarily missing; however, that requires a well-calibrated model and recurrent traffic conditions. Since the fundamental diagram is strongly non-linear near capacity, a switch between traffic regimes can introduce further estimation errors.

The third category compares the computation times of the two techniques. The computation time complexity of the ASM is nearly linear in the road length and constant in the number of measurements. The prediction step of the L-EKF performs in cubical time; however, the correction step including the matrix inversion of the Kalman Gain, which previously was the most time consuming operation, now runs in linear time. Moreover, both the ASM and the L-EKF now run in real-time for realistic networks.

Summary

Since the ASM has a low number of calibration parameters, which furthermore have a physical meaning, it is easy to apply and calibrate in practice. In fact, the two most important parameters, namely the characteristic shock wave speeds c_{cong} and c_{free} , can be calibrated by applying the Wave Speed Estimator of Chapter 3. Furthermore, the ASM is robust if traffic is flowing near capacity, since it models a smooth transition between the traffic regimes.

The L-EKF is very flexible, as it can be applied to any traffic network, including urban networks, and any traffic flow model and any sensor model can be applied. Since the L-EKF contains a prediction model, it even provides results if sensors temporarily fail or if larger parts of the network are not observed, which make it robust to missing data. A further advantage of the built-in traffic model is that the L-EKF is suitable for an extension to model-predictive control.

Therefore, despite the built-in predictive qualities of the EKF, the ASM is easier to apply and calibrate, and is furthermore more robust when traffic flows near capacity, we choose the ASM as the estimation technique for the system BOS-HbR (Figure 2.11). The ASM in combination with the WSE will be used in the case study of Chapter 7. In Section 7.2, we will also show how to derive the class-specific density based on the total density by means of the traffic flow model Fastlane, which will be defined in the next chapter.

4.5 Conclusion

In this chapter, two advanced freeway traffic state estimation techniques were analyzed and computationally improved. The first technique, the algorithm of the Adaptive Smoothing Method (ASM) was reformulated in order to solve it using the Fast Fourier Transform (FFT). Since the FFT is less complex than the conventional implementation of the ASM, it now runs within a few seconds for realistic sized freeway stretches, while fully preserving the estimation quality. The ASM technique is now applicable both in real-time for online state estimation purposes as well as for offline purposes for large batch analyses of historic traffic data.

As the second freeway state estimation technique, the Extended Kalman Filter (EKF) was sped up by applying the data only in the physical vicinity of the sensors. Since a measurement provides information only in its vicinity of a few kilometers, the measurements are used to estimate the traffic state locally. This localization drastically speeds up the computation of the correction step from an almost cubic time complexity to a linear time complexity in network size while preserving the estimation quality, so that the EKF is now applicable in real-time for realistically-sized networks.

Subsequently, the ASM and the Localized EKF were compared qualitatively. Although the EKF technique is applicable to any network topology and has predictive capabili-

ties, the ASM is much easier to calibrate due to its low number of calibration parameters. The ASM is therefore better suited and is chosen as the state estimation technique in the system BOS-HbR and is applied in the case study of this thesis.

This completes the Estimation component of the control loop of Figure 2.11. In the next chapters, the Control component is developed. The Estimation component and the Control component are then combined in a case study of the Dutch freeway A15 in Chapter 7.

Chapter 5

Multi-class Flow Analysis and Modeling

The current traffic state estimated in the previous chapter is the basis for traffic control in real-time. As shown in the control loop of Figure 2.11, the controller uses the current traffic state to compute the control signal, which is ultimately fed to the DTM measures of the traffic system. In this chapter, the theoretical framework is developed that shows how the vehicle-class specific properties affect the traffic state and the network performance. In the following chapter, multi-class DTM measures are developed to control vehicle classes individually in order to improve the network performance.

First, the multi-class traffic flow model Fastlane is presented in continuous form in Section 5.1. It is then discretized in space and time in order to solve it numerically in Section 5.2. With Fastlane, the traffic state of a network can be simulated and predicted over time. In order to evaluate the performance of a network, the traffic network performance functions of total time spent and total cost are defined in Section 5.3. Then, it is shown that the factors that largely contribute to the total cost are the travel time in free flow, the throughput at bottlenecks, and the spillback of congestion. In Section 5.4, Fastlane is used to derive the effects of these vehicle-class specific properties on the network performance. Finally, conclusions are presented in Section 5.5.

5.1 Modeling Multi-class Traffic with Fastlane

Chapter 2 demonstrated that multi-class traffic flow models capture relevant traffic phenomena by modeling multiple vehicle classes. For example, different free-speed capabilities of vehicles lead to a distribution of travel times in free flow conditions, or different vehicle lengths lead to variable capacities. The multi-class traffic flow model Fastlane captures and reproduces these phenomena. It was first developed by Van Lint et al. (2008b), and then analyzed by Van Wageningen-Kessels (2013) and applied for traffic state estimation by Yuan (2013). In this section, the Fastlane model is presented in its continuous form. In the section thereafter, the model is discretized in space and time so that it can be solved numerically. In this thesis, the model is largely used as described in the original paper. Only a part of the node model is adapted (see Section 5.2.2).

5.1.1 Basics of Macroscopic Traffic Flow Modeling

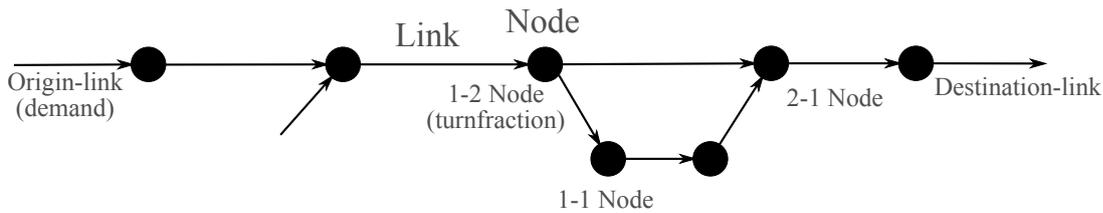


Figure 5.1: Elements of a traffic network modeled in Fastlane

The basic element of traffic flow modeling is the road network. Figure 5.1 shows which network components are implemented in Fastlane. A network consists of links representing a homogeneous stretch of the freeway, and nodes connecting these links. At the boundaries of the network, traffic enters the network at origin links, and it leaves the network at destination links. The amount of traffic that wants to enter the network is also called inflow. Furthermore, at nodes with multiple out-going links, the turnfraction describes what percentage of the traffic is flowing to what out-going link.

Fastlane is a macroscopic traffic flow model, i.e. the traffic state is described in aggregated terms over space x and time t , namely by the flow q and the density k . These two quantities determine the speed

$$v = \frac{q}{k}. \quad (5.1)$$

Furthermore, the conservation equation states that no vehicle can be created or destroyed:

$$\frac{\partial k}{\partial t} + \frac{\partial q}{\partial x} = 0. \quad (5.2)$$

Given these three variables, to solve the system, a third equation is required. The first-order traffic flow model LWR (Lighthill & Whitham, 1955; Richards, 1956) models traffic in an equilibrium, mapping a speed to each density, commonly called the fundamental diagram v^{FD} :

$$v = v^{\text{FD}}(k). \quad (5.3)$$

One fundamental diagram is defined for each link. With the assumption of flow maximization, this system of equations is uniquely solvable (Ansorge, 1990).

Single-class models are able to reproduce important traffic phenomena like the emergence, spreading and dissolution of congestion. However, they do not capture the difference of vehicle properties, such as the vehicle length and the maximum speeds, which affect the maximum throughput at a given location or the travel time, respectively.

5.1.2 Multi-class Macroscopic Traffic Flow Modeling

Like many other multi-class traffic flow models (Wong & Wong, 2002; Benzoni-Gavage & Colombo, 2003; Chanut & Buisson, 2003; Ngoduy & Liu, 2007; Logghe & Immers, 2008; Hoogendoorn, 1999), Fastlane is an extension of the LWR model. For a detailed analysis of macroscopic multi-class traffic flow models, refer to Van Wageningen-Kessels (2013).

The traffic state is modeled as class-specific density k_u , whereby u denotes the vehicle class. Both the speed (5.1)

$$v_u = \frac{q_u}{k_u} \quad (5.4)$$

and the conservation of vehicles (5.2)

$$\frac{\partial k_u}{\partial t} + \frac{\partial q_u}{\partial x} = 0 \quad (5.5)$$

hold per class u .

The fundamental diagram (5.3) is class-specific, too. In free-flowing conditions, faster vehicles can overtake slower ones. In congestion, however, the vehicles are forced to travel at the same speed, so overtaking is not possible. The fundamental diagram in Fastlane is based on the so-called total effective density K_{tot} , which will be defined soon:

$$v_u = v_u^{\text{FD}}(K_{\text{tot}}). \quad (5.6)$$

An example of a fundamental diagram used in Fastlane for two classes (cars and trucks) is presented in Figure 5.2. The free-flow speed v_1^{free} of cars (class 1) is significantly higher than free-flow speed v_2^{free} of trucks (class 2). The higher the density on the

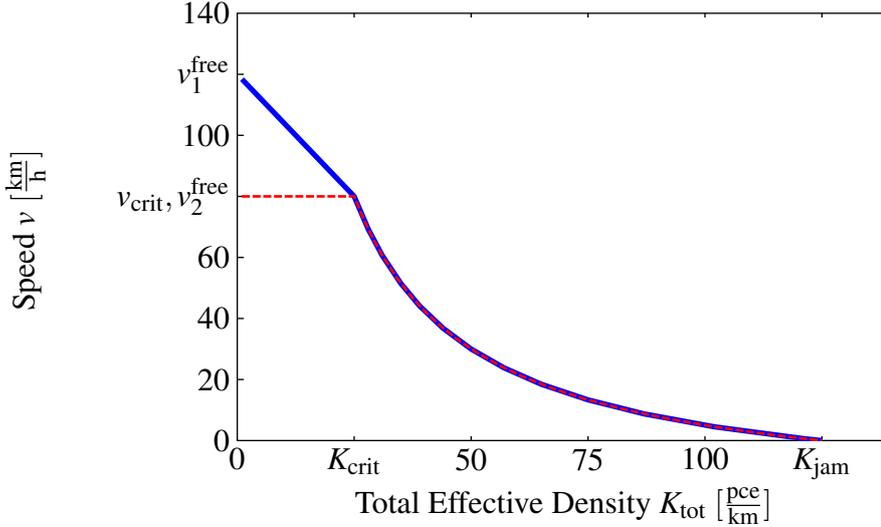


Figure 5.2: Example of a fundamental diagram in Fastlane for trucks (red dashed) and cars (blue solid)

road, the smaller the speed difference between the classes. At critical density K_{crit} , all classes travel with the same critical speed v_{crit} . In this example, the speed of trucks in free flow is insensitive to the density, since $v_2^{\text{free}} = v_{\text{crit}}$.

The critical speed v_{crit} and the critical density K_{crit} define the (effective) capacity

$$C = v_{\text{crit}} \cdot K_{\text{crit}} \quad (5.7)$$

of the link. The capacity here refers to the maximum possible throughput in $\frac{\text{pce}}{\text{h}}$.

In congestion, the speed decreases vastly, until traffic stands still at the jam density K_{jam} . The fundamental diagram in Fastlane for each class u is defined as

$$v_u^{\text{FD}}(K_{\text{tot}}) = \begin{cases} v_u^{\text{free}} - \frac{v_u^{\text{free}} - v_{\text{crit}}}{K_{\text{crit}}} \cdot K_{\text{tot}} & \text{if } K_{\text{tot}} \leq K_{\text{crit}} \\ \frac{1}{K_{\text{tot}}} \cdot \left((v_{\text{crit}} \cdot K_{\text{crit}}) - \frac{v_{\text{crit}} \cdot K_{\text{crit}}}{K_{\text{jam}} - K_{\text{crit}}} \cdot (K_{\text{tot}} - K_{\text{crit}}) \right) & \text{if } K_{\text{tot}} > K_{\text{crit}} \end{cases} \quad (5.8)$$

The total effective density K_{tot} is the aggregate of the class-specific effective densities:

$$K_{\text{tot}} = \sum_u K_u \quad (5.9)$$

The class-specific effective density

$$K_u = \pi_u(v_u) \cdot k_u \quad (5.10)$$

is based on the (vehicular) density k_u and the passenger-car equivalent (pce) value π_u defined in (2.1). Recall that the pce value is the ratio between the spacing of a vehicle of a class u and the spacing of a passenger car. The effective density is therefore expressed in passenger-car equivalents per kilometer ($\frac{\text{pce}}{\text{km}}$), whereas as the (vehicular) density is expressed in vehicles per kilometer ($\frac{\text{veh}}{\text{km}}$).

5.1.3 Dynamic PCE Value

In Fastlane, the pce value π_u (2.1) changes dynamically dependent on the traffic state. As illustrated before in Figure 2.6, the relative spacing of a truck is correlated to the traffic speed: the lower the speed, the higher the relative spacing of a truck with respect to a car. The spacing $r_u(v_u)$ at speed v_u is modeled by a simple car-following model (Pipes, 1967):

$$r_u(v_u) = r_u^{\min} + v_u \cdot h_u^{\min}, \quad (5.11)$$

parameterized with the spatial occupancy at standstill r_u^{\min} and the minimum time headway h_u^{\min} . The pce function of class u is then defined by (2.1) and (5.11) as

$$\pi_u(v_u) = \frac{r_u^{\min} + v_u \cdot h_u^{\min}}{r_{\text{car}}^{\min} + v_{\text{car}} \cdot h_{\text{car}}^{\min}}. \quad (5.12)$$

Figure 5.3 shows a typical pce function for trucks and cars.

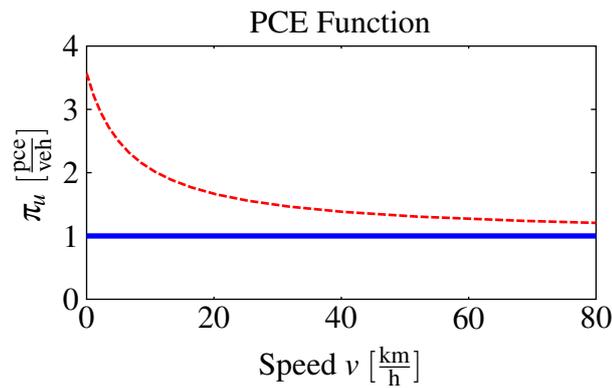


Figure 5.3: Typical pce function in Fastlane for trucks (red dashed) and cars (blue solid)

Besides the conversion between vehicular and effective density (5.10), the pce value is also used to convert the (vehicular) flow q_u to the class-specific effective flow

$$Q_u = \pi_u \cdot q_u \quad (5.13)$$

and to the total effective flow

$$Q_{\text{tot}} = \sum_u Q_u. \quad (5.14)$$

5.2 Discretization of Fastlane

In order to use the Fastlane model in computer simulations, its analytical model of the previous section is discretized in space and time. This section describes the link and the node model of Fastlane.

5.2.1 Link Model

Fastlane is discretized in space into so-called cells i with length Δx^i , and discretized in time into time steps j with length Δt . The class-specific density k_u^{ij} of each cell i evolves over the time steps j by the discretization of the conservation equation (5.2) by

$$k_u^{i,j+1} = k_u^{ij} + \frac{\Delta x^i}{\Delta t} \left(q_u^{i-\frac{1}{2},j} - q_u^{i+\frac{1}{2},j} \right), \quad (5.15)$$

denoting $q_u^{i-\frac{1}{2},j}$ as the flux of class u from cell $i-1$ to cell i (in $\frac{\text{veh}}{\text{h}}$).

As in the Cell Transmission Model (Daganzo, 1994), the fluxes are solved by Godunov's minimum demand and supply scheme (Lebaque, 1996) (the time index is omitted here for readability):

$$q_u^{i-\frac{1}{2}} = \min\{d_u^{i-1}, s_u^i\}, \quad (5.16)$$

with the class-specific demand d_u^i and the class-specific supply s_u^i of cell i (both in $\frac{\text{veh}}{\text{h}}$).

The effective demand (D_u) and supply (S_u) quantities are defined similarly to the total effective flow (5.14) and the class-specific flow (5.13):

$$d_u^i = \frac{1}{\pi_u^i} \cdot D_u^i \quad s_u^i = \frac{1}{\pi_u^i} \cdot S_u^i \quad (5.17)$$

$$D_u^i = \eta_u^i \cdot D_{\text{tot}}^i \quad S_u^i = \eta_u^{i-1} \cdot S_{\text{tot}}^i \quad (5.18)$$

with η_u as the share of class u of the total traffic flow (5.14):

$$\eta_u = \frac{Q_u}{Q_{\text{tot}}}, \quad (5.19)$$

The total demand and total supply from (5.18) are finally defined by the traffic state K_{tot}^i and Q_{tot}^i :

$$D_{\text{tot}}^{i-1} = \begin{cases} Q_{\text{tot}}^{i-1} & \text{if } K_{\text{tot}}^{i-1} \leq K_{\text{crit}}^{i-1} \\ C^{i-1} & \text{else} \end{cases} \quad S_{\text{tot}}^i = \begin{cases} C^i & \text{if } K_{\text{tot}}^i \leq K_{\text{crit}}^i \\ Q_{\text{tot}}^i & \text{else} \end{cases}, \quad (5.20)$$

with C^i as the total effective capacity and K_{crit}^i the total effective critical density of cell i .

5.2.2 Node Model

To model on-ramps, off-ramps or junctions of a network, three node types are used in Fastlane (Figure 5.4), which are explained in the following.

One-to-one nodes model a change in the network structure such as a lane drop, a change in the speed limit, or an incident. One-to-one node therefore connect links that

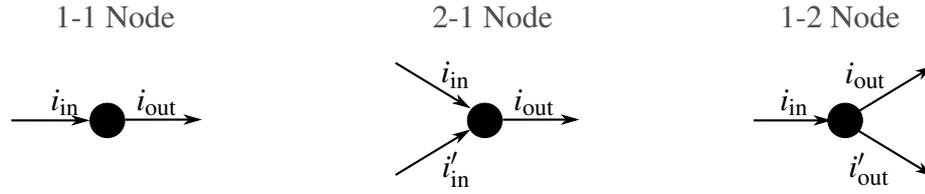


Figure 5.4: Node types in Fastlane

are characterized by different fundamental diagrams. The logic of one-to-one nodes is exactly the same as the one of the links, determining the flux between the links as the minimum of the demand of the upstream link and the supply of the downstream link (5.2.1).

Two-to-one nodes are used to model merges such as on-ramps. To determine the fluxes (5.16) from cell i_{in} to i_{out} and from i'_{in} to i_{out} , the total effective supply $S^{i_{out}}$ of (5.18) is split according to the capacities $C^{i_{in}}$ and $C^{i'_{in}}$ of the incoming links. Therefore, the supply for the incoming cells to the outgoing cells are

$$S^{i_{in} \rightarrow i_{out}} = \frac{C^{i_{in}}}{C^{i_{in}} + C^{i'_{in}}} \cdot S^{i_{out}}, \quad (5.21)$$

$$S^{i'_{in} \rightarrow i_{out}} = \frac{C^{i'_{in}}}{C^{i_{in}} + C^{i'_{in}}} \cdot S^{i_{out}}. \quad (5.22)$$

The remainder of the node logic is similar to one on the link level. In addition, if one in-link's demand is not completely served while the other in-link's demand is, then the remaining unused supply of the latter link is provided to the former one.

One-to-two nodes are used to model diverges such as off-ramps. The model of bifurcation nodes is modified with respect to the original Fastlane model by Van Lint et al. (2008b). To determine the fluxes (5.16) from cell i_{in} to i_{out} and from i_{in} to i'_{out} , the class-specific demand $d_u^{i_{in}}$ of (5.16) is split according to the specified turnfraction $\beta_u^{i_{in}}$. Therefore, the class-specific demands of the incoming cells are

$$d_u^{i_{in} \rightarrow i_{out}} = a \cdot \beta_u^{i_{in}} \cdot d_u^{i_{in}}, \quad (5.23)$$

$$d_u^{i_{in} \rightarrow i'_{out}} = a \cdot (1 - \beta_u^{i_{in}}) \cdot d_u^{i_{in}}. \quad (5.24)$$

Furthermore, in order to ensure a proper modeling of spillback, the blocking factor a describes what ratio of traffic cannot be served:

$$a = \min \left\{ 1, \left(\frac{S_{tot}^{i_{in}}}{D_{tot}^{i_{in}}} \right), \left(\frac{S_{tot}^{i'_{in}}}{D_{tot}^{i'_{in}}} \right) \right\}. \quad (5.25)$$

This is required to ensure the FIFO property at partially blocked diverges.

For details about Fastlane, refer to the original paper by Van Lint et al. (2008b) and to the dissertation by Van Wageningen-Kessels (2013).

5.3 Performance Indicators of Network Traffic

This section defines performance indicators of a traffic system. The total time spent and the total cost are defined both by microscopic and by macroscopic variables in Section 5.3.1. The significant contributing traffic phenomena to the performance indicators are then presented in Section 5.3.2.

5.3.1 Definition of Total Time Spent and Total Cost

The goal of traffic management centers is to optimize the traffic performance of a network. In this thesis, we are interested in the system optimum of a network. One performance indicator to measure the system performance is the total time spent (TTS)

$$T = \sum_{\alpha} \tau_{\alpha} \cdot 1 \text{ veh} , \quad (5.26)$$

which is the sum of the travel time τ_{α} of each vehicle α . A second performance indicator is the total cost

$$T^{\epsilon} = \sum_{\alpha} \zeta_{\alpha} \tau_{\alpha} \cdot 1 \text{ veh} , \quad (5.27)$$

which is a generalization of the total time spent, where the individual travel times are weighted by the value of time ζ_{α} .

In later chapters, the controller is applied in computer simulations conducted with Fastlane. The total time spent and the total costs will therefore be derived from macroscopic variables in discretized form. Based on the class-specific densities k_u^{ij} for each class u defined over a discretized space of size Δx^i for each cell i and discretized time of size Δt for each time step j , the total time spent is derived as

$$T = \sum_u \sum_j \sum_i k_u^{ij} \Delta x^i \Delta t . \quad (5.28)$$

Analogously, the total cost is derived as

$$T^{\epsilon} = \sum_u \zeta_u \sum_j \sum_i k_u^{ij} \Delta x^i \Delta t . \quad (5.29)$$

Equations (5.28) and (5.29) are used in later chapters to evaluate the performance of traffic conditions predicted by a traffic flow model.

5.3.2 Factors Contributing to the Total Cost

In order to control traffic towards the system optimum of a low total cost, the contributing factors that influence the total cost are derived.

The total cost (5.27) and the total time spent (5.26) depend directly on the travel time of each vehicle, which is directly related to the speed v_α and the length L of a route by

$$\tau_\alpha = \frac{L}{v_\alpha}. \quad (5.30)$$

In free flow, the speed is somewhat sensitive to the traffic conditions. The first contributing factor to the total cost is thus the *travel time* spent in free-flow.

However, when congestion emerges at a bottleneck, the *throughput* becomes important, which is the second contributing factors. The higher the throughput is, the fewer vehicles get delayed at the bottleneck, and therefore the lower the total cost is. Hegyi (2004) shows in a simple example how an increase of the throughput by 5% reduced total time spent by 14%.

The third contributing factor to the total cost is *spillback*. When congestion emerges at a bottleneck, a horizontal queue forms and spills back upstream. Figure 2.3 illustrated that spillback can block upstream offramps and thereby hinder traffic that does not want to pass the bottleneck that causes the congestion. Since this traffic gets congested too, spillback is a self-perpetuating effect. It therefore affects the total costs in two ways. Firstly, the vehicles that want to pass the bottleneck get delayed and increase the total cost. Secondly, other traffic that get hindered by the blockage of upstream infrastructure increases the total cost as well. Spillback thus has a large effect on the total cost.

To summarize, the three factors contribute to the total cost. The travel time in free flow is relevant for light traffic conditions, and can be optimized to some extent. As soon as congestion emerges, the throughput of the bottleneck should be maximized, while at the same time, the spillback of the congestion to other infrastructure like off-ramps and the underlying networks should be prevented. Each of the contributing factors will be analyzed in more detail in the following section.

5.4 Analysis of Class-specific Properties on Network Traffic Flow

This section derives the effects of the class-specific properties of free-flow speed, pce value and value of time on the network performance (Section 5.3) by means of the Fastlane model (Section 5.1). The three significant contributing factors to the total time spent and total cost are analyzed in order to derive which vehicle class has to be prioritized under which circumstances to optimize the network performance. For the sake of simplicity, the analyses are presented for the case of two classes. Furthermore, examples with realistic parameter values are shown.

5.4.1 Effects of Class-specific Properties on Throughput at an Active Bottleneck

The easiest contributing factor to analyze is the throughput at a bottleneck. The premise is that the bottleneck is active, i.e. congestion is prevalent upstream of the bottleneck, whereas downstream traffic is in free flow, as it often occurs at lane drops, for example.

Let the bottleneck have a capacity of C^{bn} . Since the bottleneck is active, the total effective flow Q_{tot} through that bottleneck is fixed to that capacity, and it is distributed to the class-specific flows q_u according to (5.14) and (5.13):

$$C^{\text{bn}} = Q_{\text{tot}} = \pi_1^{\text{cap}} \cdot q_1 + \pi_2^{\text{cap}} \cdot q_2, \quad (5.31)$$

with π_u^{cap} denoting the pce value at capacity, i.e. at critical speed: $\pi_u^{\text{cap}} = \pi_u(v_{\text{crit}})$.

The objective in multi-class traffic control is to minimize the total cost by optimally distributing the flow q_u of each class. Minimizing the total cost leads to maximizing the total monetary flow, which is defined as

$$Q_{\text{tot}}^{\text{€}} = \zeta_1 \cdot q_1 + \zeta_2 \cdot q_2. \quad (5.32)$$

(Note, that in order to minimize the total time spent, one would maximize the total (vehicular) flow $q_{\text{tot}} = q_1 + q_2$ of traffic. This is a special case of the total monetary flow, by setting the value of time ζ_u in (5.32) to one.)

This is an optimization problem, whereby the objective is to maximize the total monetary flow $Q_{\text{tot}}^{\text{€}}$ (5.32), subject to the capacity of the bottleneck C^{bn} (5.31) and positive class-specific flows q_u . Class \hat{u} then has to be prioritized exactly then if that class has the highest positive impact on the objective

$$\hat{u} = \arg \max_u \frac{\partial Q_{\text{tot}}^{\text{€}}}{\partial q_u}. \quad (5.33)$$

This optimization is solved by finding the derivative of the objective with respect to the flow of each class:

$$\frac{\partial Q_{\text{tot}}^{\text{€}}}{\partial q_1} = \zeta_1 + \zeta_2 \cdot \frac{dq_2}{dq_1}. \quad (5.34)$$

The derivative $\frac{dq_2}{dq_1}$ is inferred from the bottleneck capacity (5.31):

$$\frac{dq_2}{dq_1} = \frac{d}{dq_1} \left(\frac{1}{\pi_2^{\text{cap}}} \cdot (C^{\text{bn}} - \pi_1^{\text{cap}} \cdot q_1) \right) = -\frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}}. \quad (5.35)$$

Inserting (5.35) in (5.34) leads to a simple expression:

$$\frac{\partial Q_{\text{tot}}^{\text{€}}}{\partial q_1} = \zeta_1 - \zeta_2 \cdot \left(\frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \right); \quad (5.36)$$

and by symmetry

$$\frac{\partial Q_{\text{tot}}^{\epsilon}}{\partial q_2} = \zeta_2 - \zeta_1 \cdot \left(\frac{\pi_2^{\text{cap}}}{\pi_1^{\text{cap}}} \right). \quad (5.37)$$

For two classes holds thus

$$\text{prioritize } \hat{u} = 1 \quad \Leftrightarrow \quad \frac{\partial Q_{\text{tot}}^{\epsilon}}{\partial q_1} > \frac{\partial Q_{\text{tot}}^{\epsilon}}{\partial q_2} \quad (5.38)$$

$$\zeta_1 - \zeta_2 \cdot \left(\frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \right) > \zeta_2 - \zeta_1 \cdot \left(\frac{\pi_2^{\text{cap}}}{\pi_1^{\text{cap}}} \right) \quad (5.39)$$

$$\zeta_1 \left(1 + \frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \right) > \zeta_2 \left(1 + \frac{\pi_2^{\text{cap}}}{\pi_1^{\text{cap}}} \right) \quad (5.40)$$

$$\frac{\zeta_1}{\zeta_2} > \frac{1 + \frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}}}{1 + \frac{\pi_2^{\text{cap}}}{\pi_1^{\text{cap}}}} = \frac{\frac{\pi_2^{\text{cap}} + \pi_1^{\text{cap}}}{\pi_2^{\text{cap}}}}{\frac{\pi_1^{\text{cap}} + \pi_2^{\text{cap}}}{\pi_1^{\text{cap}}}} = \frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \quad (5.41)$$

$$\frac{\zeta_1}{\pi_1^{\text{cap}}} > \frac{\zeta_2}{\pi_2^{\text{cap}}}. \quad (5.42)$$

Class 1 should thus be prioritized if the ratio of value of time per pce value is largest.

In a practical example, when regarding cars and trucks – with $\zeta_{\text{car}} = 15 \frac{\epsilon}{\text{vehh}}$, $\pi_{\text{car}}^{\text{cap}} = 1 \frac{\text{pce}}{\text{veh}}$, $\zeta_{\text{truck}} = 45 \frac{\epsilon}{\text{vehh}}$ and $\pi_{\text{truck}}^{\text{cap}} = 1.5 \frac{\text{pce}}{\text{veh}}$ – the relation of (5.42) shows:

$$\frac{15 \frac{\epsilon}{\text{vehh}}}{1 \frac{\text{pce}}{\text{veh}}} < \frac{45 \frac{\epsilon}{\text{vehh}}}{1.5 \frac{\text{pce}}{\text{veh}}}; \quad (5.43)$$

in this case, the right-hand side is largest and therefore trucks should be prioritized to maximize the monetary flow Q^{ϵ} and to minimize the total cost T^{ϵ} .

In conclusion, in order to maximize the monetary throughput at a bottleneck, prioritize the vehicle class that has the highest ratio of monetary value per spacing.

5.4.2 Effects of the Class-specific Properties on Spillback

The second contributing factor to analyze is the spillback of congestion at a bottleneck. As stated earlier, spillback can have a larger impact on the total cost, because it can block other traffic and is a self-perpetuating effect. We analyze the growth rate of the spillback, denoted as c^{gr} , which is the speed of the shock wave that propagates upstream at the tail of the congestion.

Figure 5.5 shows the setup for the analysis. Vehicles of two classes enter the road stretch with constant flows q_1^{in} and q_2^{in} , respectively. A bottleneck is located downstream and has a sufficiently low capacity

$$C^{\text{bn}} = \pi_1^{\text{cap}} \cdot q_1^{\text{bn}} + \pi_2^{\text{cap}} \cdot q_2^{\text{bn}} \quad (5.44)$$

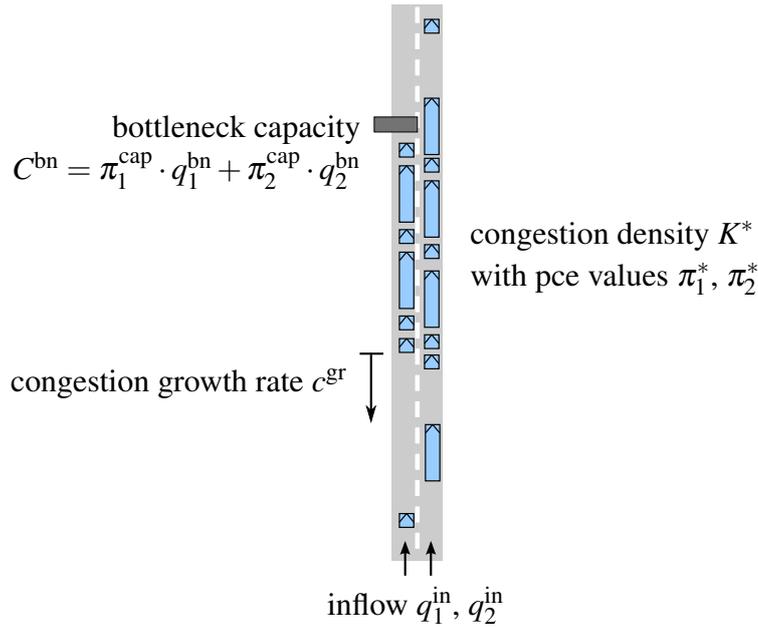


Figure 5.5: Setup for the analysis of spillback

to cause congestion to spill back. Since less traffic flows out of the bottleneck than flows into the congestion, there is a surplus of effective flow:

$$Q^{\text{gr}} = \pi_1^* \cdot q_1^{\text{gr}} + \pi_2^* \cdot q_2^{\text{gr}} \quad (5.45)$$

with the class-specific (vehicular) flow surplus

$$q_u^{\text{gr}} = q_u^{\text{in}} - q_u^{\text{bn}}. \quad (5.46)$$

Since traffic in congestion is denser than traffic at capacity, the pce value π_u^* in (5.45) is larger than the pce value π_u^{cap} in (5.44). Its exact value depends on the total effective density K^* of the congestion. This density is defined by the fundamental diagram and the bottleneck capacity. Assuming a linear flow-density fundamental diagram as in Fastlane (5.8), the effective density of the congestion is then

$$K^* = \frac{C^{\text{bn}} - C}{C} (K_{\text{crit}} - K_{\text{jam}}) + K_{\text{crit}}. \quad (5.47)$$

This density defines the speed in the congestion by the fundamental diagram (5.3):

$$v^* = v^{\text{FD}}(K^*), \quad (5.48)$$

which, in turn, defines the pce value via the pce function (5.12):

$$\pi_u^* = \pi_u(v^*). \quad (5.49)$$

By applying the continuity equation (5.1), the growth rate is derived by

$$c^{\text{gr}} = \frac{Q^{\text{gr}}}{K^*}. \quad (5.50)$$

Substitute the flow surplus (5.45):

$$c^{\text{gr}} = \frac{\pi_1^* \cdot q_1^{\text{gr}} + \pi_2^* \cdot q_2^{\text{gr}}}{K^*}, \quad (5.51)$$

substitute the class-specific flow surplus (5.46):

$$c^{\text{gr}} = \frac{1}{K^*} \left[\pi_1^* \cdot (q_1^{\text{in}} - q_1^{\text{bn}}) + \pi_2^* \cdot (q_2^{\text{in}} - q_2^{\text{bn}}) \right], \quad (5.52)$$

and finally substitute q_2^{bn} with the bottleneck flow (5.44) to derive a closed-form expression of the growth rate:

$$c^{\text{gr}} = \frac{1}{K^*} \left[\pi_1^* \cdot (q_1^{\text{in}} - q_1^{\text{bn}}) + \pi_2^* \cdot \left(q_2^{\text{in}} - \frac{C^{\text{bn}} - \pi_1^{\text{cap}} \cdot q_1^{\text{bn}}}{\pi_2^{\text{cap}}} \right) \right]. \quad (5.53)$$

The class \hat{u} that minimizes the spillback should be prioritized:

$$\hat{u} = \arg \min_u \frac{\partial c^{\text{gr}}}{\partial q_u^{\text{bn}}}. \quad (5.54)$$

For class 1, the derivative term in (5.54) leads to

$$\frac{\partial c^{\text{gr}}}{\partial q_1^{\text{bn}}} = \frac{1}{K^*} \left[-\pi_1^* + \pi_2^* \left(\frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \right) \right] \quad (5.55)$$

$$= \frac{1}{K^*} \left[\frac{\pi_1^{\text{cap}}}{\pi_2^{\text{cap}}} \cdot \pi_2^* - \pi_1^* \right], \quad (5.56)$$

and by symmetry:

$$\frac{\partial c^{\text{gr}}}{\partial q_2^{\text{bn}}} = \frac{1}{K^*} \left[\frac{\pi_2^{\text{cap}}}{\pi_1^{\text{cap}}} \cdot \pi_1^* - \pi_2^* \right]. \quad (5.57)$$

As a realistic example, let us again assume cars and trucks. The pce function of cars is constant, and the truck's pce function decreases with speed:

$$\pi_{\text{car}} \equiv 1, \quad \pi_{\text{truck}}^* > \pi_{\text{truck}}^{\text{cap}}, \quad (5.58)$$

then (5.56) reads

$$\frac{\partial c^{\text{gr}}}{\partial q_{\text{car}}^{\text{bn}}} = \frac{1}{K^*} \left[\underbrace{\frac{\pi_{\text{truck}}^*}{\pi_{\text{truck}}^{\text{cap}}}}_{>1} - 1 \right] > 0 \quad (5.59)$$

That means, the higher the flow at the bottleneck is assigned to cars, the larger the spillback is.

In conclusion, to prevent spillback, or at least to minimize the growth rate of congestion at a bottleneck, the largest vehicle class must be prioritized.

5.4.3 Effects of the Class-specific Properties on the Total Cost in Free Flow

The last contributing factor to the total cost are the travel times in free flow conditions. This factor usually is relevant only if there is little or no congestion in the network. Here, we will analyze the effects of the pce value and the class-specific fundamental diagram on the total cost.

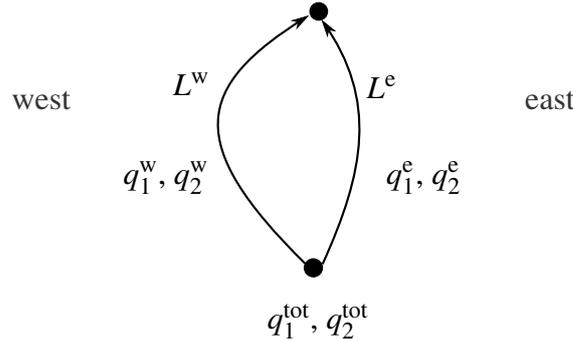


Figure 5.6: Example network for the analysis of the total cost in free flow

Consider a simple network with two possible routes between the origin and the destination, as shown in Figure 5.6. The western route is longer than the eastern route ($L^w > L^e$). There are two classes, with different fundamental diagrams, whereby class 1 is faster than class 2 (given the same total density; $v_1^{\text{FD}}(K_{\text{tot}}) \geq v_2^{\text{FD}}(K_{\text{tot}})$). The inflow q_u^{in} into the network is constant.

The objective is to minimize the total cost of the network. Since the inflow is steady, we will look at the total cost per time, denoted as \dot{T}^ϵ :

$$\dot{T}^\epsilon = \sum_u \zeta_u \sum_r \tau_u^r q_u^r, \quad (5.60)$$

which sums the product of travel times and flow for each class u and for each route $r \in \{w, e\}$. The traffic flow can be influenced by assigning the traffic classes to a route, while preserving the inflow into the network:

$$q_u^{\text{tot}} = q_u^w + q_u^e \quad . \quad (5.61)$$

The goal is thus to minimize (5.60) subject to (5.61) and positive flows q_u .

The derivatives of the objective (5.60) with respect to the control variables will show which class to prioritize. Since the derivatives are relatively complex, we assume that the speed of class 2 is constant. Since the total cost per time consists of four summands and there are two control variables (the flow q_u^w to the western route per class), there are eight derivatives, which are shown in the following. First, the derivatives with respect

to class 1:

$$\frac{\partial}{\partial q_1^w} (\zeta_1 \tau_1^w q_1^w) = \frac{\partial}{\partial q_1^w} \left(\frac{L^w \zeta_1}{v_1^w} q_1^w \right) = L^w \zeta_1 \frac{v_1^w - \frac{\partial v_1^w}{\partial q_1^w} q_1^w}{(v_1^w)^2} \quad (5.62)$$

$$\frac{\partial}{\partial q_1^w} (\zeta_2 \tau_2^w q_2^w) = \frac{\partial}{\partial q_1^w} \left(\frac{L^w \zeta_2}{v_2^w} q_2^w \right) = L^w \zeta_2 \frac{\overbrace{-\frac{\partial v_2^w}{\partial q_1^w} q_2^w}^{=0}}{(v_2^w)^2} = 0 \quad (5.63)$$

$$\frac{\partial}{\partial q_1^w} (\zeta_1 \tau_1^e q_1^e) = \frac{\partial}{\partial q_1^w} \left(\frac{L^e \zeta_1}{v_1^e} (q_1^{\text{tot}} - q_1^w) \right) = L^e \zeta_1 \frac{v_1^e - \frac{\partial v_1^e}{\partial q_1^w} (q_1^{\text{tot}} - q_1^w)}{(v_1^e)^2} \quad (5.64)$$

$$\frac{\partial}{\partial q_1^w} (\zeta_2 \tau_2^e q_2^e) = \frac{\partial}{\partial q_1^w} \left(\frac{L^e \zeta_2}{v_2^e} (q_2^{\text{tot}} - q_2^w) \right) = L^e \zeta_2 \frac{\overbrace{-\frac{\partial v_2^e}{\partial q_1^w} (q_2^{\text{tot}} - q_2^w)}^{=0}}{(v_2^e)^2} = 0 \quad (5.65)$$

and now the derivatives with respects to class 2

$$\frac{\partial}{\partial q_2^w} (\zeta_1 \tau_1^w q_1^w) = \frac{\partial}{\partial q_2^w} \left(\frac{L^w \zeta_1}{v_1^w} q_1^w \right) = L^w \zeta_1 \frac{-\frac{\partial v_1^w}{\partial q_2^w} q_1^w}{(v_1^w)^2} \quad (5.66)$$

$$\frac{\partial}{\partial q_2^w} (\zeta_2 \tau_2^w q_2^w) = \frac{\partial}{\partial q_2^w} \left(\frac{L^w \zeta_2}{v_2^w} q_2^w \right) = L^w \zeta_2 \frac{v_2^w - \frac{\partial v_2^w}{\partial q_2^w} q_2^w}{(v_2^w)^2} = L^w \zeta_2 \frac{1}{v_2^w} \quad (5.67)$$

$$\frac{\partial}{\partial q_2^w} (\zeta_1 \tau_1^e q_1^e) = \frac{\partial}{\partial q_2^w} \left(\frac{L^e \zeta_1}{v_1^e} (q_1^{\text{tot}} - q_1^w) \right) = L^e \zeta_1 \frac{-\frac{\partial v_1^e}{\partial q_2^w} (q_1^{\text{tot}} - q_1^w)}{(v_1^e)^2} \quad (5.68)$$

$$\frac{\partial}{\partial q_2^w} (\zeta_2 \tau_2^e q_2^e) = \frac{\partial}{\partial q_2^w} \left(\frac{L^e \zeta_2}{v_2^e} (q_2^{\text{tot}} - q_2^w) \right) = L^e \zeta_2 \frac{v_2^e - \frac{\partial v_2^e}{\partial q_2^w} (q_2^{\text{tot}} - q_2^w)}{(v_2^e)^2} = -L^e \zeta_2 \frac{1}{v_2^e} . \quad (5.69)$$

To minimize the total costs, class 1 thus has to be sent to route $r = w$, if the derivative of (5.60) is negative:

$$\text{increase flow } q_1^w \Leftrightarrow \frac{\partial \dot{T}^\epsilon}{\partial q_1^w} < 0 . \quad (5.70)$$

The derivative of (5.70) is computed in (5.62) to (5.65); since some of them are zero, only two components remain:

$$L^w \zeta_1 \frac{v_1^w - \frac{\partial v_1^w}{\partial q_1^w} q_1^w}{(v_1^w)^2} + L^e \zeta_1 \frac{v_1^e - \frac{\partial v_1^e}{\partial q_1^w} (q_1^{\text{tot}} - q_1^w)}{(v_1^e)^2} ? < 0 \quad (5.71)$$

This equation can be solved analytically, however it contains the derivative of the speed, which in turn depends on the effective density (5.3), which depends on the pce value (5.9), which itself depends on the speed (5.12). We therefore provide here a qualitative interpretation of this equation and its consequences for the total cost. To support our arguments, the equations and the network were implemented in Matlab's Symbolic Toolbox to find a closed expression of the total cost. Since these expressions are very long, results for realistic parameter values are presented in the following.

Interpretation of the derivatives based on realistic parameter values

The network is implemented for routes of lengths $L^w = 6$ km and $L^e = 5$ km. The capacity of each route is $C = 2000 \frac{\text{pce}}{\text{h}}$, with a critical speed of $v_{\text{crit}} = 80 \frac{\text{km}}{\text{h}}$ and a critical density of $K_{\text{crit}} = 25 \frac{\text{pce}}{\text{km}}$. The free-speed of class 1 is $v_1^{\text{free}} = 120 \frac{\text{km}}{\text{h}}$, the speed of class 2 is constant, $v_2^{\text{free}} = v_{\text{crit}}$. The inflow into the network is fixed to $q_1^{\text{in}} = 1000 \frac{\text{veh}}{\text{h}}$ and $q_2^{\text{in}} = 700 \frac{\text{veh}}{\text{h}}$; the inflow is low enough so that the freeway stays in free-flow conditions. The parameters of the pce function represent realistic cars and trucks with $h_1^{\text{min}} = h_2^{\text{min}} = 1$ s, $r_1^{\text{min}} = 7$ m and $r_2^{\text{min}} = 25$ m. Furthermore, the values of time are set to $\zeta_1 = \zeta_2 = 15 \frac{\text{€}}{\text{vehh}}$.

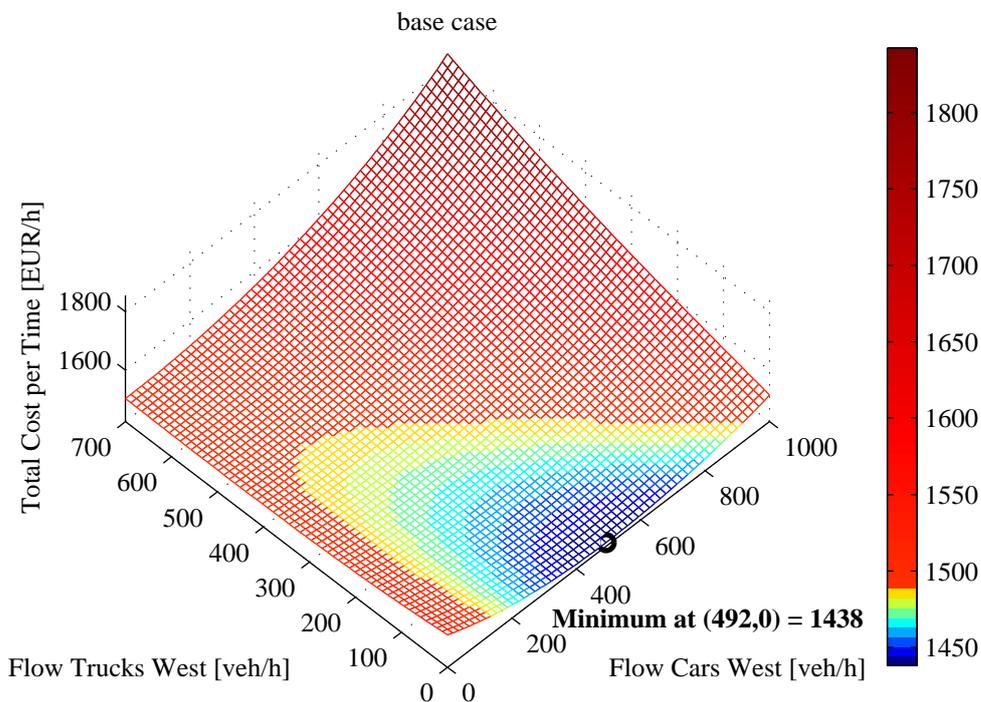


Figure 5.7: Total cost of the base case

Figure 5.7 shows the total cost dependent on the flow assignment to the western route for each class. It shows that simply assigning all traffic to the shorter western route leads to a low total cost, compared to assigning all traffic to the eastern route (at (0,0)).

However, such an assignment is not optimal. Since the speed on the empty eastern road is high, some benefits are gained when assigning some cars to the east. The black circle indicates the optimum assignment: all trucks are assigned to the short western route, whereas the cars are split between both routes (at $(492, 0)$).

In the following, the sensitivity of the total cost with respect to some of the parameters is analyzed.

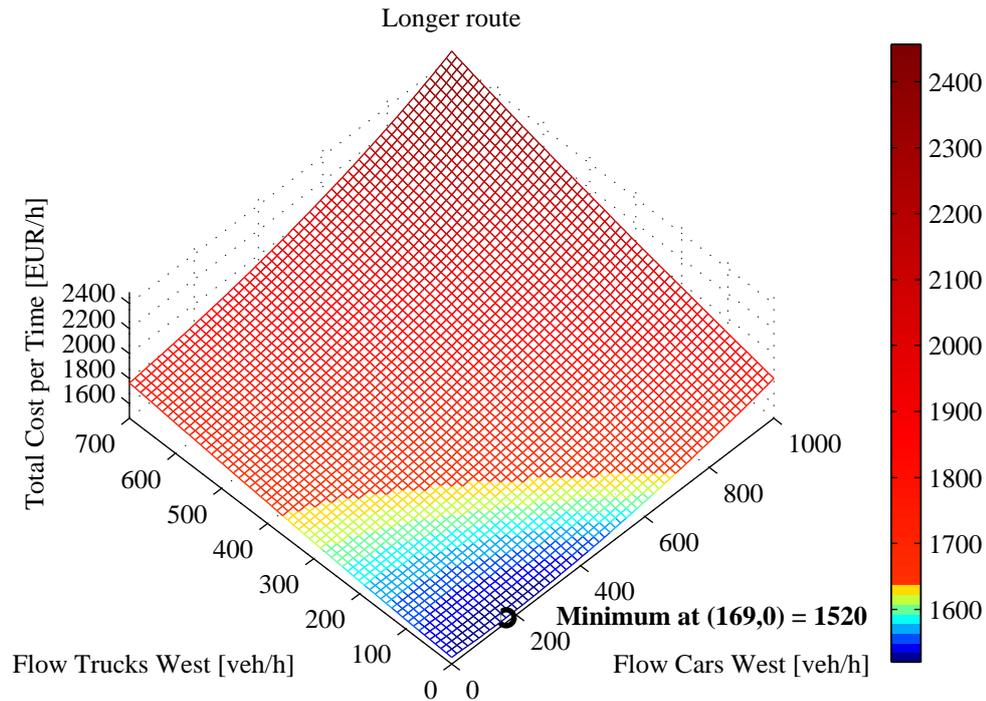


Figure 5.8: Total cost when increasing the length of the longer route

The *lengths* of the routes L^r have a large influence onto the total costs. For example, if the western route is significantly longer than the eastern route ($L^w \gg L^e$), then (5.71) is lower than zero, so that traffic is routed to the west. In the example, the length of the western route is increased to $L^w = 8$ km. Figure 5.8 shows that the optimum has significantly shifted towards the shorter eastern route (minimum at $(169, 0)$).

The *inflow* q_u^{tot} has a strong influence on the optimization of the total cost. The higher the inflow, the more vehicles have to be rerouted in order to prevent congestion. On the other hand, if inflow is low, then high speeds can be maintained even if all vehicles use the shortest route. Figure 5.9 illustrates the latter case, where the optimum is to reroute no vehicles (at $(0, 0)$).

The speed of both routes can vary between the critical speed v_{crit} and the *free speed* v_u^{free} , in realistic setup thus between $80 \frac{\text{km}}{\text{h}}$ and $120 \frac{\text{km}}{\text{h}}$ (see fundamental diagram of Figure 5.2). This is some degree of variation, but it intuitively shows that it has a less significant impact than the length of the routes. In the example, the free speed of

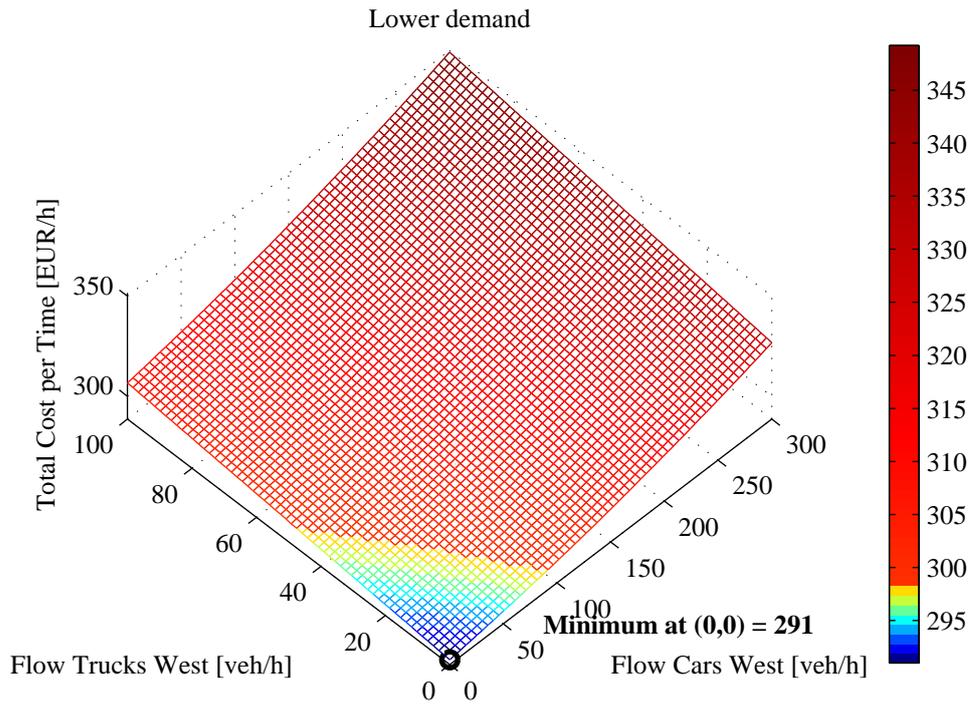


Figure 5.9: Total cost when lowering the inflow

the cars was lowered to $v_1^{\text{free}} = 100 \frac{\text{km}}{\text{h}}$. Figure 5.10 shows that the optimum has now shifted to assigning more traffic to the east (at $(322, 0)$), since there is less to gain when traveling via the longer eastern route.

The derivative of the speed $\frac{\partial v_1^w}{\partial q_1^w}$ in (5.71) is influenced by the *pce function* π (see (5.12), (5.9), (5.3)). In the example, the pce function of the trucks is reduced to one, $\pi_2 \equiv 1$. Therefore, the capacity (in $\frac{\text{veh}}{\text{h}}$) is slightly higher so that more vehicles can travel on the shorter eastern route. As Figure 5.11 indicates, there is, however, only little difference to the base case (minimum at $(441, 0)$).

The *value of time* ζ_u has some effect on the total cost. Since ζ_u is a scalar factor of the total cost (5.27), it can influence the optimal flow distribution severely. However, in realistic cases, this effect is only little. In Figure 5.12, the value of time of trucks was tripled to $\zeta_{\text{truck}} = 45 \frac{\text{€}}{\text{veh h}}$, but the optimum remains practically unchanged (minimum at $(492, 0)$).

Summary

To summarize, the effect of the multi-class parameters on the total cost in free flow conditions is relatively low. The free-speed of each class has a relatively large influence, where the influence of the pce function is almost negligible. Furthermore, the value of time has some effect on to the total cost. More important for the optimal routing is, however, the road layout, since the lengths of each route have a strong influence

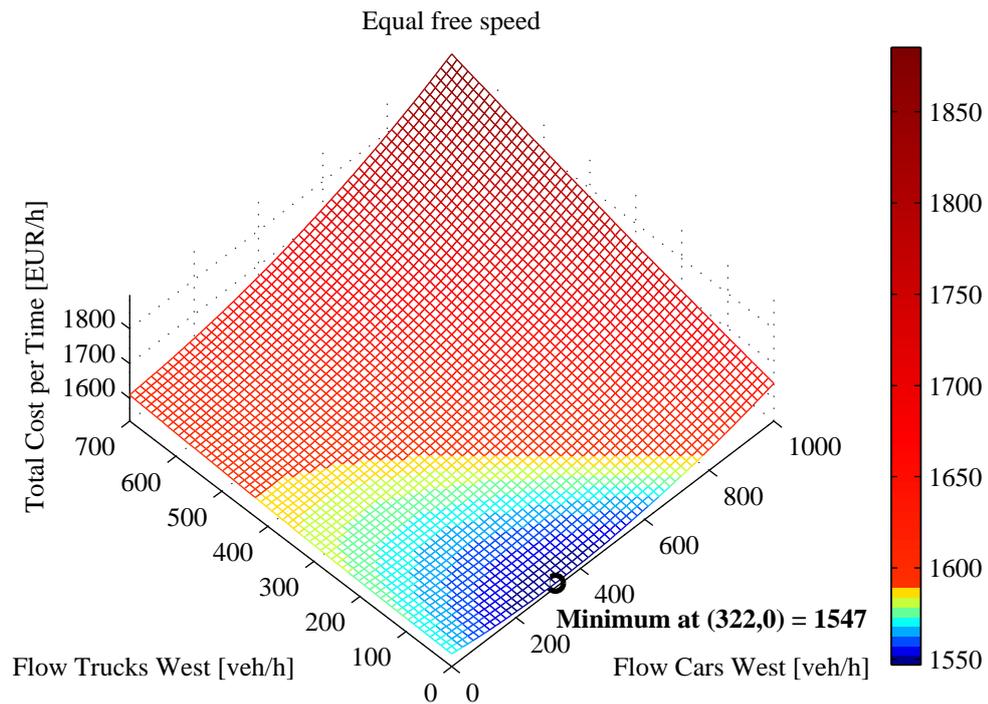


Figure 5.10: Total cost when decreasing the free speed cars

on the travel time. Furthermore, the inflow into the network is significant; if the inflow is very low, then the speed of each route is almost the free speed, so an assignment to the shortest route is optimal.

5.5 Conclusion

This chapter introduced the network performance indicators and the effects of multi-class traffic on them. The focus was on the system optimum of a low total cost, which is the sum of all vehicles' travel costs. The total cost depends mainly on three factors, namely the travel time in free flow, the throughput at a fixed bottleneck and the prevention of spillback.

To analyze the effects of these factors on the total cost, the macroscopic multi-class traffic flow model Fastlane was presented. In Fastlane, the pce value of a class is dynamic: the higher the (total effective) density, the higher the pce value. In addition, in free flow, vehicles can travel with different speeds: each class is characterized by its own free-flow speed.

Figure 5.13 summarizes which vehicle class should be prioritized in order to optimize the effects of each factor. If traffic is in free-flow, there are some ways to steer the traffic towards the system optimum of low total cost. If there are multiple routes from an origin to a destination, and if these routes are approximately of the same length,

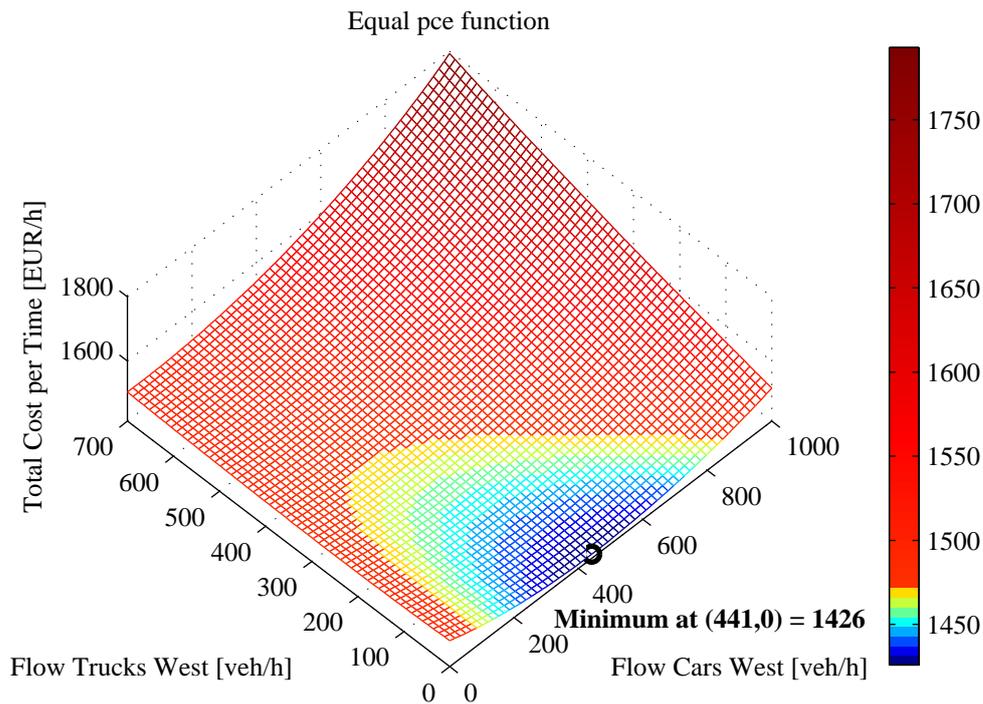


Figure 5.11: Total cost when ignoring trucks dynamics: $\pi_{\text{truck}} \equiv 1$

then traffic can be split between these routes. The free-flow speed capabilities of each class thereby have the strongest influence, which allows fast vehicles to use a longer route but with a higher speed. Vehicle classes whose speed is little sensitive to the traffic conditions usually can take the shortest route. Furthermore, the pce function and therefore the vehicle length have a negligible influence on the optimal flow distribution to minimize the total cost. All these effects, however, are relatively low.

If congestion builds up, it is more important to maximize the throughput at the bottlenecks. We looked here at the monetary throughput, which is a generalization of the flow, because it takes the value of time into account. In essence, in order to minimize the total cost, the vehicle class with the highest ratio of value of time to pce value at capacity has to be prioritized. In practice, this leads to a prioritization of trucks, since they are about three times as valuable as a car but occupy only less than double the amount of space.

If congestion grows long, it threatens to block other infrastructure such as off-ramps, thereby blocking other traffic, which leads to a large degradation of the network performance. The most important factor of the total cost is therefore to minimize the spillback when it threatens to influence other parts of the network. In essence, the longest vehicles have the strongest effect on the length of the congestion. For example, the pce value of a truck increases to value between three and four in practice. It is therefore important to prioritize trucks and keep them out of congestion, if spillback to offramps is likely to occur.

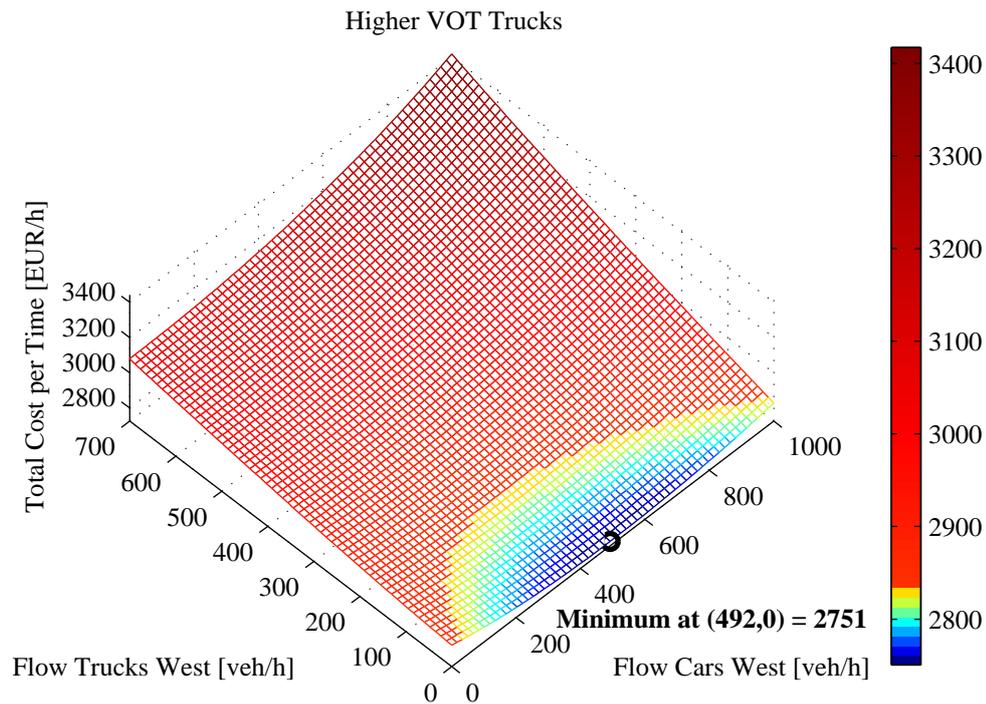


Figure 5.12: Total cost when increasing truck's value of time

These results show that different vehicle classes have to be prioritized in order to optimize the network traffic performance, dependent on the current traffic state. In the following chapter, three multi-class DTM measures will be developed and applied in simple networks which show the benefits of multi-class traffic control. Firstly, a dynamic trucklane is analyzed to determine under which traffic conditions it should be activated to optimize the network performance. Secondly, a ramp meter is generalized to multiple vehicle classes; the experiments will illustrate again the tradeoff between a high network throughput and a low spillback of congestion. Finally, route guidance is generalized to multiple vehicle classes and applied in a model-predictive control framework; among others the experiments will show that the vehicle class that has to be rerouted depends on the underlying traffic conditions. Moreover, the experiments

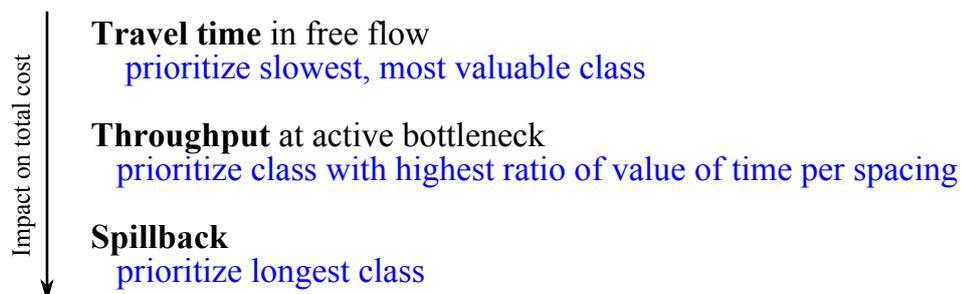


Figure 5.13: Overview of the contributing factors to the total cost and the optimal prioritization of the classes

for all three multi-class DTM measures will show that multi-class control outperforms mixed-class control.

Chapter 6

Multi-Class Traffic Control Concepts

The previous chapter showed how the properties of the vehicle-classes affect the traffic flow and which vehicle class has to be prioritized in order to optimize the network performance of the total cost. In this chapter, we develop several multi-class DTM measures in order to optimize the network performance and its contributing factors. The performance of each measure is analyzed by experiments in simple networks. The total cost is improved with up to 33 % when using multi-class DTM measures with respect to using their mixed-class counterparts.

In Section 6.1, a framework for multi-class traffic control is developed that provides an overview of the three multi-class DTM measures that are discussed in the following. The first multi-class DTM measure developed is the vehicle-class specific lane. Section 6.2 analyzes under which conditions it is beneficial to activate class-specific lanes. The next DTM measure expands conventional ramp-metering to multiple classes. In Section 6.3, multi-class ramp metering is applied in a reactive controller setup. The third DTM measure is multi-class route guidance, where each vehicle-class is guided individually. The setup in Section 6.4 is the most realistic, with changing inflows and the event of an accident; a model-predictive controller is applied to compute the optimal route-guidance signals. Section 6.5 presents the conclusions.

Section 6.2 is based on Schreiter, Pel, Van Lint, & Hoogendoorn (2012): “Modeling Monetary Costs of Multi-class Traffic Flow – Application to the Dynamic Management of Truck Lanes” and Section 6.3 is an adapted version of Schreiter, Van Lint, & Hoogendoorn (2011): “Multi-class Ramp Metering: Concepts and Initial Results”, both presented at the at the IEEE Conference on Intelligent Transportation Systems. Section 6.4 is a revised version of Schreiter, Landman, Van Lint, Hegyi, & Hoogendoorn (accepted): “Vehicle-class Specific Route-guidance of Freeway Traffic by Model-predictive Control” accepted in Transportation Research Record.

6.1 Conceptual Framework of Multi-class Traffic Control

The previous chapter showed that the network performance is influenced by three main contributing factors: the travel time in free flow, the throughput at active bottlenecks, and the spillback of congestion. In order to improve the network performance such as the total cost, this chapter develops a framework for multi-class vehicle control as shown in Figure 6.1.

There are two approaches to control traffic vehicle-class specifically. The vehicle classes can be separated within the carriageway by assigning them different lanes, or they can be separated across the network, i.e. across different links.

This chapter proposes three different vehicle-class specific control measures that apply these two approaches. Firstly, vehicle-class specific lanes dedicate a lane of the freeway to a vehicle class. In other words, one class is separated from the rest of the traffic of the carriageway so that congestion on the remaining lanes can be bypassed by the separated class.

Secondly, multi-class ramp metering combines a class-specific lane with a traffic signal: on an on-ramp of the freeway, the vehicles-classes are spatially separated into class-specific lanes, whereby traffic lights grants access to the freeway separately for each lane.

Thirdly, multi-class route guidance advises a route to each vehicle class individually

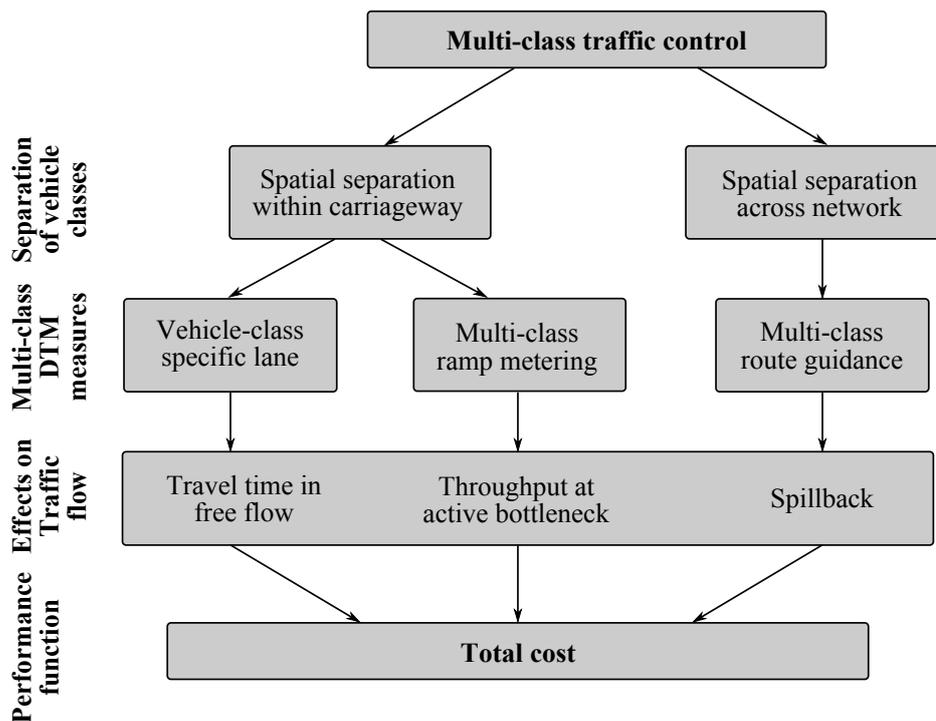


Figure 6.1: Conceptual framework of multi-class traffic control

at a bifurcation point in the network. This is therefore a measure to distribute traffic across the network.

All of these three multi-class DTM measures affect the contributing factors of the network performance. Since the factors also influence each other, a closed-form expression between the parameter values of the measure and the effects on the resulting total cost is therefore hard to calculate, numerical experiments are conducted to evaluate the performance of the DTM measures. However, in some cases, a closed-form expression between model parameters and a contributing factor is possible.

In the experiments of this section, we assume that the spatial separation of vehicle classes does not influence the capacity of the freeway. In general, we assume that the capacity of a freeway stretch is represented in the fundamental diagram so that weaving, merging and similar maneuvers are modeled in the fundamental diagram.

In the following sections, the three multi-class DTM measures are developed and validated. For each of them, a layout is proposed. The traffic flow model Fastlane presented in Sections 5.1 and 5.2 is expanded by these DTM measures so that experiments with simple networks can be conducted and their performance can be evaluated. In Section 6.5, the results of the experiments are summarized and conclusions are drawn.

6.2 Vehicle-class Specific Lanes

The first vehicle-class specific DTM measure discussed in this chapter is the class-specific lane. Such a lane may only be used by a specified vehicle class. The advantage is that a vehicle class that is considered of high value may bypass a congested area. In practice, commonly truck lanes or high occupancy vehicle (HOV) lanes are applied, especially in the United States. Trucks are considered valuable due to their freight, whereas vehicles with a high occupancy serving multiple people are valuable due to their large number of passengers. In some instances, the HOV lane is extended to a high-occupancy and toll (HOT) lane, where single drivers can buy access to it.

The traffic management center is assumed to strive for the system optimal network performance and decides when to activate or deactivate the class-specific lane. In traffic with many trucks, it is beneficial to activate a truck lane so that the valuable trucks can bypass congestion. In contrast, in traffic with nearly no trucks, it is beneficial to deactivate a truck lane and so that its capacity can be used by all traffic. The break-even point is determined in the remainder of this section.

6.2.1 Layout

A layout of vehicle-class specific lanes is proposed in Figure 6.2(a). Only a specified vehicle-class, in this case trucks, is allowed to use the dedicated lane. By installing the

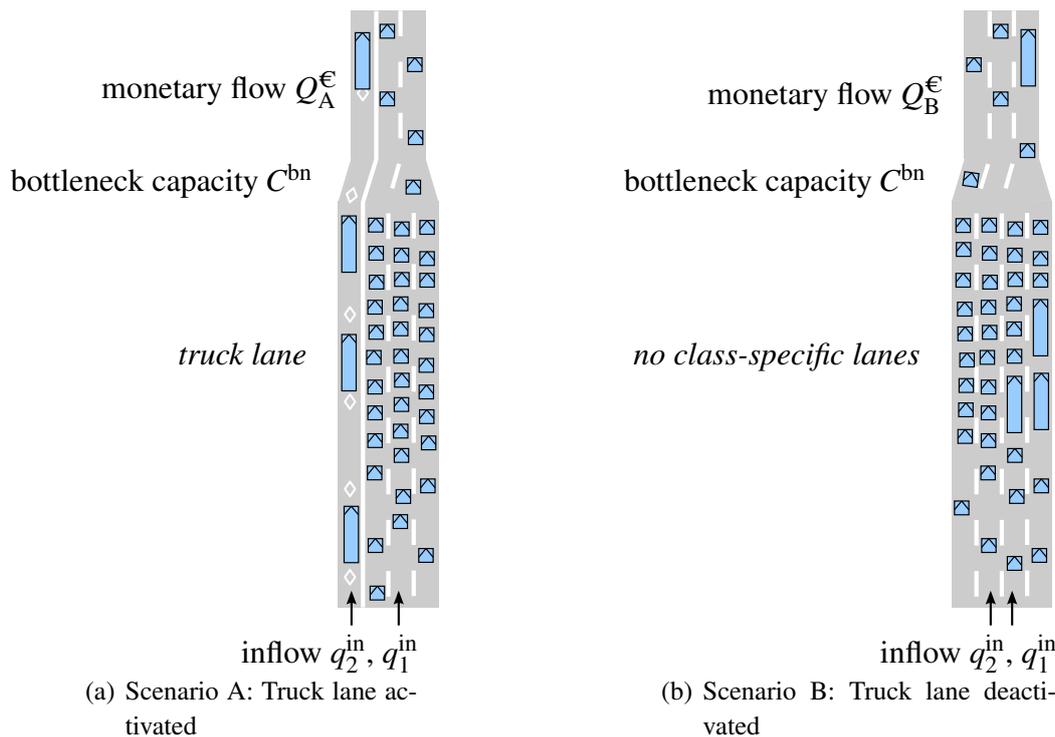


Figure 6.2: Vehicle-class specific lanes

truck-specific lane over lane drops and other bottlenecks, truck traffic can bypass the congestion that potentially emerges there.

6.2.2 Experimental Setup

The network used in the experiments is the road stretch of Figure 6.2 with a bottleneck located downstream in the form of a lane drop. The following two scenarios are considered. In scenario A, the class-specific lane is activated (Figure 6.2(a)) so that only trucks are allowed to use that lane. In scenario B, the class-specific lane is deactivated (Figure 6.2(b)) so that every vehicle class may use the lane.

If traffic is in undersaturated conditions, i.e. if no congestion is present at the bottleneck, then the throughput and therefore the performance of both scenarios is equal.

We therefore consider only oversaturated conditions, i.e. the bottleneck is active and congestion is present. In this case, the goal is to optimize the monetary throughput Q^{ϵ} (5.32) at the bottleneck.

As a typical example, consider a situation with the following realistic parameter values from the Dutch road authority (Rijkswaterstaat, 2011). Let there be a bottleneck with three lanes, whereby one lane is a truck-specific lane; i.e. the capacity is $C^{\text{bn}} = 6000 \frac{\text{pce}}{\text{h}}$ and the dedicated-lane fraction $\lambda = \frac{1}{3}$. Let the pce values at capacity be $\pi_{\text{car}}^{\text{cap}} = 1$ and $\pi_{\text{truck}}^{\text{cap}} = 1.5$, and let the values of time be $\zeta_{\text{car}} = 15 \frac{\epsilon}{\text{vehh}}$ and $\zeta_{\text{truck}} = 45 \frac{\epsilon}{\text{vehh}}$.

In these experiments, the spatial dynamical behavior of traffic such as the growth of congestion is out of the scope. Since only the throughput at a point is of interest, the Fastlane model (Section 5.1) here reduces to the effective flow equations (5.14) and (5.13) and the pce value (5.12) at critical speed.

6.2.3 Results

This section presents first the analytical results, then the numerical results based on the parameter values given above.

6.2.3.1 Analytical Results

The monetary throughput Q_A^ϵ of scenario A is derived (5.32) by the sum of the class-specific flows q_u at the bottleneck, weighted by the value of time ζ_u :

$$Q_A^\epsilon = \zeta_{\text{truck}} \cdot q_{\text{truck}} + \zeta_{\text{car}} \cdot q_{\text{car}} . \quad (6.1)$$

Since class 2 can flow unhindered, its bottleneck flow equals the inflow: $q_{\text{truck}} = q_{\text{truck}}^{\text{in}}$. Since class 1 is oversaturated, its flow is fixed to the capacity of the remaining lanes: $q_{\text{car}} = \frac{(1-\lambda)C^{\text{bn}}}{\pi_{\text{car}}^{\text{cap}}}$. The monetary flow of scenario A is thus

$$Q_A^\epsilon = \zeta_{\text{truck}} \cdot q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \cdot \frac{(1-\lambda)C^{\text{bn}}}{\pi_{\text{car}}^{\text{cap}}} . \quad (6.2)$$

Since empirical research has shown that the (effective) capacity increases if vehicle-class are spatially separated (Cassidy et al., 2009), Q_A^ϵ is a lower bound for the monetary flow of scenario A, so the results in the remainder are also interpreted as lower bounds.

The monetary throughput Q_B^ϵ of scenario B is defined by the class-specific flows at the bottleneck. Since traffic is oversaturated, the flows q_u at the bottleneck are a fraction a of the inflows q_u^{in} :

$$Q_B^\epsilon = \zeta_{\text{truck}} \cdot q_{\text{truck}} + \zeta_{\text{car}} \cdot q_{\text{car}} \quad (6.3)$$

$$= \zeta_{\text{truck}} \cdot a q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \cdot a q_{\text{car}}^{\text{in}} \quad (6.4)$$

$$= a \cdot (\zeta_{\text{truck}} \cdot q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \cdot q_{\text{car}}^{\text{in}}) , \quad (6.5)$$

whereby the factor a in (6.5) is defined by the total flow at the bottleneck q_{tot} and the total inflow $q_{\text{tot}}^{\text{in}}$:

$$a = \frac{q_{\text{tot}}}{q_{\text{tot}}^{\text{in}}} . \quad (6.6)$$

The total bottleneck flow q_{tot} in (6.6) is derived from the bottleneck capacity C^{bn} , whereby the vehicle classes share the total flow according to their inflow composition $\eta_u = \frac{q_u^{\text{in}}}{q_{\text{tot}}^{\text{in}}}$:

$$C^{\text{bn}} = \pi_{\text{truck}}^{\text{cap}} q_{\text{truck}} + \pi_{\text{car}}^{\text{cap}} q_{\text{car}} \quad (6.7)$$

$$= \pi_{\text{truck}}^{\text{cap}} \eta_{\text{truck}} q_{\text{tot}} + \pi_{\text{car}}^{\text{cap}} \eta_{\text{car}} q_{\text{tot}}. \quad (6.8)$$

The total flow in (6.8) is therefore

$$q_{\text{tot}} = \frac{C^{\text{bn}}}{\pi_{\text{truck}}^{\text{cap}} \eta_{\text{truck}} + \pi_{\text{car}}^{\text{cap}} \eta_{\text{car}}}. \quad (6.9)$$

Combining (6.9), (6.6) and (6.5) leads to

$$Q_B^{\text{€}} = \frac{q_{\text{tot}}}{q_{\text{tot}}^{\text{in}}} \cdot \left(\zeta_{\text{truck}} \cdot q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \cdot q_{\text{car}}^{\text{in}} \right) \quad (6.10)$$

$$= \frac{1}{q_{\text{tot}}^{\text{in}}} \cdot \frac{C^{\text{bn}}}{\pi_{\text{car}}^{\text{cap}} \frac{q_{\text{car}}^{\text{in}}}{q_{\text{tot}}^{\text{in}}} + \pi_{\text{truck}}^{\text{cap}} \frac{q_{\text{truck}}^{\text{in}}}{q_{\text{tot}}^{\text{in}}}} \cdot \left(\zeta_{\text{truck}} \cdot q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \cdot q_{\text{car}}^{\text{in}} \right); \quad (6.11)$$

the total inflow $q_{\text{tot}}^{\text{in}}$ in (6.11) gets neutralized so that the monetary flow of scenario B is

$$Q_B^{\text{€}} = \frac{\zeta_{\text{car}} \cdot q_{\text{car}}^{\text{in}} + \zeta_{\text{truck}} \cdot q_{\text{truck}}^{\text{in}}}{\pi_{\text{car}}^{\text{cap}} \cdot q_{\text{car}}^{\text{in}} + \pi_{\text{truck}}^{\text{cap}} \cdot q_{\text{truck}}^{\text{in}}} \cdot C^{\text{bn}}. \quad (6.12)$$

The break-even point between the two scenarios is determined by the comparing the monetary flows $Q_A^{\text{€}}$ (6.2) and $Q_B^{\text{€}}$ (6.12):

$$Q_A^{\text{€}} - Q_B^{\text{€}} = \zeta_{\text{truck}} q_{\text{truck}}^{\text{in}} + \zeta_{\text{car}} \frac{(1-\lambda)C^{\text{bn}}}{\pi_{\text{car}}} - \frac{\zeta_{\text{car}} q_{\text{car}}^{\text{in}} + \zeta_{\text{truck}} q_{\text{truck}}^{\text{in}}}{\pi_{\text{car}}^{\text{cap}} q_{\text{car}}^{\text{in}} + \pi_{\text{truck}}^{\text{cap}} q_{\text{truck}}^{\text{in}}} C^{\text{bn}} \quad (6.13)$$

$$= \zeta_{\text{truck}} q_{\text{truck}}^{\text{in}} + \left[(1-\lambda) \frac{\zeta_{\text{car}}}{\pi_{\text{car}}} - \frac{\zeta_{\text{car}} q_{\text{car}}^{\text{in}} + \zeta_{\text{truck}} q_{\text{truck}}^{\text{in}}}{\pi_{\text{car}}^{\text{cap}} q_{\text{car}}^{\text{in}} + \pi_{\text{truck}}^{\text{cap}} q_{\text{truck}}^{\text{in}}} \right] C^{\text{bn}}. \quad (6.14)$$

A positive value of $Q_A^{\text{€}} - Q_B^{\text{€}}$ (6.14) thus indicates that an activated truck lane is beneficial.

6.2.3.2 Results Based on Realistic Parameter Values

Figure 6.3 presents the monetary flows over the class-specific inflows. If the truck lane is activated (Figure 6.3(a)), then the monetary flow $Q_A^{\text{€}}$ (6.2) increases linearly with the truck inflow $q_{\text{truck}}^{\text{in}}$; furthermore, $Q_A^{\text{€}}$ is insensitive to the car inflow $q_{\text{car}}^{\text{in}}$, since the cars queue up at the end of the congestion and therefore do not influence the flow at the bottleneck.

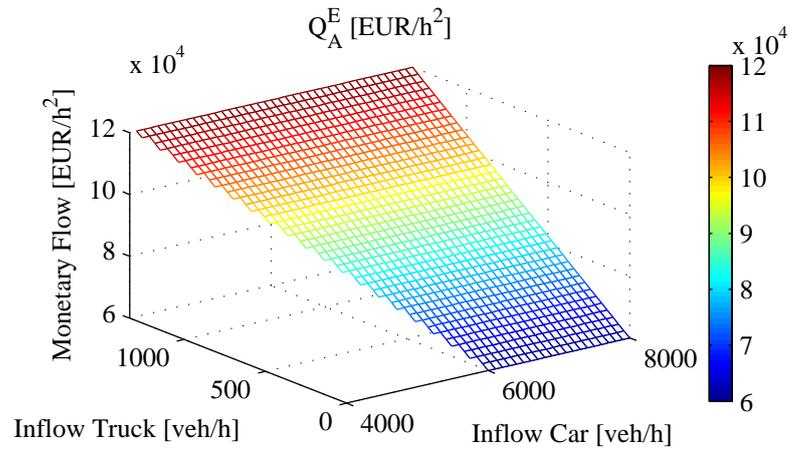
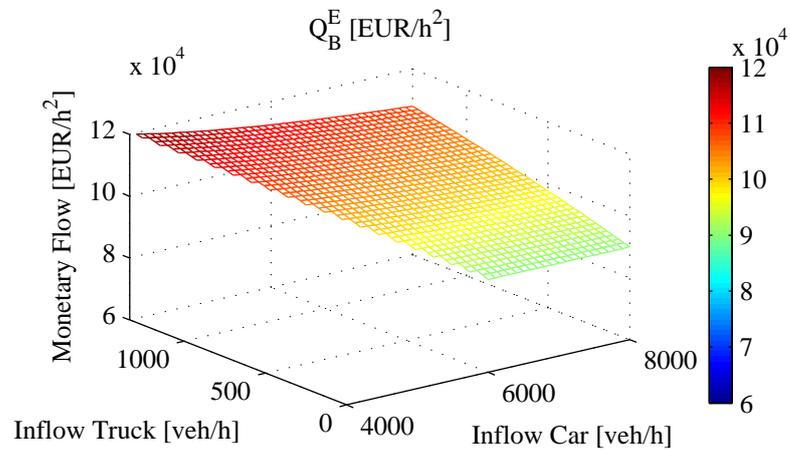
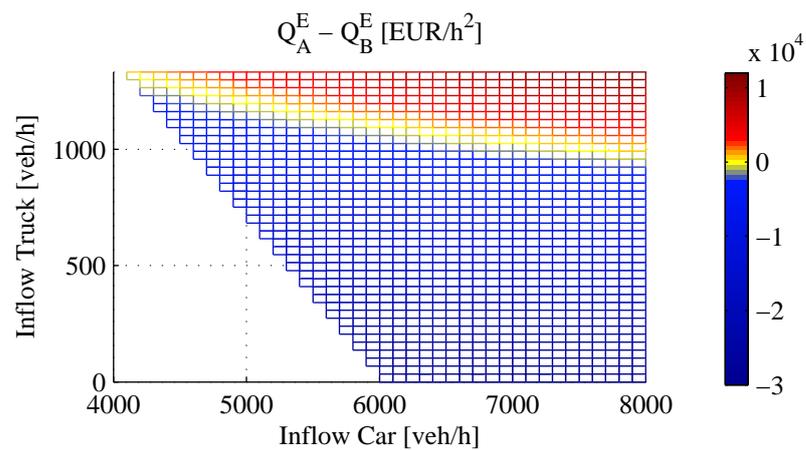
(a) Monetary flow Q_A^E (truck lane activated)(b) Monetary flow Q_B^E (truck lane deactivated)(c) Difference of both cases $Q_A^E - Q_B^E$

Figure 6.3: Results of the trucklane experiments: total monetary flow for both cases and the difference thereof

If the truck lane is deactivated (Figure 6.3(b)), then the monetary flow Q_B^ϵ increases with the truck inflow $q_{\text{truck}}^{\text{in}}$. In contrast, if the car inflow $q_{\text{car}}^{\text{in}}$ increases, then the cars hinder valuable trucks so that the monetary flow decreases. Therefore, the more car traffic there is, the more beneficial a truck lane is.

Figure 6.3(c) shows the difference of both scenarios $Q_A^\epsilon - Q_B^\epsilon$ (6.14). Positive values indicate that an active trucklane is beneficial. The yellow line indicates the break-even point between both scenarios. As can be seen, the larger the inflow, the more beneficial a truck lane is. In the case of heavy demand (the inflow exceeds the capacity by approximately one third), a truck lane is beneficial if it is used to at least three quarters of its capacity. Since the capacity of the bottleneck is likely to be higher if vehicle-classes are spatially separated, the switching point is even slightly lower.

6.2.4 Conclusion

A dedicated truck lane is beneficial if congestion is emerging at a bottleneck and a high percentage of trucks is present. It can even improve the performance if it is running below capacity. Based on a realistic setup, the experiment showed that the total cost is improved if the dynamic truck lane is activated when it is used at approximately three-quarters of its capacity.

6.3 Multi-class Ramp Metering

The second vehicle-class specific DTM measure discussed in this chapter is multi-class ramp metering (MCRM). In current practice, some on-ramps are metered to restrict the inflow onto the freeway. The aims are to keep traffic in free-flow near capacity without breaking down to congestion and to limit the growth of congestion on the freeway. In current practice, ramp meter installations operate on all vehicles equally, not distinguishing between vehicle classes.

This section extends the conventional ramp meter to a multi-class ramp meter that controls the inflow of each vehicle class separately. The traffic management center can specify the desired inflow composition $\hat{\eta}$. By this means, different objectives such as maximizing the throughput of the on-ramp or minimized the spillback can be achieved. In the following, Section 6.3.1 presents the layout, the algorithm and the effects of the multi-class ramp meter, which are based on the theory of dynamic passenger-car equivalence (pce) values. The experiment is set up in Section 6.3.2; its results are discussed in Section 6.3.3. Finally, conclusions are presented in Section 6.3.4.

6.3.1 Layout and Implementation

This section presents the theory of multi-class ramp metering (MCRM): the layout of physical implementation, the control algorithm and the implementation in the traffic

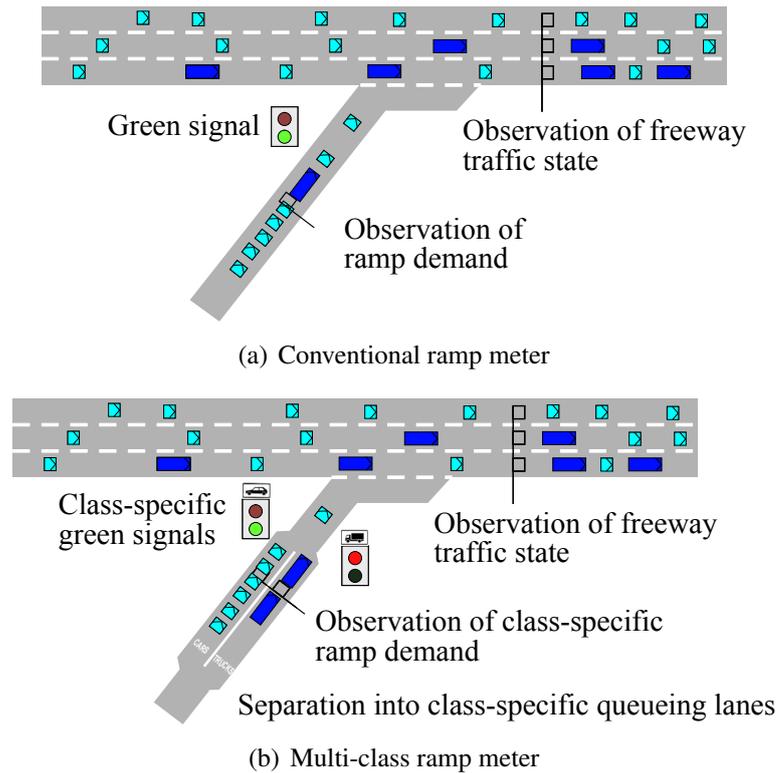


Figure 6.4: Physical layout of ramp meters

flow model Fastlane.

6.3.1.1 Layout

The conventional ramp meter is illustrated in Figure 6.4(a). At an on-ramp, a traffic signal is installed. The inflow into the freeway is controlled by setting the green and red time of the traffic signal. The goal is usually to keep the traffic on the freeway in free flow. Therefore, the traffic state of the freeway is measured and serves as feedback to the ramp meter.

Figure 6.4(b) illustrates the physical layout of the multi-class ramp meter for two vehicle classes. The metered on-ramp is split into multiple queuing lanes, one for each vehicle class. Downstream of the queuing lanes, traffic signals supply green times separately to each vehicle class.

The traffic state is observed at two locations. At the on-ramp, the demand d_u^{rm} of each queue is observed. On the freeway, the traffic state is observed, for example in the form of occupancy or total density.

6.3.1.2 Proposed Algorithm

The control algorithm consists of two parts:

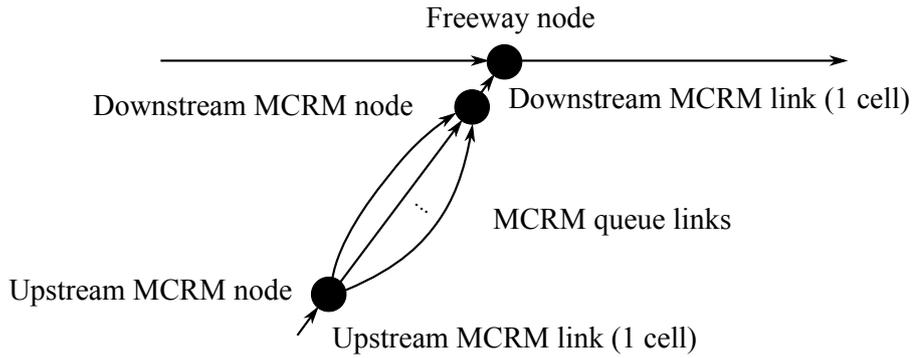


Figure 6.5: Implementation of multi-class ramp meter in Fastlane

1. Calculation of the total ramp supply $S_{\text{tot}}^{\text{rm}}$ based on the freeway traffic state
2. Distribution of the ramp supply to the vehicle classes in the waiting queues according to the desired share $\hat{\eta}$

In the first part – the calculation of the total ramp supply $S_{\text{tot}}^{\text{rm}}$ – a conventional mixed-class ramp meter algorithm is used. The implicit goal is to control the traffic flow towards a set-point density \hat{K}_{tot} , which is usually slightly below the critical density in order to keep traffic in free-flow conditions and to prevent it from breaking down.

In the second part, the ramp supply $S_{\text{tot}}^{\text{rm}}$ is distributed to each vehicle class proportionally on the specified desired share $\hat{\eta}$ specifying the desired inflow composition:

$$S_u^{\text{rm}}(t) = \hat{\eta}_u \cdot S_{\text{tot}}^{\text{rm}}(t), \quad (6.15)$$

which denotes the supply for each vehicle class u in $\frac{\text{pcc}}{\text{h}}$. This class-specific effective supply is then converted into $\frac{\text{veh}}{\text{h}}$ by the current pcc value $\pi_u(t)$ of the freeway:

$$s_u^{\text{rm}}(t) = \frac{S_u^{\text{rm}}(t)}{\pi_u(t)}. \quad (6.16)$$

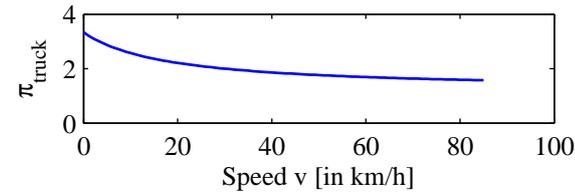
The class-specific fluxes $q_u^{\text{rm}}(t)$ are determined by the class-specific demand $d_u^{\text{rm}}(t)$ and the class-specific supply $s_u^{\text{rm}}(t)$ (6.16):

$$q_u^{\text{rm}}(t) = \min\{s_u^{\text{rm}}(t), d_u^{\text{rm}}(t)\}. \quad (6.17)$$

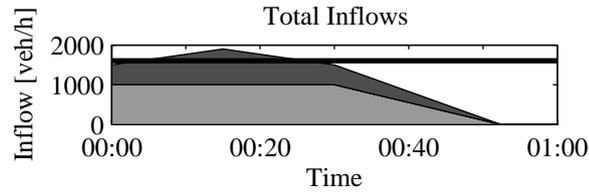
If the supply $s_u^{\text{rm}}(t)$ exceeds the corresponding demand $d_u^{\text{rm}}(t)$, then the remaining supply $s_u^{\text{rm}}(t) - d_u^{\text{rm}}(t)$ is redistributed to the other vehicle classes at the ramp according to the desired share. The resulting class-specific flux is finally converted to green and red times of the traffic signals.

6.3.1.3 Implementation in Fastlane

Fastlane is expanded to simulate multi-class ramp metering. Figure 6.5 shows the network layout for a multi-class ramp meter. The metered ramp is split into multiple links and nodes.



(a) PCE value of trucks



(b) Inflows into the freeway (light) and the on-ramp (dark), the black line denotes the approximate capacity downstream of the on-ramp

Figure 6.6: Experimental setup of multi-class ramp meter

Each waiting queue lane is modeled by a link. Since Fastlane does not model nodes with more than three adjacent links, a special node model for MCRM is developed. At the upstream node, the vehicle classes are distributed into the queuing links according to their class. The downstream node handles the actual MCRM logic as presented above, computing the flux between the queue links and the downstream cell. The ramp meter supply is based on the traffic state of the first cell downstream of the freeway node.

6.3.2 Experimental Setup

The multi-class ramp meter is tested in a network with one on-ramp as in Figure 6.4(b). The length of the freeway section is 4 km; the on-ramp is located halfway. To prevent congestion spilling back to the boundary of the network, the on-ramp is 5 km long. The inflow composition is 50% trucks and 50% cars. Figure 6.6(a) plots the pce function used.

An adjusted version of the Alinea ramp meter algorithm (Papageorgiou et al., 1991) is used as the first part of the control algorithm to determine the total ramp supply by

$$S_{\text{tot}}^{\text{rm}}(j) = S_{\text{tot}}^{\text{rm}}(j-1) + \kappa \cdot (K_{\text{tot}}(j) - \hat{K}_{\text{tot}}) . \quad (6.18)$$

It is based on the observed downstream freeway effective total density $K_{\text{tot}}(t)$ and the previous on-ramp supply $S_{\text{tot}}^{\text{rm}}(t-1)$. The variable κ is a parameter to adjust the stability and reaction speed of the controller. The set-point density \hat{K}_{tot} is set slightly below the critical density of the freeway. The original Alinea controller is based on occupancies instead of total effective densities.

No.	Remark	$\hat{\eta}_{\text{car}}$	$\hat{\eta}_{\text{truck}}$
1	absolute priority trucks MCRM	0	1
2	absolute priority cars MCRM	1	0
3	mixed priority MCRM	0.2	0.8
4	conventional mixed-class RM	n/a	n/a

Table 6.1: Ramp meter policies applied in both experiments

The inflows into the network over the one hour of simulation time are shown in Figure 6.6(b). In the first half hour, the traffic conditions are oversaturated at the on-ramp so that the ramp meter has to be activated and a queue has to build up. After 00:30, the traffic on the freeway is reduced, allowing for a higher ramp supply and the dissolution of the congestion built up.

The desired share $\hat{\eta} = [\hat{\eta}_{\text{car}} \ \hat{\eta}_{\text{truck}}]$ at the ramp meter is varied in four scenarios. Table 6.1 presents three MCRM policies and, for comparison, one policy with conventional mixed-class ramp metering that are simulated.

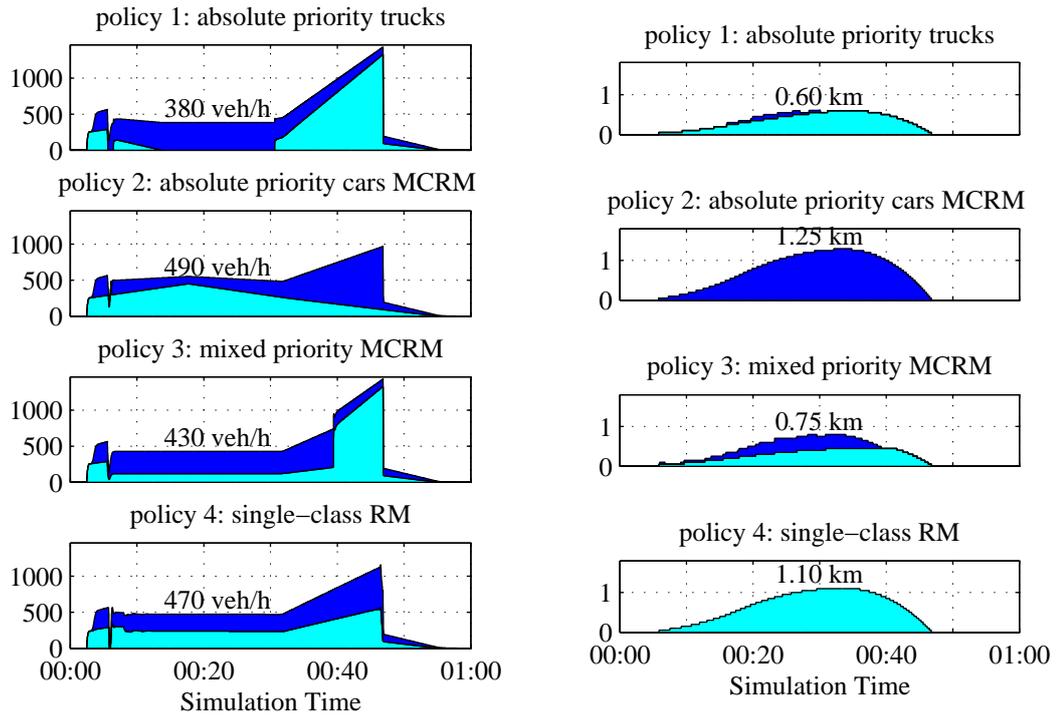
Each policy is evaluated by the class-specific queue length, the class-specific travel time, the total time spent T (5.26), and the monetary costs $T^{\text{€}}$ (5.27), where the loss of one vehicle hour of a car counts for $\zeta_{\text{car}} = 15 \frac{\text{€}}{\text{vehh}}$, and that of a truck for $\zeta_{\text{truck}} = 45 \frac{\text{€}}{\text{vehh}}$.

6.3.3 Results

Figure 6.7 and Figure 6.8 present the results of all four policies in terms of the outflow of the ramp per vehicle class over time, the queue length at the on-ramp, the number of vehicles on the on-ramp, the travel times of both classes and the total time spent and the total cost.

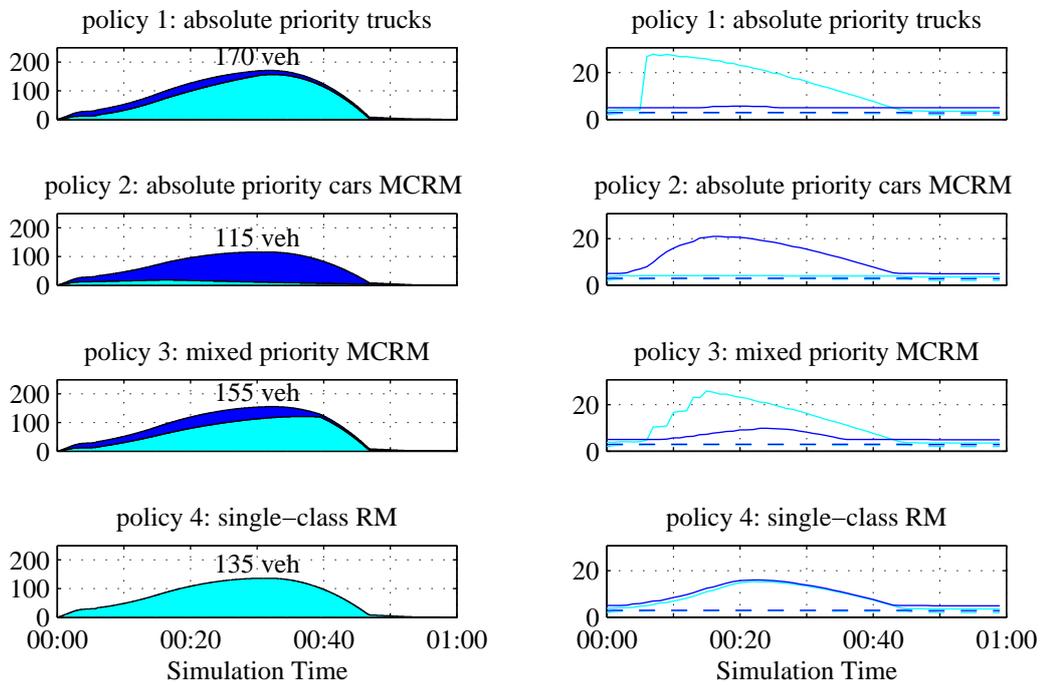
Figure 6.7(a) shows the class-specific outflow over time for each policy. Due to the different prioritization policies, different outflows at the ramps do emerge. If trucks are prioritized (policy 1), then the truck flow (dark blue) reaches a maximum quickly. Since downstream of the on-ramp the freeway is operating near capacity, some traffic has to be held back at the on-ramp. Since trucks are prioritized, the flow of the cars (light blue) dies out. The total ramp flow at that point is $380 \frac{\text{veh}}{\text{h}}$. When the traffic situation gets lighter, first all trucks are served; then, at 00:30, the car queue is served and the car outflow increases sharply. After 00:45, all traffic demand is served.

Analogously, if cars are prioritized (policy 2), first the car flow stays high, then the built-up truck queue is served. Since cars are shorter than trucks, more vehicles can be served ($490 \frac{\text{veh}}{\text{h}}$ at 00:30). Since all the car demand is served, some of the truck traffic can flow onto the freeway as well.



(a) Ramp flow [veh/h] over simulation time

(b) Queue length [km] simulation time



(c) Number of vehicles [veh] the on-ramp over simulation time

(d) Travel times [min] the freeway (dashed) and the on-ramp (solid) over simulation time

Figure 6.7: Simulation results of multi-class ramp metering, car data (light blue) and truck data (dark blue)

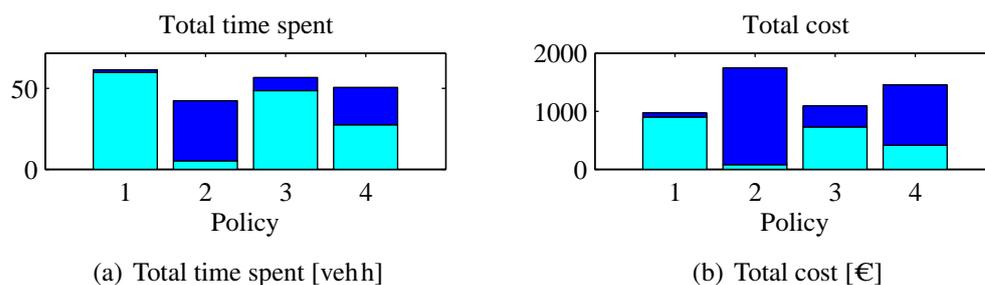


Figure 6.8: Simulation results of multi-class ramp metering, car data (light blue) and truck data (dark blue) (cont.)

In the mixed priority policy (policy 3), the results are between the previous policies. Both trucks and cars can flow into the highway, preventing any vehicle-class from starving on the on-ramp. The total ramp flow at 00:30 is $430 \frac{\text{veh}}{\text{h}}$.

Finally, the conventional, mixed-class ramp meter (policy 4) serves all vehicles with the same priority; since the traffic composition is half cars and half trucks, their ramp flows are equal; the total ramp flow is $470 \frac{\text{veh}}{\text{h}}$.

The length of the waiting queue and the number of vehicles on the on-ramp are presented in Figure 6.7(b) and Figure 6.7(c), respectively. Depending on the policy, not only the composition of the queues differ, but also their joint length. In the truck prioritization policy, the queue consists almost exclusively of cars, its length at 00:30 is 0.6 km, consisting of 170 vehicles. In the car prioritization policy, the queue consists of trucks only, with a length of 1.25 km and 115 vehicles. Notice that there are two important differences between these policies. Firstly, when cars are prioritized, the queue is significantly longer. This is due to the differences in vehicle lengths and therefore to the growing pce value of trucks when the speed decreases. Secondly, however, at the same time fewer vehicles are queued. The reason is that cars are significantly shorter than trucks, and therefore the vehicular capacity of car traffic is naturally higher than the truck's capacity. The other two policies are in between the previous two. (In the figures, the mixed-class plots show only one class, which contains both trucks and cars.)

The prioritization of cars leads to several consequences. Since more vehicles are served compared to the other policies, fewer vehicles are delayed, which results in a lower travel time (Figure 6.7(d)). Furthermore, the total time spent is lower (Figure 6.8(a)). A car-prioritization policy therefore is favored if many cars have to be served and the total time spent has to be minimized. Figure 6.8(b) shows the monetary cost. Although fewer vehicles are served, the truck-prioritization policy outperforms the other policies. For example the car-prioritization policy causes 75 % higher total costs than the truck-prioritization policy.

6.3.4 Conclusion

Multi-class Ramp Metering enables class-specific control of the inflow into the freeway. A parameter of the MCRM is the desired inflow composition, which can be set by the traffic management center. Simulations with three different inflow composition policies of MCRM and one conventional, mixed-class ramp meter showed large differences in the outflow of the ramp, the number of queuing vehicles and the queue lengths. Consequently, the travel times, the total time spent and the total cost differ significantly. This enables the traffic management center to set the control policy in order to reach certain objectives. If the objective is to minimize the total time spent, then cars should be prioritized, since they enable a higher throughput. If the total cost is to be minimized, then trucks should be prioritized; as a positive side effect, the total queue length is reduced, which reduces spillback to the underlying network.

6.4 Multi-class Route Guidance

The third vehicle-class specific controller discussed in this chapter is multi-class route guidance (MCRG). A route-guidance controller advises traffic to take a specified route. In common practice, traffic is controlled regardless of the vehicle-class. In this section, the conventional route-guidance controller is expanded to multi-class route guidance, where each vehicle class is guided separately. To analyze the multi-class route-guidance controller thoroughly, an optimal and predictive control approach is chosen. By applying model-predictive control (MPC), the route guidance controller optimizes the signal over a short period of time.

The remainder of this section is structured as follows. Section 6.4.1 presents the layout of MCRG and the implementation in the traffic flow model Fastlane. Experiments are set up in Section 6.4.2. The results are presented and discussed in Section 6.4.3. Finally, conclusions are drawn in Section 6.4.4.

6.4.1 Layout and Implementation

Figure 6.9 shows the layout of a route guidance controller in a freeway network. At a bifurcation point in the network, a fraction of traffic can be rerouted by advising a route. In mixed-class route guidance (Figure 6.9(a)), all reroutable traffic is advised the same route. Multi-class route guidance (Figure 6.9(b)) advises a route advice to each vehicle class individually. If a variable message sign is used, then each vehicle class is assigned a route in an all-or-nothing way. By using in-car navigation, each vehicle is advised individually, which enables the rerouting of fractions of each vehicle-class specific flow.

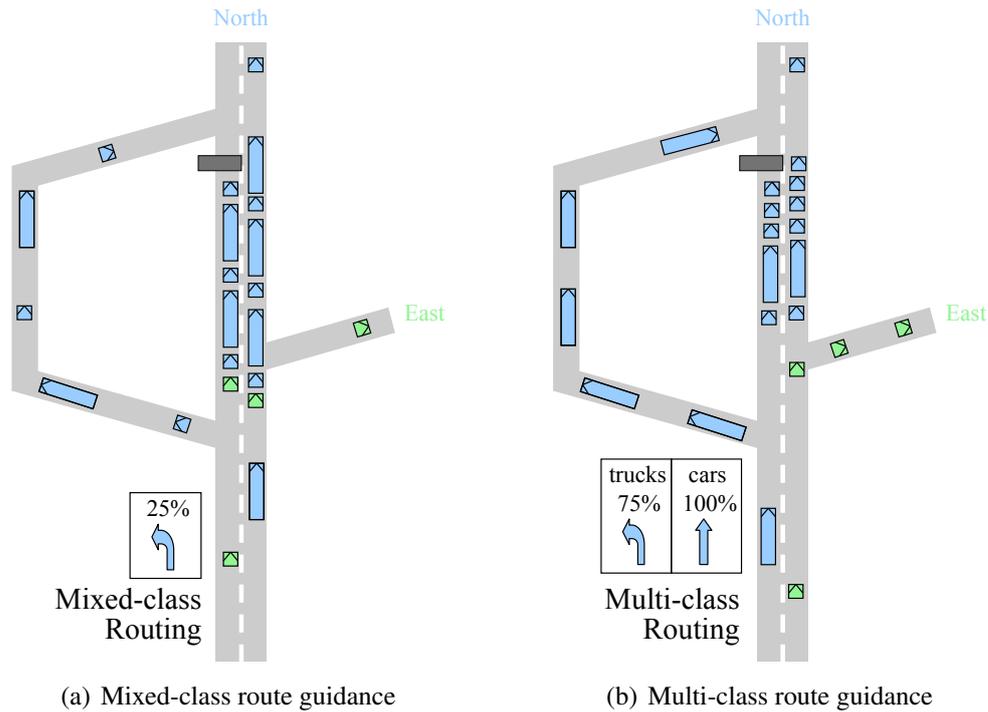


Figure 6.9: Layout of a multi-class route guidance controller

Multi-class route guidance is implemented in Fastlane by setting the class-specific turnfraction at bifurcation points in the network. The turnfraction of each controlled class is simply set to the route guidance advice.

6.4.2 Experimental Setup

To compare the performance of multi-class control against mixed-class control, several experiments based on computer simulations are conducted.

6.4.2.1 Traffic Setup

The network shown in Figure 6.10(a) is used. It has one origin in the south and two destinations, one in the north with two possible routes, and one in the east with one possible route. The main route to the north has a large capacity of $C_{\text{main}} = 5610 \frac{\text{pcc}}{\text{h}}$ so that in normal conditions traffic can flow unhindered without causing congestion. However, an incident with strength ω occurs so that the capacity is reduced to $C_{\text{inc}} = (1 - \omega) \cdot C_{\text{main}}$. Dependent on ω , congestion can occur and spill back which threatens to block the off-ramp to the east. Different incident strengths are simulated within a range of $\omega \in [0, 0.9]$. The incident occurs at the beginning of the simulation and lasts through the whole simulation period.

The alternative route is used to guide the northbound traffic around the incident. It is 5 km longer than the main route and has a capacity of $C_{\text{alt}} = 1870 \frac{\text{pcc}}{\text{h}}$. On both main and

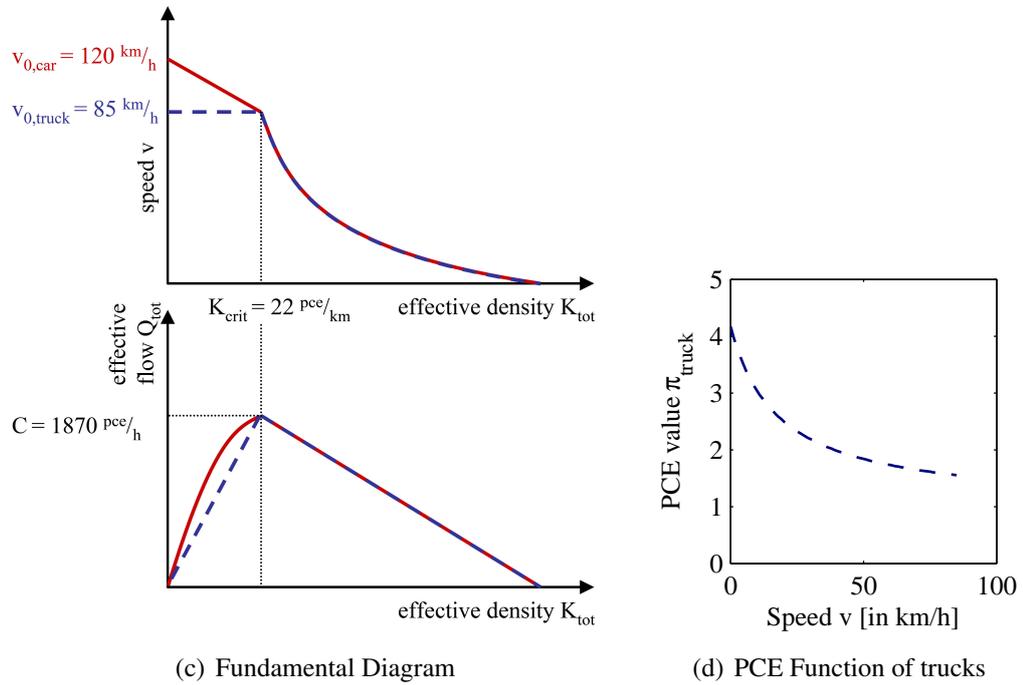
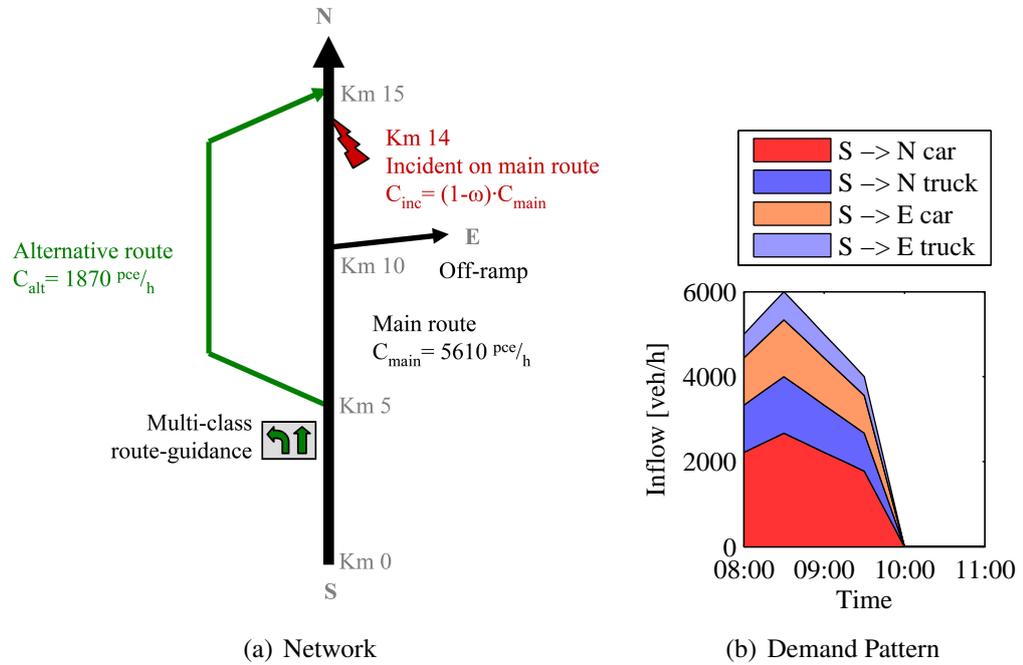


Figure 6.10: Experimental setup of multi-class route guidance

alternative route, vehicles can overtake each other as long as the traffic conditions are in free flow. The fundamental diagram for the alternative route is shown in Figure 6.10(c). For different capacities, the fundamental diagram is simply scaled. The network is modeled in Fastlane (Sections 5.1 and 5.2) with a discretization of $\Delta x = 500\text{m}$ and $\Delta t = 15\text{s}$.

To guide the vehicles, route-guidance is applied at the bifurcation to the alternative route where each vehicle is advised individually. Every vehicle complies to the advice given.

The demand pattern over the $T_{\text{sim}} = 3\text{h}$ of simulation is shown in Figure 6.10(b). The values of time are based on the values from the Dutch road authority (Rijkswaterstaat, 2011), with a car's VOT of $\zeta_{\text{car}} = 15 \frac{\text{€}}{\text{veh h}}$ and a truck's VOT of $\zeta_{\text{truck}} = 45 \frac{\text{€}}{\text{veh h}}$. The pce function π_u (5.12) is parameterized with $h_{\text{truck}}^{\text{min}} = 1.2\text{s}$, $h_{\text{car}}^{\text{min}} = 1.2\text{s}$, $r_{\text{truck}}^{\text{min}} = 25\text{m}$ and $r_{\text{car}}^{\text{min}} = 6\text{m}$ (Figure 6.10(d)).

6.4.2.2 Model-predictive Control

The model-predictive control loop is shown in Figure 6.11. Sensors observe the traffic process to gather data that are used to estimate the current traffic state, which is fed into the control component. The control component computes the optimal control signal w_* , which is applied to the DTM measures to influence the traffic process. A control signal w is a (column) vector that defines the action for each controlled parameter of a DTM measure. In the example of route guidance measures, a controlled parameter is the turnfraction of a vehicle class. The DTM measures are actuated at every control interval Δt_c , usually in the order of several minutes. This (outer) loop of the MPC scheme is thus executed every Δt_c .

The optimal control signal w_* is found based on the predictions by a traffic flow model, which predicts the traffic state over a horizon T_p , usually in the order of one hour. Since the prediction horizon is larger than the control interval, multiple signals are optimized for each controlled parameter. These signals are called the optimal signal trajectory \mathbf{w}_* and take the form of a matrix with $\frac{T_p}{\Delta t_c}$ columns, whereby the l -th column denotes the control signals at time $(l-1)\Delta t_c$.

The optimal control signal trajectory \mathbf{w}_* is the control signal trajectory that minimizes the objective function J :

$$\mathbf{w}_* = \min_{\mathbf{w}} J(\mathbf{w}) . \quad (6.19)$$

The objective function J defines how the performance of the system is evaluated. In this thesis, the total cost (5.27).

This optimization is usually solved in a iterative procedure, where a control signal trajectory \mathbf{w} is selected and fed into the traffic prediction model. The model then predicts the future traffic state based on these DTM actions and the current traffic

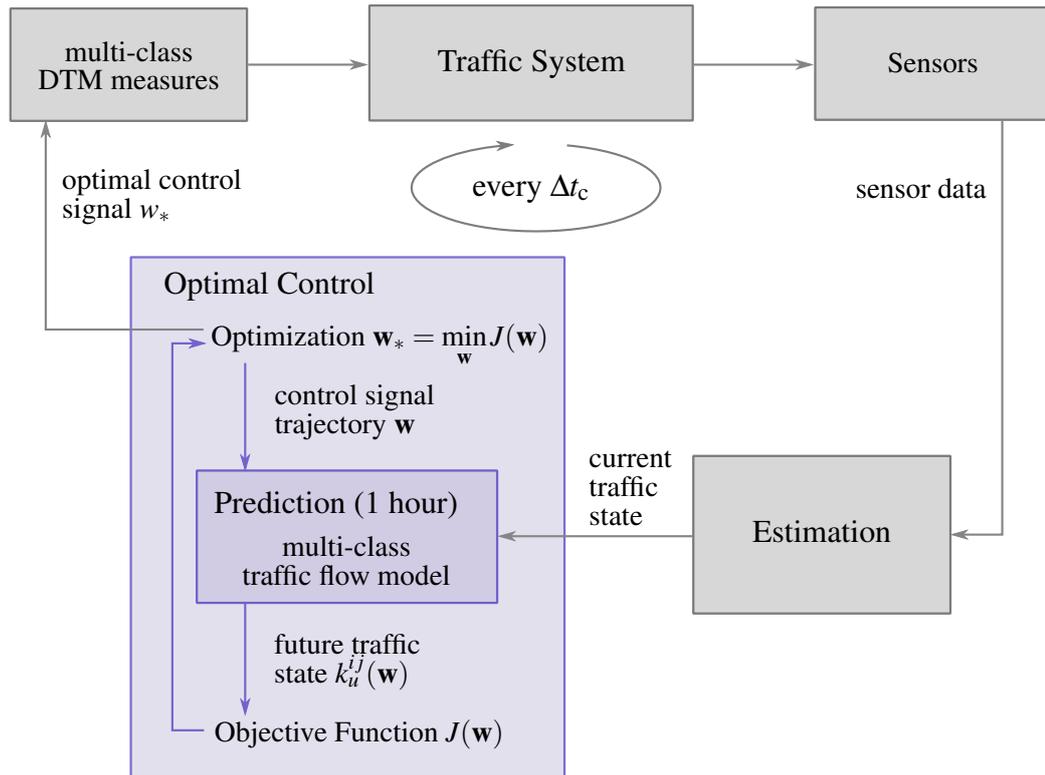


Figure 6.11: The model-predictive control loop

state. The resulting class-specific spatiotemporal density $k_u^{ij}(\mathbf{w})$ is used to evaluate the objective function $J(\mathbf{w})$.

Many optimization algorithms have been developed. In this experiment, we use the active-set method as it implemented by the `fmincon` function of Matlab. This optimization method determines the optimal control signals based by an iterative approach. The (local) optimum is found, when the difference between the solutions of two iterations is close to zero and lower than a specified stopping criterion.

The optimal control signal w_* is the first sample of the optimal control signal trajectory \mathbf{w}_* . In other words, the first column vector of \mathbf{w}_* is applied to the controlled parameters of the DTM measures and the remaining signals are discarded. This completes one cycle of the MPC loop. It is repeated every Δt_c .

For computation time efficiency, a so-called control horizon $T_c < T_p$ can be defined that declares that the control signals only until the period to T_c are optimized, whereas the control signals between T_c and T_p are held constant. The control signal trajectory then reduces to a matrix with only $\frac{T_c}{\Delta t_c}$ columns, which speeds up the optimization procedure. For a formal definition of MPC for these cases refer to Hegyi (2004).

6.4.2.3 Controller Setup

The following four controllers are simulated and compared:

1. Multi-class model-predictive control (MC MPC): Cars and trucks are controlled separately.
2. Mixed-class model-predictive control (SC MPC): All traffic is controlled by one signal.
3. Mixed-class reactive control by measured travel times (SC TT): The fastest route is advised to all vehicles
4. No Control: All traffic uses the main route.

Due to the network layout, only traffic that flows from south to the north can be rerouted, i.e. the eastbound traffic is not guided.

Multiple scenarios with different objective functions J are simulated. The first objective is to minimize the total time the vehicles spent in the network as defined in (5.29) based on the class-specific spatio-temporal densities k_u^{ij} :

$$J^{\text{TTS}} = T = \sum_u \sum_j \sum_i k_u^{ij} \Delta x^i \Delta t^j . \quad (6.20)$$

The second objective is to minimize the monetary costs as defined in (5.28):

$$J^{\text{€}} = T^{\text{€}} = \sum_u \zeta_u \sum_j \sum_i k_u^{ij} \Delta x^i \Delta t^j . \quad (6.21)$$

For the MPC parameters, calibration and testing have shown that a prediction horizon length of $T_p = 60\text{min}$ and a control interval of $\Delta t_c = 10\text{min}$ are sufficient. Each control signal is bound by 0, indicating the main route, and 1, indicating the alternative route.

6.4.3 Results

This section presents the results and a discussion of the experiments, split into three topics. First, the performance of the four controllers is compared. Second, the influence of the value of time onto the multi-class MPC is presented. Third, the influence of the incident strength ω onto the multi-class MPC is analyzed.

6.4.3.1 Comparison of the Controllers

The four controllers are compared for both objective functions and for multiple different incident strengths. Example simulation results of all four controllers for the case of an incident strength of $\omega = 0.64$ and the optimization of the total time spent J^{TTS} (6.20) are shown in Figure 6.12. In these simulations, the incident strength is large enough to let congestion emerge at its location at Km 14.

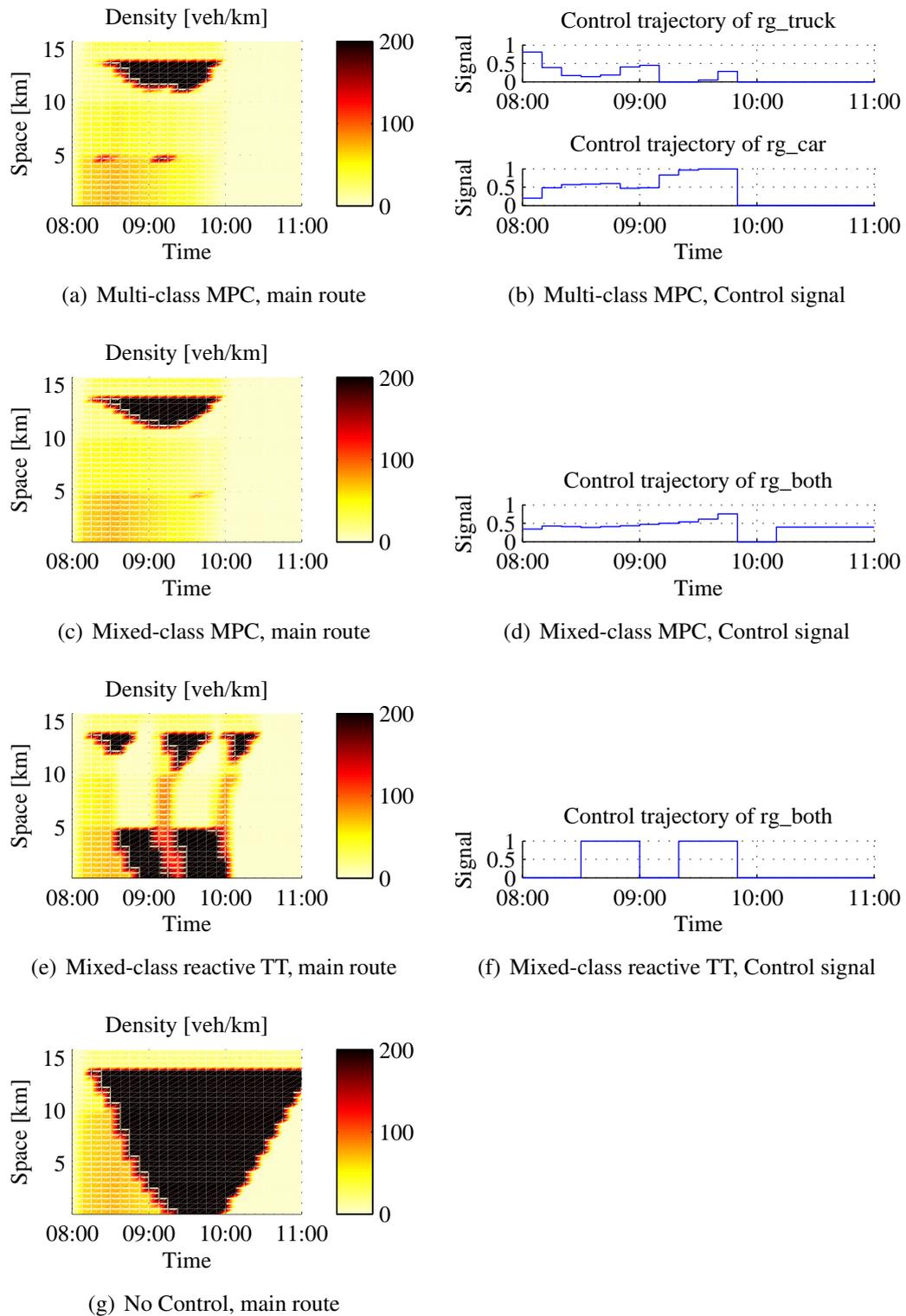


Figure 6.12: Results of multi-class route guidance of the four controllers (incident strength $\omega = 0.64$, objective function total time spent J^{TTS})

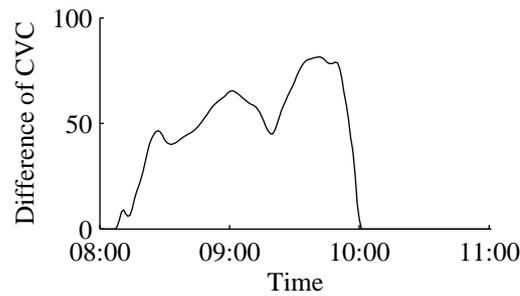


Figure 6.13: Difference of total vehicle count at northern exit between MC MPC and SC MCP

The *multi-class MPC* reroutes some of the traffic via the alternative route, as the control signal in Figure 6.12(b) shows (“rg_truck” is the routing signal for trucks, “rg_car” the one for cars). Between 08:30 and 09:30 the traffic demand exceeds the joint capacity of the main road and the alternative road. Congestion therefore emerges at the bottleneck, as the spatio-temporal density plot in Figure 6.12(a) shows. However, the congestion stays at the bottleneck and does not affect other parts of the network. Both routes are used at their respective capacities. The performance of the multi-class MPC is $J_{MC MPC}^{TTS} = 1950 \text{ vehh}$.

As can be seen in the control signals, the share of traffic being rerouted is not equally distributed between the vehicle classes. Significantly more cars were rerouted via the alternative route, whereas only a few truck were rerouted during peak traffic. Later in this paper, we will analyze the effect of the traffic state and the incident strength in more detail.

The *mixed-class MPC* performs slightly worse with $J_{SC MPC}^{TTS} = 2040 \text{ vehh}$. As with the MC MPC, the alternative route is used at capacity during the congestion. The congestion pattern (Figure 6.12(c)) is similar to the one of the MC MPC.

The difference in performance between the MC MPC and the SC MPC can therefore only be explained by the traffic composition of the two routes. Due to the nature of the SC MPC, it guides the traffic to the alternative route regardless of class (Figure 6.12(d)). The MC MPC, however, sends more cars than trucks via the alternative route. This is because during congestion, the travel time of the alternative route is shorter than that of the main road. Furthermore, since cars are shorter than trucks, more cars than trucks can flow through any given bottleneck. In other words, the throughput in $\frac{\text{veh}}{\text{h}}$ is higher the more the traffic is composed of cars. Guiding cars via the faster, alternative route therefore leads to a higher total throughput. Figure 6.13 shows the difference of the cumulative vehicle count between the MC MCP and SC MPC scenario at the northern exit over time. For example, at 09:00, 60 more vehicles more had arrived at the northern exit in the MC MPC than in the SC MPC. The vehicles in the MC MPC therefore arrived faster at the destinations. A faster arrival means a lower total time spent and a better performance.

The *reactive mixed-class controller* can guide the traffic only in an all-or-nothing way.

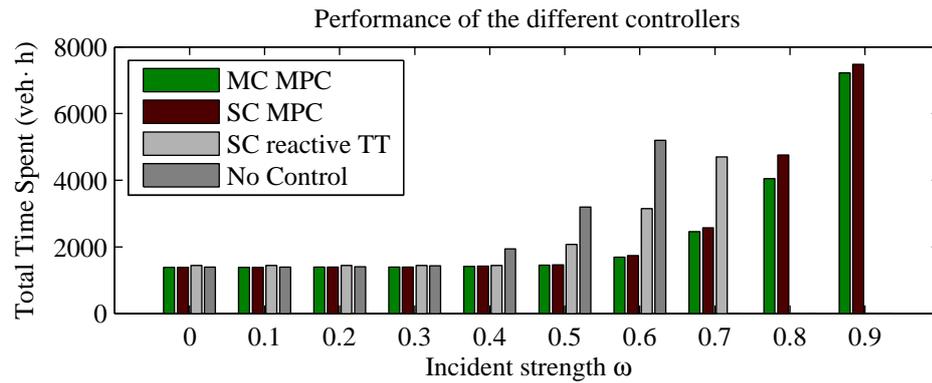
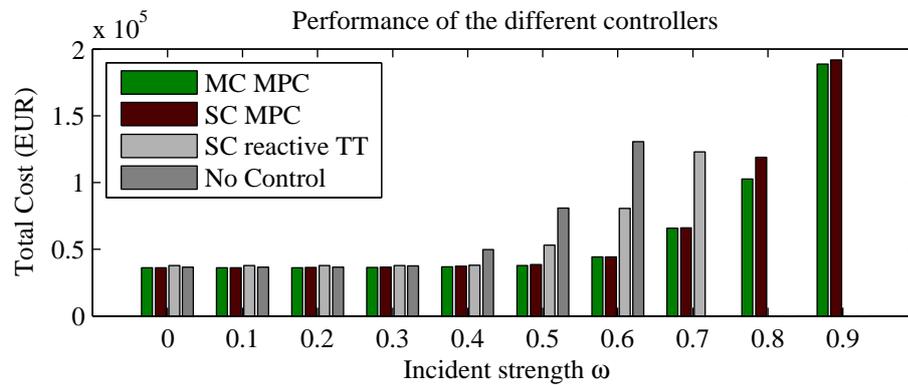
(a) Objective: minimize total time spent J^{TTS} (b) Objective: minimize monetary costs $J^{\text{€}}$

Figure 6.14: Performance of the four route-guidance controllers for varying incident strengths

If the travel time on the alternative route is shorter, then all traffic is advised to use that route, as it happens at 08:30, as Figure 6.12(f) shows. Since the demand exceeds the alternative route's capacity, congestion emerges at Km 5 (Figure 6.12(e)). Since no traffic then uses the main road, the congestion at the incident location dissolves. This causes the travel time to decline, and the controller switches back to the main route at 09:00. This effect repeats multiple times, leading to oscillations in the control signal.

The total time spent $J_{SC\ TT}^{TTS} = 3750 \text{ veh h}$ of the reactive controller is significantly higher than the one of the predictive controllers. One reason for this is that the reactive controller always acts delayed, not being able to anticipate the future traffic state. Another reason is the binary zero-one signal, which always leads to an underutilization of the infrastructure. Finally, since congestion emerges at the bifurcation to the alternative road, the eastbound traffic is blocked as well, which drastically increases the TTS.

Although a zero-one signal is not very realistic, this scenario shows the disadvantage of the oscillations of the control signal. Furthermore, the controller only reacts on the travel time and does not directly optimize the objective of total time spent. It therefore cannot take the eastbound traffic into account and is insensitive to its amount.

For comparison, the base case of *no control* is shown in Figure 6.12(g). Since the traffic is not guided at all, the alternative route is not used. The traffic therefore queues up at the bottleneck and severe congestion emerges. The performance with $J_{\text{NoCtrl}}^{\text{TTS}} = 6260$ vehh is therefore inferior to the other scenarios.

In summary, the multi-class MPC performs slightly better than the mixed-class MPC. Both MPC controllers outperform the reactive controller. (Note that the comparison with the reactive control is somewhat simplified, since the reactive controller is relatively simple, such as rerouting in an all-or-nothing fashion, or reacting with relatively large control intervals.) The no-control case is the worst.

This trend is also visible for other incident strengths and for the objective of minimizing the monetary costs, especially for the cases in which congestion cannot be prevented ($\omega \geq 0.6$). Figure 6.14 shows the performance of the controller over different incident strengths ω for the total time spent J^{TTS} (5.29) and for the monetary costs $J^{\text{€}}$ (5.28). (For readability, the performance results of the non-MPC controllers have been omitted for heavy incidents, which led to a very high objective values.)

6.4.3.2 The Influence of the Value of Time

This section analyzes the results of the multi-class MPC for the two objective functions, aiming at the minimization of the total time spent J^{TTS} (6.20), and the minimization of the monetary costs $J^{\text{€}}$ (6.21). Note that the difference between these two is only the class-specific value of time ζ_u . In the following, four example simulation runs are presented and compared. Figure 6.15 shows the control trajectories of the MC MPC for both objective functions and for different incident strengths.

Let us first analyze the control signal during a *light incident* ($\omega = 0.4$). Congestion is prevented by rerouting some of the traffic via the alternative route. For the objective of total time spent (essentially, a truck is as valuable as a car), trucks are rerouted via the slower alternative route, as shown in Figure 6.15(a). Therefore, there are many short vehicles (cars) on the main route, which leads to a high throughput on the main route, and thus to a low TTS.

In contrast, for the objective of total cost (essentially, a truck is three times as valuable as a car), cars are rerouted instead, as Figure 6.15(b) shows. Then, trucks experience a short travel time. The loss of throughput (in $\frac{\text{veh}}{\text{h}}$) on the main route is compensated for by the higher value of time of the trucks, which thus leads to low costs.

Let us now analyze the control signal during a *heavy incident* ($\omega = 0.7$). In this case, congestion is severe and leads to long travel times on the main road. In the case of the objective of total time spent, cars are guided via the alternative route. Since the alternative route is faster, throughput is maximized there by using it for short vehicles, which leads to a low TTS.

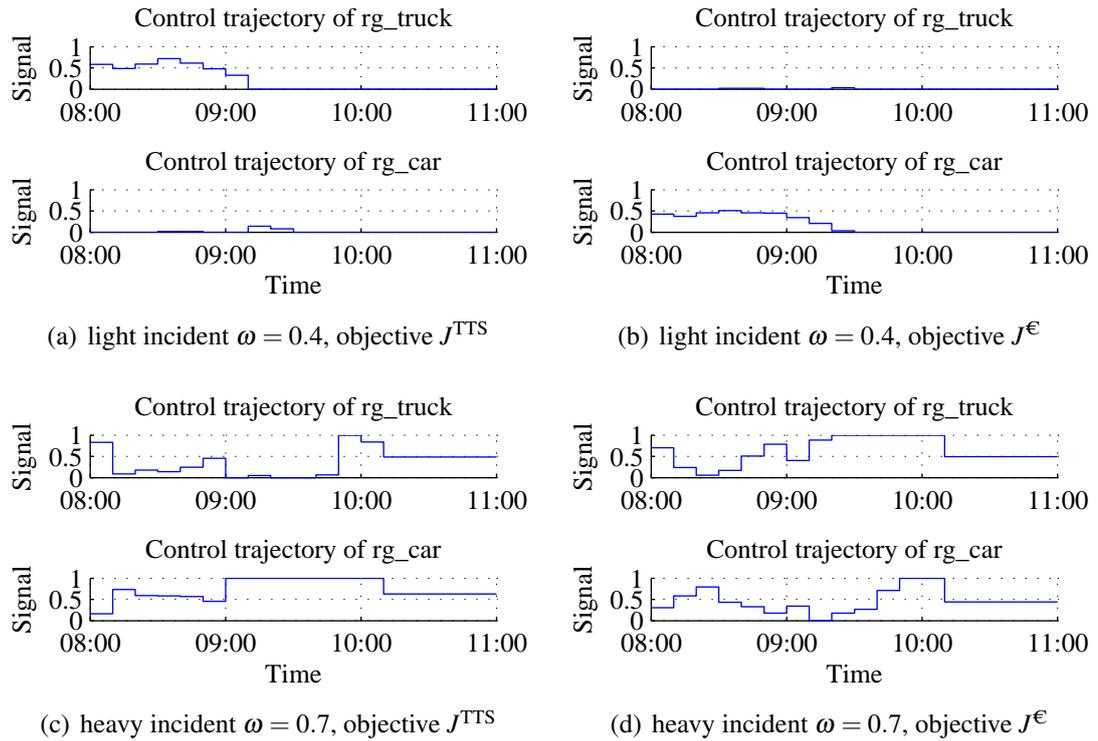


Figure 6.15: Control signals of the multi-class MPC for different values of time and different incident strengths for trucks (rg_truck) and cars (rg_car)

In contrast, for the objective of total cost, the valuable trucks are rerouted instead. Similarly to the case of the light incident, to minimize the costs it is advantageous to enable a short travel time for the valuable vehicles.

In summary, the multi-class controller uses the ability of realizing different prioritizations depending on the value of time ζ_u . The more valuable a vehicle is, the faster it should travel through the network. The control signals therefore can change drastically if the value of time is changed.

6.4.3.3 The Influence of the Incident Strength

This section analyzes the influence of the incident strength ω on the control signals of the multi-class MPC. The objective function used is the total cost with an adjustment: the eastbound traffic is considered very important ($\zeta_{\text{truck, east}} = 10\zeta_{\text{car}}$, $\zeta_{\text{truck, north}} = \zeta_{\text{car}}$), and delays of it lead to a sharp increase of the total cost.

In the case of *no incident* ($\omega = 0$), the capacity of the main route is sufficiently large so that the traffic can flow freely without causing congestion (Figure 6.16(a)) and (almost) no rerouting is necessary (Figure 6.16(b)).

In the case of a *tiny incident* ($\omega = 0.3$), some of the traffic is rerouted (Figure 6.16(d)) via the alternative route to prevent congestion. Figure 6.16(c) presents the travel times

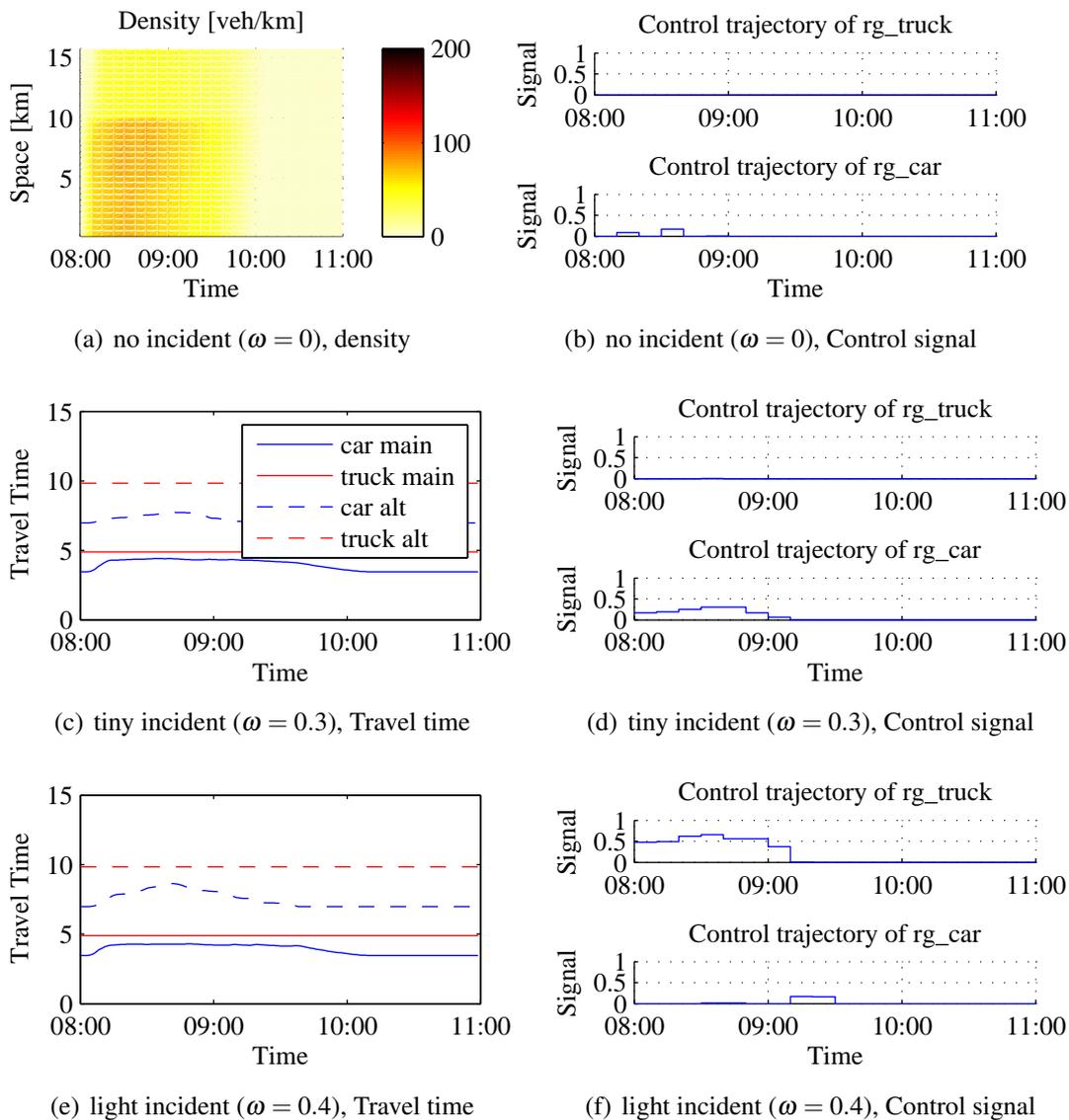


Figure 6.16: Results of multi-class MPC for different incident strengths for trucks (rg_truck) and cars (rg_car), including travel time on both the main route (main) and the alternative route (alt)

for both vehicle classes on both routes. As can be seen, cars traverse the alternative route in 7.5 min (blue dashed line), whereas trucks need 10 min (red dashed line). Cars are thus much faster than trucks on that route. Rerouting cars via the fast alternative route therefore leads to the optimal TTS.

In the case of a *light incident* ($\omega = 0.4$), more vehicles have to be rerouted to prevent congestion. Since many vehicles now have to use the alternative route, the density is nearly critical so that all vehicles drive at nearly the same speed. As Figure 6.16(e) shows, at 08:30, the travel time for cars is 9 min, and for trucks it is 10 min. The travel times are thus almost equal on that route. Therefore, it is optimal to keep the small vehicles on the fastest route in order to maximize the throughput (Figure 6.16(f)). The

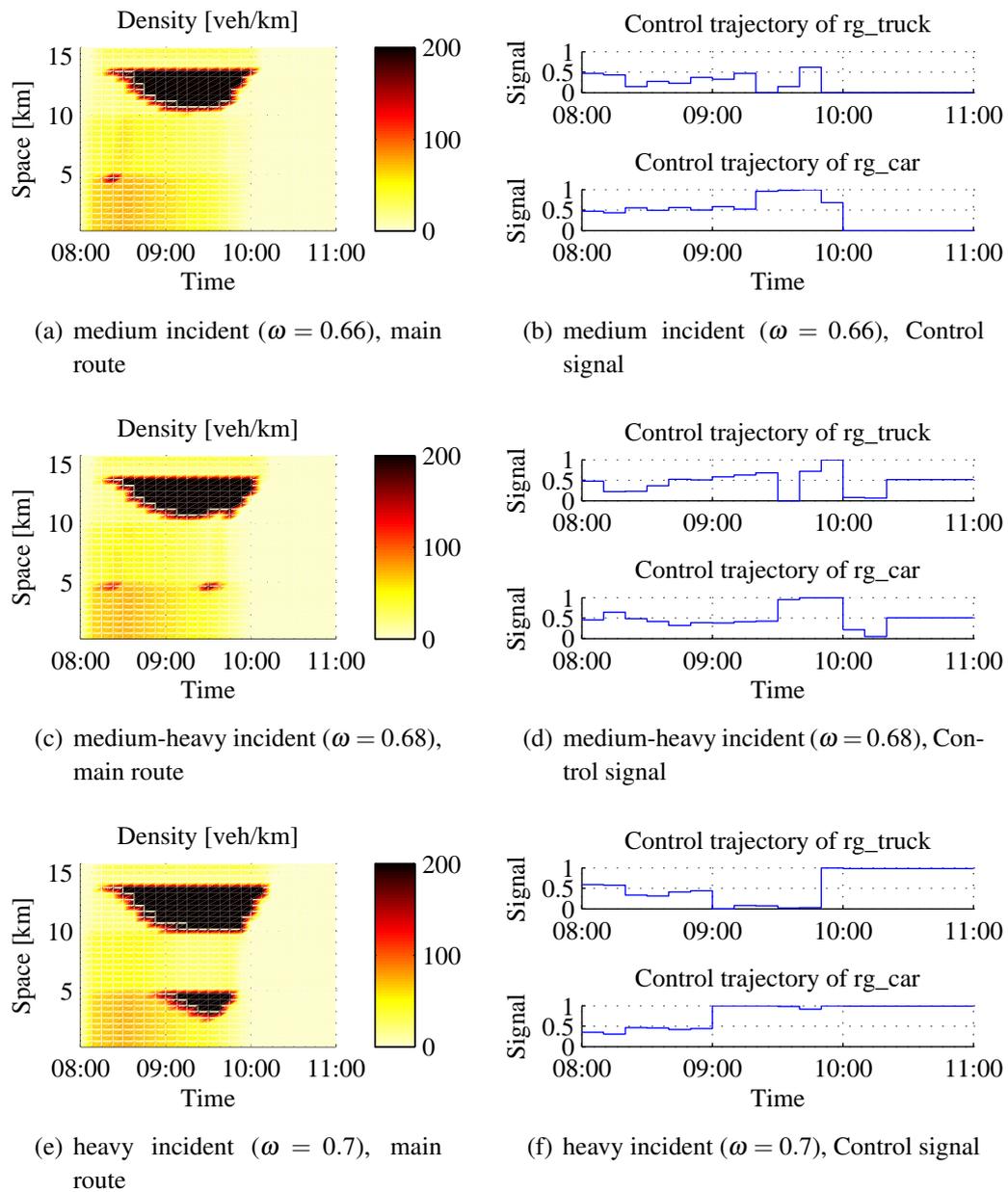


Figure 6.17: Results of multi-class MPC for different incident strengths (cont.)

fastest route is the main route, where the travel is 5 min for both classes.

In the case of a *medium incident* ($\omega = 0.66$), the demand exceeds the joint capacity of both routes, and congestion emerges at the incident location (Figure 6.17(a)). Since the alternative route is now faster than the main route, more cars than trucks are rerouted via the alternative route (Figure 6.17(b)).

The more the congestion increases, the more cars are expected to be rerouted via the alternative route. However, another effect becomes decisive in the case of a *medium-heavy incident* ($\omega = 0.68$). The congestion threatens to spill back over the off-ramp at Km 10 (Figure 6.17(c)), and thereby blocking the eastbound traffic. The controller manages to prevent such a spillback by rerouting more trucks via the alternative route

Incident strength	Class rerouted	Reason
no incident	—	no delay
tiny incident	cars	on alternative route, much shorter TT for cars compared to trucks
light	trucks	same TT on alternative route, while higher throughput on faster main route
medium	cars	alternative route faster than main route
medium-heavy	trucks	prevent spillback
heavy	cars	cannot prevent spillback, alternative route faster than main route

Table 6.2: Rerouting of the vehicle class by the multi-class MPC dependent on the incident strength

(Figure 6.17(d)) and instead placing short cars into the congestion. The reason for this is the dynamic pce value of the trucks: as illustrated earlier, in congestion, a truck occupies significantly more space than a car; in free-flow (as is present on the alternative route), a truck occupies only slightly more space than a car. Consequently, the throughput of the alternative route decreases. In other words, the prevention of spillback to the off-ramp outweighs the reduction of throughput of the alternative route.

In the case of a *heavy incident* ($\omega = 0.70$), congestion reaches back to the on-ramp and spillback can no longer be prevented after 09:00 (Figure 6.17(e)) so that all vehicles including the eastbound traffic experience congestion. Since the eastbound traffic is now delayed, the total time spent increases severely. The throughput is maximized by rerouting the small cars via the alternative route (Figure 6.17(f)). Given the constraints by this heavy incident, the solution is optimal, when the controller puts the congestion at the bifurcation point. It controls the traffic in such a way that the downstream congestion does not block the off-ramp.

In summary, the rerouting of the vehicle classes by the controller depends on the severity of the incident. The rerouting thereby is not monotonically related to the incident strength, but varies for different scenarios. Table 6.2 summarizes the results of the rerouting of the multi-class MPC dependent on the incident strength.

6.4.4 Conclusion

This section generalized route guidance to multiple vehicle classes and tested it in simulations with a model-predictive controller. The multi-class MPC outperforms the

mixed-class MPC with up ten percent. Especially during congestion, the multi-class controller is able to distribute the vehicle classes over different routes, in order to enable a short travel time for one of the classes and to reduce spillback effects. Furthermore, both the multi-class and the mixed-class MPC outperform a reactive controller based on measured travel times that does not take the value of time and the pce value into account.

The control signal of the multi-class MPC is sensitive to the value of time (VOT). If the total time spent is optimized (equal VOT for each class), then cars are prioritized because the throughput is maximized in terms of vehicles per hour. In contrast, if the monetary costs are optimized (high VOT of truck), then trucks are prioritized because the throughput is maximized in terms of Euros per hour.

Furthermore, the multi-class control signal is sensitive to the incident strength. In cases of light incidents, a low total time spent is achieved by routing the shortest vehicles (cars) to the fastest route. If the incident is very light so that no congestion emerges, cars are therefore guided via the main route, whereas trucks are rerouted via the alternative. If the incident is stronger so that congestion emerges, then cars are rerouted via the alternative, since it is faster than the main route. In the case of a heavy incident with severe congestion, the minimization of spillback of congestion to upstream infrastructure is crucial so that long vehicles should be kept out of congestion and therefore should be rerouted.

6.5 Conclusion

In this chapter, we developed multi-class DTM measures and showed in experiments that they improve the traffic performance. We focused here on traffic flow, notably the effects on throughput, spillback and travel time, and ultimately on the total cost of the traffic.

The DTM measures were first categorized into two approaches. A DTM measure can separate the vehicle classes within a carriageway, or it can distribute the vehicle classes across the network. Given these approaches, three multi-class DTM measures were developed, tested and analyzed based on the traffic flow model Fastlane.

The first multi-class DTM measure developed was vehicle-class specific lanes, which separates the classes on the carriageway by dedicating a lane to a specified class. Here, we looked at the effects on the monetary throughput of truck lanes. Analytic experiments with realistic parameter values based on the values of the Dutch road authority showed that truck lanes are beneficial if the truck lane is used to at least three quarters of its capacity.

The second multi-class DTM measure developed was multi-class ramp metering, which separates the vehicle classes by using dedicated lanes and controls the access to the

freeway by a traffic signal. The traffic management center is able to optimize different objective by specifying a desired vehicle-class composition that flows into the network. By prioritizing cars, the throughput at the on-ramp, the number of vehicles delayed and therefore the total time spent are optimized. In contrast, by prioritizing trucks, the spillback of congestion on the on-ramp and the total costs are optimized. In the experiments, the multi-class ramp metering reduced the total cost by 33% compared to mixed-class ramp metering.

The third multi-class DTM measure developed was multi-class route guidance, which separates the classes across the network by advising a route to each vehicle class at a bifurcation point. Experiments with a model-predictive controller show that the performance of the network is slightly improved if class-specific routing is applied. The multi-class controller is able to distribute the vehicles in order to leverage the differences in travel time and to minimize spillback. We found performance improvements of up to 10%.

The experiments showed that all of the multi-class DTM measures developed here improve the traffic flow and the network performance and outperform their mixed-class counterparts. Moreover, we showed that the traffic management center can decide which objective to optimize by specifying the inflow composition in the case of multi-class ramp metering, or by specifying the turnfraction of the multi-class route guidance. Furthermore, the multi-class route-guidance was implemented in a model-predictive control framework so that a given network performance function of total cost was optimized.

Other costs like the implementation of the control devices or externalities like emissions or safety were beyond the scope, since they do not influence the traffic flow. A further assumption is that the (effective) capacity is modeled statically in the fundamental diagram, i.e. weaving and similar vehicle maneuvers do not further affect the capacity, although empirical research has shown that the spatial separation of vehicle classes by a dedicated lane leads to an increase of capacity (Cassidy et al., 2009). The results obtained are thus a lower bound so that the benefits of multi-class control might potentially even be higher.

This concludes the Control component of BOS-HbR (Figure 2.11). The following chapter combines the multi-class control concepts developed and tested in this chapter with the state estimator computationally optimized in Chapter 4. In a case study, the multi-class route guidance controller and the multi-class ramp meter are applied to a model of the Dutch A15 in order to optimize traffic during an incident and under regular conditions.

Chapter 7

Case Study: Multi-class Control of the Dutch Freeway A15

In this chapter, we evaluate how multi-class control improves the traffic performance of a realistic freeway . The three components of the control loop outlined in Figure 2.11 – Estimation, Prediction and Control – are combined into the system BOS-HbR and applied to the Dutch freeway A15. In a case study, the effects of the multi-class DTM measures developed in the previous chapter are simulated in regular and in incidental traffic conditions, and are then compared to their mixed-class counterparts. The results show that the total costs of traffic is slightly reduced if predictive multi-class control is applied.

The site of the Dutch A15 eastbound near the harbor of Rotterdam is presented in Section 7.1. The Estimation component of the control loop is set up in Section 7.2 and an example result is presented. For the Prediction component, the traffic flow model Fastlane shown in Sections 5.1 and 5.2 is used to simulate the traffic conditions of the A15. Its parameters are calibrated based on historic data in Section 7.3; it is then validated both under regular and under incidental conditions in Section 7.4. The multi-class Control component is set up in Section 7.5. The case study is prepared both for normal and incidental conditions in Section 7.6. The results in Section 7.7 show that multi-class control improves the traffic performance. In these experiments, the setup is chosen in such a way that the accuracy of the controller is high, at the expense of the computation time. Nevertheless, Sections 7.8 and 7.9 discuss ways to improve the performance and the computation time of BOS-HbR, respectively. This chapter closes with conclusions in Section 7.10.

This chapter is an extended version of Schreiter, Van Lint, & Hoogendoorn (2013): “Vehicle-class Specific Control of Freeway Traffic” presented at the 92nd Annual Meeting of the Transportation Research Board.



Figure 7.1: The network of the A15 site for the case study

7.1 A15 Site Description

BOS-HbR is developed for the A15 near the harbor area of Rotterdam and is used in this case study. This section describes the network of the site, the traffic problems that often arise there, and the data sources that observe the traffic.

7.1.1 Network of the Site

Figure 7.1 shows the network that is used in this case study. The main route is the freeway A15 eastbound near the harbor of Rotterdam, the Netherlands. This freeway is the main connection between the harbor and the hinterland. Because of the harbor, many trucks travel via the A15. Figure 7.2 shows the median truck percentage over time, based on three weeks of working days. Ten to fifteen percent of the traffic is composed of trucks during peak hours; this share is even higher during the off-peak hours in the early afternoon. Furthermore, the narrow bandwidth of the truck percentage during the day (the quartiles are close to the median) indicates that this truck percentage is fairly stable over multiple working days. The remaining traffic is mostly composed of cars, while the share of the other vehicle classes like buses or motorbikes is negligibly small.

The highest traffic demand is observed between the Km 32 near Rozenburg and Km 58 near the interchange Vaanplein. In fact, the traffic demand is so high that congestion emerges at Km 44 near Spijkennisse and at Km 56 near Charlois on a regular basis during the evening peak of a normal workday.

There is an alternative route via the underlying road structure. The route from Spijkennisse via the Botlek bridge (Botlekbrug) provides one lane. This alternative route connects again to the A15 after the bridge. Furthermore, the alternative route also continues via the Vondelingenweg, which is first one and then two lanes wide. The alternative route finally connects back to the A15 at the on-ramp Charlois. The main purpose of the alternative route is to connect the surrounding harbor area to the road network. Nevertheless, especially during off-peak hours, the alternative route can be used for rerouting traffic if an accident blocks the traffic on the A15.

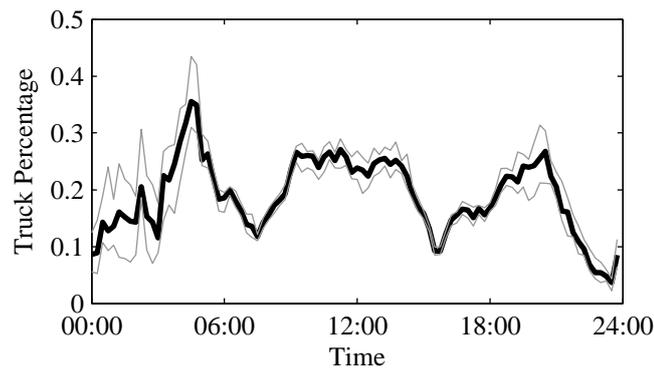


Figure 7.2: Truck percentage of the A15, median (thick black) and quartiles (thin gray) of the workdays observed at Km 45 near Spijkenisse

7.1.2 Recurring and Nonrecurring Traffic Problems on the Site

The A15 is subject to three major problems. The first problem is the recurrent congestion that emerges at the two bottlenecks at Km 44 (Spijkenisse) and Km 56 (Charlois). The congestion lasts for several hours during the evening peak and leads to high total costs of traffic.

The second problem is incidents. On average, incidents occur every three days on the A15 (Van Zuylen et al., 2007) and are sometimes so severe that congestion emerges. If these occur between Spijkenisse and Charlois, then some of the traffic can be rerouted via the alternative route.

The third problem is caused by construction. Over the next years, the A15 will be expanded to increase the capacity, which aims at improving the traffic conditions. During the construction, however, the network structure is changing, which will affect the traffic flow; it can even create new bottlenecks.

The model-predictive control approach chosen in Section 2.1 is suitable to work under incidental conditions, since their effects on traffic flow are directly taken into account by the Prediction component. It furthermore is flexible to deal with the changing infrastructure that will occur in the coming years.

7.1.3 Data Sources for Estimation and Calibration

Multiple sensors are installed on the site to collect traffic data. Table 7.1 lists the sensor types that are used in this thesis. (Note that there are more data sources available in the harbor area such as Bluetooth sensors or weigh-in-motion detectors, which are applicable for Dynamic Traffic Management, too, although they are out of the scope of this thesis. The Port of Rotterdam Authority operates a number of these data sources and provides information in real-time on the website of Roportis (2012).)

Sensor type	Data provided	Data availability
induction loops RWS normal	aggregated speed and flow	online, offline
induction loops HbR	individual vehicle data	online, offline
induction loops RWS detailed	individual vehicle data	offline
camera from helicopter	video	offline

Table 7.1: Sensors providing traffic data of the site

The induction loops by the Dutch road authority Rijkswaterstaat (RWS) cover the freeway of the A15 to provide speed and flow information. These data are aggregated over one minute. The distance between the detectors is half a kilometer on average so that a close estimation of the traffic state is possible. These data are available in real-time; the data base Regiolab Delft (2012) operated by the Delft University of Technology provides this information within a few minutes.

The induction loops by the Port of Rotterdam authority (HbR) observe the traffic state at some locations of the underlying network structure. These sensors provide individual vehicle data such as the vehicle length and the passing time, which allows estimating class-specific data such as the traffic composition. These data are gathered by the data base Roportis (2012) by the Port of Rotterdam in real-time.

To calibrate the prediction model well (Section 7.3), the vehicle-class specific data such as the traffic composition of the A15 are required. For that purpose, the induction loops by Rijkswaterstaat can be run in a detailed mode so that they provide individual vehicle data, similar to the HbR induction loops. We gathered offline data at some locations of the A15 over a period of a few weeks.

Furthermore, to gather detailed data, we captured videos from a helicopter perspective. Among other data, the spacings of the vehicles and the trajectories can be estimated. Figure 2.5 shows a frame a video of the A15.

7.1.4 Multi-class DTM Measures Used in the Case Study

Three DTM measures are simulated in this case study to test the potential benefits of multi-class control. One multi-class route guidance measure (MCRG, Section 6.4) is set up at the bifurcation point between the main route and the alternative route near Spijkenisse. A second MCRG is located at the alternative route after the Botlekbridge, where the alternative route is connected to the main route. The turnfractions for both vehicle classes are controlled.

Furthermore, a multi-class ramp meter (MCRM, Section 6.3) is set up at the on-ramp connecting the A4 to the A15. The desired share $\hat{\eta}$ and the set point density \hat{K}_{tot} (6.18) of the ramp meter algorithm Alinea are controlled.

7.1.5 The System BOS-HbR Applied to the A15

The goal of the system BOS-HbR (Figure 2.11) is to gather sensor data, estimate the current traffic state, and optimize the control signals for the three DTM measures by an model-predictive control approach.

In practical application, the control signals computed by BOS-HbR can be advised to an operator in a traffic management center. The operator then can decide whether to implement the signals or not. In addition, the future traffic states both under the optimal control case and under the no-control case can be shown to the operator.

In the case study of this chapter, the traffic process is simulated by a traffic flow model, and the control signals are directly applied to the DTM measures and the traffic process. In the following sections, the components of BOS-HbR are set up and then the case study is performed.

7.2 Setup of Estimation Component

The Estimation component is the first part of BOS-HbR. It collects the data from the sensor data bases and fuses them to calculate the current traffic state. Only those sensors that gather the data online are used as data sources for the state estimation process, namely the inductive loops of Rijkswaterstaat and of the Port of Rotterdam.

As traffic state estimator, we apply the computationally improved version of the Adaptive Smoothing Method (ASM) described in Section 4.2. Its kernels defined by (4.4) are parameterized with sizes of $\sigma = 300\text{m}$ and $\tau = 30\text{s}$. Furthermore, the characteristic wave speeds c_{cong} and c_{free} are estimated online by the Wave Speed Estimator developed in Chapter 3.

To illustrate an example result of the Estimation component, Figure 7.3 shows the spatiotemporal speed plot of a working day at 16:15. The figure shows that both regular bottlenecks at Km 56 (Charlois) and at Km 44 (Spijkenisse) are active so that congestion emerged. Furthermore, a heavy incident occurred at Km 41 at 16:05 so that the capacity is significantly reduced there. In the figure, the emerging congestion is visible, as the speed suddenly drops to nearly zero. Due to the low capacity at the incident site, the congestion at the regular bottleneck at Km 44 dissolved.

Conversion from mixed-class to multi-class traffic state

Since the online loops of Rijkswaterstaat do not measure the truck percentage, only mixed-class data are available. However, since the variation of the truck share η varies little over the working days (see Figure 7.2), it can be estimated by historic data. This part shows in four steps how the estimated mixed-class traffic state expressed in speed v and flow q_{tot} is combined with the truck share η to compute the multi-class traffic state

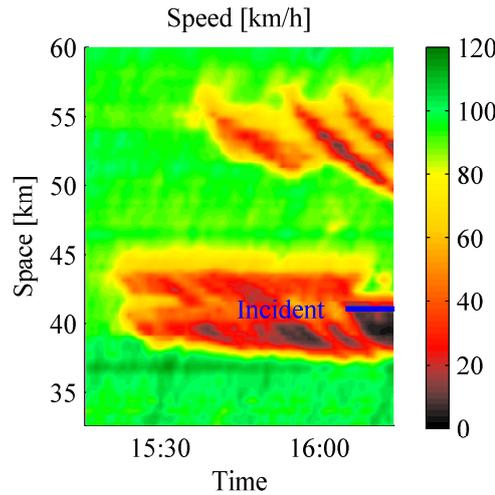


Figure 7.3: Result of the Estimation component for a day (06-07-2011 at 16:15), speed contour over space and time; clearly visible are the recurrent congestions at Km 44 (Spijkenisse) and Km 56 (Charlois); an incident occurred at Km 41 at 16:05

in terms of the class-specific effective densities K_u for each class u . In the following, only the algorithm for the conversion is shown, whereas the proof is shown in Appendix B.

In the first step, the estimated average speed v and the total vehicular flow q_{tot} of the most recent data, i.e. the state vector over space at 16:15 in case of Figure 7.3, determine the total vehicular density by

$$k_{\text{tot}} = \frac{q_{\text{tot}}}{v} . \quad (7.1)$$

In the second step, the total effective density K_{tot} is determined by the solution of

$$\begin{aligned} & \left[\frac{\eta}{1-\eta} v_{\text{car}}^{\text{FD}}(K_{\text{tot}}) + v_{\text{truck}}^{\text{FD}}(K_{\text{tot}}) \right] K_{\text{tot}} \\ & - \left[\frac{\eta}{1-\eta} \pi_{\text{truck}}(K_{\text{tot}}) v_{\text{car}}^{\text{FD}}(K_{\text{tot}}) + v_{\text{truck}}^{\text{FD}}(K_{\text{tot}}) \right] k_{\text{tot}} = 0 \end{aligned} \quad (7.2)$$

for $K_{\text{tot}} \in [0, K_{\text{jam}}]$, based on the fundamental diagram v_u^{FD} (5.8) and the pce function π_{truck} (5.12).

In the third step, the class-specific speeds v_u are determined by the fundamental diagram:

$$v_u = v_u^{\text{FD}}(K_{\text{tot}}) . \quad (7.3)$$

In the fourth and final step, the class-specific vehicular densities are calculated by

$$k_{\text{car}} = \frac{v_{\text{truck}}}{\frac{\eta}{1-\eta} v_{\text{car}} + v_{\text{truck}}} k_{\text{tot}} , \quad (7.4)$$

$$k_{\text{truck}} = k_{\text{tot}} - k_{\text{car}} . \quad (7.5)$$

Parameter	Data source	Section
Network topology	map	7.3.1
Flow of user classes		7.3.2
Total inflow into network	loops RWS, HbR	7.3.3
Total turnfraction at bifurcation nodes	loops RWS, HbR	
Traffic composition	loops RWS detailed	
Pce function	video	7.3.4
Capacity at non-bottlenecks	areal map	7.3.5
Capacity at regular bottlenecks	loops RWS	7.3.6

Table 7.2: Calibration parameters of the Predictions component of the case study

The current traffic state defined by the class-specific vehicular densities k_u is used in the Prediction component to initialize the traffic flow model. Before the Prediction component can be used, however, its parameters must be calibrated so that the traffic state predicted by the model matches the true traffic state as closely as possible. This is the topic of the following section.

7.3 Setup and Calibration of the Prediction Component

In the Prediction component of BOS-HbR, the multi-class traffic flow model Fastlane explained in Sections 5.1 and 5.2 is used to simulate and reproduce the traffic conditions of the site. This section describes how Fastlane is calibrated to the A15. Section 7.4 validates the model.

Table 7.2 summarizes which parameters have to be calibrated. The network topology is modeled based on maps of the area. Section 7.3.1 provides the network used in Fastlane. Since some of the traffic can be rerouted via the alternative route, the flow of traffic is distinguished by modeling multiple user classes in Section 7.3.2.

Traffic enters the network at its boundaries via the origin links, which are modeled by class-specific inflows over time. These are calibrated based on historic data gathered by induction loops. Similarly, traffic is split at bifurcation nodes, which are modeled by class-specific turnfractions. Analogously to the inflows, the turnfractions are estimated based on historic loop data. Since the RWS loops provide only quantities aggregated over all vehicle classes, individual loop data are required to estimate the traffic composition and thereby the class-specific inflows and turnfractions. Section 7.3.3 calibrates both the inflow and the turnfractions for each user class.

The pce function of Fastlane requires detailed data about the spacings of the vehicles. Loop data provide information only about the time headway, and only with a resolution

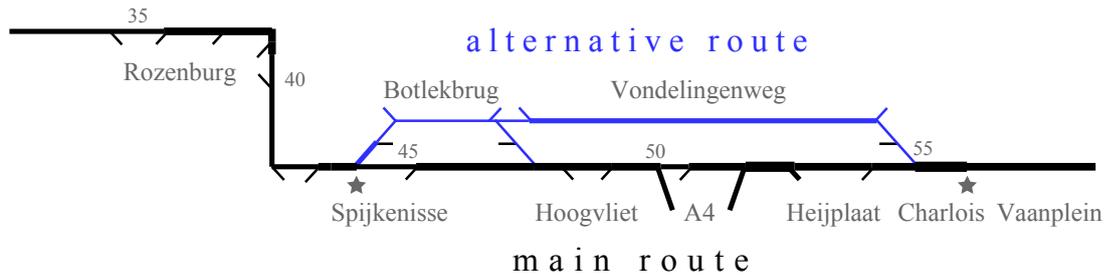


Figure 7.4: Network model of the site; the width of each line indicates the number of lanes modeled, the stars indicate the location of the regular bottlenecks

of 1 s, which is too coarse to estimate the pce function well. Video data, however, directly provide spatial information with a high resolution. The pce function is calibrated in Section 7.3.4.

Finally, the fundamental diagram of each part of the network is estimated. Since the capacity can only be observed at bottlenecks, the fundamental diagram at non-bottlenecks is based on the number of lanes at that location (Section 7.3.5). The capacity of the regular bottlenecks is estimated by an iterative optimization method (Section 7.3.6). The remainder of this section shows the calibration procedure for each of these parameters. The parameter values found are then validated in Section 7.4.

7.3.1 The Network Model

The network layout is directly visible in maps, for example in the ones provided by Google Maps (2012). Figure 7.4 shows the network model. The width of the lines indicates the number of lanes modeled, varying from one lane for most of the on- and off-ramps to four lanes between the on-ramp Charlois and the bottleneck Charlois. Each on-ramp is modeled as an origin-link, for which the inflow has to be calibrated. Each of the off-ramps is modeled as a destination-link, for which the turnfraction of the adjacent bifurcation node has to be calibrated (Section 7.3.3).

The alternative route is shown in blue. To abstract from all inflows and outflows on the alternative route, its background traffic is summarized in the following way: background traffic enters the alternative route at the on-ramp near the beginning and leaves it at the next off-ramp near its end. Since the alternative route consists of the two parts of the Botlekbrug and the Vondelingenweg, two pairs of such on- and off-ramps are modeled. Furthermore, since Fastlane models only nodes with at most three adjacent links, the off-ramp Spijkenisse and the on-ramps Heijplaat and Charlois are modeled as part of the alternative route. The following section explains in detail how the traffic flows of both routes are modeled by user classes.

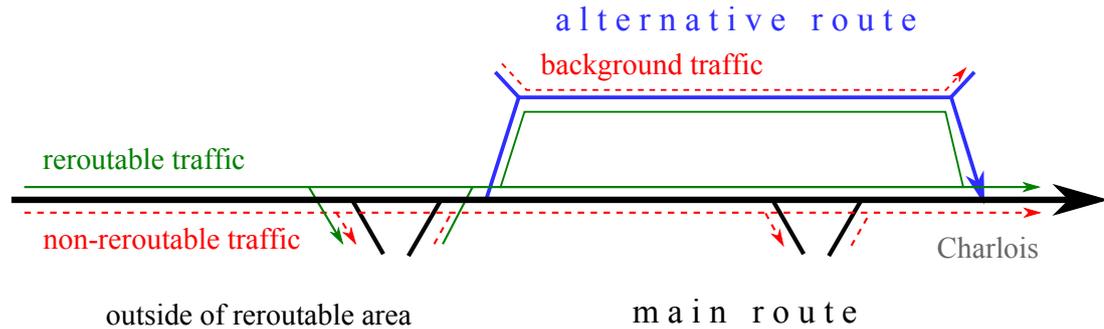


Figure 7.5: Modeling of destination classes: Traffic passing Charlois is reroutable (green solid), traffic that leaves the network earlier is not reroutable (red dashed); furthermore, there is background traffic on the alternative route (red dashed)

7.3.2 Model of the User Classes and Their Flow Through the Network

Multiple user classes are modeled in the case study. In the following, they are distinguished by vehicle classes and by destination classes.

The traffic of the A15 is composed of multiple vehicle classes. We distinguish two vehicle classes, namely cars and trucks. As Figure 2.6 illustrated, trucks are significantly longer than cars, and the relative spacing between them is dynamically dependent on the traffic conditions. As explained in Section 5.1, the traffic behavior of the vehicle classes is modeled by the pce function π (5.12); it is calibrated in Section 7.3.4.

A further distinction is made dependent on the destination. Traffic either passes the point where the alternative route connects back to the A15 at Charlois or it leaves the network before that point. This leads to multiple destination classes, as Figure 7.5 illustrates. Traffic that passes Charlois can be rerouted via the alternative route (green solid). Conversely, traffic that leaves the network before Charlois cannot be rerouted (red dashed). Furthermore, there is background traffic on the alternative route, which is another destination class. However, since this background traffic of the alternative route does not interfere with the non-reroutable traffic of the main route, it is modeled by the non-reroutable destination class (red dashed) as well.

In total, there are thus two vehicle classes and two destination classes. We model all combinations thereof by four user classes. Table 7.3 summarizes these four user classes. The state of the network traffic, the inflows and the turnfractions are thus defined for each of the four user classes.

7.3.3 Calibration of Inflows and Turnfractions

Traffic is entering the network at its boundaries via the origin-links, and it is split at bifurcation nodes of the network. These are modeled by class-specific inflows and

User class	Vehicle class	Destination class	Reroutable
1	car	after Charlois	yes
2	truck	after Charlois	yes
3	car	before Charlois or background traffic	no
4	truck	before Charlois or background traffic	no

Table 7.3: User classes in the case study

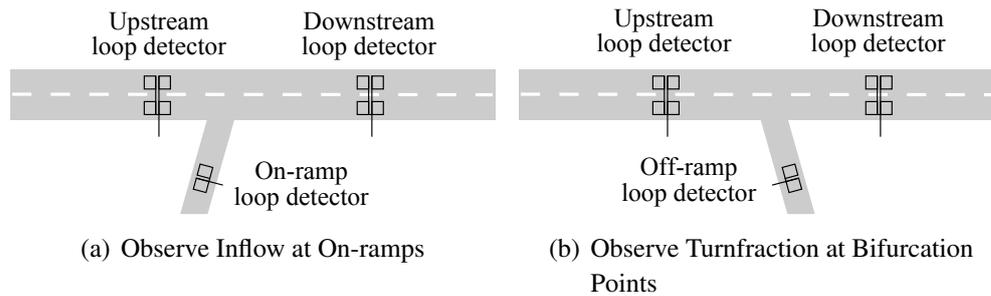


Figure 7.6: Detector configuration for calibrating inflow and turnfraction

class-specific turnfractions, respectively. They are calibrated based on offline data from induction loops.

The calibration of the inflows and turnfractions is divided into two steps. First, the inflow and turnfraction per location are calibrated in total, i.e. as the sum of all classes. Then, they are split into multiple user classes.

Calibration of Total Inflow and Total Turnfractions

The total inflows and total turnfractions are calibrated by offline loop data, based on the total flow. Both inflow and turnfractions can be observed in two different ways, dependent on the detector configuration. Figure 7.6 shows detector configurations at on-ramps and bifurcation points occurring in practice.

Figure 7.6(a) shows a detector configuration to observe the inflow at an on-ramp. If a loop-detector is placed on the on-ramp, then the inflow is directly measurable. If the on-ramp is not equipped with a detector, then the inflow is observed indirectly as the difference of flows between the downstream and the upstream detector.

Figure 7.6(b) shows a detector configuration to observe the turnfraction at a bifurcation point, such as an off-ramp or an interchange. If the off-ramp is equipped with a detector then the turnfraction is calculated directly by the off-ramp flow divided by the upstream flow. If the off-ramp is not equipped, then the turnfraction is determined by one minus the downstream flow divided by the upstream flow.

For either case, data from the whole year of 2010 are used to calibrate the total inflows and turnfractions. First, the data are aggregated into intervals of 15 min. Then the data

are categorized into seven categories, one for each day of the week. Finally, for each of the 15-min-intervals and for each weekday, the median is applied. This result is the total inflow or turnfraction, respectively.

Two examples of calibrated total inflows and total turnfractions are shown in Figure 7.7. The inflow pattern for each weekday of the on-ramp from the A4 is shown in Figure 7.7(a). The thin grey lines are the observed aggregated data (i.e. one line for each of the 52 weeks of the year). The thick black line is the median, which is used as the calibrated total inflow. Clearly visible is the typical bimodal M-shape of the working days, indicating the morning and the evening peak caused by the commuter traffic. During the weekend, traffic demand is much lower and practically no commuting traffic is present. Similarly, Figure 7.7(b) shows the turnfraction for off-ramp to the A4. Since the turnfraction is a relative measure, the commuting traffic is not visible there.

Calibration of Class-Specific Inflows and Class-specific Turnfractions

The class-specific inflows and class-specific turnfraction are determined based on their total quantities estimated above. They depend both on the user class and on the location in the network.

At some locations, vehicle-class specific data were gathered from the RWS loops which provide the truck share η_{truck} at that section of the road. If no truck share is available for some locations, then a default truck share of $\eta_{\text{truck}} = 15\%$ is used in the calculations below.

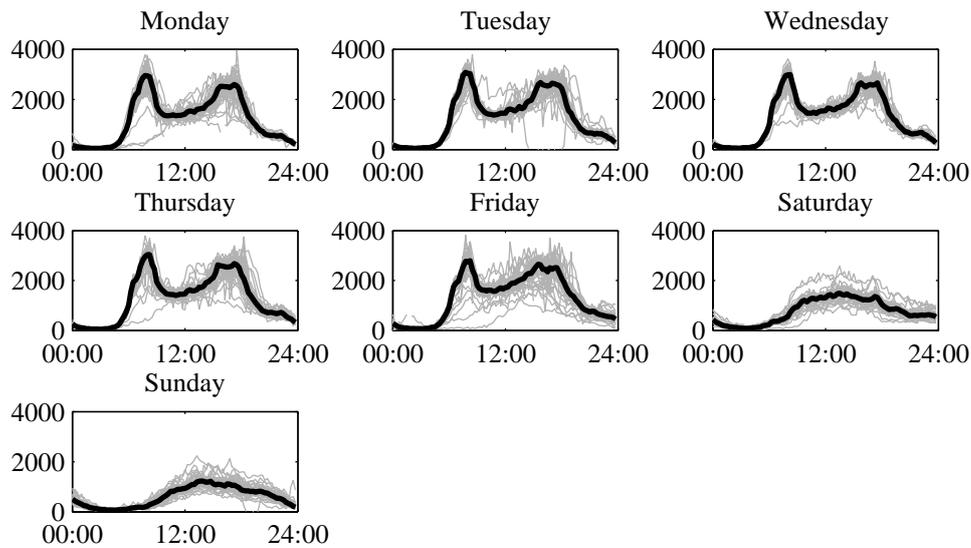
Furthermore, the classes are split according to their destination, as was illustrated above in Figure 7.5. A part of the total flow estimated at the bifurcation point to the alternative route (at Km 44 at Spijkenisse) wants to pass Charlois. We assume that this part is at least $\bar{\eta} = 18\%$, i.e. in the model at most 18% of the traffic is reroutable.

In the following, the user-class specific inflow compositions and user-class specific turnfractions are calculated based on their location, given the truck percentage η_{truck} and the reroutable share $\bar{\eta}$. Table 7.4 summarizes how the inflow composition and turnfraction for all user-classes are calculated.

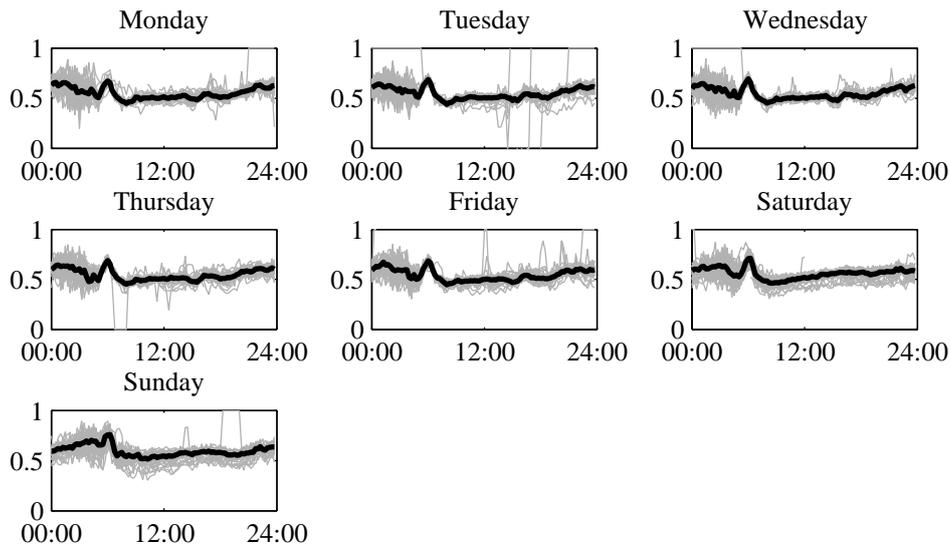
Outside of the reroutable area, i.e. before the bifurcation at Spijkenisse, all four user classes flow into the freeway. The inflow composition there is split according to the reroutable share $\bar{\eta}$ and the truck percentage η_{truck} .

On the main route, no reroutable traffic enters the network so that the inflow composition of these classes is set to 0. The remaining two classes share the inflow according to the truck percentage η_{truck} .

On the alternative route, only background traffic is entering the network. Therefore, only non-reroutable classes enter the network so that the inflow composition on the alternative route is defined the same as the one on the main route.



(a) Total inflow (in $\frac{\text{veh}}{\text{h}}$) in the A15 from the interchange with the A4



(b) Total turnfraction from to A15 to the interchange with the A4

Figure 7.7: Example of calibrated total inflow and total turnfraction for the case study; thin grey lines: observed quantities gathered each week; thick black lines: median thereof

The class-specific turnfractions outside of the reroutable area are the same as the total turnfraction. If no class-specific data are available, then the results from the loop data are used directly. If class-specific turnfraction data β_{truck} and β_{car} are available, then these are used.

On the main route, the destination classes are distinguished. Those classes who pass

Location	User class	Inflow composition	Turnfraction
outside of reroutable area	1	$\bar{\eta} \cdot (1 - \eta_{\text{truck}})$	β_{car}
	2	$\bar{\eta} \cdot \eta_{\text{truck}}$	β_{truck}
	3	$(1 - \bar{\eta}) \cdot (1 - \eta_{\text{truck}})$	β_{car}
	4	$(1 - \bar{\eta}) \cdot \eta_{\text{truck}}$	β_{truck}
on main route	1	0	0
	2	0	0
	3	$1 - \eta_{\text{truck}}$	$\beta_{\text{car}}^{\text{adjusted}}$
	4	η_{truck}	$\beta_{\text{truck}}^{\text{adjusted}}$
on alternative route	1	0	0
	2	0	0
	3	$1 - \eta_{\text{truck}}$	1
	4	η_{truck}	1

Table 7.4: Calibration of the user-class specific inflows and turnfractions the case study

Vaanplein and are therefore reroutable do not take any off-ramps; therefore their turnfractions are set to 0. The turnfractions of the non-reroutable classes are determined based on the total turnfraction and the base share $\bar{\eta}$. To keep the focus on the calibration in general, the detailed calculations of the adjusted turnfractions for cars $\beta_{\text{car}}^{\text{adjusted}}$ and trucks $\beta_{\text{truck}}^{\text{adjusted}}$ are presented in Appendix A.

On the alternative route, the reroutable classes stay in the network; their turnfractions are therefore set to 0. The other classes model the background traffic of the alternative route. Since the alternative route is modeled in such a way that its on- and off-ramps are summarized by one on-ramp in the beginning and one off-ramp at the end, all background traffic leaves the alternative route at the bifurcation node; therefore, the turnfractions of the reroutable traffic are set to 1.

Summary

In summary, the inflows and turnfractions are based on historic data gathered from induction loops. Data of one year provide the historic patterns, categorized into the seven days of the week. Since four user classes are modeled, the total inflows and turnfractions are split to generate the user-class specific patterns. These patterns depend on the vehicle class, the availability of vehicle-class specific data and on the location in the network. These patterns are then directly applied in the Prediction component to simulate the class-specific inflows at origin-links and the class-specific turnfractions at bifurcation nodes.

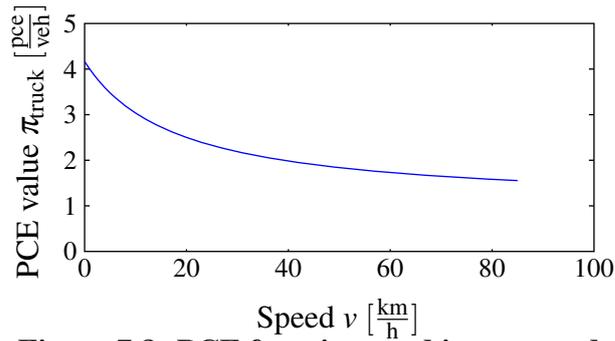


Figure 7.8: PCE function used in case study

7.3.4 Calibration of the PCE Function

The videos gathered by the helicopter camera provide data about the relation between the two vehicle classes of cars and trucks. Figure 2.5 shows a frame of a video of the A15. Clearly visible are the different spacings between cars and trucks. Such videos are used to calibrate the parameter values of the pce function π_{truck} (5.12).

At standstill, the vehicle classes showed average spacings of $r_{\text{car}}^{\text{min}} = 7 \text{ m}$ and $r_{\text{truck}}^{\text{min}} = 25 \text{ m}$. The relative spacings at capacity showed an average pce value of $\pi_{\text{truck}} = 1.5 \frac{\text{pce}}{\text{veh}}$. Minimum headways of $h_{\text{car}}^{\text{min}} = 1 \text{ s}$ and $h_{\text{truck}}^{\text{min}} = 1 \text{ s}$ lead to a realistic pce function. Figure 7.8 shows the pce function calibrated with these parameter values, which is used in the case study.

7.3.5 Calibration of the Fundamental Diagram at Non-bottleneck Locations

The fundamental diagram v^{FD} Fastlane was defined in (5.8); an example of a fundamental diagram was shown in Figure 5.2. Since the capacity at non-bottlenecks cannot be observed directly, we chose a heuristic approach that defines the effective capacity and the effective jam density based on the number of lanes n_{lane} . We calibrated the effective capacity as

$$C = n_{\text{lane}} \cdot 2250 \frac{\text{pce}}{\text{h}}, \quad (7.6)$$

and the jam density as

$$K_{\text{jam}} = n_{\text{lane}} \cdot \frac{1}{r_{\text{car}}^{\text{min}}}. \quad (7.7)$$

Areas with a merge lane such as off-ramps are not accounted for in the fundamental diagram, since vehicles tend to merge as soon as merging is allowed. Furthermore, weaving lanes have a lower capacity than normal lanes; we model them as half a normal lane. For example, the section of the A15 between the A4 and off-ramp Heijplaat

consists of three main lanes and two weaving lanes, which is therefore modeled as four lanes.

Finally, the free-flow speeds of the classes are modeled as $v_{\text{car}}^{\text{free}} = 110 \frac{\text{km}}{\text{h}}$ and $v_{\text{truck}}^{\text{free}} = 85 \frac{\text{km}}{\text{h}}$.

7.3.6 Calibration of the Fundamental Diagram at Active Bottle-necks

The remaining parameters to calibrate are the capacities of the regular bottlenecks. Since these have a large effect on the traffic conditions, a different calibration approach than for the other parameters is chosen. The capacities are calibrated by automatic iterative optimization with the goal that the predictions of the Prediction component closely match the true traffic conditions.

The optimization objective is that the predicted congestion γ is at the same location and at the same time as the true congestion $\tilde{\gamma}$:

$$J_{\text{reg}} = \sum_i \sum_j |\gamma^{ij} - \tilde{\gamma}^{ij}| \Delta x^i \Delta t \quad (7.8)$$

whereby the γ^{ij} indicates the traffic regime of cell i at time step j :

$$\gamma^{ij} = \begin{cases} 1 & \text{if } v^{ij} < v_{\text{crit}}^{ij} \\ 0 & \text{else} \end{cases} \quad (7.9)$$

In other words, the difference between predicted and true traffic regime in space and time is minimized. We chose the traffic regime, since the correct location of the traffic jams is the most relevant criterion for the control with the objective to minimize the total cost presented later in Section 7.5. Moreover, the traffic regime is not sensitive to short-term traffic phenomena. Optimizing the speed, for example, is very sensitive to stop-and-go waves, which often occur during congestion. However, the exact location of stop-and-go waves does not much affect the total cost; instead, the location and duration of the congestion is essential.

Calibration Procedure

A training data set of 25 days during the evening peak was selected, with starting times at 15:30, 16:00, 16:30 and 17:00, leading to 100 calibration data sets. For each data set, the capacities of both regular bottlenecks were optimized by applying the built-in optimization of Matlab. In all simulation in this chapter, the Fastlane model is discretized (5.15) with an approximate cell length of $\Delta x \approx 100\text{m}$ and a time step length of $\Delta t = 2.7\text{s}$.

For each of the data sets, the calibration of the capacity can either succeed or fail. A calibration is successful if the predicted traffic conditions closely match the true traffic

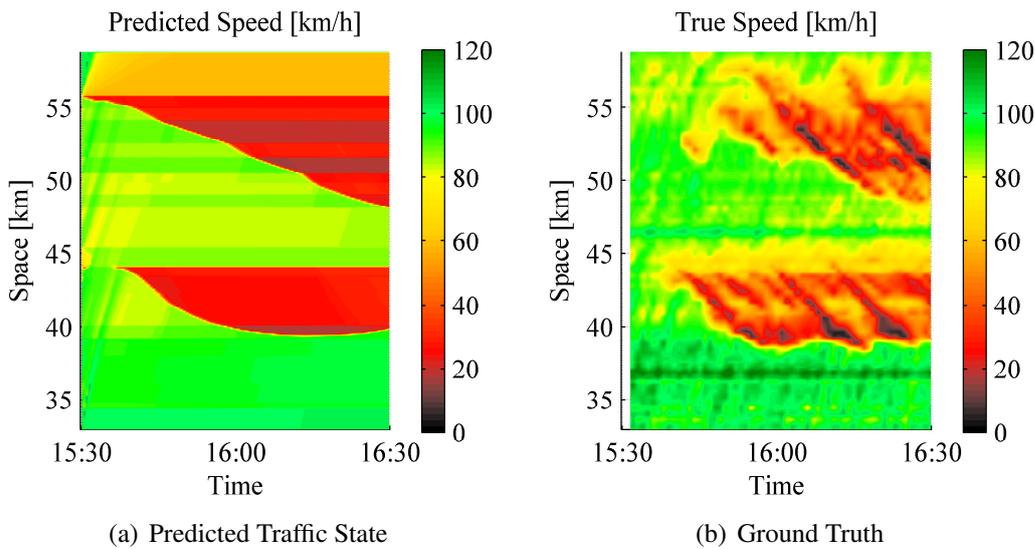


Figure 7.9: Example of a successful capacity calibration of a data set (24-01-2011 at 15:30)

conditions. Figure 7.9 shows an example of a successfully calibrated data set. Figure 7.9(a) shows the speed prediction of the model over space and time; congestion is predicted at both regular bottlenecks at Km 56 and Km 44, respectively. For comparison, Figure 7.9(b) shows the true traffic conditions. The capacity values of the successful data sets are then used for further calibration.

However, the calibration of some data sets failed. Figure 7.10 shows an example of a failed case, where the jam predicted at Km 56 is much stronger than the true congestion. The capacity value calculated is much lower than a realistic value. The optimization algorithm thus did not find the optimal capacity value. Reasons for these calibration failures are due to the local minimum being small so that the optimization algorithm did not find it. Furthermore, the objective function is not continuous, but step-wise constant; i.e. a small change in capacity may not change the spatiotemporal traffic regime, thereby falsely indicating that the local minimum is found. In either case, a manual classification in successful and failed calibrations for each data set is therefore necessary. The capacity values of the failed data sets are then discarded in the remaining calibration procedure.

Finally, the capacity values of the successful data sets were averaged and used as the calibrated capacities in the Prediction component. These values are $4200 \frac{\text{pcc}}{\text{h}}$ for the bottleneck at Km 44 (Spijkensisse) and $4800 \frac{\text{pcc}}{\text{h}}$ for the bottleneck at Km 56 (Charlois).

This concludes the calibration of the Prediction component. The following section validates the Prediction component with a different data set, to test whether it predicts the true traffic conditions well.

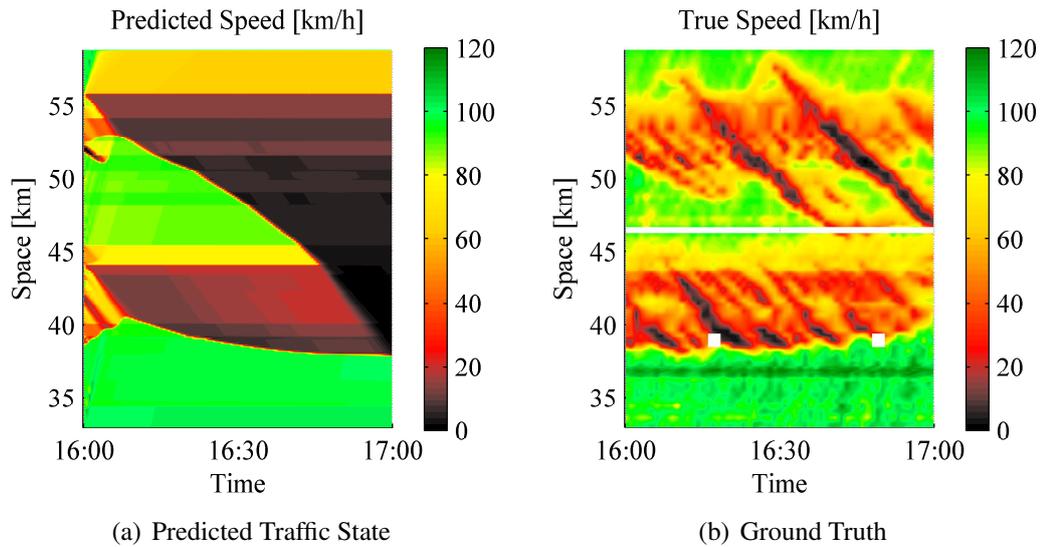


Figure 7.10: Example of a failed capacity calibration of a data set (24-01-2011 at 16:00)

7.4 Validation of the Prediction Component

The previous section determined the values of the parameters of the Prediction component. This section validates whether these values lead to realistic predictions of the traffic conditions of the A15. For this purpose, a validation data set of 20 days is used that is disjunct from the calibration data set used to determine the capacity values in Section 7.3.6.

In the following, the Prediction component is validated for regular congestion emerging on normal weekdays during the evening peak, and for incidental conditions where an accident occurred so that the freeway was partially blocked. Thereafter, the validation results are discussed. The performance measure is the difference of the traffic regime between predicted and true traffic J_{reg} (7.8), as it was used to calibrate the capacities as bottlenecks (Section 7.3.6).

7.4.1 Validation During Regular Congestion

First, the Prediction component is validated during regular traffic conditions during the evening peak, i.e. congestion emerged at the two bottlenecks. Ten days combined with four different starting times at 15:30, 16:00, 16:30 and 17:00 are selected for the test, resulting in 40 data sets.

Figure 7.11 shows the validation of one data set. The congestion of both regular bottlenecks are at the correct locations. Furthermore, the length of the congestions is close to the true congestion lengths.

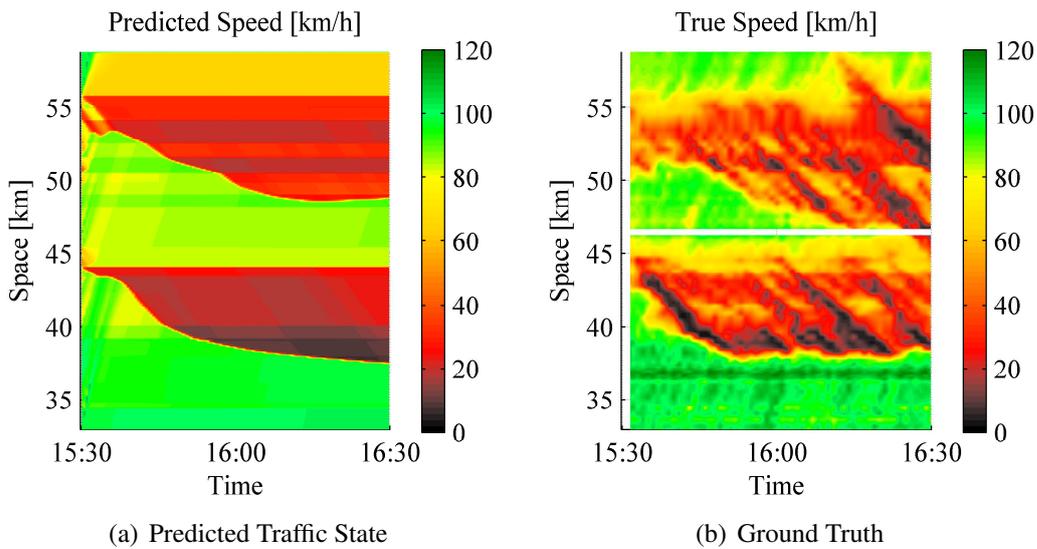


Figure 7.11: Validation of the Prediction component during regular conditions (30-03-2011 at 15:30)

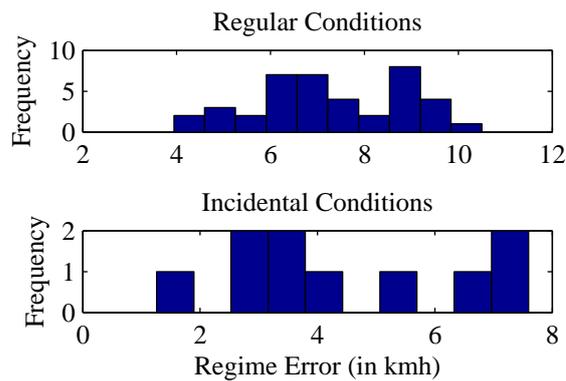


Figure 7.12: Performance J_{reg} of the validation

Out of the 40 validation data sets, 31 of them show a good match with reality, whereby six out of them closely match the true traffic conditions. In nine cases, the prediction results were far off the true traffic conditions. Figure 7.12 shows the histogram of the performance of the data sets. For comparison, the performance of the data set of Figure 7.11 is 6.1 kmh.

7.4.2 Validation During Incidental Conditions

Next, the Prediction component is validated for incidental conditions. Ten days where an incident occurred were selected as validation data sets. The location of the incident, the remaining capacity and the duration of it were fed into the Prediction component.

Figure 7.13 shows the results for of a data set (06-07-2011 at 16:15) where an incident occurred at Km 41 and lasted until 16:40. As can be seen, the congestion caused by the

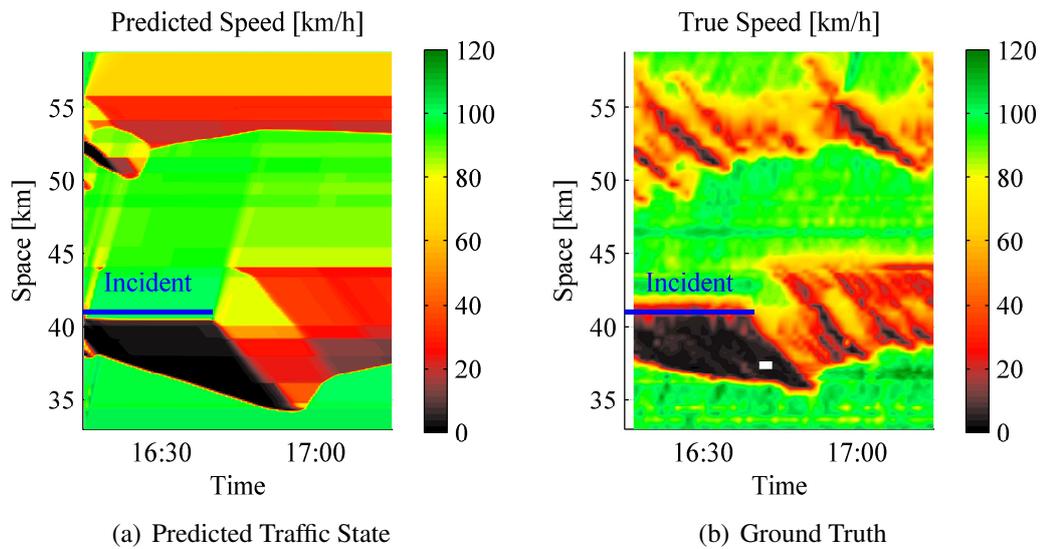


Figure 7.13: Validation of the Prediction component during incidental conditions (06-07-2011 at 16:15)

accident is predicted accurately. Furthermore, when the accidents clears, the resulting regular congestion emerging at Km 44 (Spijkenisse) is predicted well. (Note that the incident in the ground truth appears to be located further downstream. This is due to the finite acceleration, which need some hundreds of meters to accelerate to a high speed. This finite acceleration is not modeled in the Prediction component where the vehicles immediately reach a high speed.)

Six out of the ten data sets are predicted well, only one was far off, and three showed a mediocre prediction quality. The histogram in Figure 7.12 shows the performance of the validation date sets. For comparison, the performance of the data set of Figure 7.13 is 7.3 km/h.

7.4.3 Discussion of Validation

To conclude, the validation of the Prediction component shows sufficiently accurate results under many conditions. For the validation during incidents, the location, the remaining capacity and the duration of the accident was fed into the model. The location of the incident can directly be seen in the spatiotemporal plots produced by the Estimation component (Figure 7.3). The remaining capacity can be measured immediately downstream of the accident. Though the duration of the accident is unknown beforehand, we used the true duration to validate the model. The results show that the model predicts the congestion caused by incidents well. With the Prediction component, a traffic management center can thus predict what the traffic state will be if the accident is expected have a certain duration.

Since the main objective of BOS-HbR is to predict the congestion well, this validation

was evaluated by the difference of predicted and true traffic regime, i.e. the predicted congestion is at the right place at the right time. We did not analyze the difference of speeds, for instance, because it fluctuates in congested conditions due to stop-and-go waves and therefore provides only limited information about the location of congestion.

The capacity values calibrated in Section 7.3.6 are relatively low, especially the one at Km 56 (Charlois). Because this bottleneck is caused by a lane drop from four to three lanes, we therefore expected an effective capacity of around $6000 \frac{\text{pcc}}{\text{h}}$. The calibrated value of $4800 \frac{\text{pcc}}{\text{h}}$ is significantly lower than the value we expected. Nevertheless, the validation shows that the realistic traffic conditions are predicted so that using these capacity values in the prediction model is justified.

Therefore, under the objective of providing a realistic prediction, we regard the Prediction component to be validated. Now we are therefore ready to perform the actual case study, where multi-class control of the site is simulated.

7.5 Setup of the Control Component

This section describes the setup of the Control component of the control loop of Figure 2.11. The objective of the controller is to minimize the total cost (5.27) during the prediction horizon. Since there is no simple way of converting the traffic state in the form of the class-specific densities to the control objective in the form of the total cost, we only use the running costs that occur within the prediction horizon, and ignore any terminal costs at the end of it.

As explained in Section 7.1.4, three DTM measures are used to control the traffic of the A15. The signals of the two multi-class route guidance controllers represent the turnfraction. They can assume values between 0, indicating to stay on the route, and 1, indicating to use the off-ramp. The multi-class ramp meter at the A4 is controlled by the desired share of the ramp outflow and the set point density. The signal for the desired share can vary between 0, indicating full priority to trucks, and 1, indicating full priority to cars. The set point density for the freeway traffic state can vary between 0 and $200 \frac{\text{pcc}}{\text{km}}$, whereby the critical density of the link is $112 \frac{\text{pcc}}{\text{km}}$. The control interval is set to $\Delta t_c = 10 \text{ min}$.

The optimization is solved by the interior-point algorithm implemented in Matlab. This algorithm is a gradient-based method that always respects the boundaries of the control signal range, i.e. this method avoids signals that have no physical meaning and would lead to undefined prediction model behavior, such as negative turnfractions.

The setup for the values of the prediction horizon T_p and the control horizon T_c determine the quality of the controller both in terms of performance and computation time. In general, the more control signals have to be optimized, the finer-grained and the

more precise the traffic system can be controlled and therefore the better the performance will be. However, at the same time, the control problem becomes more difficult to solve since the objective function might then contain multiple local optima. Since we use a gradient-based method, the global optimum might not be found if many control signals are used. A further drawback of using a long prediction and a long control horizon is the computation time.

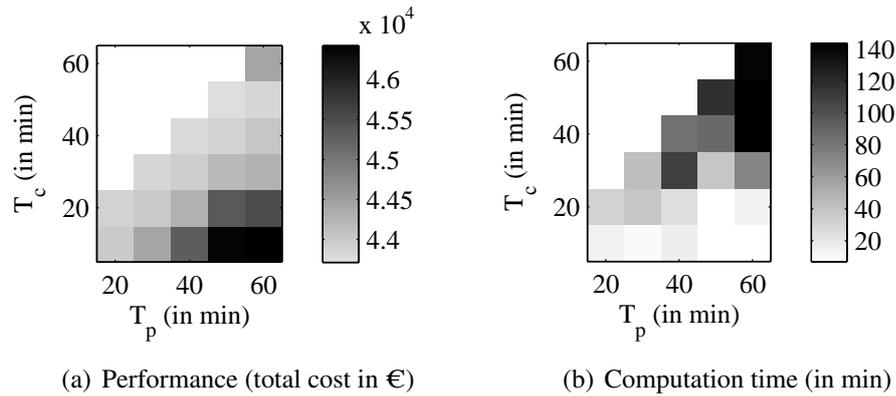


Figure 7.14: Performance and computation time of the controller dependent on the lengths of the prediction horizon and the control horizon

The effects of the length of the prediction horizon and the control horizon on the performance and the computation time is shown in Figure 7.14. Intuitively, increasing the control horizon T_c leads to a better performance (Figure 7.14(a)). Furthermore, increasing only the prediction horizon deteriorates the performance. This counter-intuitive result is due to the setup: the simulation time was fixed, and the controller was activated only if the remaining simulation time was longer or equal to the prediction horizon time. For example in the case, of $(T_p, T_c) = (60 \text{ min}, 10 \text{ min})$, the DTM measures where controlled only for the first 10 min; its performance is therefore relatively bad. A very good performance is achieved in the cases where the control horizon equals the prediction horizon. However, as can be seen in the case of $T_c = T_p = 60 \text{ min}$, the performance is worse than for shorter horizons. In this case, the controller did not find the global optimum. When taking the computation times into account (Figure 7.14(b)), shorter horizons are preferable. We therefore choose a trade-off between performance and computation time in the form of $T_p = T_c = 30 \text{ min}$.

The number of signals being optimized in the prediction horizon thus is the number of input signals times the number of control intervals. Since there are six input signals (two turnfractions for each of the two MCRG, and the share and the density for the MCRM) and three control intervals, 18 signals are optimized in the case of the multi-class controller.

Since the prediction model is validated for a period of one hour, the simulation time of the experiment is set to that value.

7.6 Setup of the Case Study

In the following sections, the control loop of Figure 2.11 is applied to simulate traffic control of the A15. This case study applies a multi-class controller in both regular and incidental conditions to minimize the total cost of the traffic of the A15. The multi-class controller is compared to a mixed-class controller and to the default no-control situation.

The experiment consists of multiple scenarios. Firstly, the multi-class controller is compared to the mixed-class controller and the no-control case. Secondly, different values of time are used for the truck class: $\zeta_{\text{truck}} \in \{\frac{1}{3}, 1, 3, 100\} \cdot \zeta_{\text{car}}$, with a value of time for cars of $\zeta_{\text{car}} = 15 \frac{\text{€}}{\text{vehh}}$. Thirdly, the controllers are applied to multiple data sets: ten days were selected for the experiments under incidental conditions, and two days with four different starting times were selected for the experiments under regular traffic conditions. The performance is compared in terms of total costs and the computation time is measured.

7.7 Results of the Case Study

This section presents the results for regular conditions in Section 7.7.1 and for incidental conditions in Section 7.7.2.

7.7.1 Results of Multi-class Control Under Regular Conditions

This section presents the results of the experiments performed under regular conditions. The results are illustrated in the figures by the data set of 30-03-2011 at 15:30, for a value of time of $\zeta_{\text{truck}} = 3\zeta_{\text{car}}$. In the following, the results for the no control case, for the mixed-class controller and for the multi-class controller are shown. For comparison, Table 7.5 summarizes the performance of all examples.

	Regular Conditions	Incident Conditions
Multi-class Control	43 800 €	39 800 €
Mixed-class Control	44 200 €	40 000 €
No Control	47 900 €	45 500 €

Table 7.5: Results of the case study, comparison of total cost of example cases

No Control Under Regular Conditions

Since the dataset is the same as in the example of the validation in Section 7.4.1, the predicted traffic conditions were already shown in Figure 7.11; the two bottlenecks at

Km 44 and Km 56 are active so that congestion emerges and grows at these locations. The performance of the no control case is 47 900 €.

Mixed-Class Control Under Regular Conditions

The results of the mixed-class controller are shown in Figure 7.15. The spatio-temporal speed plot shows that the congestion is significantly shorter.

The reason is that the route guidance measure at the first bottleneck guided a part of the reroutable traffic via the alternative route. As a result, fewer traffic has to pass the bottleneck at Spijkenisse so that the congestion is shorter there. Since the traffic stays on the alternative route (the route-guidance signal after the Botlekbridge is zero), less traffic is flowing into the congestion at Charlois. Of course, this congestion is only relocated to the alternative route. The ramp meter at the A4 is not activated.

Since the congestion at Spijkenisse is reduced, fewer vehicles are queued so that the performance of the mixed-class controller is better than that of the no control case. The performance of the mixed-class controller is 44 200 €, which is a relative improvement of 9.2 %.

Multi-class Control Under Regular Conditions

Figure 7.16 presents the results of the multi-class controller. The congestion at both bottlenecks are shorter than in the no control case, and even slightly shorter than in the mixed-class controller case.

Traffic is rerouted via the underlying network, again leading to less congestion on the main route compared to the no-control case. In contrast to the mixed-class controller, the multi-class controller splits the routing dependent on the vehicle class. Only cars are rerouted; furthermore, a small part of the rerouted cars is sent back to the main route after the Boktlekbrug. By these measures, the more valuable trucks stay on the main route, and the less valuable cars take the alternative route to reduce the demand at the bottleneck, which reduced the congestion at Km 44. Here, too, the ramp meter at the A4 is not activated.

Since the vehicles can be guided class-specifically, the valuable trucks stay on the main route, which improves the performance. The performance of the multi-class controller is 43 800 €, which is an improvement of 9.1 % compared to the no-control case, and 1.0 % compared to the mixed-class case.

7.7.2 Results of Multi-class Control Under Incidental Conditions

This section presents the results of the experiments performed under incidental conditions. During the morning of 18-04-2011, an incident occurred at Km 54, which

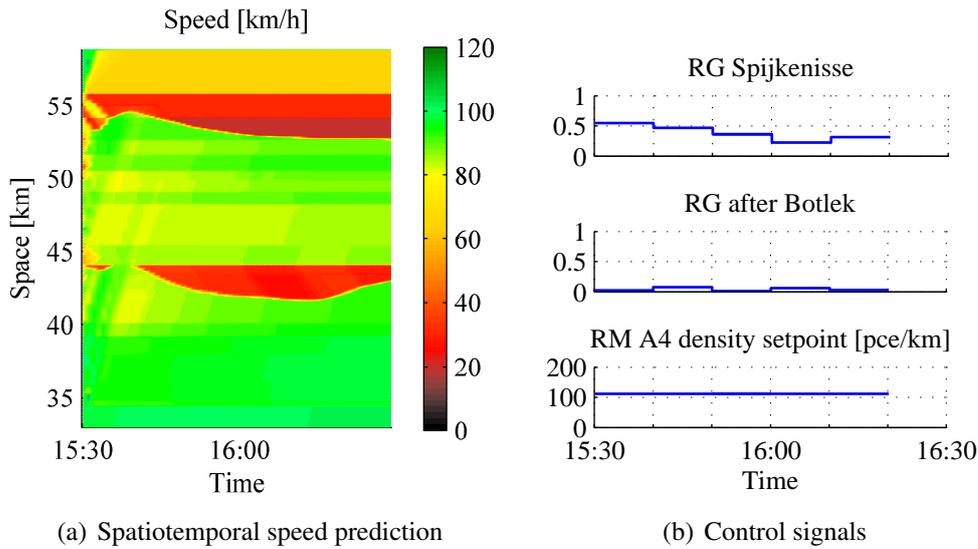


Figure 7.15: Results of mixed-class control during regular conditions

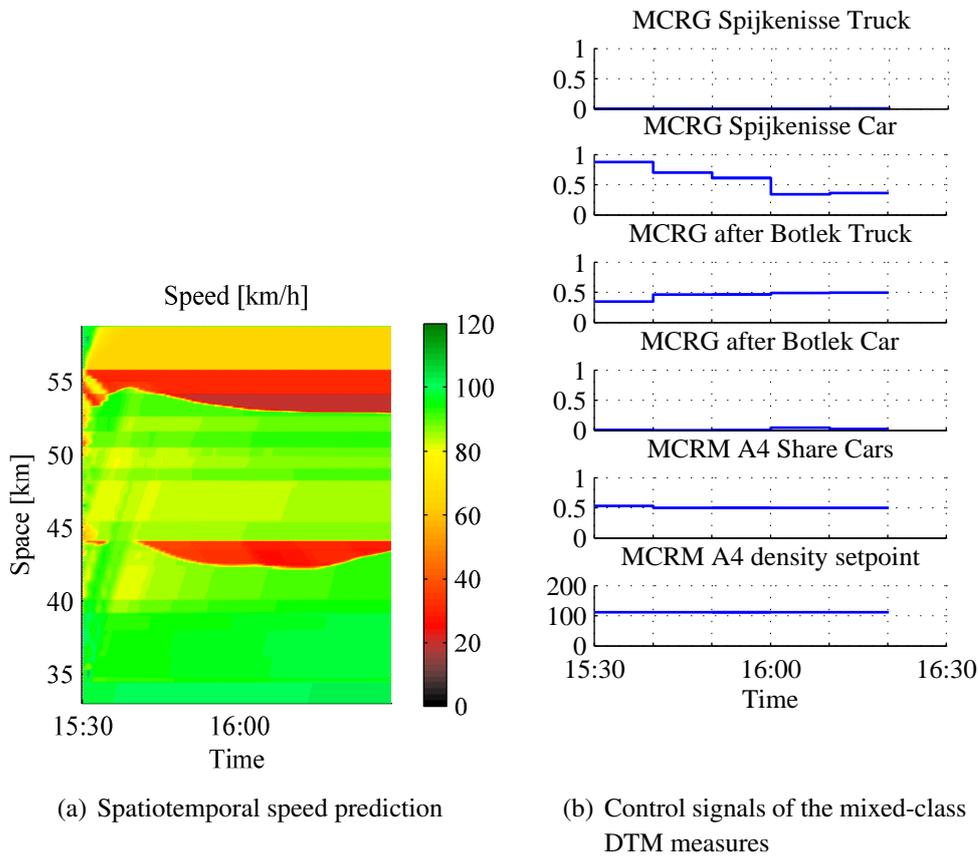


Figure 7.16: Results of multi-class control under regular conditions

caused congestion. In the following, the results for the no control case, for the mixed-class controller and for the multi-class controller are shown for a value of time of $\zeta_{\text{truck}} = 3\zeta_{\text{car}}$.

No Control Under Incidental Conditions

The predicted traffic conditions of the no-control case are shown in Figure 7.17. The congestion caused by the incident grows heavily during the duration of the incident. The spillback partially blocks upstream off-ramps, which worsens the congestion even more. When the incident clears at 09:45, the congestion dissolves slowly over more than 20 min. The incident thus affects the traffic on the A15 long after it has been cleared. The performance of the no control case is 45 500 €.

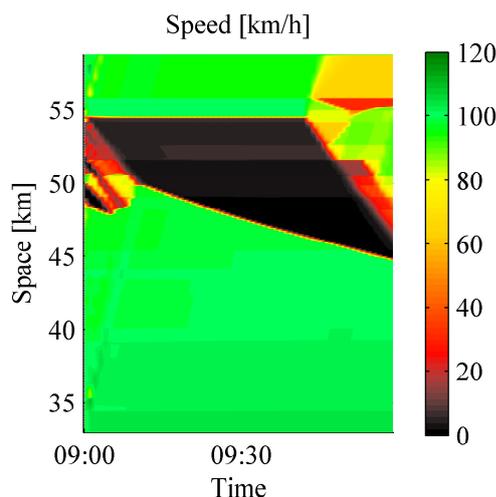


Figure 7.17: Results of no control under incidental conditions; spatiotemporal speed

Mixed-Class Control Under Incidental Conditions

Figure 7.18 shows the results for the mixed-class control case. The congestion caused by the incident is significantly shorter than in the no-control case. This reduces the spillback and the blocking of the off-ramps so that the congestion stays short. Consequently, the congestion dissolves within a few minutes after the incident is cleared.

All of the reroutable traffic is guided via the alternative route. Since the congestion spills back over the on-ramp of the A4, the ramp meter is activated and holds back traffic from the A4. After the incident is cleared and the congestion is dissolved, the ramp meter releases the congestion from the A4.

The performance of the mixed-class control case is 40 000 €, which is an improvement of 11.9% compared to the no-control case.

Multi-class Control Under Incidental Conditions

The results of the multi-class controller are shown in Figure 7.19. The traffic state on the A15 is very similar to that of the mixed-class controller. Here, too, traffic is

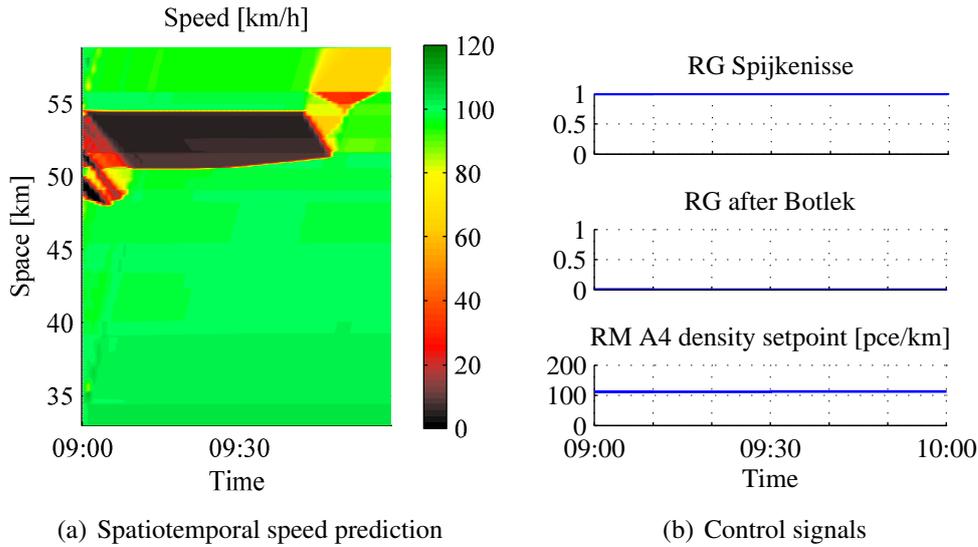


Figure 7.18: Results of mixed-class control at an incident

rerouted via the underlying road and the ramp meter is activated due to the congested A15.

However, the multi-class controller improves the traffic conditions when congestion on the A15 is dissolved and the congestion from the A4 is released. The share of the MCRM is set to 0, which means that trucks are prioritized. As a consequence, first the valuable trucks are released from the queue and thereafter the less valuable cars, as shown in Figure 7.19(c) by means of the number of queued vehicles.

The performance of the multi-class controller is 39 800 €, which is an improvement of 12.5 % with respect to the no-control case and 0.7 % with respect to the mixed-class controller.

7.8 Discussion of the Performance

This section first analyzes the performance of the controllers applied in the case study. Then, directions for improving the performance are outlined.

7.8.1 Analysis of the Performance

The performance of the controllers under regular and under incidental conditions is similar. To illustrate this, the performance of all scenarios of the incident experiment is shown in Figure 7.20. The absolute performance (Figure 7.20(a)) suggests that the performance of the controllers differs only little. A comparison of the performance differences with respect to the no-control case (Figure 7.20(b)) reveals, however, that predictive control substantially improves the traffic conditions on the site. Dependent

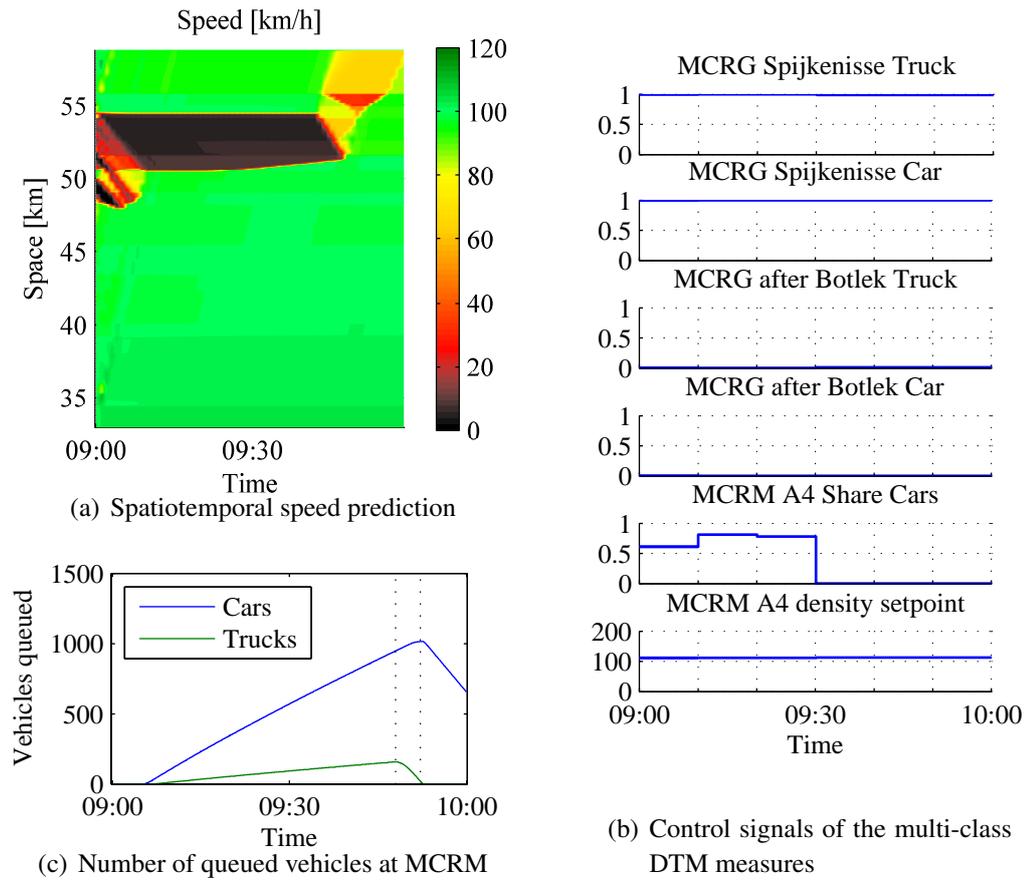


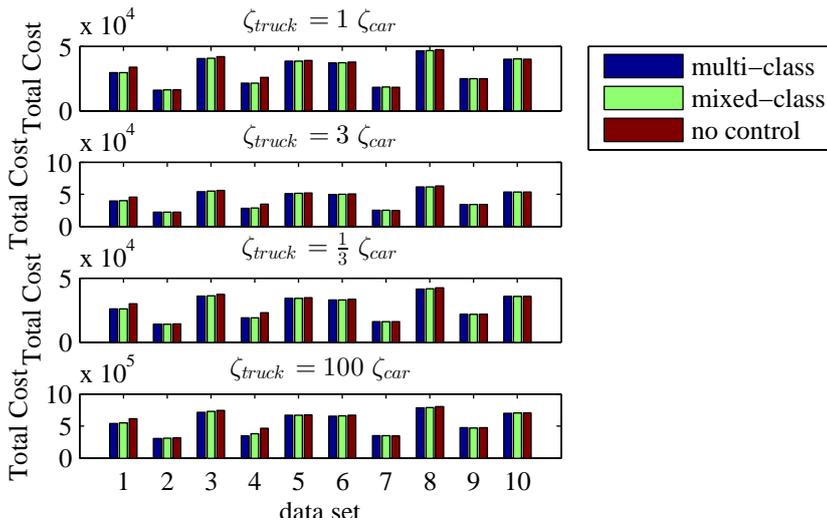
Figure 7.19: Results of multi-class control at an incident

on the location and strength of the incident, the network can perform substantially better if control is applied, as is the case for data set 1, which was presented in Section 7.7.2. The performance is even more improved if traffic is controlled user-class specifically.

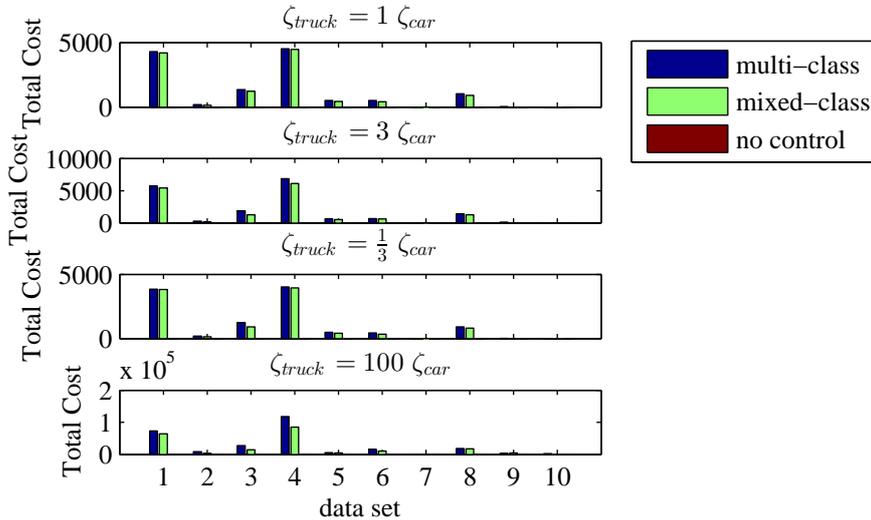
These improvements are due to two reasons. Firstly, the capacity of the alternative route is not fully used, so that rerouting can substantially reduce the demand at the bottleneck and therefore decrease the congestion. Secondly, the ramp meter at the A4 can decrease spillback on the A15 so that upstream off-ramps are not blocked, or at least that the blocking is delayed. Since the on-ramp of the A4 is multiple kilometers long, it offers a lot of buffer space so that (multi-class) ramp metering is possible there without congestion spilling back to upstream infrastructure.

If the DTM measures are expanded to multi-class measures, then the properties of the vehicle classes can be exploited accordingly. For example, if only a part of the traffic can be rerouted, then leaving the valuable vehicles on the fastest route and rerouting the less valuable ones via the longer route improves the traffic performance. Another example is multi-class ramp metering, where first the queue of the valuable vehicle class is dissolved and then the queue of the less valuable one.

These different performances show that predictive control can substantially improve



(a) Absolute performance



(b) Performance improvement compared to no-control

Figure 7.20: Comparison of performance during incidental conditions for different data sets and different values of time (in €)

the traffic on the A15. Dependent on the strength, time and location of the incident, these improvements can vary largely. In the experiments, we found values between 0 and 25%. The application of multi-class control improves the traffic state even more, whereby these improvements are in the order of 1%.

This improvement is relatively low compared to the values of up to 33% of the idealized experiments in Chapter 6. Several reasons may explain this small difference. Firstly, the location of the bottlenecks, especially the one at Km 56 (Charlois) is located far downstream, where no alternative enables a rerouting around the bottleneck location.

Secondly, the truck percentage in the experiments of Chapter 6 is significantly higher

than in the case of the A15. Naturally, the higher heterogeneity of those experiments lead to a larger improvement of the traffic performance than this case study.

Finally, the network of the previous experiments was smaller than the one of the case study. The relative improvements of the small networks are therefore stronger pronounced. A remedy of this last point is to increase the number of DTM measures in the network so that more control of the network traffic is possible. For example, all on-ramps of the A15 could be equipped with multi-class ramp meters.

7.8.2 Potential Approaches for Improving the Performance

In the following, several ways to improve the performance quality of the control loop are outlined. The ideas outlined here suggest to use existing scientific methods that might improve the quality of the controller. Later, in the conclusions chapter of this thesis, Section 8.2 outlines general directions for further research.

Improving the State Estimation

Improve sensors The current sensor configuration outlined in Section 7.1.3 densely covers the freeway of the A15 but provides only mixed-class data. Conversely, the sensors on the alternative route gather individual data, but only at a few locations. A sensor configuration that both observes vehicle-individual data and densely covers the network will therefore improve the quality of the estimation. By those means, the multi-class traffic state of both the freeway and the alternative route can be estimated with a high precision. Such multi-class data could furthermore be stored in a data base, which would provide historical multi-class data. These would be valuable for the ex-post evaluation of the traffic performance of the network and could further be used for better calibration of the prediction component.

Use information about sensor noise The Estimation component can further be expanded to use more available information, such as the noise distribution of the sensors. If the sensor data and their noise distributions are combined with a prediction by a traffic flow model and its noise distributions, then the estimation quality could be improved. A recursive Bayesian estimator as outlined in Section 2.3 provides a framework for this. A current challenge is, however, to calibrate such an estimator well.

Improving the Prediction Model

Adjust calibrated inflows and turnfractions online Currently, the inflows and turnfractions applied in the prediction model are the medians of historic data patterns, i.e.

they only depend on the day of the week and the time, but not on the traffic conditions. To improve the quality, these could be adjusted based on the current traffic state (see e.g. Smith et al. (2002)).

Calibrate fundamental diagram online Currently, the parameter values of the Prediction component stay fixed once they have been calibrated offline. Alternatively, the parameter values could be estimated online. An example of a joint estimation of the parameters of the fundamental diagram and the traffic state with Fastlane is shown by Van Lint et al. (2008a).

Reduce discretization cell lengths of Fastlane The prediction quality of Fastlane can be improved if the cell size Δx and the time step length Δt in (5.15) are reduced. By this means, the numerical diffusion is reduced, which means that the result of the discretized model is closer to the continuous model (Leclercq et al., 2007). A drawback of this approach is that the computation time increases quadratically in the inverse of the cell length.

Replace Euler coordinate system with Laplace coordinate system Fastlane is currently used in Eulerian coordinates, i.e. the cells are fixed with respect to the road (Sections 5.1 and 5.2). Another way to reduce the numerical diffusion is to use a Lagrangian formulation of Fastlane, i.e. where the cells (or “platoons”) comprise a fixed number of vehicles and travel downstream with the traffic. Van Wageningen-Kessels et al. (2010) and Van Wageningen-Kessels (2013) reformulate Fastlane to Lagrangian coordinates; Yuan et al. (in press) and Yuan (2013) apply the Lagrangian Fastlane model to recursive traffic state estimation.

Improving Traffic Control Component

Reduce control interval for finer-grained control Currently, the controller is set up in such a way that the DTM measures are controlled every $\Delta t_c = 10$ min (Section 7.5). A finer-grained control by reducing the control interval would yield a better control performance, although the improvement is likely to be only little. Similarly, the prediction horizon and the control horizon can be increased to cover the whole one hour the prediction model is validated for. All of these measures have the potential to improve the performance of the controller, however, these also would increase the computation time.

Use multiple starting points for optimization Since the optimization algorithm is not guaranteed to find the global optimum, running multiple optimizations with different starting points increases the chances of finding a better local optimum and therefore increasing the performance.

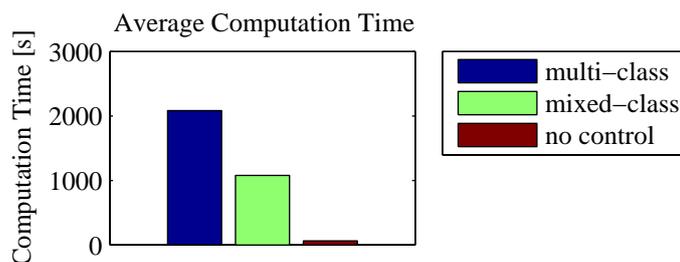


Figure 7.21: Average computation time of the Control component

Summary

This section outlined approaches to improve the performance quality of BOS-HbR. Some of the approaches named require an extensive calibration of the parameters, others are the subject of current research. Most of the approaches, however, pose a dilemma between a high performance quality and a low computation time.

7.9 Discussion of the Computation Time

This section first analyzes the computation time of the case study. Then, it outlines directions to improve the computation time.

7.9.1 Analysis of the Computation Time

Figure 7.21 shows the average computation time of the different controllers used in the case study, as they are implemented in Matlab and executed on a laptop with 3.5 GB memory and a 2.5 GHz dual-core processor. The multi-class controller computes the signals within one hour. The mixed-class controller is significantly faster, since fewer control signals are optimized. In the case of no control, only the Prediction component and the file handling and visualization are executed. These procedures take one minute.

The high computation times of the predictive controllers are due to the setup of the experiment. The experiment was very detailed, including a lot of signals to optimize and an optimization stopping criterion that requires small changes of the results of the iterations (Section 6.4.2.2). The result is that the computation time of the optimization is relatively high, so that this setup is not suitable for online applications in practice. The following section outlines steps to improve the computation time. Section 8.5 presents a setup of BOS-HbR where a cycle of the inner control loop performs within a few minutes and runs online.

7.9.2 Potential Approaches for Improving the Computation Time

This section discusses potential ways to improve the computation time of BOS-HbR.

Improving the State Estimation

Deactivate calibration of Adaptive Smoothing Method by Wave Speed Estimator

The Estimation component now consists of the Adaptive Smoothing Method (Section 4.2) and the Wave Speed Estimator (Chapter 3), which calibrates the ASM parameters online. Instead, typical values for the characteristic wave speed, for example estimated by historic averages, can be used as parameter values so that the WSE can be deactivated. This would reduce the computation time of the Estimation component from one minute to a few seconds.

Reduce time horizon of Adaptive Smoothing Method Furthermore, the time horizon of the ASM is now set to 4h, mainly for visualization purposes. However, only data of a few minutes have a significant effect on the current traffic state. A reduction of the horizon to, say, 5 min would therefore speed up the Estimation component even more. Though, since the ASM already runs within a few seconds, this measure would only have a marginal effect.

Improving the State Prediction

Increase discretization cell lengths of Fastlane The discretization parameters of Fastlane (notably the cell length as described earlier) and the number of user classes affect the computational speed. Reducing the number of user classes linearly decreases the computation speed. The network model is currently discretized with cells of circa $\Delta x = 100\text{m}$ and a time step of circa $\Delta t = 3\text{s}$. Increasing the length of a cell would automatically increase the time step length (Courant et al., 1928)). This increase would thus decrease the computation time quadratically. The maximum length of a cell is limited by the length of the shortest link of the network, however.

Replace Euler coordinate system with Laplace coordinate system A Lagrangian formulation of the Fastlane model reduces and simplifies the scheme. Currently, in Eulerian coordinates, the traffic state of a cell in the next cell depends on the traffic state of three cells, namely itself and its two neighbors, since traffic characteristics can propagate both upstream and downstream. In Lagrangian formulation, the traffic state of a cell (or “platoon”) in the next time steps depends on only two cells, namely itself and its downstream neighbor, since the characteristics propagate only upstream. Van Wageningen-Kessels et al. (2010) developed the Lagrangian link model and Yuan et al. (in press) developed a multi-class Lagrangian node model for Fastlane.

Replace traffic flow model with a simpler one A simpler traffic flow model could be applied altogether (see Section 2.2). However, this approach, too, poses a trade-off between computational speed and quality of the controller.

Improving the Traffic Control Component

Limit the computation time of the optimization procedure In the case study, the optimization procedure is set up to find the optimal control signals with a high precision (Section 6.4.2.2). For online applications, a low computation time is crucial, so that a more relaxed stopping criterion can be chosen. In addition, the number of iterations of the optimization procedure can be capped, effectively limiting the computation time of the control component. Furthermore, the initial value of the optimization procedure can be set to the optimal signal found in the previous iteration of the control loop.

Use a faster optimization algorithm Another approach to improve the computation time is to use a different optimization algorithm altogether. The active-set algorithm is usually faster than the interior-point algorithm used in the case study, but sometimes evaluates points outside of the specified bounds (such as negative turnfractions) which causes the prediction model to crash. Another optimization algorithm is the feasible direction method; Kotsialos et al. (2002) apply this method to optimize traffic network-wide within a few seconds.

Reformulate the control problem A reformulation of the control problem can lead to a faster optimization. Reformulating it to the optimization by the co-state problem (Wang et al., 2012), or optimizing the marginal costs (Zuurbier, 2010) can lead to a faster optimization. Approximating the optimization problem by a piece-wise linear model (e.g. Lin et al. (2009)) usually leads to a much faster optimization, though the performance can decrease since the prediction model might no longer incorporate non-linear traffic phenomena any more.

Increase control interval for coarser-grained control The computation time depends significantly on the number of signals that are optimized. To lower the number of control signals, the control interval length of the model-predictive controller can be increased to and the control horizon can be decreased (Section 7.6).

Decrease the number of DTM measures A further way to lower the number of signals is to remove some of the DTM measures. For instance, currently there is no ramp meter installed at the interchange with the A4; removing this ramp meter from the experiments reduces the number of DTM measures to the two route guidance measures that guide the traffic via the alternative route.

Combine model-predictive control and rule-based control We chose model-predictive control to find the optimal signals that minimize the total cost. However, MPC

can be combined with other control approaches, such as the one discussed in Section 2.1. For example, the high performance of MPC can be combined with the computation speed of a rule-based system. The MPC is used to develop and validate the rules offline. These rules are then applied in the rule-based controller in operation. The drawback of such an approach is, however, that it is not applicable to incidental conditions if these are not represented in the rule base. Nevertheless, applying a validated rule-based controller during regular conditions would free up computation time. The model-predictive controller would then only be used online under incidental conditions, and offline for the development of the rule base.

Improving the Program Implementation

Since the purpose of the BOS-HbR prototype is to show that the concept of multi-class DTM improves the traffic conditions, little attention has been paid to optimizing the computation time of the non-traffic related parts of BOS-HbR. There are thus many parts of the prototype which can be implemented more efficiently.

Parallelize control optimization procedure The control loop can be parallelized in several ways so that multiple processors or even multiple machines solve the control loop. One way is to parallelize the optimization procedure, as the gradients can be computed simultaneously. In an extreme case, only one optimization iteration would be executed so that the whole objective function is evaluated at specified sampling points at once, whereby one machine evaluates one sampling point. Of course, the larger the sample set is, the more machines are required.

Parallelize components of the control loop A different method of parallelization is to run the different processes of BOS-HbR independently of each other. For example, the data gathering, the Estimation component and the Control component can each run on a dedicated machine. Then, the Control component would use the most recent traffic state provided by the Estimation Component, but would further not interfere with it.

Improve communication with data base, file handling and input/output procedures The gathering of the data from the data bases and sensors can be improved, or the saving of the results to files and visualizations can be optimized or even partially omitted. A large part of the computation time of the no-control case shown in Figure 7.21 is due to these input and output operations.

Use a programming language that is more computation time efficient The current implementation environment is Matlab. Its strength is the easy manipulation and

visualization of matrix data, which makes it very suitable for scientists and engineers to create prototypes. To produce a fast version of BOS-HbR, the implementation environment can be changed to a more runtime-efficient language.

Professional implementation with focus on online application Further large improvements of computation time are expected if BOS-HbR is implemented by a professional software engineering approach with the goal of developing an online application that is deployed in a traffic management center. Such a professional implementation would also include a more stable execution and better error handling. Furthermore, a graphical user interface (GUI) would enable easy interaction with the traffic operators. Such an interface would also include an intuitive way to modify the configuration, such as implementing new (multi-class) DTM measures at different locations.

Summary

This section presented approaches to improve the computation time of BOS-HbR. Some of them affect only the values parameters, others change the approach or model that was used in the components of the control loop, whereas some suggest a better implementation of the BOS-HbR.

While all of the adjustments have the potential to increase the computational speed, they also have potential drawbacks. If they do affect the model or approach, usually the validity and the performance of the controller will decrease. A new implementation of the prototype by a professional software development approach, however, will increase the computation speed, and preserve or even increase the performance quality.

General directions for further research are outlined in the conclusions chapter of this thesis in Section 8.2.

7.10 Conclusion

This chapter combined the components developed in the previous chapters of this thesis to the system BOS-HbR and applied it to the Dutch freeway A15 and a part of the underlying road network. The data from the road-side sensors are gathered and fused by the Estimation component to estimate the traffic state of the A15 in real-time. Subsequently, this current traffic state is used to initialize the Prediction component, which predicts the traffic state of the following hour in real-time. These predictions are the basis for the Control component, where the control signals of multi-class DTM measures like route guidance and ramp metering are optimized to reduce the total cost of traffic.

The experiments show that a predictive control approach with two route guidance and one ramp meter measure outperforms the no-control case. The total cost of traffic is

reduced up to the order of 10%. Furthermore, if cars and trucks are controlled class-specifically, the performance is improved even further within the order of 1%.

Currently, BOS-HbR has been developed as a prototype to estimate and predict the traffic state of the A15 in real-time. We outlined ways to improve its performance quality and its computation time. Whereas most ways pose a dilemma between those two goals, the estimation of the traffic state with a recursive Bayesian model, an implementation of Fastlane in Lagrangian coordinates, the replacement of the optimization procedure with a genetic algorithm, and a professional implementation of BOS-HbR have the potential to improve both goals. The challenge of these approaches is, however, that they require a long development or a long calibration due to their complexity.

This chapter showed that multi-class control, including all three components of the control loop, has the ability to improve the performance of traffic in a real network. The final chapter summarizes the ideas developed in this thesis and gives recommendations for practice and science. Furthermore, it shows an example of BOS-HbR applied online.

Chapter 8

Conclusions and Recommendations

In this thesis, we developed multi-class Dynamic Traffic Management (DTM) for freeway traffic. In the first part, we reformulated existing traffic state estimators so that they run efficiently in real-time, and we developed a method to calibrate them automatically. In the second part, multi-class control concepts were developed to improve the traffic performance of freeways. Based on the multi-class traffic flow model Fast-lane, the three class-specific properties of vehicle length, free-flow speed and value of time were analyzed to quantify their impact on the traffic flow and the network performance. Then, existing DTM measures were generalized so that they can influence each vehicle class individually. Finally, the estimation and control concepts of multi-class DTM were combined in a case study based on simulations of the Dutch A15.

In this final chapter, we first summarize the main findings and conclusions of each of the previous chapters in Section 8.1. Then, directions for future research are outlined in Section 8.2. Recommendation for practical use are presented in Section 8.3. In Section 8.4, we propose a workflow that moves from the current state-of-the-practice of case-based traffic control to multi-class proactive traffic control. In Section 8.5, this thesis finishes with an example of multi-class DTM in the form of the system BOS-HbR as it is currently running online.

8.1 Main Findings and Conclusions

This section summarizes the main findings and conclusions of each chapter.

In Chapter 2, we studied the state of the art of estimating, predicting and controlling the traffic flow of freeways. Most of the existing methods pose a dilemma between accuracy and computational speed. We concluded that an model-predictive control approach is the best one suited for analyzing the effects of controlling vehicle classes individually. To predict the effects of multi-class traffic and multi-class control accurately and quickly, we decided to use a macroscopic multi-class traffic flow model. We use the model Fastlane, which furthermore captures the dynamics of the spacings between different classes. For the traffic state estimation, we decided on two candidate methods. Firstly, the Adaptive Smoothing Method is a data-driven method that interpolates the traffic data in space and time according to the propagation speed of the traffic characteristics. Secondly, the Extended Kalman Filter is a model-driven approach that optimally combines the sensor data with the prediction of a traffic flow model. The final decision of a suitable estimation method was made in favor of the ASM later in Chapter 4. The estimation, prediction and control methods were then combined to form the control loop for multi-class DTM.

In Chapter 3, we analyzed spatiotemporal traffic data to extract the propagation speeds of shock waves. We found that treating spatiotemporal traffic data as a two-dimensional image and analyzing it by image processing tools enables the estimation of shock waves and their propagation speeds with high accuracy. This method can be used to automatically estimate shock waves and their speeds from both empirical and synthetic data, and can be used to support the calibration of traffic state estimators that are parameterized by characteristic shock wave speeds, the partial calibration of the fundamental diagram, and the evaluation of traffic flow models that reproduce shock waves. Furthermore, the experiment showed that the characteristic shock wave speed of congested traffic of the Dutch A13 ranges between $-16 \frac{\text{km}}{\text{h}}$ and $-23 \frac{\text{km}}{\text{h}}$. This confirms the findings of previous empirical studies that this characteristic speed is within that range.

The goal of Chapter 4 was to estimate the traffic state within a few seconds given raw sensor data. Reformulating two well-known traffic state estimation techniques resulted in significant efficiency gains without a proportional loss of estimation accuracy. The main conclusions are, firstly, that the Adaptive Smoothing Method can be reformulated to employ the Fast Fourier Transform. This speeds up its computation by nearly two orders of magnitude so that it now estimates the traffic state within a few seconds. Secondly, localizing the Extended Kalman Filter so that sensor data are used to correct the traffic state in their physical vicinity reduces the computation time complexity. The Extended Kalman Filter method now estimates the traffic state within a few seconds, too. We then qualitatively compared the ASM and the EKF in order to decide on a traffic state estimator for the case study. We decided to use the ASM, because currently it is easier to calibrate and more robust when traffic is switching between free-flow and

congestion.

In Chapter 5, the traffic flow model Fastlane was outlined first in continuous and then in discretized form. Then, the network performance indicator of total cost was analyzed, which is a generalization of the total time spent, whereby the travel time of each vehicle is weighted by its value of time. We found that the total cost is significantly influenced by the traffic throughput at bottlenecks, the spillback of congestion, and the travel time in free flow. To reduce the total cost, short and valuable vehicles must be prioritized at bottlenecks to increase the throughput, long vehicles must be rerouted around congestion to reduce spillback, and fast vehicles must be routed via non-congested routes to use their high speed in free flow. The prioritization of vehicle classes therefore cannot be determined in general, but instead poses a dilemma. An example of such a dilemma arises when prioritizing cars at a bottleneck, which leads to a high throughput at bottlenecks, but also to large spillback of congestion due to queued trucks.

In Chapter 6, multi-class DTM measures were developed and their impact on the traffic flow and the network performance was analyzed. We found that conventional, mixed-class ramp meters and route guidance measures can be generalized to control each vehicle class individually. A multi-class ramp meter provides a dedicated queuing lane for each vehicle class and grants access to the freeway vehicle-class specifically. It is a measure to increase the vehicular throughput and reduce the number of queued vehicles by prioritizing cars, or to decrease the total cost and decrease the spillback on the on-ramp by prioritizing trucks. Experiments showed reductions of total cost of 33 % for the multi-class controller with respect to the mixed-class controller. A multi-class route guidance measure controls the turnfraction of each vehicle class individually. It is used, for example, to keep trucks out of congestion or to steer cars via a longer but non-congested route. Experiments show improvements in the order of 10%. We furthermore found that the prioritization of the vehicle classes depends on the traffic state. In addition, we analyzed a dynamic truck lane. In the case of using realistic values of time and realistic demands, we found that a trucklane is beneficial in terms of total costs if it is used at at least three quarters of its capacity. We conclude that the generalization of mixed-class DTM measures to multiple vehicle classes significantly improves the traffic performance.

The case study in Chapter 7 combined the methods developed in the previous chapter and applied the control loop for multi-class DTM to the Dutch A15 near the harbor of Rotterdam. The traffic state of the Dutch A15 can be estimated within one minute, including the download of the sensor data and the calibration of the estimator, using the state estimation approach put forward in this thesis. We furthermore found that the traffic state can be predicted one hour ahead both under regular conditions and under incidental conditions. The prediction itself is computed within a few seconds. Furthermore, control advice for three multi-class DTM measures can be determined within one hour. In simulations, we found that multi-class control improves the traffic performance of total cost of the A15 compared to mixed-class control. We therefore

conclude that multi-class DTM is beneficial for optimizing the total cost of traffic.

The setup of the case study of Chapter 7 is not feasible for traffic control in real-time, however, since the computation time is too high. Nevertheless, Section 8.5 shows a real-time application, where the number of DTM measures is reduced and other parameter values are adjusted so that a cycle of the control loop takes approximately five minutes and is therefore feasible for real-time applications in a traffic management center.

8.2 Recommendations for Further Research

This section outlines directions for further research, split for estimation, prediction and control.

8.2.1 Traffic State Estimation

Estimation of traffic composition from mixed-class data Many freeways are set up with sensors that measure only mixed-class data, which excludes the traffic composition. Yuan et al. (2012) developed a method to extract multi-class data from aggregate data; however, this method requires calibration based on historical data. Another approach that would not require calibration is to use the shock wave speeds of traffic. As shown by (3.3), the truck percentage influences the shock wave speeds. The Wave Speed Estimator of Chapter 3 is able to identify the shock wave speeds from spatiotemporal traffic and can therefore be used to infer the truck percentage. The major question is with what precision the truck percentage can be estimated.

Generalization of the Adaptive Smoothing Method to multi-class data The Adaptive Smoothing Method currently takes the traffic data of speed, flow and density into account; those propagate mainly with one of two possible characteristic speeds. To increase the estimation quality of the ASM, it could be generalized to take multi-class data such as the truck percentage into account as well. In contrast to speed or flow, the truck percentage propagates always downstream, but mainly with the speed of traffic itself. They propagate therefore within a large range between zero and free-flow speed, which would require a large number smoothing kernels. A reformulation of the ASM to take multi-class traffic into account might therefore be necessary.

Calibration of recursive Bayesian estimators The calibration of recursive Bayesian estimators is essential for an accurate estimation of the traffic state. Besides the calibration of the underlying traffic flow model, the noise distributions of the system and the measurement model, or more precisely the covariance matrices thereof are essential. The actual noise distribution of practical systems are usually unknown, however, which makes an estimation of them difficult.

Improvement of the memory efficiency of recursive Bayesian estimators Recursive Bayesian estimators such as the Extended Kalman Filter require a quadratic space complexity in the network size, namely for the storage of the covariance matrices. Similarly to the computation time reduction of Section 4.3, a partial storage of the covariance matrices to avoid large regions of near-zero covariance values could be sufficient, since the errors of the estimated traffic state of network parts that are far away from each other are not correlated. This would drastically reduce the computation space complexity and would allow for the traffic state estimation of large networks.

Quantitative comparison of Adaptive Smoothing Method with Kalman filter techniques In Chapter 4, we compared the ASM and the Localized EKF qualitatively. Comparing both estimators quantitatively would answer under which circumstances which methods performs better and is therefore is more suitable for DTM applications. Similarly, other Kalman filter techniques should be included in this quantitative comparison.

8.2.2 Traffic State Prediction

Modeling of the capacity drop in multi-class traffic flow models The capacity drop reduces the throughput at active bottlenecks and therefore has an effect on the network performance. In order to reproduce traffic flow and the network performance more realistically, multi-class traffic flow models like Fastlane could be expanded to reproduce the capacity drop and therefore describe traffic even more realistically.

Fastlane solved in Lagrangian coordinates for faster and better predictions The numerical diffusion and the computational speed of macroscopic traffic flow models can be improved by replacing the Eulerian formulation, which is a coordinate system fixed to the road, by a Lagrangian formulation, which is a coordinate system that travels with the vehicles. Currently, the Lagrangian formulation for Fastlane is being developed by Van Wageningen-Kessels (2013) and applied for traffic state estimation by Yuan (2013). The Lagrangian formulation could therefore lead to a better and faster prediction of the traffic state. Due to its simpler numerical scheme, it can also improve the optimization for traffic control.

8.2.3 Traffic Control

Analysis of the effects of the capacity drop on the network performance The capacity drop is caused by congestion and has a negative effect on throughput and therefore on the traffic performance. The analysis of the effects of delaying or omitting the capacity drop by multi-class control measures could lead to class-specific control strategies that improve the network performance.

Analysis of other traffic performance functions Traffic management centers, policy makers or travelers may strive for other objectives than the minimization of the total cost of traffic. The analysis of other performance functions such as emissions, asphalt degradation, safety or travel time guaranties could lead to class-specific control strategies that improve the network performance defined in these terms.

Analysis of other vehicle-class specific properties In this thesis, we analyzed the effects of vehicle-class specific characteristics of speed, spacing and value of time. The analysis of other class-specific characteristics such as fuel consumption, weight and acceleration would make it possible to address the total cost or other performance functions in more detail.

Development and analysis of other multi-class DTM measures Traffic is also influenced by other mixed-class DTM measures that could be expanded to multi-class measures to improve the traffic performance. Examples are dynamic road pricing, mobility management, lane-change prohibitions or carpooling.

Expansion of the case study The case study of Chapter 7 included two multi-class route guidance measures and one multi-class ramp meter. More DTM measures could be used to improve the traffic conditions. For example, multi-class ramp meters could be installed on all on-ramps. Coordinating them might make it possible to reduce a large part of the congestion by preventing spillback, thereby significantly decreasing the total cost.

Improvement of the optimization procedure for faster computation The optimization of the control signal trajectory is currently performed by Matlab's built-in optimization algorithms. These use the traffic prediction model Fastlane in its current form. A reformulation of Fastlane and the control problem might lead to a faster solution algorithm of the control problem. A further direction is to assess the performance of fast optimization algorithms for Fastlane such as the feasible direction method, which was already applied by Kotsialos et al. (2002) for the Metanet model.

Distribution of the controller In the case study, the control signals for all DTM measures were computed globally. As the computation times showed, this approach does not scale well in the number of control signals and quickly becomes infeasible for practical applications. This problem can be remedied by splitting the network into multiple sub-networks and solving their control problems locally. The main question of this approach is how to enable the communication between the sub-networks and how to achievement the global optimum of such a distributed controller.

Avoidance of latent effects by anticipatory control Travelers adapt to regular congestion on a day-to-day basis and change their behavior; they therefore adapt to the control signals, which could lead to unexpected latent effects such as rat running. The application of anticipatory control which would take the travelers' decision into account could therefore prevent unexpected latent effects.

Effects of partial compliance The compliance of the travelers was assumed to be perfect in this research. Analysis of the acceptance of the travelers towards multi-class DTM measures and therefore the partial compliance of them would lead to more realistic results.

Coordination of traffic control with underlying network The underlying network, such as arterial roads and their traffic lights are usually controlled independently of the freeway network and its DTM measures. A coordination of these two systems can improve the network performance. For example, the traffic lights of the arterial can be coordinated with the ramp meter rate of the freeway to influence the length of the waiting queue at the ramp and to prevent it from spilling back to the underlying network.

Implementation issues To move towards implementation of multi-class traffic control, practical topics like the installation and maintenance of multi-class DTM measures, and the conversion of the traffic management centers and the education of traffic operators should be addressed.

8.3 Recommendations for Practical Use

This thesis showed that multi-class DTM improves the traffic conditions of freeway networks. The application of class-specific DTM requires an adjustment of the existing infrastructure. This thesis showed that multi-class ramp metering can be installed by providing a lane for each vehicle class. Multi-class route guidance is even easier to implement, as in-car devices become more and more popular, which makes them both easier to localize and easier to advise. This enables a fine-grained route advice for many different classes. With such an infrastructure, a road authority can manage the usage of its roads and the service it offers to its travelers in a fine-grained manner.

Besides these general remarks, more recommendation for practice are given for each component of the control loop.

8.3.1 Traffic State Estimation

Since both reformulated versions of the Adaptive Smoothing Method and the Extended Kalman Filter are now able to estimate the traffic state of realistic networks within a few seconds and they take the propagation of traffic flow characteristics into account, they can be used in practice.

Shock wave speeds can be extracted automatically by the Wave Speed Estimator both from empirical and synthetic data. Applications are the automatic calibration of traffic state estimators and the automatic evaluation of traffic flow models. The latter can be used in the automatic calibration of traffic flow models, for example, if the correct modeling of stop-and-go waves is an objective.

During the calibration of the prediction model and the estimation of the traffic state for the case study, valuable data could not be observed online. On the freeway, the traffic composition is not measured online. On the underlying road, the traffic state is observed only at a few locations. The traffic state thus cannot be estimated precisely. Therefore, probably the most important comment for practice is if traffic should be improved by the means of Dynamic Traffic Management, then sufficient traffic data have to be gathered, archived and be made available. The project of the National Data Warehouse that is currently being set up archives the traffic data of the Netherlands, including multi-class information at some locations (Nationale Databank Wegverkeergegevens, 2012).

8.3.2 Traffic State Prediction

We recommend the use of a multi-class prediction model for traffic networks that are subject to multiple vehicle classes. This enables the prediction of congestion, its spill-back, the travel time, the expected costs and many other variables. Furthermore, it is a crucial component for a proactive multi-class controller. Nevertheless, multi-class prediction can also be used for mixed-class DTM.

The validated prediction model of the A15 of Section 7.4 can be used in a traffic management center to predict the traffic state one hour ahead. Based on this predictions, information can be used to inform travelers or companies in the harbor area about the expected short-term traffic conditions of the freeway.

Furthermore, the validated model can support the development of control rules for the A15. It can be used to simulate control scenarios or to test rules for a rule-based controller. In addition, this system is flexible for the expansion of the infrastructure of the network, such as the reconstruction that is currently in progress or the installation of new DTM measures.

8.3.3 Traffic Control

The use of multi-class control improves the traffic conditions, as was shown in the case study. We developed concepts for multi-class ramp metering and multi-class route guidance and showed their benefits for DTM.

Furthermore, proactive control greatly improves the traffic conditions. The case study showed improvements in the order of 10%. It coordinates the DTM measures and is able to predict traffic conditions caused by rare events, such as incidents or a platoon of trucks. It also predicts the effects of DTM measures and is therefore capable of optimizing the signals for the DTM measures in order to optimized a given objective.

The next section outlines how the concepts developed in this thesis could be implemented in practice for application in a traffic management center. Section 8.5 shows an example for multi-class control for the A15.

8.4 Workflow Towards Implementation of Multi-class Dynamic Traffic Management in Practice

Current state of the practice: reactive rule-based control with triggers The current state of the practice of traffic management centers is a two-layered system. The top layer uses a rule-based controller, where traffic control scenarios are activated or deactivated based on simple triggers, such as the presence of congestion at specified locations. Instead of estimating the traffic state completely, the sensor data are used directly to trigger the control scenarios. In the bottom layer, local DTM measures are activated or deactivated. The DTM measures themselves are controlled locally without communicating to other DTM measures.

Here, we present a workflow towards an implementation of the system presented in the case study of Chapter 7 for use in traffic management centers in practice. The result of each step of the workflow itself contributes to the state of the practice. The following workflow therefore presents an incremental path towards proactive optimal vehicle-class specific traffic control where the product of each step itself can be applied in practice.

Implementation of a traffic state estimator for the whole network The first step is to estimate the current traffic state over the whole network in real-time. The current traffic state includes the traffic density over the whole network, which is required to initialize a traffic flow prediction model. Example traffic state estimators are the ones developed and validated in Chapters 3 and 4, which are able to estimate the traffic state in real time; Section 7.2 showed an example for the A15.

Calibration and validation of a traffic flow model for short-term traffic flow prediction The next step is to implement a traffic flow model that predicts how the traffic conditions will develop over the next hour. This allows for prediction of the effects of incidents and the resulting traffic conditions like congestion, spillback, travel time or throughput. Most traffic flow models represent the traffic in the form of density; such a model is therefore initialized with the current estimated traffic density. The implementation of a traffic flow model includes its calibration and validation, i.e. the tuning of its parameters so that the predicted traffic state is close to the true traffic state. Sections 7.3 and 7.4 showed how the Fastlane model presented in Sections 5.1 and 5.2 was calibrated and validated for the A15.

Application of predictive rule-based control Instead of activating the scenarios of the rule-based controller based on the current traffic state, they can be activated based on future traffic state predicted by the validated traffic flow model. By these means, the negative effects of congestion can be anticipated, and traffic control scenarios can be activated earlier than in a reactive approach in order to counter the congestion.

Application of model-predictive control Once optimal control proves to yield a better traffic performance than rule-based control, it can be implemented in a traffic management center. This requires the application of the control signals to the actual DTM measures. Since the feedback loop of the control loop would then be closed, this approach is a full-fledged model-predictive controller. Online model-predictive control requires a fast optimization of the control problem, so that the control signals are computed within a few minutes. Paths towards a fast optimization include the reformulation of the control problem, the limitation of the number of DTM measures to a few, or the usage of efficient hardware to solve the control problem.

Generalization to vehicle-class specific control Finally, the mixed-class DTM measures can be generalized to multi-class DTM measures to optimize the traffic performance as shown in this thesis. This requires the installation and operation of multi-class DTM measures like the ones in Sections 6.3 and 6.4 so that multi-class control can be performed.

Result: proactive optimal vehicle-class specific traffic control The result is a proactive traffic controller that coordinates multiple multi-class DTM measures and computes the optimal control signals individually for each vehicle class. The case study of Chapter 7 shows the application of a simulation of such a controller and its positive effects on the traffic performance of the network. The next section shows an example of how the developed system of the case study can be used in real-time.

8.5 Application of BOS-HbR in Practice

This thesis concludes with an example of multi-class optimal control as it can be applied in practice. The system BOS-HbR was developed as a prototype to estimate, predict and control the traffic of the A15 near the harbor of Rotterdam. It is currently running online on a dedicated computer at Delft University of Technology. The results including the optimal control signals are computed in real-time and are presented on the BOS-HbR Website (2012), a screenshot is shown in Figure 8.1.

Project Summary

The aim of the software prototype BOS-HbR (Beslissingsondersteunend Systeem voor het Havenbedrijf Rotterdam) is to show that the traffic conditions near the harbor of Rotterdam can be improved by applying proactive Dynamic Traffic Management (DTM). The network encompasses the main route of the A15 and a part of the underlying parallel structure, namely the Botlekbrug and the Vondelingenweg.



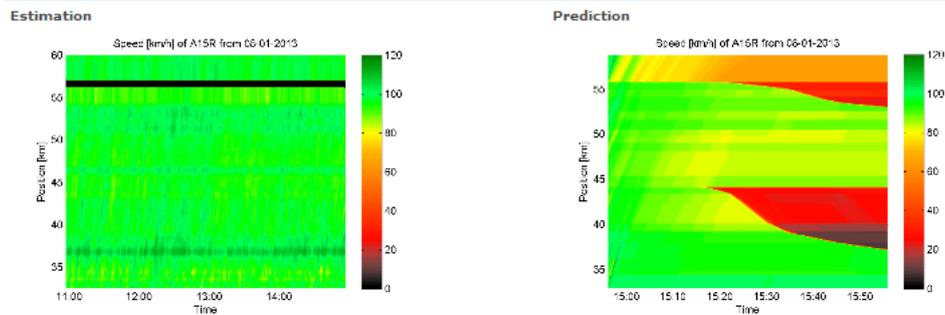
Dependent on the current traffic conditions, BOS-HbR advises to reroute a part of the traffic via the underlying parallel structure. A route-guidance controller is simulated at the bifurcation to the alternative route at Spijkenisse, a second route-guidance controller is simulated after the Botlekbrug. Furthermore, the controllers advice a route for each vehicle class; BOS-HbR distinguishes between cars and trucks.

The control values are computed by three components. The first component of BOS-HbR estimates the current traffic of the A15 and the underlying road. The second component predicts the traffic by means of a traffic flow model over the next hour. This component can furthermore simulate the effect of the class-specific route-guidance controllers. The third component optimizes the route guidance signals in such a way that the predicted vehicle loss hours of traffic are minimized, whereby the time of a trucks counts three times than the one of a car.

This page shows the estimated traffic state of the A15, the predicted traffic state if no control is done, the optimal route-guidance signals to minimize the vehicle loss hour and the corresponding future traffic state of the A15.

Realtime Data of the A15

Estimation and Prediction without Control



Prediction with proactive Control

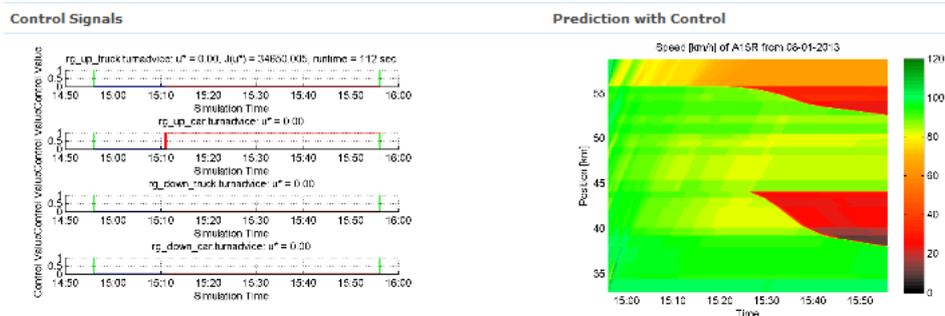


Figure 8.1: Screenshot of the BOS-HbR Website (2012)

It is a simplified version of the case study presented in Chapter 7. Only the two multi-

class route-guidance measures are simulated to reroute traffic via the alternative route parallel to the A15. Furthermore, the control interval is increased to 15 min, and the number of control intervals in the horizon is reduced to two, leading to eight control signals being optimized so that the computation time is significantly lower than in the case study. The average computation time of a cycle of the control loop of BOS-HbR is 6 min.

BOS-HbR can be the basis for further development. For example, its performance can be compared with that of the current state of the practice, further DTM measures can be simulated and optimized, the network of the harbor or the freeway ring of Rotterdam can be implemented to optimize traffic streams during regular and incidental conditions, or further vehicle classes such as empty and loaded trucks could be simulated to optimize the logistics of the harbor. If extended with vehicle emissions model, the system can also be used to minimize the total emission of the network.

Bibliography

- Abdullah, S., M. Khalid, R. Yusof, K. Omar (2007) Comparison of feature extractors in license plate recognition, in: *Modelling & Simulation, 2007. AMS'07. First Asia International Conference on*, pp. 502–506.
- Almejalli, K., K. Dahal, M. Hossain (2007) Intelligent traffic control decision support system, in: Giacobini, M., ed., *Applications of Evolutionary Computing*, vol. 4448 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 688–701, 10.1007/978-3-540-71805-5-75.
- Anagnostopoulos, C., I. Anagnostopoulos, V. Loumos, E. Kayafas (2006) A license plate-recognition algorithm for Intelligent Transportation System applications, *IEEE Transactions on Intelligent Transportation Systems*, 7(3), p. 377.
- Ansorge, R. (1990) What does the entropy condition mean in traffic flow theory?, *Transportation Research Part B: Methodological*, 24(2), pp. 133 – 143.
- Bando, M., K. Hasebe, K. Nakanishi, A. Nakayama (1998) Analysis of optimal velocity model with explicit delay, *Phys. Rev. E*, 58(5), pp. 5429–5435.
- Benzoni-Gavage, S., R. Colombo (2003) An n -populations model for traffic flow, *European Journal of Applied Mathematics*, 14(05), pp. 587–612.
- Bertini, R. L., M. T. Leal (2005) Empirical study of traffic features at a freeway lane drop, *Journal of Transportation Engineering*, 131(6), pp. 397–407.
- Blandin, S., A. Couque, A. Bayen, D. Work (2012) On sequential data assimilation for scalar macroscopic traffic flow models, *Physica D: Nonlinear Phenomena*, 241(17), pp. 1421 – 1440.
- Bogenberger, K., H. Keller, S. Ritchie (2001) *Adaptive Fuzzy Systems for Traffic Responsive and Coordinated Ramp Metering*, Ph.D. thesis, Technische Universität München.
- Buckeye, K. R. (2012) Innovations on managed lanes in Minnesota, *Public Works Management & Policy*.
- Carlson, R. C., I. Papamichail, M. Papageorgiou, A. Messmer (2010) Optimal mainstream traffic flow control of large-scale motorway networks, *Transportation Research Part C: Emerging Technologies*, 18(2), pp. 193 – 212.

- Cassidy, M., C. Daganzo, K. Jang, K. Chung (2009) Spatiotemporal effects of segregating different vehicle classes on separate lanes, *Transportation and Traffic Theory 2009: Golden Jubilee*, pp. 57–74.
- Cassidy, M. J., M. Mauch (2001) An observed traffic pattern in long freeway queues, *Transportation Research Part A: Policy and Practice*, 35(2), pp. 143 – 156.
- Chanut, S., C. Buisson (2003) Macroscopic model and its numerical solution for two-flow mixed traffic with different speeds and lengths, *Transportation Research Record: Journal of the Transportation Research Board*, 1852(-1), pp. 209–219.
- Chi, S., C. H. Caldas (2011) Automated object identification using optical video cameras on construction sites, *Computer-Aided Civil and Infrastructure Engineering*, 26(5), pp. 368–380.
- Chiabaut, N., L. Leclercq (2011) Wave velocity estimation through cumulative vehicle count curves automatic analysis, *Transportation Research Record: Journal of the Transportation Research Board*, 2249(-1), pp. 1–6.
- Chiabaut, N., L. Leclercq, C. Buisson (2010) From heterogeneous drivers to macroscopic patterns in congestion, *Transportation Research Part B: Methodological*, 44(2), pp. 299 – 308.
- Chung, K., J. Rudjanakanoknad, M. J. Cassidy (2007) Relation between traffic density and capacity drop at three freeway bottlenecks, *Transportation Research Part B: Methodological*, 41(1), pp. 82 – 95.
- Coifman, B., S. Kim (2009) Speed estimation and length based vehicle classification from freeway single-loop detectors, *Transportation Research Part C: Emerging Technologies*, 17(4), pp. 349 – 364.
- Coifman, B., Y. Wang (2005) Average velocity of waves propagating through congested freeway traffic, in: *Transportation and Traffic Theory. Flow, Dynamics and Human Interaction. 16th International Symposium on Transportation and Traffic Theory*, pp. 165–179.
- Cormen, T., C. Leiserson, R. Rivest, C. Stein (2001) *Introduction to Algorithms*, The MIT press, 2nd edition (September 1, 2001).
- Courant, R., K. Friedrichs, H. Lewy (1928) Über die partiellen Differenzgleichungen der mathematischen Physik, *Mathematische Annalen*, 100(1), pp. 32–74.
- Daganzo, C. F. (1994) The Cell Transmission Model: A dynamic representation of highway traffic consistent with the hydrodynamic theory, *Transportation Research Part B: Methodological*, 28(4), pp. 269 – 287.
- Dahlkamp, H., A. Kaehler, D. Stavens, S. Thrun, G. Bradski (2006) Self-supervised monocular road detection in desert terrain, in: *Proc. of Robotics: Science and Systems (RSS)*.

- De Palma, A., M. Kilani, R. Lindsey (2008) The merits of separating cars and trucks, *Journal of Urban Economics*, 64(2), pp. 340 – 361.
- De Schutter, B., S. P. Hoogendoorn, H. Schuurman, S. Stramigioli (2003) A multi-agent case-based traffic control scenario evaluation system, in: *Intelligent Transportation Systems, 2003. Proceedings. 2003 IEEE*, vol. 1, pp. 678 – 683 vol.1.
- Dijker, T. (2002) FOSIM (Freeway Operations SIMulation), URL www.fosim.nl.
- Edie, L. (1965) Discussion of traffic stream measurements and definitions, in: *Proceedings*, Organisation for Economic Co-operation and Development, p. 139.
- Elefteriadou, L., D. Torbic, N. Webster (1997) Development of passenger car equivalents for freeways, two-lane highways, and arterials, *Transportation Research Record: Journal of the Transportation Research Board*, 1572, pp. 51–58.
- Evensen, G. (2003) The Ensemble Kalman Filter: Theoretical formulation and practical implementation, *Ocean dynamics*, 53(4), pp. 343–367.
- Google Maps (2012) <http://maps.google.com/>.
- Graves, T., A. Karr, N. Rouphail, P. Thakuriah (1998) Real-time prediction of incipient congestion on freeways from detector data, *Technical Report, National Institute of Statistical Sciences, ResearchTriangle Park, NC*.
- Hegyi, A. (2004) *Model predictive control for integrating traffic control measures*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics.
- Hegyi, A., B. De Schutter, H. Hellendoorn (2005) Model predictive control for optimal coordination of ramp metering and variable speed limits, *Transportation Research Part C: Emerging Technologies*, 13(3), pp. 185 – 209.
- Herrera, J. C., A. M. Bayen (2010) Incorporation of Lagrangian measurements in freeway traffic state estimation, *Transportation Research Part B: Methodological*, 44(4), pp. 460 – 481.
- Hodge, V., R. Krishnan, T. Jackson, J. Austin, J. Polak (2011) Short-term traffic prediction using a binary neural network, in: *43rd Annual UTSG Conference, Open University, Milton Keynes, UK*.
- Homepage Texas Transportation Institute (2012) <http://tti.tamu.edu/2011/03/01/the-wsdot-express-lanes-project-research-into-practice>.
- Hoogendoorn, S. P. (1999) *Multiclass continuum modelling of multilane traffic flow*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics, Delft.

- Hoogendoorn, S. P., B. De Schutter, H. Schuurman (2003) Decision support in Dynamic Traffic Management. Real-time scenario evaluation, *European Journal of Transport and Infrastructure Research*, 3(1), pp. 21–38.
- Hough, P. (1962) Method and means for recognizing complex patterns, US Patent 3069654.
- Huber, M., U. Hanebeck (2007) The hybrid density filter for nonlinear estimation based on hybrid conditional density approximation, in: *Information Fusion, 2007 10th International Conference on*, pp. 1–8.
- Inter Visual Systems (2012) Traffic management center Rhooon, Source: http://www.inter.nl/upload/Image/Control_Room_Systems/Groot/rijkswaterstaat_3_1_Large.jpg.
- Jazwinski, A. (1970) *Stochastic processes and filtering theory*, Academic Pr.
- Julier, S., J. Uhlmann (1997) A new extension of the Kalman Filter to nonlinear systems, in: *Int. Symp. Aerospace/Defense Sensing, Simul. and Controls*, vol. 3, Spie Bellingham, WA, p. 26.
- Kalman, R. (1960) A new approach to linear filtering and prediction problems, *Journal of basic Engineering*, 82(1), pp. 35–45.
- Kastrinaki, V., M. Zervakis, K. Kalaitzakis (2003) A survey of video processing techniques for traffic applications, *Image and Vision Computing*, 21(4), pp. 359–381.
- Kerner, B. S. (1998) Experimental features of self-organization in traffic flow, *Physical Review Letters*, 81(17), pp. 3797–3800.
- Kerner, B. S., H. Rehborn (1997) Experimental properties of phase transitions in traffic flow, *Physical Review Letters*, 79(20), pp. 4030–4033.
- Kesting, A., M. Treiber (2008) Calculating travel times from reconstructed spatiotemporal traffic data, in: *Proceedings of the 4th International Symposium Networks for Mobility, Stuttgart*, iISBN: 978-3-921882-24-5.
- Knoop, V. L., S. P. Hoogendoorn, H. J. Zuylen (2009) Empirical differences between time-mean speed and space-mean speed, *Traffic and Granular Flow07*, pp. 351–356.
- Kotsialos, A., M. Papageorgiou, M. Mangeas, H. Haj-Salem (2002) Coordinated and integrated control of motorway networks via non-linear optimal control, *Transportation Research Part C: Emerging Technologies*, 10(1), pp. 65–84.
- Laval, J. (2010) Hysteresis in the fundamental diagram: Impact of measurement methods, in: *Transportation Research Board 89th Annual Meeting*, 10-1597.
- Lebacque, J. (2003) Two-phase bounded-acceleration traffic flow model. Analytical solutions and applications, *Transportation Research Record*, 1852, pp. 220–230.

- Lebaque, J. (1996) The Godunov scheme and what it means for first order traffic flow models, in: *Proceedings of the 13th International Symposium on Transportation and Traffic Theory. JP Lesort (ed.) Pergamon*, Pergamon Press, pp. 647–677.
- Leclercq, L., J. Laval, E. Chevallier (2007) The Lagrangian coordinates and what it means for first order traffic flow models, in: Allsop, R., M. Bell, B. Heydecker, eds., *Transportation and Traffic Theory 2007*, Elsevier, Oxford, UK, pp. 735–753.
- Leutzbach, W., R. Wiedemann (1986) Development and applications of traffic simulation models at the Karlsruhe Institut für Verkehrswesen, *Traffic engineering & control*, 27(5), pp. 270–278.
- Lighthill, M., G. Whitham (1955) On kinematic waves. II. A theory of traffic flow on long crowded roads, *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences (1934-1990)*, 229(1178), pp. 317–345.
- Lin, S., B. De Schutter, Y. Xi, J. Hellendoorn (2009) A simplified macroscopic urban traffic network model for model-based predictive control, in: *Proceedings of the 12th IFAC Symposium Control Transportation Systems*, pp. 286–291.
- Logghe, S., L. Immers (2008) Multi-class kinematic wave theory of traffic flow, *Transportation Research Part B: Methodological*, 42(6), pp. 523 – 541.
- Lou, Y., Y. Yin, J. A. Laval (2011) Optimal dynamic pricing strategies for high-occupancy/toll lanes, *Transportation Research Part C: Emerging Technologies*, 19(1), pp. 64 – 74.
- Malinovski, Y., J. Zheng, Y. Wang (2009) Model-free video detection and tracking of pedestrians and bicyclists, *Computer-Aided Civil and Infrastructure Engineering*, 24(3), pp. 157–168.
- Mammar, S., A. Messmer, P. Jensen, M. Papageorgiou, H. Haj-Salem, L. Jensen (1996) Automatic control of variable message signs in Aalborg, *Transportation Research Part C: Emerging Technologies*, 4(3), pp. 131 – 150.
- Masher, D., D. Ross, P. Wong, P. Tuan, H. Zeidler, S. Petracek (1975) Guidelines for design and operation of ramp control systems, Tech. rep., Stanford Research Institute, Menid Park, CA.
- Messmer, A., M. Papageorgiou (1990) METANET: A macroscopic simulation program for motorway networks, *Traffic Engineering and Control*, 31, pp. 466–70.
- Messmer, A., M. Papageorgiou, N. Mackenzie (1998) Automatic control of variable message signs in the interurban Scottish highway network, *Transportation Research Part C: Emerging Technologies*, 6(3), pp. 173 – 187.
- Mihaylova, L., R. Boel, A. Hegyi (2007) Freeway traffic estimation within Particle Filtering framework, *Automatica*, 43(2), pp. 290 – 300.

- Munroe, R. (2012) xkcd Nr. 1007: "Sustainable", URL <http://xkcd.com/1007/>.
- Nationale Databank Wegverkeergegevens (2012) <http://www.ndw.nu>.
- Newell, G. F. (2002) A simplified car-following theory: A lower order model, *Transportation Research Part B: Methodological*, 36(3), pp. 195 – 205.
- Ng, M., S. T. Waller (2010) A computationally efficient methodology to characterize travel time reliability using the Fast Fourier Transform, *Transportation Research Part B: Methodological*, In Press, Corrected Proof.
- Ngoduy, D. (2008) Applicable filtering framework for online multiclass freeway network estimation, *Physica A: Statistical Mechanics and its Applications*, 387(2-3), pp. 599 – 616.
- Ngoduy, D., R. Liu (2007) Multiclass first-order simulation model to explain non-linear traffic phenomena, *Physica A: Statistical Mechanics and its Applications*, 385(2), pp. 667 – 682.
- Papageorgiou, M., H. Hadj-Salem, J. Blosseville (1991) ALINEA: A local feedback control law for on-ramp metering, *Transportation Research Record*, 1320(1320), pp. 194–198.
- Papamichail, I., A. Kotsialos, I. Margonis, M. Papageorgiou (2010) Coordinated ramp metering for freeway networks – A model-predictive hierarchical control approach, *Transportation Research Part C: Emerging Technologies*, 18(3), pp. 311 – 331, 11th IFAC Symposium: The Role of Control.
- Papamichail, I., M. Papageorgiou (2008) Traffic-responsive linked ramp-metering control, *Intelligent Transportation Systems, IEEE Transactions on*, 9(1), pp. 111 –121.
- Payne, H. (1971) Models of freeway traffic and control, *Mathematical Models of Public Systems*, 1(1), pp. 51–61.
- Pipes, L. (1967) Car following models and the fundamental diagram of road traffic, *Transportation Research*, 1(1), pp. 21–29.
- Qing, O. (2011) *Fusing Heterogeneous Traffic Data: Parsimonious Approaches using Data-Data Consistency*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics.
- Regiolab Delft (2012) <http://www.regiolab-delft.nl/>.
- Richards, P. (1956) Shock waves on the highway, *Operations research*, 4(1), pp. 42–51.
- Rijkswaterstaat (2011) Economische kengetallen, URL http://www.rijkswaterstaat.nl/kenniscentrum/economische_evaluatie/kengetallen/index.aspx.

- Roportis (2012) <http://www.roportis.com/default.aspx>.
- Schönhof, M., D. Helbing (2009) Criticism of three-phase traffic theory, *Transportation Research Part B: Methodological*, 43(7), pp. 784 – 797.
- Schreiter, T., R. L. Landman, J. W. C. Van Lint, A. Hegyi, S. P. Hoogendoorn (accepted) Vehicle-class-specific route guidance of freeway traffic by model-predictive control, *Transportation Research Record*.
- Schreiter, T., A. J. Pel, J. W. C. Van Lint, S. P. Hoogendoorn (2012) Modeling monetary costs of multi-class traffic flow – application to the dynamic management of truck lanes, in: *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*.
- Schreiter, T., J. W. C. Van Lint, S. P. Hoogendoorn (2011) Multi-class ramp metering: Concepts and initial results, in: *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, pp. 885 –889.
- Schreiter, T., J. W. C. Van Lint, S. P. Hoogendoorn (2013) Vehicle-class specific control of freeway traffic, in: *Transportation Research Board 92nd Annual Meeting*, 13-0585.
- Schreiter, T., J. W. C. Van Lint, M. Treiber, S. P. Hoogendoorn (2010a) Two fast implementations of the Adaptive Smoothing Method used in highway traffic state estimation, in: *Proceedings of IEEE ITSC, Funchal, Portugal*, pp. 1202–1208.
- Schreiter, T., J. W. C. Van Lint, Y. Yuan, S. P. Hoogendoorn (2010b) Propagation wave speed estimation of freeway traffic with image processing tools, in: *in proceeding of the Transportation Research Board, 89th Annual Meeting, Washington, D.C., DVD*.
- Shapiro, V., G. Gluhchev, D. Dimov (2006) Towards a multinational car license plate recognition system, *Machine Vision and Applications*, 17(3), pp. 173–183.
- Smith, B. L., B. M. Williams, R. K. Oswald (2002) Comparison of parametric and nonparametric models for traffic flow forecasting, *Transportation Research Part C: Emerging Technologies*, 10(4), pp. 303 – 321.
- Sorenson, H. (1985) *Kalman filtering: theory and application*, IEEE.
- Strassen, V. (1969) Gaussian elimination is not optimal, *Numerische Mathematik*, 13(4), pp. 354–356.
- Taale, H., S. P. Hoogendoorn (2012) Network-wide traffic management with integrated anticipatory control, in: *Transportation Research Board 91st Annual Meeting*, 12-3654.
- Taale, H., F. Middelham (2000) Ten years of ramp-metering in the Netherlands, in: *Road Transport Information and Control, 2000. Tenth International Conference on (Conf. Publ. No. 472)*, pp. 106 –110.

- Tampère, C. M. J., L. H. Immers (2007) An Extended Kalman Filter application for traffic state estimation using CTM with implicit mode switching and dynamic parameters, in: *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*, pp. 209–216.
- Treiber, M., D. Helbing (2002) Reconstructing the spatio-temporal traffic dynamics from stationary detector data, *Cooperative Transportation Dynamics*, 1(3), pp. 3.1–3.21.
- Treiber, M., A. Hennecke, D. Helbing (2000) Congested traffic states in empirical observations and microscopic simulations, *Physical Review E*, 62(2), pp. 1805–1824.
- Treiber, M., A. Kesting (2012) Validation of traffic flow models with respect to the spatiotemporal evolution of congested traffic patterns, *Transportation Research Part C: Emerging Technologies*, 21(1), pp. 31–41.
- Treiber, M., A. Kesting, R. E. Wilson (2011) Reconstructing the traffic state by fusion of heterogeneous data, *Computer-Aided Civil and Infrastructure Engineering*, 26:6, pp. 408–419.
- Tyburski, R. (1988) A review of road sensor technology for monitoring vehicle traffic, *ITE (Institute of Transportation Engineers) Journal;(USA)*, 59(8).
- UJMP (2010) <http://www.ujmp.org>.
- Van Hinsbergen, C. P. I. (2010) *Bayesian Data Assimilation for Improved Modeling of Road Traffic*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics.
- Van Hinsbergen, C. P. I., T. Schreiter, F. S. Zuurbier, J. W. C. Van Lint, H. J. Van Zuylen (2012) Localized Extended Kalman Filter for scalable real-time traffic state estimation, *Intelligent Transportation Systems, IEEE Transactions on*, 13(1), pp. 385–394.
- Van Hinsbergen, C. P. I., F. S. Zuurbier, J. W. C. Van Lint, H. J. Van Zuylen (2008) Using an lwr model with a cell based extended kalman filter to estimate travel times, in: *Third International Symposium of Transport Simulation*, 65, Surfers Paradise, QLD, Australia.
- Van Lint, J. W. C. (2010) Empirical evaluation of new robust travel time estimation algorithms, *Transportation Research Record*, 2160 / 2010(2160), pp. 50–59.
- Van Lint, J. W. C., S. P. Hoogendoorn (2010) A robust and efficient method for fusing heterogeneous data from traffic sensors on freeways, *Computer-Aided Civil and Infrastructure Engineering*, 25(8), pp. 596–612.
- Van Lint, J. W. C., S. P. Hoogendoorn, A. Hegyi (2008a) Dual EKF state and parameter estimation in multi-class first-order traffic flow models, in: *Proceedings of the 17th IFAC World Congress*.

- Van Lint, J. W. C., S. P. Hoogendoorn, M. Schreuder (2008b) Fastlane: New multi-class first-order traffic flow model, *Transportation Research Record: Journal of the Transportation Research Board*, 2088, pp. 177–187.
- Van Lint, J. W. C., N. J. Van der Zijpp (2003) An improved travel-time estimation algorithm by using dual loop detectors, *Transportation Research Record: Journal of the Transportation Research Board*, 1855, pp. 41–48.
- Van Wageningen-Kessels, F. L. M. (2013) *Multi class continuum traffic flow models: Analysis and simulation methods*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics, Delft, The Netherlands.
- Van Wageningen-Kessels, F. L. M., J. W. C. Van Lint, S. P. Hoogendoorn, C. Vuik (2010) Lagrangian formulation of a multi-class kinematic wave model, *Transportation Research Record: Journal of the Transportation Research Board*, 2188, pp. 29–36.
- Van Zuylen, H. J., F. Viti, V. L. Knoop, H. Tu (2007) Analysis of accidents for the a15 Dutch motorway, Tech. rep., Delft University of Technology.
- Wahle, J., A. L. C. Bazzan, F. Klügl, M. Schreckenberg (2000) Decision dynamics in a traffic scenario, *Physica A: Statistical Mechanics and its Applications*, 287(34), pp. 669 – 681.
- Wallace, G. (1992) The JPEG still picture compression standard, *IEEE Transactions on Consumer Electronics*, 38(1).
- Wang, M., S. P. Hoogendoorn, W. Daamen, R. G. Hoogendoorn, B. V. Arem (2012) Driver support and cooperative systems control design: Framework and preliminary results, in: *American Control Conference, 2012. Proceedings of the 2012*.
- Wang, Y., M. Papageorgiou (2005) Real-time freeway traffic state estimation based on Extended Kalman Filter: A general approach, *Transportation Research Part B*, 39(2), pp. 141–167.
- Wang, Y., M. Papageorgiou, A. Messmer (2003) Predictive feedback routing control strategy for freeway network traffic, *Transportation Research Record: Journal of the Transportation Research Board*, 1856(-1), pp. 62–73.
- Wang, Y., M. Papageorgiou, A. Messmer (2007) Real-time freeway traffic state estimation based on Extended Kalman Filter: A case study, *Transportation Science*, 41(2), pp. 167–181.
- Wang, Y., M. Papageorgiou, G. Sarros, W. Knibbe (2006) Feedback route guidance applied to a large-scale express ring road, *Transportation Research Record: Journal of the Transportation Research Board*, 1965(-1), pp. 79–88.

- BOS-HbR Website (2012) <http://www.regiolab-delft.nl/boshbr>.
- Weissel, F., T. Schreiter, M. F. Huber, U. D. Hanebeck (2008) Stochastic model predictive control of time-variant nonlinear systems with imperfect state information, in: *Multisensor Fusion and Integration for Intelligent Systems, 2008. MFI 2008. IEEE International Conference on*, pp. 40–46.
- Wikimedia Commons (2012) <http://commons.wikimedia.org>.
- Wong, G. C. K., S. C. Wong (2002) A multi-class traffic flow model – an extension of LWR model with heterogeneous drivers, *Transportation Research Part A: Policy and Practice*, 36(9), pp. 827–841.
- Work, D., O. Tossavainen, S. Blandin, A. Bayen, T. Iwuchukwu, K. Tracton (2008) An Ensemble Kalman Filtering approach to highway traffic estimation using GPS enabled mobile devices, *Proceeding of the 47th IEEE Conference on Decision and Control*, 47, pp. 5062–5068.
- Yuan, Y. (2013) *Lagrangian Multi-class Traffic State Estimation*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics, Delft, The Netherlands.
- Yuan, Y., J. W. C. Van Lint, F. L. M. Van Wageningen-Kessels, S. P. Hoogendoorn (in press) Network-wide Lagrangian traffic state estimation using loop and floating car data, *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*.
- Yuan, Y., R. E. Wilson, J. W. C. Van Lint, S. P. Hoogendoorn (2012) Estimation of multi-class and multi-lane counts from aggregate loop detector data, *Transportation Research Record*, p. In press.
- Zhang, X., J. Rice (1999) Visualizing loop detector data, Tech. rep., University of California, Berkeley.
- Zheng, Z., S. Ahn, D. Chen, J. Laval (2011a) Applications of wavelet transform for analysis of freeway traffic: Bottlenecks, transient traffic, and traffic oscillations, *Transportation Research Part B: Methodological*, 45(2), pp. 372–384.
- Zheng, Z., S. Ahn, D. Chen, J. Laval (2011b) Freeway traffic oscillations: Microscopic analysis of formations and propagations using wavelet transform, *Transportation Research Part B: Methodological*, 17, pp. 702–716.
- Zuurbier, F. (2010) *Intelligent Route Guidance*, Ph.D. thesis, Delft University of Technology; Netherlands Research School for Transport, Infrastructure and Logistics.
- Zuurbier, F. S., J. W. C. V. Lint, V. L. Knoop (2006) State estimation using an Extended Kalman Filter and the first order traffic flow model DSMART, in: *Eleventh IFAC Symposium on Control in Transportation Systems*, Delft, The Netherlands.

Appendices

Appendix A

Calculation of the Class-specific Turnfraction on the Main Route

The calibration of the Fastlane model for the Prediction component requires the turnfractions of the user classes. Section 7.3.3 presented the calibration of both inflows and turnfraction. This appendix shows how the turnfractions on the main route are calculated (c.f. Figure 7.5 and Table 7.4). The goal is thus to determine the class-specific turnfractions of the reroutable destination classes β_r and non-reroutable destination classes β_{nr} at a bifurcation point.

Let the following traffic variables be given: total turnfraction β_{tot} and total flow q_{tot} at that bifurcation point, the base share $\bar{\eta}$ of the reroutable class, and the total flow before the bifurcation to the alternative route q_{Sp} . The following system of equations is given:

$$\text{total outflow } q_{tot}^{out} = \beta_{tot} \cdot q_{tot} \quad (\text{A.1})$$

$$\text{class-specific turnfraction } \beta_u = \frac{q_u^{out}}{q_u} \quad (\text{A.2})$$

$$\text{flow of reroutable class } q_r = \bar{\eta} \cdot q_{Sp} \quad (\text{A.3})$$

$$\text{flow of non-reroutable class } q_{nr} = q_{tot} - q_r \quad (\text{A.4})$$

by definition of the classes from Figure 7.5 follows

$$q_r^{out} = 0 \quad (\text{A.5})$$

$$q_{nr}^{out} = q^{out} \quad (\text{A.6})$$

The class-specific turnfraction are thus

$$\beta_r = 0 \quad (\text{A.7})$$

$$\beta_{nr} = \frac{q_{nr}^{out}}{q_{nr}} \quad (\text{A.8})$$

The two terms of the right-hand side of (A.8) are then

$$q_{nr}^{out} = q_{tot}^{out} - q_r^{out} = q_{tot}^{out} = \beta_{tot} \cdot q_{tot} \quad (\text{A.9})$$

$$q_{nr} = q_{tot} - q_r = q_{tot} - \bar{\eta} q_{Sp} \quad (\text{A.10})$$

so that (A.8) leads to

$$\beta_{nr} = \frac{q_{tot}}{q_{tot} - \bar{\eta} q_{Sp}} \cdot \beta_{tot} \quad (\text{A.11})$$

To calculate the user-class specific turnfractions $\beta_{car}^{adjusted}$ and $\beta_{truck}^{adjusted}$, respectively, apply the vehicle-class specific turnfractions β_{car} and β_{truck} , respectively, as the total turnfraction β_{tot} in (A.1).

Appendix B

Conversion of Traffic Data to Initialize Fastlane

B.1 Problem Description

Traffic data are given in the form of total vehicular densities and flow shares per class. From this data the class-specific vehicular densities have to be derived. This will be shown for the case of two classes. In the following we will use these symbols:

k_{tot}	total vehicular density
K_{tot}	total effective density
k_u	vehicular density of class u
v_u	speed of class u
q_u	vehicular flow of class u
$v_u^{\text{FD}}(K_{\text{tot}})$	fundamental diagram for class u
η_u	share of class u
π_u^*	pce value of class u
$\pi_u(v_1, v_u)$	pce function of class u (function of speeds)
$\bar{\pi}_u(K_{\text{tot}})$	pce function of class u (function of effective density)

B.2 Solution Procedure

The traffic data are converted from total vehicular density and share to class-specific densities in the following three steps.

step 1 Solve for $K_{\text{tot}} \in [0, K_{\text{jam}}]$:

$$\left[\frac{\eta_2}{\eta_1} v_1^{\text{FD}}(K_{\text{tot}}) + v_2^{\text{FD}}(K_{\text{tot}}) \right] K_{\text{tot}} - \left[\frac{\eta_2}{\eta_1} \bar{\pi}_2(K_{\text{tot}}) v_1^{\text{FD}}(K_{\text{tot}}) + v_2^{\text{FD}}(K_{\text{tot}}) \right] k_{\text{tot}} = 0. \quad (\text{B.1})$$

step 2 Use the fundamental diagram to find the speeds:

$$v_1 = v_1^{\text{FD}}(K_{\text{tot}}) \text{ and } v_2 = v_2^{\text{FD}}(K_{\text{tot}}). \quad (\text{B.2})$$

step 3. Find k_1 and k_2 by using:

$$k_1 = \frac{v_2}{\frac{\eta_2}{\eta_1}v_1 + v_2} k_{\text{tot}} \text{ and } k_2 = k_{\text{tot}} - k_1. \quad (\text{B.3})$$

B.3 Proof

The following relations are used:

$$k_{\text{tot}} = k_1 + k_2, \quad (\text{B.4})$$

$$\eta_1 = \frac{q_1}{q_1 + q_2}, \quad (\text{B.5})$$

$$\eta_2 = \frac{q_2}{q_1 + q_2}, \quad (\text{B.6})$$

$$q_1 = k_1 v_1, \quad (\text{B.7})$$

$$q_2 = k_2 v_2, \quad (\text{B.8})$$

$$v_1 = v_1^{\text{FD}}(K_{\text{tot}}), \quad (\text{B.9})$$

$$v_2 = v_2^{\text{FD}}(K_{\text{tot}}), \quad (\text{B.10})$$

$$\pi_2^* = \pi_2(v_1, v_2), \quad (\text{B.11})$$

$$K_{\text{tot}} = k_1 + \pi_2^* k_2. \quad (\text{B.12})$$

Equations (B.9)–(B.11) refer to modeling functions in Fastlane. Their parameters are supposed to be known. Furthermore, the total density k_{tot} and the shares η_1 and η_2 are supposed to be known.

Proof of step 2 Trivial.

Proof of step 3 Once v_1 is known (from step 1 and 2), equations (B.4)–(B.8) can be used to find k_1 and k_2 . From (B.5) and (B.6) it follows that:

$$\left. \begin{array}{l} \text{from (B.5): } q_1 = \eta_1(q_1 + q_2) \\ \text{from (B.6): } q_1 + q_2 = \frac{q_2}{\eta_2} \end{array} \right\} \Rightarrow q_1 = \frac{\eta_1}{\eta_2} q_2. \quad (\text{B.13})$$

We now substitute (B.7) on the left-hand side of (B.13) and (B.8) on the right-hand side:

$$k_1 v_1 = \frac{\eta_1}{\eta_2} k_2 v_2. \quad (\text{B.14})$$

Substituting (B.4) on the right-hand side gives:

$$k_1 v_1 = \frac{\eta_1}{\eta_2} (k_{\text{tot}} - k_1) v_2 . \quad (\text{B.15})$$

Rewriting ((B.15) gives:

$$k_1 = \frac{\eta_1}{\eta_2} (k_{\text{tot}} - k_1) \frac{v_2}{v_1} \Leftrightarrow \quad (\text{B.16})$$

$$\left(1 + \frac{\eta_1 v_2}{\eta_2 v_1}\right) k_1 = \frac{\eta_1}{\eta_2} k_{\text{tot}} \frac{v_2}{v_1} \Leftrightarrow \quad (\text{B.17})$$

$$k_1 = \frac{\frac{\eta_1}{\eta_2} k_{\text{tot}} \frac{v_2}{v_1}}{1 + \frac{\eta_1 v_2}{\eta_2 v_1}} \Leftrightarrow \quad (\text{B.18})$$

$$k_1 = \frac{v_2}{\frac{\eta_2}{\eta_1} v_1 + v_2} k_{\text{tot}} . \quad (\text{B.19})$$

In the last step the nominator and denominator of the right-hand side were both multiplied with $\frac{\eta_2}{\eta_1} v_1$. Once k_1 is known, k_2 is defined by (B.4).

Proof of step 1 Substituting (B.4) into (B.12) and subsequently rewriting gives:

$$K_{\text{tot}} = k_1 + \pi_2^* (k_{\text{tot}} - k_1) \Leftrightarrow \quad (\text{B.20})$$

$$K_{\text{tot}} = -(\pi_2^* - 1) k_1 + \pi_2^* k_{\text{tot}} \Leftrightarrow \quad (\text{B.21})$$

$$(\pi_2^* - 1) k_1 - \pi_2^* k_{\text{tot}} + K_{\text{tot}} = 0 . \quad (\text{B.22})$$

Substituting (B.19) and subsequently rewriting gives:

$$(\pi_2^* - 1) \frac{v_2}{\frac{\eta_2}{\eta_1} v_1 + v_2} k_{\text{tot}} - \pi_2^* k_{\text{tot}} + K_{\text{tot}} = 0 \Leftrightarrow \quad (\text{B.23})$$

$$\left[(\pi_2^* - 1) \frac{v_2}{\frac{\eta_2}{\eta_1} v_1 + v_2} - \pi_2^* \right] k_{\text{tot}} + K_{\text{tot}} = 0 . \quad (\text{B.24})$$

Multiply both sides with $\frac{\eta_2}{\eta_1} v_1 + v_2$:

$$\left[(\pi_2^* - 1) v_2 - \pi_2^* \left(\frac{\eta_2}{\eta_1} v_1 + v_2 \right) \right] k_{\text{tot}} + K_{\text{tot}} \left(\frac{\eta_2}{\eta_1} v_1 + v_2 \right) = 0 . \quad (\text{B.25})$$

Rearrangement gives:

$$\left[\pi_2^* v_2 - v_2 - \pi_2^* \frac{\eta_2}{\eta_1} v_1 - \pi_2^* v_2 \right] k_{\text{tot}} + \left[\frac{\eta_2}{\eta_1} v_1 + v_2 \right] K_{\text{tot}} = 0 \Rightarrow \quad (\text{B.26})$$

$$\left[\frac{\eta_2}{\eta_1} v_1 + v_2 \right] K_{\text{tot}} - \left[\frac{\eta_2}{\eta_1} \pi_2^* v_1 + v_2 \right] k_{\text{tot}} = 0 . \quad (\text{B.27})$$

Rewriting the pce function as a function of the effective density by substituting (B.9) and (B.10) in (B.11) and defining the new function gives:

$$\pi_2^* = \pi_2 (v_1^{\text{FD}}(K_{\text{tot}}), v_2^{\text{FD}}(K_{\text{tot}})) , \quad (\text{B.28})$$

$$\bar{\pi}_2(K_{\text{tot}}) \stackrel{\text{def}}{=} \pi_2 (v_1^{\text{FD}}(K_{\text{tot}}), v_2^{\text{FD}}(K_{\text{tot}})) . \quad (\text{B.29})$$

This new function expresses the pce value as a function of the effective density.

Substituting (B.29) and (B.9) and (B.10) into (B.27) gives:

$$\left[\frac{\eta_2}{\eta_1} v_1^{\text{FD}}(K_{\text{tot}}) + v_2^{\text{FD}}(K_{\text{tot}}) \right] K_{\text{tot}} - \left[\frac{\eta_2}{\eta_1} \bar{\pi}_2(K_{\text{tot}}) v_1^{\text{FD}}(K_{\text{tot}}) + v_2^{\text{FD}}(K_{\text{tot}}) \right] k_{\text{tot}} = 0 . \quad (\text{B.30})$$

The only unknown in (B.30) is the effective density K_{tot} .

Summary

The increase of mobility of the past decades has led to substantial congestion on the freeways. Traffic jams emerge both on a daily basis at the same location, as well as during accidents when a part of the freeways is temporarily blocked. In those cases, traffic management centers intervene into traffic in order to reduce or even dissolve congestion. This is called Dynamic Traffic Management (DTM). Common DTM measures are rerouting traffic, ramp metering at the on-ramps of freeways and opening peak-hour lanes.

DTM is performed in two steps. First, the current traffic state is estimated by fusing sensor data, usually from dual-inductive loops or from in-car sensors. The traffic state describes where the traffic is located in the network and how fast it travels. Second, based on the current traffic state, a controller determines appropriate actions for each DTM measure in order to improve the traffic performance.

In current practice, DTM controls traffic as a whole, not differentiating between the different vehicle classes. However, vehicles can be classified according to length, maximum speed or value of time. The vehicle classes therefore have different effects on the network performance. For example, short vehicles can travel with a shorter time headway than longer ones. Consequently, more short vehicles can pass any given location than long one. The capacity for shorter vehicles is therefore larger than for long ones.

In this thesis, Dynamic Traffic Management is generalized to take the properties of different vehicle classes into account. The effects of the vehicle classes on the traffic flow and the network performance are analyzed based on the macroscopic multi-class traffic flow model Fastlane. Furthermore, existing DTM measures are generalized in order to control traffic vehicle-class specifically. A multi-class ramp meter is developed that is able to meter each vehicle class individually. Prioritizing short vehicles increases the network throughput; conversely, prioritizing valuable vehicles decreases the total cost. Multi-class route guidance enables the routing of a vehicle class around a congested area. At bottleneck locations, a class-specific lane makes it possible to keep a specified vehicle class in free-flow.

In order to apply DTM in real-time, two existing traffic state estimators are analyzed and reformulated so that they now estimate the traffic state of realistically-size freeways within a few seconds. The Adaptive Smoothing Method is reformulated to use

the Fast Fourier Transform. The Extended Kalman Filter technique is localized so that measurements are used to correct the system state only in the vicinity of the measurement instead of correcting the state of the whole network. Furthermore, a tool is developed that extracts the position and speed of shock waves from spatiotemporal traffic data, which are used to calibrate traffic state estimators or traffic flow models.

The developed components of estimation and control are then combined in a case study to optimize class-specific control advices for the Dutch freeway A15 near the harbor of Rotterdam. The case study shows that multi-class DTM improves the network performance compared to conventional, mixed-class DTM. Currently, those control advices are calculated continuously and are published at a website in real-time. In addition, the current traffic state, the future traffic for the next hour without control, and the future traffic with multi-class control are published.

Samenvatting

De mobiliteitstoename in de afgelopen decennia heeft tot veel congestie op de snelwegen geleid. Files ontstaan zowel dagelijks op dezelfde locatie als bij ongelukken waarbij een gedeelte van de weg tijdelijk geblokkeerd is. In zulke gevallen grijpen verkeersmanagementscentra in om de congestie te verminderen of zelfs op te lossen. Dit wordt Dynamisch Verkeersmanagement (DVM) genoemd. Veel gebruikte DVM-maatregelen zijn het herrouteren van verkeer, toeritdosering bij toeritten van snelwegen en het openen van spitsstroken.

DVM bestaat uit twee stappen. Ten eerste wordt de huidige verkeerstoestand geschat door het fuseren van data, meestal afkomstig van inductielussen en van sensoren in voertuigen. De verkeerstoestand beschrijft waar het verkeer zich in het netwerk bevindt en hoe snel het reist. Ten tweede bepaalt een regelaar gepaste acties voor iedere DVM-maatregel. Dit gebeurt op basis van de huidige verkeerstoestand en met als doel om de prestatie van het verkeer te verbeteren.

In de huidige praktijk wordt het verkeer als geheel geregeld door DVM-maatregelen, zonder rekening te houden met de verschillende voertuigklassen. Echter, voertuigen kunnen geclassificeerd worden volgens hun lengte, maximum snelheid en tijdswaarde. De voertuigklassen hebben daarom verschillende effecten op de netwerkprestatie. Er kunnen bijvoorbeeld meer korte voertuigen voorbij een bepaalde locatie rijden dan lange. De capaciteit voor korte voertuigen is daarom groter dan voor lange voertuigen.

In dit proefschrift wordt Dynamisch Verkeersmanagement gegeneraliseerd om rekening te houden met de eigenschappen van verschillende voertuigklassen. De effecten van de voertuigklassen op de verkeersstroom en de netwerkprestatie worden geanalyseerd op basis van Fastlane, een macroscopisch verkeerstrommodel met meerdere klassen. Bovendien worden bestaande DVM-maatregelen gegeneraliseerd om het verkeer klasse-specifiek te regelen. Een toeritdosering wordt ontwikkeld voor meerdere klassen die in staat is om iedere klasse apart te regelen. Het prioriteren van korte voertuigen verhoogt de netwerkdoorstroom. Anderzijds, het prioriteren van waardevolle voertuigen verlaagt de totale kosten. Routegeleiding voor meerdere klassen maakt het mogelijk om een bepaalde voertuigklasse om een gebied met congestie te leiden. Op flessenhalslocaties zorgt een strook voor een specifieke klasse er voor dat deze klasse in vrije afwikkeling blijft.

Om DVM in real-time toe te passen, worden twee bestaande verkeerstoestandschatters

geanalyseerd en geherformuleerd zodat ze nu de verkeerstoestand van een snelweg van realistische afmetingen binnen een paar seconden schatten. De Adaptive Smoothing Method word geherformuleerd om gebruik te maken van snelle Fouriertransformatie. De Extended Kalman Filter-techniek wordt plaatsgebonden gemaakt zodat metingen alleen gebruikt worden om de systeemtoestand te corrigeren in de nabijheid van de meting in plaats van het corrigeren van de toestand van het gehele netwerk. Bovendien wordt er een gereedschap ontwikkeld die de positie en snelheid van schokgolven uit ruimte-tijd-verkeersdata haalt. De positie en snelheid worden gebruikt om verkeers-toestandschatters en verkeerstroomodellen te kalibreren.

De ontwikkelde componenten van schatting en regeling worden dan gecombineerd in een gevalsanalyse voor het optimaliseren van de klasse specifieke-regeladviezen voor de Nederlandse snelweg A15 bij de haven van Rotterdam. De gevalsanalyse laat zien dat DVM voor meerdere klassen de netwerkprestatie verbetert in vergelijking met conventionele DVM voor gemengde klassen. Momenteel worden deze adviezen continu berekend en in real-time gepubliceerd op een website. Bovendien worden de huidige verkeerstoestand, de toekomstige verkeerstoestand voor het komende uur zonder regeling en de toekomstige verkeerstoestand met regeling voor meerdere klassen gepubliceerd.

Zusammenfassung

Die starke Zunahme der Mobilität führte dazu, dass heutzutage viele Autobahnen von Staus betroffen sind. Diese Staus treten sowohl regelmäßig jeden Tag an der gleichen Stelle auf, als auch unregelmäßig an verschiedenen Stellen wie sie zum Beispiel durch Unfälle verursacht werden. Verkehrszentralen greifen in diesen Fällen in den Verkehr ein, um die auftretenden Staus aufzulösen oder zumindest zu verkleinern. Dies wird als Dynamisches Verkehrsmanagement (DVM) bezeichnet. Übliche DVM-Maßnahmen umfassen das Umleiten des Verkehrs, das Einschränken des Zuflusses an den Autobahnauffahrten oder das Öffnen eines zusätzlichen Fahrstreifens.

DVM erfolgt in zwei Schritten. Zuerst wird der aktuelle Verkehrszustand auf Grund von Sensordaten geschätzt, die üblicherweise von Induktionsschleifen oder von den Fahrzeugen selbst gewonnen werden. Der Verkehrszustand beschreibt die Position des Verkehrs im Netzwerk und seine aktuelle Geschwindigkeit. Danach werden über eine Regelung auf Basis des aktuell geschätzten Verkehrszustands geeignete DVM-Maßnahmen ausgewählt um die Verkehrslage zu verbessern.

Momentan bezieht sich diese Regelung auf den gesamten Verkehr, ohne zwischen verschiedenen Fahrzeugklassen zu unterscheiden. Fahrzeuge können jedoch nach Länge, Maximalgeschwindigkeit oder Zeitwert unterschieden werden und haben daher unterschiedliche Einflüsse auf die Netzwerkeffizienz. Kurze Fahrzeugen können zum Beispiel mit einem kürzeren zeitlichen Abstand fahren. Bei gleichen Umständen können daher mehr kleine Autos eine Stelle passieren als große. Die Kapazität für kleine Fahrzeuge ist daher höher als für große.

In dieser Dissertation wird Dynamisches Verkehrsmanagement für mehrere Fahrzeugklassen verallgemeinert. Unter anderem werden die Effekte der Fahrzeugeigenschaften auf den Verkehrsfluss untersucht. Dazu wird das makroskopische Verkehrsflussmodell Fastlane herangezogen, das zwischen mehreren Fahrzeugklassen unterscheidet. Weiterhin werden bestehende DVM-Maßnahmen verallgemeinert um den Verkehr klassenspezifisch zu regeln. Die hier entwickelte klassenspezifische Zuflusskontrolle ermöglicht, dass Fahrzeuge je nach ihrer Klasse unterschiedlich starken Zufluss zur Autobahn bekommen. Das Priorisieren von kurzen Fahrzeugen erhöht den Netzwerkdurchsatz. Umgekehrt vermindert das Priorisieren von wertvollen Fahrzeugen die totalen Kosten. Klassenspezifische Umleitung ermöglicht das Umleiten einer Fahrzeugklasse um einen Stau. Klassenspezifische Fahrstreifen ermöglichen hohe Geschwindigkeiten für eine Fahrzeugklasse.

Um DVM möglichst schnell ausführen zu können, werden außerdem zwei bestehende Methoden der Verkehrszustandsschätzung analysiert und so umformuliert, dass sie den Zustand von Autobahnenabschnitten realistischer Größe innerhalb von wenigen Sekunden berechnen können. Die Adaptive Smoothing Method wird so umformuliert, sodass die Schnelle Fourier-Transformation benutzt werden kann. Das Extended Kalman-Filter wird lokalisiert, sodass die Messdaten dazu benutzt werden um nur den Verkehrszustand in der Nähe des Sensors korrigiert wird, anstatt des Zustands des gesamten Netzwerks. Weiterhin wird eine Methode entwickelt, die die Position und die Geschwindigkeit von Schockwellen auf Basis von raumzeitlichen Verkehrsdaten schätzt. Diese Merkmale werden unter anderem benutzt, um Zustandsschätzer oder Verkehrsflussmodelle zu kalibrieren.

In einer Fallstudie werden die beiden hier entwickelten Elemente Schätzung und Regelung kombiniert, um fahrzeugklassenspezifische Regelsignale für die Regelung der niederländischen Autobahn A15 in der Nähe des Rotterdamer Hafens zu optimieren. Die Fallstudie zeigt, dass klassenspezifische Regelung die Kosten des Verkehrs im Vergleich zu konventioneller, nicht-klassenspezifischer Regelung verringert. Zur Zeit werden diese Regelsignale kontinuierlich berechnet und auf einer Webseite in Echtzeit veröffentlicht. Zu sehen sind außerdem der aktuelle Verkehrszustand der Autobahn, der erwartete Zustand in einer Stunde ohne Regelung, und der erwartete Zustand mit Regelung.

TRAIL Thesis Series

The following list contains the most recent dissertations in the TRAIL Thesis Series. For a complete overview of more than 100 titles see the TRAIL website: www.rsTRAIL.nl.

The TRAIL Thesis Series is a series of the Netherlands TRAIL Research School on transport, infrastructure and logistics.

Wageningen-Kessels, F.L.M. van, *Multi-class continuum traffic flow models: Analysis and simulation methods*, T2013/7, March 2013, TRAIL Thesis Series, the Netherlands

Taneja, P., *The Flexible Port*, T2013/6, March 2013, TRAIL Thesis Series, the Netherlands

Yuan, Y., *Lagrangian Multi-Class Traffic State Estimation*, T2013/5, March 2013, TRAIL Thesis Series, the Netherlands

Schreiter, T., *Vehicle-Class Specific Control of Freeway Traffic*, T2013/4, March 2013, TRAIL Thesis Series, the Netherlands

Zaerpour, N., *Efficient Management of Compact Storage Systems*, T2013/3, February 2013, TRAIL Thesis Series, the Netherlands

Huibregtse, O.L., *Robust Model-Based Optimization of Evacuation Guidance*, T2013/2, February 2013, TRAIL Thesis Series, the Netherlands

Fortuijn, L.G.H., *Turborotonde en turboplein: ontwerp, capaciteit en veiligheid*, T2013/1, January 2013, TRAIL Thesis Series, the Netherlands

Gharehgozli, A.H., *Developing New Methods for Efficient Container Stacking Operations*, T2012/7, November 2012, TRAIL Thesis Series, the Netherlands

Duin, R. van, *Logistics Concept Development in Multi-Actor Environments: Aligning stakeholders for successful development of public/private logistics systems by increased awareness of multi-actor objectives and perceptions*, T2012/6, October 2012, TRAIL Thesis Series, the Netherlands

Dicke-Ogenia, M., *Psychological Aspects of Travel Information Presentation: A psychological and ergonomic view on travellers' response to travel information*, T2012/5, October 2012, TRAIL Thesis Series, the Netherlands

Wismans, L.J.J., *Towards Sustainable Dynamic Traffic Management*, T2012/4, September 2012, TRAIL Thesis Series, the Netherlands

Hoogendoorn, R.G., *Empirical Research and Modeling of Longitudinal Driving Behavior under Adverse Conditions*, T2012/3, July 2012, TRAIL Thesis Series, the Netherlands

Carmona Benitez, R., *The Design of a Large Scale Airline Network*, T2012/2, June 2012, TRAIL Thesis Series, the Netherlands

Schaap, T.W., *Driving Behaviour in Unexpected Situations: a study into the effects of drivers compensation behaviour to safety-critical situations and the effects of mental workload, event urgency and task prioritization*, T2012/1, February 2012, TRAIL Thesis Series, the Netherlands

Muizelaar, T.J., *Non-recurrent Traffic Situations and Traffic Information: determining preferences and effects on route choice*, T2011/16, December 2011, TRAIL Thesis Series, the Netherlands

Cantarelli, C.C., *Cost Overruns in Large-Scale Transportation Infrastructure Projects: a theoretical and empirical exploration for the Netherlands and Worldwide*, T2011/15, November 2011, TRAIL Thesis Series, the Netherlands

Vlies, A.V. van der, *Rail Transport Risks and Urban Planning: solving deadlock situations between urban planning and rail transport of hazardous materials in the Netherlands*, T2011/14, October 2011, TRAIL Thesis Series, the Netherlands

Pas, J.W.G.M. van der, *Clearing the Road for ISA Implementation? Applying adaptive policymaking for the implementation of intelligent speed adaptation*, T2011/13, October 2011, TRAIL Thesis Series, the Netherlands

Zegeye, S.K., *Model-Based Traffic Control for Sustainable Mobility*, T2011/12, October 2011, TRAIL Thesis Series, the Netherlands

Mahr, T., *Vehicle Routing under Uncertainty*, T2011/11, September 2011, TRAIL Thesis Series, the Netherlands

Pel, A.J., *Transportation Modelling for Large-scale Evacuations*, T2011/10, July 2011, TRAIL Thesis Series, the Netherlands

Zheng, F., *Modelling Urban Travel Times*, T2011/9, July 2011, TRAIL Thesis Series, the Netherlands

Vlassenroot, S.H.M., *The Acceptability of In-vehicle Intelligent Speed Assistance (ISA) Systems: from trial support to public support*, T2011/8, June 2011, TRAIL Thesis Series, the Netherlands

Kroesen, M., *Human Response to Aircraft Noise*, T2011/7, June 2011, TRAIL Thesis Series, the Netherlands

About the Author

Curriculum Vitae



Thomas Schreiter was born in Zwickau, Germany, in 1981. He moved to Karlsruhe, Germany, in 2002 to study computer science with the focus on telematics, theoretical computer science and transportation at Karlsruhe University of Technology (now Karlsruhe Institute of Technology). He was awarded his diploma in 2008 and was among the top ten students.

In December 2008, Thomas started the PhD program at Transport & Planning, faculty of Civil Engineering and Geosciences of Delft University of Technology, the Netherlands. The PhD project “The MultiModal Port Traffic Centre” was funded by the Harbor of Rotterdam (Havenbedrijf Rotterdam) and the Traffic Management Company (Verkeersonderneming). Its goal was to assess ways to make the port more accessible by controlling the traffic of the A15 corridor and the underlying network. During the program, Thomas organized the Trail course “Modeling Traffic Flow Phenomena,” published parts of the work in conference proceedings and journals, and supervised several master’s students during their final project.

Currently, Thomas is a researcher at the University of California at Berkeley on the topic of estimating, predicting and controlling of freeway and arterial traffic flow.

Publications

Journal Articles

Schreiter, Thomas; Landman, Ramon; Van Lint, Hans; Hegyi, Andreas; Hoogendoorn, Serge: *Vehicle-class Specific Route-guidance of Freeway Traffic by Model-predictive Control*, Transportation Research Record (TRR), (accepted)

Landman, Ramon; Schreiter, Thomas; Hegyi, Andreas; Van Lint, Hans; Hoogendoorn, Serge: *Policy based service level-oriented route guidance in road networks a comparison with system optimal route guidance*, Transportation Research Record (TRR), 2012

Van Hinsbergen, Chris, Schreiter, Thomas; Zuurbier, Frank; Van Lint, Hans; Hoogendoorn, Serge: *The Localized Extended Kalman Filter for Scalable, Real-Time Traffic State Estimation*, Journal of IEEE Transactions on Intelligent Transport Systems (T-ITS), 2012

Peer-reviewed Conference Contributions

Schreiter, Thomas; Van Lint, Hans; Hoogendoorn, Serge: *Vehicle-class Specific Control of Freeway Traffic*, 92nd Annual Meeting of the Transportation Research Board (TRB), Washington, D.C., 2013

Schreiter, Thomas; Van Wageningen-Kessels, Femke; Yuan, Yufei, Van Lint, Hans; Hoogendoorn, Serge: *Fastlane – Traffic flow modeling and multi-class dynamic traffic management*, 12th International Congress of the Research School Transportation, Infrastructure and Logistics (TRAIL), Rotterdam, The Netherlands, 2012

Schreiter, Thomas; Van Lint, Hans; Hoogendoorn, Serge; Muhurdarevic, Zlatan: *Een generieke aanpak voor DVM – Proactief regelen van de snelweg A15 in het havengebied met BOS-HbR*, Nationaal Verkeerskundecongres (NVC), Nieuwegein, The Netherlands, 2012

Schreiter, Thomas; Pel, Adam; Van Lint, Hans; Hegyi, Andreas; Hoogendoorn, Serge: *Modeling Monetary Costs of Multi-class Traffic Flow – Application to the Dynamic Management of Truck Lanes*, IEEE Intelligent Transport Systems Conference (ITSC), Anchorage, AK, 2012

Schreiter, Thomas; Landman, Ramon; Van Lint, Hans; Hegyi, Andreas; Hoogendoorn, Serge: *Vehicle-class Specific Route-guidance of Freeway Traffic by Model-predictive Control*, Proceedings of the 91st Annual Meeting of the Transportation Research Board (TRB), Washington, D.C., 2012

- Landman, Ramon; Schreiter, Thomas; Hegyi, Andreas; Van Lint, Hans; Hoogendoorn, Serge: *Policy based service level-oriented route guidance in road networks – a comparison with system optimal route guidance*, Proceedings of the 91st Annual Meeting of the Transportation Research Board (TRB), Washington, D.C., 2012
- Schreiter, Thomas; Van Lint, Hans; Hoogendoorn Serge: *Multi-class Ramp Metering: Concepts and Initial Results*, Proceedings of IEEE Intelligent Transport Systems Conference (ITSC), Washington, D.C., 2011
- Yuan, Yufei; Van Lint, Hans; Hoogendoorn, Serge; Vrancken, Jos; Schreiter, Thomas: *Freeway Traffic State Estimation Using Extended Kalman Filter for First-Order Traffic Model in Lagrangian Coordinates*, Proceedings of IEEE International Conference on Network, Sensing and Control (ICNSC), Delft, The Netherlands, 2011
- Schreiter, Thomas; Van Lint, Hans; Hoogendoorn, Serge: *A Real-time Multi-class Traffic State Prediction Framework for Incident Management on Highways*, Models and Technologies for Intelligent Transport Systems, Leuven, Belgium, 2011
- Van Wageningen-Kessels, Femke; Schreiter, Thomas; Van Lint, Hans; Hoogendoorn, Serge: *Modeling Traffic Phenomena*, Proceeding of 2nd International Conference on Models and Technologies for Intelligent Transport Systems (MT-ITS), Leuven, Belgium, 2011
- Schreiter, Thomas; Muhurdarevic, Zlatan; Scheerder, Ernst; Van Lint, Hans; Hoogendoorn, Serge: *Proactieve Regeling op de A15 met een Voorspellend Verkeersafwikkelingsmodel*, Proceedings of the 9e Symposium Dynamisch Verkeersmanagement, Rotterdam, The Netherlands, 2011
- Schreiter, Thomas; Smits, Erik-Sander; Van Lint, Hans; Hoogendoorn, Serge: *The Cell-Transmission Model with Capacity Drop*, Proceedings of the 11th International Congress of the Research School Transportation, Infrastructure and Logistics (TRAIL), Rotterdam, The Netherlands, 2010
- Schreiter, Thomas; Van Lint, Hans; Treiber, Martin; Hoogendoorn, Serge: *Two Fast Implementations of the Adaptive Smoothing Method used in Highway Traffic State Estimation*, Proceedings of IEEE Intelligent Transport Systems Conference (ITSC), Madeira, Portugal, 2010
- Van Hinsbergen, Chris; Schreiter, Thomas; Zuurbier, Frank; Van Lint, Hans; Hoogendoorn, Serge: *Fast Traffic State Estimation with the Localized Extended Kalman Filter*, Proceedings of IEEE Intelligent Transport Systems Conference (ITSC), Madeira, Portugal, 2010
- Schreiter, Thomas; Van Lint, Hans; van Hinsbergen, Chris; Zuurbier, Frank; Hoogendoorn, Serge: *Data-Model Synchronization in Online Traffic State Estimation*, Proceedings of Traffic Flow Theory and Characteristics Committee (AHB45) Summer Meeting of the Transportation Research Board, France, 2010

- Schreiter, Thomas; Yuan, Yufei; Van Lint, Hans; Hoogendoorn, Serge: *The Adaptive Smoothing Method with Spatially Varying Kernels*, Proceedings of the Seventh Triennial Symposium on Transportation Analysis (Tristan7), Tromsø, Norway, 2010
- Van Hinsbergen, Chris; Schreiter, Thomas; Van Lint, Hans; Hoogendoorn, Serge; Van Zuylen, Henk: *Online estimation of Kalman Filter parameters for traffic state estimation*, Proceedings of the Seventh Triennial Symposium on Transportation Analysis (Tristan7), Tromsø, Norway, 2010
- Yuan, Yufei; Van Lint, Hans; Schreiter, Thomas; Hoogendoorn, Serge and Vrancken, Jos: *Automatic Speed-Bias Correction with Flow-Density Relationships*, Proceedings of the IEEE International Conference on Networking, Sensing and Control (ICNSC), Chicago, IL, 2010
- Schreiter, Thomas; Van Lint, Hans; Yuan, Yufei; Hoogendoorn, Serge: *Propagation Wave Speed Estimation of Freeway Traffic with Image Processing Tools*, Proceedings of the 89th Annual Meeting of the Transportation Research Board (TRB), Washington, D.C., 2010
- Weissel, Florian; Schreiter, Thomas; Huber, Marco; Hanebeck, Uwe: *Stochastic Model Predictive Control of Time-Variant Nonlinear Systems with Imperfect State Information*, Proceedings of IEEE Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), Seoul, Korea, 2008

Technical Report

- Van Lint, Hans; Schreiter, Thomas; Yuan, Yufei: *Technische en functionele haalbaarheid CHECK-algoritme voor de productie van statistische verkeersgegevens en indicatoren*, Technische Universiteit Delft, The Netherlands, 2009

Propositions

Pertaining to the dissertation
Vehicle-class Specific Control of Freeway Traffic

Thomas Schreiter

18 March 2013

1. Congestion on freeways is reduced by extending Dynamic Traffic Management with user-class specific measures.
2. The Adaptive Smoothing Method is more suitable than Kalman Filtering for Dynamic Traffic Management applications. (Chapter 4)
3. The more car traffic there is, the more a truck lane reduces the total cost of traffic. (Chapter 6)
4. Mobile phones and social networks play a key role in reducing congestion.
5. If one thinks a program is bug-free, then one has not analyzed the code long enough.
6. Affirmative action is unjust.
7. Laziness in combination with foresight leads to prosperity.
8. In Europe “freedom” is interpreted as safety and equality, whereas in the United States “freedom” is interpreted as the ability to act without restriction.
9. The writing of a thesis takes longer than expected, independent of the current point of progress.
10. The lingua franca of the world is not English. The lingua franca of the world is bad English.

These propositions are considered opposable and defensible and have been approved as such by the promotor Prof. dr. ir. S.P. Hoogendoorn.

Stellingen

Behorend bij het proefschrift
Vehicle-class Specific Control of Freeway Traffic

Thomas Schreiter

18 Maart 2013

1. Congestie op snelwegen wordt verminderd door het uitbreiden van Dynamisch Verkeersmanagement met klassespecifieke maatregelen.
2. De Adaptive Smoothing Method is beter geschikt dan Kalmanfiltertechnieken voor Dynamisch Verkeersmanagementtoepassingen. (Chapter 4)
3. Hoe meer autoverkeer, hoe meer een vrachtwagenrijstrook de totale verkeerskosten vermindert. (Chapter 6)
4. Mobiele telefoons en sociale netwerken spelen een sleutelrol in het verminderen van congestie.
5. Als je denkt dat een programma geen bugs bevat, dan heb je de code niet lang genoeg geanalyseerd.
6. Positieve discriminatie is onrechtvaardig.
7. Luiheid in combinatie met vooruitziendheid leidt tot voorspoed.
8. In Europa wordt “vrijheid” geïnterpreteerd als veiligheid en gelijkheid, daarentegen wordt “vrijheid” in de Verenigde Staten geïnterpreteerd als de mogelijkheid zonder beperking te handelen.
9. Het schrijven van een proefschrift duurt langer dan verwacht, onafhankelijk van de huidige voortgang.
10. De lingua franca van de wereld is niet Engels. De lingua franca van de wereld is slecht Engels.

Deze stellingen worden oponeerbaar en verdedigbaar geacht en zijn als zodanig goedgekeurd door de promotor Prof. dr. ir. S.P. Hoogendoorn.