

**Traffic Management and Control  
in Intelligent Vehicle Highway Systems**

L.D. Baskar



# **Traffic Management and Control in Intelligent Vehicle Highway Systems**

## **Proefschrift**

ter verkrijging van de graad van doctor  
aan de Technische Universiteit Delft,  
op gezag van de Rector Magnificus prof. dr. ir. J.T. Fokkema,  
voorzitter van het College van Promoties,  
in het openbaar te verdedigen op woensdag 18 november 2009 om 12.30 uur  
door Lakshmi Dhevi BASKAR,  
master of science in computer science, Otto-von-Guericke Universität Magdeburg,  
geboren te Erode, Tamil Nadu, India.

Dit proefschrift is goedgekeurd door de promotoren:

Prof.dr.ir. J. Hellendoorn

Prof.dr.ir. B. De Schutter

Samenstelling promotiecommissie:

Rector Magnificus

Prof.dr.ir. J. Hellendoorn

Prof.dr.ir. B. De Schutter

Dr. Z. Papp

Prof.dr.ir. S.P. Hoogendoorn

Prof.dr. C. Witteveen

Prof.dr.ir. B. van Arem

Prof.Dr.-Ing. F. Busch

voorzitter

Technische Universiteit Delft, promotor

Technische Universiteit Delft, promotor

TNO Science and Industry, Delft

Technische Universiteit Delft

Technische Universiteit Delft

Technische Universiteit Delft

Technische Universität München



The research described in this thesis was supported by the VIDI project “Multi-Agent Control of Large-Scale Hybrid Systems” (DWV.6188) of the Dutch Technology Foundation STW and the BSIK project “Transition to Sustainable Mobility (TRANSUMO)”

**TRAIL Thesis Series T2009/12, the Netherlands TRAIL Research School**

P.O. Box 5017

2600 GA Delft, The Netherlands

T: +31 (0) 15 278 6046

T: +31 (0) 15 278 4333

E: info@rstrail.nl

Published and distributed by: L.D. Baskar

E-mail: lakshmibaskar@gmail.com

ISBN 978-90-5584-120-2

Keywords: intelligent vehicle highway systems, intelligent transportation systems, intelligent vehicles, model predictive control

Copyright © 2009 by L.D. Baskar

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

Printed in the Netherlands

# Preface

Hurray, I made it! This is first impression I got when I completed my thesis writing. This was not an easy task for a person who comes from a different educational background. My PhD research involved learning of new technical skills as well discovering myself. I had this research opportunity, which many people as deserving as me did not get. So I worked hard to achieve it. I owe a lot of thanks to many people who made this research career, first possible and also an experience to hold on.

It is usual to say that I thank my supervisors Hans and Bart for believing in me and for their support, guidance, positively told criticisms, and reviews provided during my PhD. But for me, this is not just meant to be in words. Apart from motivating me, stream lining the ideas to carry out the research in a proper direction, I thank Hans for his simplicity and his easily approachable character. Hans never forgets to appreciate the person for what they have done, for instance, even if it is a presentation or after every thesis chapter is completed. Hans makes a point to ask how my research is going as well as to ask how my vacation was.

I am really grateful to my daily supervisor Bart for his persistent support during both bad and good times of my profession and in my personal life. Bart, I thank you for making me comfortable by stating that the research is a team work. Though he is occupied the whole day, Bart has always time to answer my doubts, even at late nights. All I can say is, Bart as a supervisor and as an understanding human gave a helping hand in all professional ways to encourage me to do my research. He drops in my office just to ask me if I am in a relaxed mind to carry out the research.

I thank my PhD committee members Dr. Zoltan Papp, Prof. Bart van Arem, Prof. Serge Hoogendoorn, Prof. Cees Witteveen, Prof. Fritz Busch, Prof. Hans Hellendoorn and Prof. Bart De Schutter for their valuable time and suggestions provided to improvise my thesis.

I also thank the students Mernout Burger and Berend de Graaf, whom I supervised during their final work. Their work provided a substantial contribution to this thesis.

My special thanks to my beloved parents Baskaran and Shanthi Baskaran, my brother Santosh, and my best, adorable friends Prasanna in Brussels, and Mani and Shanthi Baskaran in India. Other important personalities are Mrs. Arsha Timmermans, Mrs. Aletta Wubben, and Mrs. Tineke Holkema. I greatly appreciate them for their interests in making me realise the need for a balance in my research and personal life. I take this opportunity to thank my all DCSC colleagues and my friends. In particular, I wish to thank Alina, Amol, Andreas, Dang, Jianfei, Monique, Navin, Ronald, Rudy, Samira, Solomon, Shu, Mrs. Kitty and Mrs. Ellen from DCSC and Mrs. Conchita from TRAIL, and Gayatri, Sampath, Rohini, Raghu, Joe and Abhijeet in Delft.

L.D. Baskar, Delft, November 2009.



# Contents

<b>Preface</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation and goals . . . . .	2
1.2 General overview of the thesis . . . . .	6
1.2.1 Outline of the thesis . . . . .	6
1.2.2 Main contributions . . . . .	8
1.2.3 Road map . . . . .	9
<b>2 Traffic Management and Intelligent Vehicle Highway Systems: State-of-the-Art</b>	<b>11</b>
2.1 Introduction . . . . .	11
2.2 Control design methods . . . . .	13
2.2.1 Static feedback control . . . . .	14
2.2.2 Optimal control and model predictive control . . . . .	16
2.2.3 Artificial intelligence techniques . . . . .	19
2.2.4 Comparison . . . . .	22
2.3 Intelligent vehicles and traffic control . . . . .	24
2.3.1 Intelligent vehicles . . . . .	24
2.3.2 IV-based traffic management . . . . .	25
2.3.3 IV-based control measures . . . . .	26
2.4 Control frameworks and architectures for IVHS . . . . .	27
2.4.1 PATH framework . . . . .	27
2.4.2 Dolphin framework . . . . .	29
2.4.3 Auto21 CDS framework . . . . .	30
2.4.4 CVIS . . . . .	32
2.4.5 SafeSpot . . . . .	32
2.4.6 PReVENT . . . . .	33
2.5 Comparison of the IVHS frameworks . . . . .	34
2.6 Outlook . . . . .	36
2.6.1 Application to IVHS . . . . .	36
2.6.2 Challenges and open issues . . . . .	37
2.7 Summary . . . . .	39

<b>3</b>	<b>A New Traffic Management Framework for IVHS</b>	<b>41</b>
3.1	Introduction . . . . .	41
3.2	IV-based control framework . . . . .	41
3.2.1	Proposed control framework . . . . .	42
3.3	Main improvements and extensions . . . . .	43
3.3.1	Integrated in-vehicle and roadside control measures . . . . .	44
3.4	Contributions of our approach . . . . .	45
3.5	Open problems . . . . .	45
3.6	Summary . . . . .	46
<b>4</b>	<b>Roadside Controller</b>	<b>47</b>
4.1	Introduction . . . . .	47
4.2	A hierarchical framework for IV-based traffic management . . . . .	48
4.3	Model Predictive Control (MPC) . . . . .	50
4.4	MPC for IVHS . . . . .	51
4.4.1	States and control inputs . . . . .	51
4.4.2	Performance criterion and constraints . . . . .	51
4.4.3	Optimisation methods . . . . .	52
4.4.4	Prediction models for IVHS . . . . .	52
4.4.5	Platoon-based prediction model . . . . .	55
4.5	Case study . . . . .	56
4.5.1	Set-up . . . . .	56
4.5.2	Scenario . . . . .	57
4.5.3	Models . . . . .	57
4.5.4	Control problem . . . . .	58
4.5.5	Results and analysis . . . . .	61
4.6	Summary . . . . .	67
<b>5</b>	<b>Area Controller</b>	<b>69</b>
5.1	Introduction . . . . .	69
5.2	Intelligent vehicle highway systems (IVHS) . . . . .	71
5.3	Optimal route choice control in IVHS using mixed-integer linear programming	72
5.3.1	Approach . . . . .	72
5.3.2	Set-up . . . . .	73
5.3.3	Static case with sufficient network capacity . . . . .	74
5.3.4	Static case with queues at the boundaries of the network only . . . . .	74
5.3.5	Dynamic case with queues at the boundaries of the network only . . . . .	75
5.3.6	Dynamic case with queues inside the network . . . . .	77
5.3.7	Approximation based on mixed-integer linear programming . . . . .	79
5.3.8	Case study . . . . .	81
5.4	Optimal routing for IVHS using a macroscopic traffic flow model . . . . .	83
5.4.1	Macroscopic traffic flow characteristics . . . . .	84
5.4.2	Macroscopic traffic flow for IVs . . . . .	85
5.4.3	A METANET-like macroscopic model for IVs . . . . .	86
5.4.4	Model predictive route choice control . . . . .	92
5.4.5	Case study . . . . .	93



---

5.5	Interface between area and roadside controllers . . . . .	95
5.6	Summary . . . . .	96
<b>6</b>	<b>Traffic Management with Semi-Autonomous Intelligent Vehicles</b>	<b>99</b>
6.1	Introduction . . . . .	99
6.2	Preliminaries . . . . .	100
6.2.1	Model Predictive Control (MPC) . . . . .	100
6.2.2	In-vehicle measures . . . . .	101
6.3	MPC for semi-autonomous traffic . . . . .	101
6.3.1	Prediction model . . . . .	101
6.3.2	Performance criteria and constraints . . . . .	104
6.4	Simulation model - Paramics . . . . .	104
6.5	Coupling Paramics and Matlab . . . . .	107
6.5.1	Interface plugin . . . . .	107
6.5.2	ISA plugin . . . . .	109
6.6	Case study . . . . .	111
6.6.1	Set-up . . . . .	111
6.6.2	Scenario . . . . .	112
6.6.3	Models . . . . .	113
6.6.4	Control problem . . . . .	116
6.6.5	Results and analysis . . . . .	116
6.7	Summary . . . . .	122
<b>7</b>	<b>Conclusions and Recommendations</b>	<b>125</b>
7.1	Main contributions . . . . .	125
7.2	Conclusions . . . . .	126
7.2.1	Main conclusions . . . . .	126
7.2.2	Conclusions per chapter . . . . .	126
7.3	Open problems and recommendations for future research . . . . .	128
	<b>Bibliography</b>	<b>133</b>
	<b>Glossary</b>	<b>145</b>
	<b>Samenvatting</b>	<b>149</b>
	<b>Summary</b>	<b>151</b>
	<b>Curriculum vitae</b>	<b>153</b>
	<b>TRAIL Thesis Series</b>	<b>155</b>



# Chapter 1

## Introduction

Most of the problems faced by today's traffic networks are caused by the ever-increasing usage of the traffic system. Traffic congestion is considered to be one of the prominent issues that needs attention. Traffic control and management experts and policy makers have come up with many possible solutions to solve the traffic congestion problem. Some of these solutions focused either on increasing the number of roads or lanes to cope with the demand, or on limiting the traffic demand by levying tolls and raising taxes for using the system. Also, due to political concerns and feasibility constraints, both of these options did not offer a promising solution. Another solution is to use the current system in a more efficient way. This option offers high benefits and potential both on the short term and the long term. This approach is worked out in this thesis, with a particular focus on the long term.

In terms of conventional traffic control approaches, efficient utilisation is made possible by controlling and managing the roadside infrastructure intelligently, which in turn can improve the traffic performance. Currently, this intelligence is introduced in the traffic systems by means of roadside based measures and control handles such as dynamic route guidance panels, ramp metering systems, dynamic speed limits, and also by means of infrastructure equipment such as sensors and actuators. Meanwhile, the other important element in the traffic system — i.e., the vehicles — have become much more intelligent. By this intelligence, we mean that the vehicles are equipped with a number of on-board sensors that help in gathering information such as their position and speed, and with many fast devices that process and present the obtained information in a meaningful and usable form [21]. These techniques can then assist or control the driver actions to sustain a safe and better driving operation.

The traffic management industry thus recognised the importance of the technologies from the fields of telecommunication, control, and information sciences, and decided to put them to use along with the current roadside infrastructure and equipment. This resulted in an Intelligent Vehicle Highway System (IVHS) – a basic medium to distribute the intelligence between vehicles and the roadside infrastructure, and to improve the traffic performance in a reliable and efficient manner. This performance can be expressed in terms of safety, throughput, travel time, fuel emissions, etc.

An IVHS is not a newly developed system, but a next level of traffic control and man-

agement approach that efficiently combines and coordinates systems in both the roadside infrastructure and in the vehicles. In general, IVHS includes roadside-based activities such as controlling and managing the traffic network, and vehicle-based activities such as controlling and steering individual vehicles.

Intelligent Vehicles (IVs) in the IVHS can sense the driving environment using sensors and can provide assistance to the driver (via warnings or advisories) or can take complete control of the vehicle itself to achieve an efficient vehicle operation. These vehicle control systems can thus shift driving tasks such as steering, braking, and throttle control from drivers to the on-board controllers in the vehicles. This complete control of driving tasks leads to an automated driving, which in turn allows the vehicles' activities to be fully controlled by the traffic control and management systems. Such IVs with complete automation also reduce the negative effects of driver delays and errors.

This fully automated handling of IVs allows the vehicles to be arranged in a closely spaced group called platoons. In a platoon, the first vehicle is called a leader and the remaining are followers. In this approach, the vehicles travel as platoons on the highways with high speeds. For safety reasons, larger distances are maintained between the platoons. The vehicles inside the platoon maintain a small safety distance to avoid crashes between them. By travelling at a high speed and with short distances between vehicles, more vehicles can be accommodated on the network, which in turn also yields an improvement in the traffic flow. Although the platooning approach offers a significant improvement in the traffic flow, the implementation of this approach in practice was considered to be impossible in the 1990s. Due to the increasing penetration of advanced in-vehicle technologies in the market and the availability of faster computers, we believe that the difficulties regarding safety, acceptance of automation, and legal issues can be handled in a much better way in course of time, so that ultimately IVHS can actually be implemented and widely deployed in practice.

Till now, much attention has been given to the development of advanced vehicle control systems and to conventional traffic control approaches. Although IVHS contain both roadside infrastructure and IVs, the link that connects these elements has obtained less attention. In this thesis we will address this issue and bridge this gap between the roadside infrastructure and the automated IVs by developing a framework and approach for IVHS-based traffic management and control that integrates roadside-based and IV-based traffic control measures.

## 1.1 Motivation and goals

In this thesis, we will investigate the possibilities to implement a next generation traffic control and management approach. This approach shifts away from a global roadside traffic management to a more vehicle-based and user-specific network-based traffic management. The primary areas of interest that served as a basis to our research involve the following elements:

- IV-based traffic control measures,
- Control structure,
- Control methodology.

Based on this discussion, we will then formulate the objectives of this thesis.

## IV-based traffic control measures

Currently applied traffic control methods influence the traffic system solely using roadside-based control tools. Although these conventional approaches are striving to improve the performance of traffic systems, there is still plenty of room for new contributions.

On the other hand, a slow but steady deployment of IVs into the market has shown their acceptance by the users and their effectiveness in assisting the driving operations in a more safe and robust manner [22, 111, 151]. Some functionalities that currently exist and that are available on the market are Adaptive Cruise Control (ACC) and dynamic route guidance. As more and more vehicle support technologies that are intelligent enough are introduced into the current system, we believe that IV techniques — when chosen and utilised properly along with the conventional traffic control measures such as dynamic speed limits, route guidance, ramp metering, lane closures, etc — have the potential to bring significant improvements to the existing traffic system.

Since we opt to use the IVHS approach, we will work with automated IVs in combination with the platooning approach. Full automation of driving tasks in platoons can enhance the performance of the traffic system. This envisioned improvement in the system performance motivated us to use IV-based control handles and measures for traffic management and control purposes. These IV-based control measures will then act as a control interface between the roadside infrastructure and the platoons of IVs. The roadside traffic controllers will use IV control measures such as route guidance systems, Intelligent Speed Adaptation (ISA) along with the conventional traffic control measures to control platoons of IVs. In particular, every platoon leader is controlled by the on-board controller and the rest of the vehicles in platoons follow their respective leaders. By this approach, the intelligence is distributed between roadside traffic control measures and in-vehicle technologies, and is integrated to provide a roadside-vehicle-specific management rather than just a roadside-infrastructure management. Also, the IV measures should be selected such that their functionality does not counteract with each other and also with the existing conventional roadside control measures. Hence, the new set of all possible control measures should be coordinated so as to serve the same objective.

## Control structure

Having discussed the use of IV-based measures for traffic control and management purposes, the next question that arises is how to develop a control methodology that can handle and coordinate these control measures, and that can be implemented in a large-scale traffic network. To obtain a tractable methodology, we should first select an appropriate control structure to tackle a large-scale traffic problem [4]. In general, appropriateness of any control structure for a large-scale system can be analysed in terms of scalability, computational complexity, communication overhead, and etc. Scalability refers to the ability of the structure to handle the growing size of the problem both in space and in time. The computational complexity is analysed through the amount of memory and time required to solve the problem. Communication overhead is determined by the bandwidth requirements and by the time involved in communications rather than in finding a solution for the problem. We will classify the most commonly used control structures as follows:

- Centralised,

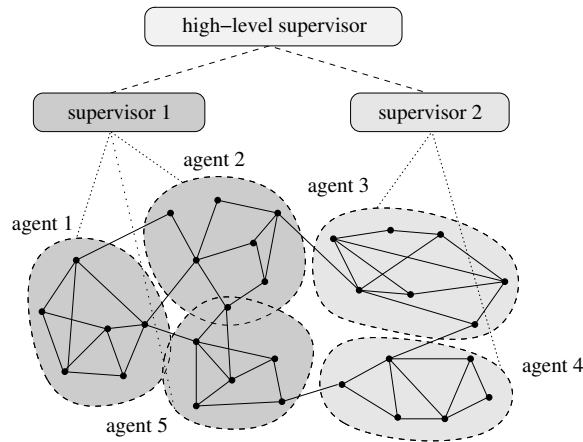


Figure 1.1: General representation of a hierarchical, distributed control structure

- Distributed,
- Hierarchical

Up to now, most control methods are based on a centralised control paradigm. However, in practice, centralised control of large-scale traffic systems is often infeasible due to lack of scalability, computational complexity, and communication overhead. For these reasons, we prefer a distributed approach, which can overcome the shortcomings of the centralised approach [52, 134]. In a distributed approach, the control problem is divided into many subproblems, with each subproblem being managed by a local controller. The local controller is responsible for the problems that occur in its area and it has the complete freedom to choose the ways to solve the problems. Finally, all these subproblem solutions can be summed up to yield the global solution. An additional advantage of a distributed control approach is that it results in a controlled system that is more robust to system failures.

To control a large-scale traffic system, a distributed approach with hierarchical or multi-layered structure can make the task of controlling manageable and can deal with the increasing complexity of the system by adding layers, without changing the complete structure. The key factors such as coordination and system tractability, offered by the hierarchical and distributed control structure motivated us to use this approach for controlling a large-scale traffic system. Such an approach is based on the principle of “divide and conquer” and allows us to split the task of controlling the network into several subproblems (either at the same level (for a distributed approach) or at several levels (for a hierarchical approach)). Thus, by deploying this structure, the activities of the local traffic centres can be coordinated by a supervisory traffic centre and this will result in managing the traffic system in a feasible manner. Such a multi-level control structure with local control agents at the lowest level, and one or more higher supervisory control level, is shown in Figure 1.1.

For illustration purposes, let us apply these approaches to a communication network. Consider a client-server system for watching online movies on the Internet based on a centralised design. As the number of users accessing the web site increases, the server gets overloaded, faces buffering problems, and becomes too busy for further access. Thus the

centralised system cannot serve the users when the demand is large. To solve this, one can make use of peer-to-peer networks, and this network can be viewed as a distributed system. This approach provides a reliable operation even when the number of users that simultaneously access the system increases, as the resource can be found in various peers without relying on a centralised server. In a distributed approach, the failures that occur in one subsystem can be dealt with without causing much problems to the other subsystems in the network that are coupled to the faulty subsystem. Hence, a single point of failure in the system is avoided in the distributed system.

While applying these approaches to control and to manage a large-scale traffic system, a centralised approach will face similar problems as in the client-server system and might produce infeasible solutions. Hence, we prefer to use a distributed control approach. This approach will then have local traffic control centres operating at main cities and these centres are responsible for controlling the traffic activities in their respective areas. However, when a local controller has an unresolved problem, or two traffic control centres have to share the same resources, then the involved control centres should not affect other neighbours with their adverse effects. In these cases, coordination plays a key role to obtain a peaceful and fruitful agreement among the neighbours, and this process can be carried out by a supervisory controller through communications and negotiations.

## Control methodology

Now moving on to the actual control methodology, the control design should be flexible and adaptable to dynamic changes in the demands and working conditions, and in the structure of the system. The changes in the system behaviour can be caused by unexpected events such as incidents or accidents on the highways. Usually, the demand and traffic conditions are time-variant and characterised by a high degree of uncertainty. Hence, static or fixed control strategies usually do not suit our purpose, and an adaptable, robust, and feedback-based control method is required. In traffic control, the process of feedback is considered as an essential property to handle unknown disturbances that act on the system in time and to reduce their negative effects.

Due to platoon-specific management, the possible combinations of control measures that a roadside traffic controller has to consider can be an integer or a real-valued problem or a combination of both. These combinations can make the traffic problem much more complex. In general, the control method should be able to handle this complexity, to deal with multiple objective functions, and to satisfy those constraints that are imposed on the system while determining optimal solutions for the problem. The resulting multi-objective constrained control or optimisation problem should result in optimal values for and coordination of the selected control measures.

All these requirements lead us to choose a control technique that is advanced, robust, adaptive, predictive, and model-based, and that can operate in real-time. Our work is inspired by the work of Papageorgiou and his co-workers [90, 93, 104] and of Bellemans and Hegyi [17, 18, 64, 66, 67], where a model-based approach such as Model-based Predictive Control (MPC) is used for traffic control and management. The traffic controller uses MPC for predicting the future behaviour of the system due to external inputs and demands, and for predicting the effects of various control measures on the system. The underlying concept of the MPC controller is based on on-line optimisation and uses an explicit

prediction model to obtain the optimal actions for the control measures subject to system dynamics and constraints. However, this technique also comes with its own advantages and disadvantages: The advantage is that the MPC is a feedback control algorithm which can handle constrained, complex dynamical systems. However, since it is a model-based approach, certain criteria on the model need to be satisfied. The model that will be used for prediction purposes should be fast, does not need to represent fine details of the system, but should reproduce the traffic evolution with a sufficient degree of accuracy. We will discuss these issues in more detail in the next chapters.

## Goals

The goals of this thesis are to design a framework to control large-scale traffic networks using a multi-level control structure and to determine a tractable control design methodology for use within this framework. In particular, our framework will be designed to suit a variant of the IVHS setup. The IVHS set-up we consider in addition uses the monitoring and control capabilities offered by automated IVs that support the platooning approach and combines them with *conventional* traffic control measures. Once the framework is designed, we will develop a structured and tractable design methodology for robust control of the IVHS.

Our focus will be on traffic control and management strategies for the IVHS, with special attention at the roadside infrastructure. We are mainly interested in developing ways to design an appropriate distributed, hierarchical control structure for our IVHS, to determine control design methods that can integrate the IV-based techniques along with existing traffic control measures, to adapt the currently used control measures to fit the new IVHS framework, and also to make a selection of control methods that are suitable to be applied at different levels of control. A detailed analysis of the performance and complexity of the methods and of the resulting controllers has to be made using relevant case studies.

Furthermore, we have made some assumptions in our work. We will not deal with legal and user acceptance issues of the new system, and neither with the development of protocols for communication and coordination purposes. We also assume that the automated IVs and the roadside infrastructure are equipped with the necessary communication and control components to provide and respond to the roadside-IV control measures. Also, we assume that the vehicle and platoon controllers to be present and designed according to literature [77, 139, 147].

## 1.2 General overview of the thesis

In this section, we give an overview of the topics that will be dealt with in each chapter of this thesis. In particular, Section 1.2.1 provides a brief summary of the contents of each chapter, and briefly discusses the contributions made by this thesis to the state-of-the-art. Next, the road map in Section 1.2.3 guides the readers to understand the structure of the thesis.

### 1.2.1 Outline of the thesis

In this thesis, we propose a multi-level framework for IVHS-based traffic management and control. In particular, we will implement IV-based control design methods at different levels



of the roadside infrastructure in our new IVHS-framework and analyse their potential contribution to improve the performance of the system. This thesis is organised as follows:

- **Chapter 2** deals with two main streams of topics. First, we give an overview of control design methodologies that are most often used to control the traffic on highways. Most of the control methods that have been presented in the literature deal with vehicles that are driven by humans. On the basis of this survey, we make a comparative analysis of these methodologies and also sketch how these methods could fit in an IVHS-based traffic control framework. Next, we discuss intelligent vehicles and describe how the IV-based control measures and handles can be used to improve the traffic flow. Subsequently, we discuss various existing traffic management architectures for IVHS such as PATH, Dolphin, Auto21 CDS, CVIS, SafeSpot, and PREVENT. As a result of the survey on IVs and IVHS-frameworks, we provide a qualitative comparison of these frameworks based on their strong and weak points. The last section of the chapter discusses some open issues regarding IVHS as well as the practical challenges in their deployment.
- Subsequently in **Chapter 3**, we introduce a new framework for IVHS that meets our goals stated in Section 1.1 and that considers and combines the strong points of other existing IVHS-based control architectures discussed in Chapter 2. Our new framework is capable of distributing the intelligence between vehicles and roadside infrastructure, and integrates IV-based measures and the conventional roadside traffic control measures. We describe the main features of our new multi-level framework and also sketch the contributions that our approach can provide for the improvement of traffic performance. Our framework also allows vehicle-vehicle and vehicle-roadside communications. Due to the legal and practical acceptance of this system and implementation issues that may be involved, we can say that the newly proposed framework, in general, is a research result aimed at the future.
- In **Chapter 4**, we specifically focus on the roadside controller, which integrates the identified IV-based control measures and conventional traffic control measures, and which assigns traffic control actions based on the platooning approach. The implementation of this controller is the first and crucial step towards the deployment of our framework. The actual control strategy that will be used by the controller is the MPC approach. Each roadside controller will control a stretch of a highway in the network and it will provide optimal control commands such as dynamic speed limits, lane allocations for each platoon on the highway stretch, and release times for the platoons at on-ramps. These commands are then realised by the platoons of automated IVs on the highways by means of Intelligent Speed Adaptation (ISA), Adaptive Cruise Control (ACC), and other necessary sensors and communication protocols. To analyse the potential benefits of the new framework, we compare the proposed approach both for traffic scenarios with human driven IVs and with IV-based platoons.
- Next, we move up one level in the hierarchical IVHS-based traffic control framework. In **Chapter 5**, we deal with the area controller, which controls a set of interconnected highways and which coordinates the activities of various roadside controllers in its area. The area controller is responsible for assigning optimal routes to the platoons in the IVHS. Since in general the optimal routing problem results in a mixed-integer

problem, which is hard to solve, we propose two approximate models (one solely based on flows and queue lengths, and one that is an adaptation of METANET, a macroscopic traffic flow model for human drivers, to the case of platoons of intelligent vehicles). This results in simplified but fast simulation models that — when used for optimal routing — yield mixed-integer *linear* or nonlinear *real-valued* optimisation problems, for both of which efficient solvers exist.

- **Chapter 6** considers an intermediate situation between the current conventional traffic management and system and the fully automated IVHS-based traffic management system of the future. In this chapter, we primarily investigate the possibility of using semi-autonomous vehicles with advanced in-car systems for traffic control. For illustrating such a system, we consider a traffic system with vehicles that are equipped with fully autonomous ACC and advisory ISA. However, in contrast to the system considered in Chapters 3, 4, and 5, these vehicles do not travel in platoons. In this set-up the roadside controller calculates the dynamic speed limits and issues these limits using the speed limit display panels. We use a traffic simulation software package PARAMICS to represent the current traffic system, and we develop plug-ins to program the behaviour of the partially automated vehicles and their react to the control measures. Matlab is chosen as the software environment to implement the roadside controller. Since there is no existing interface between the two software programs, we create a plug-in to connect Matlab and PARAMICS. We then consider various scenarios in which we investigate and compare the effect of control on a traffic system driven by humans only and on a system with semi-automated IVs only.
- Finally, **Chapter 7** summarises the main contributions of this thesis, followed by some suggestions for future research.

Some of the material included in this thesis has already been published. In particular, Chapter 3 is based on [9, 10, 16], Chapter 4 is based on [11–15], and Chapter 6 is based on the MSc thesis of Mernout Burger [29].

### 1.2.2 Main contributions

The contributions of this thesis to the state-of-the-art are:

- The study and analysis of the existing IVHS frameworks. Using a general structure of the IVHS traffic system, and based on the comparative analysis of the control approaches, we also suggest which control methodologies can be used at different levels of the framework.
- The development of a new multi-level IVHS framework that distributes the intelligence between roadside and platoons of automated IVs and that uses IV-based control measures for controlling and managing the traffic system.
- The development of a control approach based on model-based predictive control that can be used by the roadside and area controllers and that allows to integrate and coordinate various IV-based and roadside control measures such as ramp metering, dynamic speed limits, route guidance, etc.

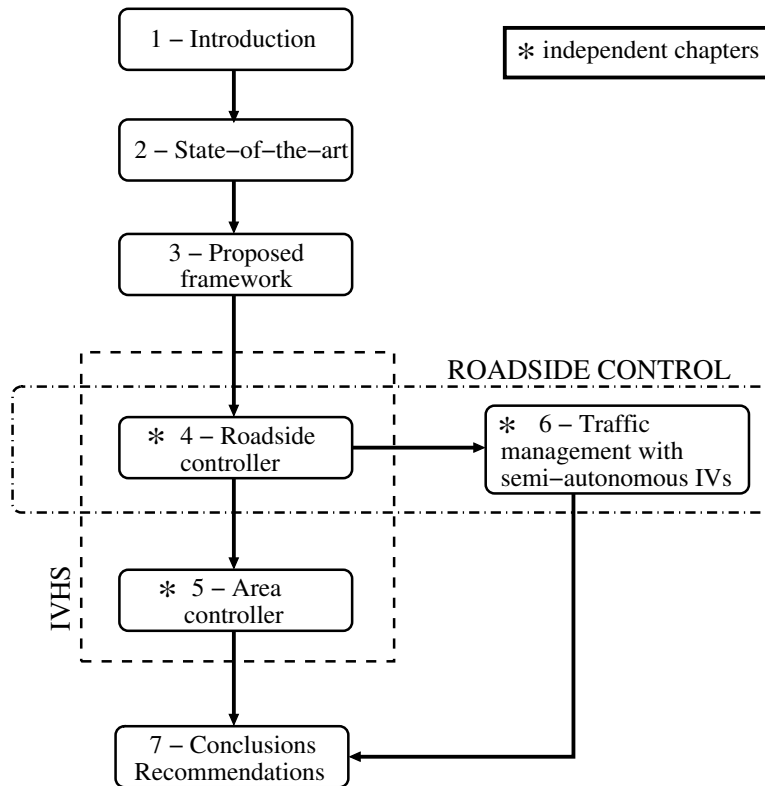


Figure 1.2: A road map for the chapters

### 1.2.3 Road map

Figure 1.2 illustrates the organisation and the relations between the chapters of the thesis and also represents the way in which the chapters can be read. Some of the chapters of the thesis, viz. Chapters 4, 5, and 6 are written in such a way that the reader can read and understand that chapter independent of other chapters. These chapters are self-contained in the sense that they briefly recapitulate all the material of the preceding chapters that is necessary to understand the contents of that particular chapter.

Before starting to read the thesis, we recommend the reader to have a first glance at the basic structure of the thesis. The chapters can be read by following the flowchart shown in Figure 1.2. Chapter 1 gives a general introduction to the topics dealt in this thesis and puts emphasis on the research problem and goals. In Chapter 2 we discuss the state-of-the-art of the conventional control methodologies and the control architectures, particularly for IVHS-based traffic management. We suggest the reader to go through Chapter 3, as this chapter introduces our newly proposed IV-based framework and gives a description of control responsibilities assigned at each level of the proposed framework. The self-contained Chapters 4 and 5 also briefly summarise our newly proposed IVHS-framework and mainly focus on issues related to implementation of the control approaches at the roadside controller and area controller respectively. Chapter 6 is a different genre compared to Chapters 4

and 5, as this chapter shows the possibility of implementing IV systems in the current traffic system. Chapter 7 summarises the contributions of this thesis and gives directions for future research.

## Chapter 2

# Traffic Management and Intelligent Vehicle Highway Systems: State-of-the-Art

Traffic congestion in highway networks is one of the main issues to be addressed by today's traffic management schemes. Automation combined with the increasing market penetration of on-line communication, navigation, and advanced driver assistance systems will ultimately result in intelligent vehicle highway systems (IVHS) that distribute intelligence between roadside infrastructure and vehicles and that — in particular on the longer term — are one of the most promising solutions to the traffic congestion problem. In this chapter, we present a survey on traffic management and control frameworks for IVHS. First, we give a short overview of the main currently used traffic control methods for freeways. Next, we discuss IVHS-based traffic control measures. Then, various traffic management architectures for IVHS such as PATH, Dolphin, Auto21 CDS, etc. are discussed and a comparison of the various frameworks is presented. Finally, we sketch how existing traffic control methodologies could fit in an IVHS-based traffic control set-up.

### 2.1 Introduction

Due to the ever-increasing traffic demand, modern societies with well-planned road management systems, and sufficient infrastructures for transportation still face the problem of traffic congestion. This results in loss of travel time, and huge societal and economic costs. Constructing new roads could be one of the solutions for handling the traffic congestion problem, but it is often less feasible due to political and environmental concerns. An alternative would be to make more efficient use of the existing infrastructure. In this chapter, we will consider the latter approach with a special focus on freeway traffic management and control.

Traffic management and control approaches are used to control the traffic flows and to prevent or reduce traffic jams, or more generally to improve the performance of the traffic system. Possible performance measures in this context are throughput, travel times, safety,

fuel consumption, emissions, reliability, etc. Currently implemented traffic management approaches primarily make use of roadside-based traffic control measures (such as ramp metering, traffic signals, dynamic route information panels, and dynamic speed limits) and infrastructure-based equipment (including sensors and traffic control centres). These measures and the corresponding equipment will be indicated by the term “roadside infrastructure” in the remainder of the chapter.

As an extension to the current traffic control approaches, advanced technologies in the field of communication, control, and information systems have been combined with the existing transportation infrastructure and equipment [80]. This marks the emergence of a next level/next generation of traffic control and management approaches and serves as the motivation for “Intelligent Transportation Systems” (ITS) or “Intelligent Vehicle Highway Systems” (IVHS) [50, 136]. ITS/IVHS incorporate intelligence in both the roadway infrastructure and in the vehicles with the intention of reducing congestion and environmental impact, and of improving traffic performance, by exploiting the distributed nature of the system and by making use of cooperation and coordination between the various vehicles and the various elements of the roadside infrastructure. In the remainder of this chapter, we will use IVHS as a generic word to indicate these systems.

IVHS comprise traffic management systems, driver information systems, and vehicle control systems. In particular, these vehicle control systems are aimed at developing an automated vehicle-highway system that shifts the driver tasks from the driver to the vehicle [149]. These driver tasks include activities such as steering, braking, and making control decisions about speeds and safe headways. Automated Highway Systems (AHS) go one step further than IVHS and involve complete automation of the driving task. For better (network-wide) coordination of traffic activities, AHS also distribute the intelligence between the vehicles and the roadside infrastructure. In this chapter, we will focus on AHS and on the relations and interactions between the vehicles in the AHS and the roadside infrastructure. In particular, we will consider the control aspects of these systems.

An important component of IVHS and AHS are the intelligent vehicles (IVs), which sense the environment around them using sensors (such as radar, lidar, or machine vision techniques) and strive to achieve more efficient vehicle operation either by assisting the driver (via advisories or warnings) or by taking complete control of the vehicle (i.e., partial or full automation) [21]. These IVs also support vehicle-vehicle and vehicle-roadside communication.

Based on the extent to which the roadside and vehicle could work together (without human driver intervention), we can discern different types of AHS [37, 75] as follows:

- Autonomous vehicle systems: Vehicles are equipped with sensors and computers to operate without roadside infrastructure assistance and without coordination with neighbouring vehicles.
- Cooperative vehicle systems: Vehicles use sensors and wireless communication techniques to coordinate their manoeuvres with neighbouring vehicles without any roadside intervention.
- Infrastructure-supported systems: Vehicles communicate with each other and guidelines for decision making purposes are provided by the roadside infrastructure.

- Infrastructure-managed systems: Vehicles indicate their desired actions such as lane changes, exits, and entries to the roadside infrastructure. The roadside system then provides the instructions for inter-vehicle coordination of these manoeuvres.
- Infrastructure-controlled systems: The roadside infrastructure takes entire control of the vehicle operations, monitors the traffic, and optimises the vehicle operations in such a way that the network is utilised as well as possible.

Each of these categories is characterised by its own time scale and the “region” covered, i.e., the number of other vehicles and traffic control measures involved in the process. In this context autonomous and cooperative systems typically operate in the milliseconds to seconds range and cover small groups of neighbouring vehicles. On the other hand, the infrastructure-based systems (i.e., managed and controlled) typically work in the seconds to hours range and involve a much larger number of vehicles and several freeway links or even entire freeway networks.

The objectives of this chapter are twofold. The first objective is to provide a survey of traffic control frameworks for IVHS that integrate the intelligence of both the roadside infrastructure and the IVs to improve the traffic performance. The second objective is to discuss the potential application of control design methods that are currently used for traffic control purposes to IVHS-based traffic management and control systems. More specifically, this chapter is organised as follows. Section 2.2 presents an overview of the current control methods that are applied for traffic control and management. In Section 2.3 we give an overview of intelligent vehicles and IV-based control measures. In Section 2.4 we present existing IVHS frameworks that combine roadside infrastructure and vehicles for efficient traffic management, and in Section 2.5 we provide a comparative analysis of these frameworks. Finally, Section 2.6 briefly sketches how the currently used control design methods presented in Section 2.2 could potentially be applied in these IV-based traffic control frameworks. This section also presents topics for future work.

## 2.2 Control design methods

In the literature different control methodologies have been presented for controlling and managing a traffic network in which vehicles are driven by humans [43, 81, 114]. In this section, we will discuss the control design methodologies for freeway traffic control that are currently most often used in practice such as

- Static feedback control,
- Optimal control and model predictive control,
- Artificial intelligence (AI) techniques.

Throughout this section, we will consider ramp metering as a typical application for freeway traffic control and use it to illustrate the different control methods on a common example. Note, however, that the methods can also be used for other traffic control measures such as variable speed limits, lane closures, shoulder lane openings, etc.

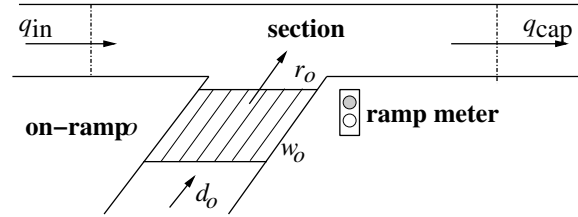


Figure 2.1: Ramp metering

## Ramp metering

Ramp metering [88, 112, 140] is a traffic control measure that can be applied at the on-ramps of freeways and that is basically implemented via traffic signals located at the freeway entrances. The purpose of a ramp metering installation is to regulate the flow of traffic that is allowed to enter the freeway from the on-ramp such that the freeway capacity is utilised efficiently.

For example, let us consider a freeway section with a mainstream origin and an on-ramp origin. The on-ramp origin  $o$  is receiving a traffic demand  $d_o$  (veh/h). Let  $r_o$  (veh/h) be the ramp flow admitted by the traffic signal, let  $q_{cap}$  be the freeway capacity (veh/h), and let  $q_{in}$  be the upstream freeway flow (veh/h). When traffic congestion occurs, the maximum outflow from the traffic jam  $q_{con}$  (veh/h) is usually less than when compared to free-flow traffic  $q_{cap}$ . This phenomenon is called capacity drop [58, 102] and should be prevented whenever possible. If no ramp metering is applied, then there is a possibility for congestion on the freeway due to the on-ramp vehicles in case the demand  $d_o$  and the upstream flow  $q_{in}$  are high. Using ramp metering, breakdown of the flow can be avoided and the outflow can be kept at a high level due to prevention of capacity drop, which in its turn can reduce the amount of time spent by the vehicles in the network. Obviously, applying ramp metering could result in a queue with queue length  $w_o$  on the on-ramp as shown in Figure 2.1. Hence, a trade-off has to be maintained between reducing the congestion on the freeway and keeping the on-ramp queues short.

The control methods we discuss below are operating in discrete time. This means that at each sample time instant  $t = kT$  where  $T$  is the sampling interval and the integer  $k$  is the discrete-time sample step, measurements of the traffic are performed and fed to the traffic controller. The controller then uses this information to determine the control signal to be applied during the next sampling interval.

### 2.2.1 Static feedback control

#### General concepts

Dynamical systems can be controlled in two ways: using open-loop control and using closed-loop control. In an open-loop system, the control input does not depend on the output of the system, whereas in a closed-loop system, the control action is a function of the output of the system. Feedback or closed-loop control systems are suited for applications that involve uncertainties or modelling errors.



In “static<sup>1</sup>” feedback control methods, the controller gets measurements from the system and determines control actions based on the current state of the system in such a way that the performance of the system is improved. The main examples of static feedback controllers are state feedback controllers (where the feedback gain can be computed using, e.g., pole placement) and PID controllers (for which several tuning rules exist, such as the Ziegler-Nichols rules) [3]. However, the static feedback strategy in general does not handle any external constraints. This is a major drawback of this control scheme.

### Control application for conventional traffic

A ramp metering system based on static feedback derives control actions (the metering rates) using real-time traffic measurements collected from the vicinity of the ramp meter. The two main local ramp metering strategies that use static feedback are [117, 135]:

- The demand-capacity strategy,
- ALINEA.

The demand-capacity strategy can be formulated as follows (see also Figure 2.2(a)):

$$r_o(k) = \begin{cases} q_{\text{cap}} - q_{\text{in}}(k-1) & \text{if } o_{\text{out}}(k) \leq o_{\text{cr}} \\ q_{\text{min}} & \text{otherwise} \end{cases}$$

where  $r_o(k)$  is the flow rate on the on-ramp at sample step  $k$ ,  $o_{\text{out}}(k)$  is the freeway occupancy<sup>2</sup> downstream of the on-ramp,  $o_{\text{cr}}$  is the critical occupancy at which the traffic flow tends to be at its maximum, and  $q_{\text{min}}$  is the prespecified minimum ramp flow value. This strategy determines the flow rate value as the difference between the downstream capacity and the upstream flow, until the critical occupancy is reached. Once this critical occupancy is reached, the on-ramp flow rate is reduced to a minimum value to prevent the occurrence of congestion. Also this demand strategy attempts to solve or to prevent the congestion problem only after the freeway is operating at its maximum usage.

One of the best known ramp metering strategies is ALINEA [117] (or one of its extensions such as the METALINE algorithm [116]). The basic ALINEA strategy is a closed-loop algorithm that determines the ramp metering rate in such a way that the downstream occupancy from the on-ramp is kept at a desired or prespecified value  $o_{\text{set}}$  (see also Figure 2.2(b)):

$$r_o(k) = r_o(k-1) + K_r(o_{\text{set}} - o_{\text{out}}(k))$$

where  $r_o(k)$  is the flow rate on the ramp at sample step  $k$ ,  $o_{\text{out}}(k)$  is the freeway occupancy downstream of the ramp, and  $K_r$  is a regulator parameter. One of the difficulties of this method is the determination of appropriate set-points.

Although the static feedback control strategies discussed above are based on real-time measurements, they are implemented at a local level only, based on the information that is obtained locally (i.e., in the vicinity of an on-ramp).

<sup>1</sup>By “static” we mean here that the control parameters of the feedback controller are taken to be fixed.

<sup>2</sup>The occupancy  $o$  of a highway or a lane can be defined as the fraction of time that the detector is occupied by every individual vehicle.

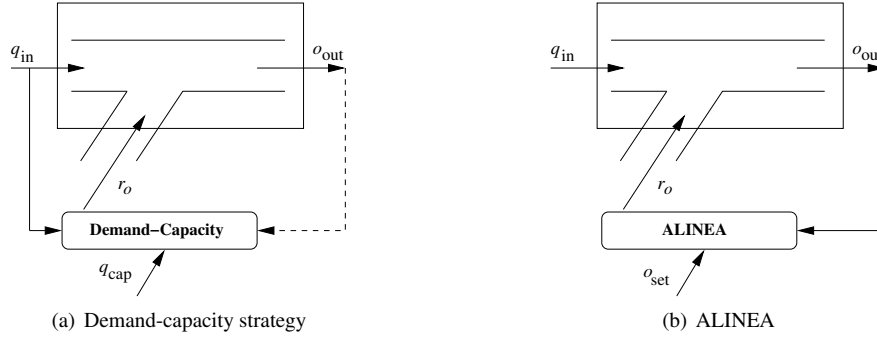


Figure 2.2: Demand-capacity strategy and ALINEA. The dashed line in Figure 2.2(a) indicates that this measurement is used for comparison only.

## 2.2.2 Optimal control and model predictive control

Now we discuss two dynamic control methods that apply optimisation algorithms to determine optimal control actions based on real-time measurements: optimal control and model predictive control.

### General concepts

#### General concepts of optimal control

Optimal control determines a sequence of admissible control actions that optimise a performance function by considering future demands and by satisfying the constraints [33, 85, 137]. A general discrete-time optimal control problem contains the following elements:

1. Dynamical system model equations,
2. An initial state  $\mathbf{x}_0$ ,
3. An initial time  $t_0$ ,
4. Constraints,
5. Measurements,
6. A performance index  $J$ .

More specifically, consider a multi-input multi-output dynamical system expressed by the following equation:

$$\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), \mathbf{d}(k))$$

where  $\mathbf{x} \in \mathbb{R}^n$  is an  $n$ -vector of states,  $\mathbf{u} \in \mathbb{R}^m$  is an  $m$ -vector of manipulatable control inputs,  $\mathbf{f}$  is a continuously differentiable function, and  $\mathbf{d}$  is the disturbance vector.

For a given time horizon  $K$ , the optimal control problem consists in determining a sequence of control vectors  $\mathbf{u}(0), \mathbf{u}(1), \dots, \mathbf{u}(K-1)$  in such a way that the performance index  $J$  takes on the minimum possible value subject to the initial conditions, system dynamics, and constraints, i.e.,

Minimise

$$J = \vartheta[\mathbf{x}(K)] + \sum_{k=0}^{K-1} \varphi[\mathbf{x}(k), \mathbf{u}(k), \mathbf{d}(k)]$$

subject to

$$\begin{aligned} \mathbf{x}(0) &= \mathbf{x}_0 \\ \mathbf{x}(k+1) &= \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), \mathbf{d}(k)) && \text{for } k = 0, \dots, K-1, \\ \mathbf{u}_{\min}(k) &\leq \mathbf{u}(k) \leq \mathbf{u}_{\max}(k) && \text{for } k = 0, \dots, K-1, \\ \mathbf{c}(\mathbf{x}(k), \mathbf{u}(k), k) &\leq \mathbf{0} && \text{for } k = 0, \dots, K-1, \end{aligned}$$

where  $\vartheta$  and  $\varphi$  are twice differentiable, nonlinear functions and are called the terminal cost and Lagrangian respectively,  $\mathbf{u}_{\min}$  and  $\mathbf{u}_{\max}$  are bounds for the control variables,  $\mathbf{c}$  expresses path constraints imposed on the state  $\mathbf{x}$ , and the control trajectories  $\mathbf{u}$  and the disturbance vector  $\mathbf{d}$  are assumed to be known over the period  $[k, k+KT]$  for  $k=0, \dots, K-1$ . Thus the performance criterion depends on the initial state, and the whole time history of the system's state and the control variables. There are two basic approaches to solve the above optimal control problem: calculus of variations [56, 69] and dynamic programming [19].

The main drawback of optimal control is that the method is essentially an open-loop control approach and thus suffers from disturbances and model mismatch errors. Next, we will discuss model predictive control, which uses feedback and a receding horizon approach to overcome some of the drawbacks optimal control.

### General concepts of model predictive control

Model Predictive Control (MPC) [30, 101, 123] has originated in the process industry and it has already been successfully implemented in many industrial applications. MPC is a feedback control algorithm that can handle constrained, complex dynamical systems [35, 38]. The main difference between optimal control and MPC is the rolling horizon approach used in MPC (this essentially means that the optimal control is performed repeatedly but over a limited horizon). On the one hand, this results in a suboptimal performance compared to optimal control (at least in the absence of disturbances). However, on the other hand, the rolling horizon approach introduces a feedback mechanism, which allows to reduce the effects of possible disturbances and of model mismatch errors.

The underlying concept of the MPC controller is based on on-line optimisation and uses an explicit prediction model to obtain the optimal values for the control measures subject to system dynamics and constraints. At each time step  $k$ , the MPC controller first measures or determines the current state  $\mathbf{x}(k)$  of the system. Next, the controller uses (on-line) optimisation and an explicit prediction model to determine the optimal values for the control measures over a given prediction period determined by the prediction horizon  $N_p$  (see Figure 2.3(b)). In order to reduce the computational complexity of the problem, one often introduces a constraint of the form  $\mathbf{u}(k+j) = \mathbf{u}(k+j-1)$  for  $j = N_c, \dots, N_p-1$ , where  $N_c$  is called the control horizon.

The optimal control inputs are then applied to the system in a receding horizon approach as follows. At each control step  $k$  only the first control sample  $\mathbf{u}^*(k)$  of the optimal control sequence  $\mathbf{u}^*(k), \dots, \mathbf{u}^*(k+N_c-1)$  is applied to the system. Next, the prediction horizon

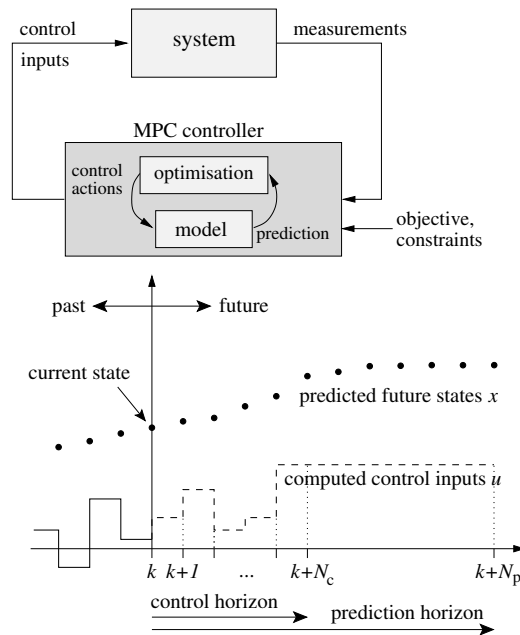


Figure 2.3: Model predictive control

is shifted one step forward, and the prediction and optimisation procedure over the shifted horizon is repeated using new system measurements.

MPC for linear systems subject to a quadratic objective function and linear constraints can be solved using quadratic programming. Other types of MPC problems in general require global or multi-start local optimisation methods such as sequential quadratic programming, pattern search, or simulated annealing [118].

Just as optimal control MPC can take into account constraints on the inputs and outputs, and it can also deal with multi-input multi-output systems. MPC has an advantage over optimal control due to receding horizon approach. This feedback mechanism of MPC makes the controlled system more robust to uncertainties and disturbances. Nevertheless, MPC still has some of the drawbacks of optimal control such as computational complexity, the need of an explicit model for prediction purposes, and the fact that the external inputs and disturbances need to be known fairly accurately in advance for the entire prediction horizon.

### Control application for conventional traffic

Optimal control can be applied to ramp metering as follows [89, 91, 92]. A second-order macroscopic METANET model can be used to model a freeway traffic system. The METANET model represents a network as a directed graph with links (corresponding to freeway stretches with uniform characteristics) and vertices (corresponding to on-ramps, intersections, etc.). In METANET each link is divided into segments. The state vector  $\mathbf{x}$  for the control problem consists of densities  $\rho_{m,i}$ , and mean speeds  $v_{m,i}$  of every segment  $i$  of every link  $m$ , and queue lengths  $w_o$  at every origin  $o$  of the network. The control vector

$\mathbf{u}$  consists of the ramp metering rates  $r_o$  of every on-ramp  $o$  under control. The process disturbance vector  $\mathbf{d}$  contains demands  $d_o$  at the origin links. All these values are assumed to be known over the entire simulation period. When controlling traffic, the controller needs to have a goal or objective to achieve, such as minimising the total time spent in a traffic network, minimising the total distance travelled, minimising the total fuel consumption, minimising the queue lengths, maximising the safety of vehicles in the network, etc., or a combination of these objectives.

Using the optimal control algorithm, the controller computes the optimal control values, based on the assumed demands  $d_o$ . In reality, these demands cannot be known in advance. However, often the future demand can be estimated reasonably well from upstream and downstream measurements in combination with historical data.

MPC for ramp metering and for conventional road-side based non-IV traffic management has been developed and implemented in [18, 66, 67]. In their research, similar models as described for optimal control approach are considered for MPC approach. The optimisation problem can be solved in the same way as for optimal control. The main differences between the optimal approach and that in MPC is that a shorter control horizon is used — such that the control signals (in this case, the metering rates  $r_o$  of every on-ramp  $o$  under control) are allowed to vary till the control horizon, after which the rates are taken to remain constant (till the prediction horizon), — and that the optimal control inputs are applied in a moving horizon scheme. The latter feature results in a more robust operation of the traffic control system.

### 2.2.3 Artificial intelligence techniques

#### General concepts

Artificial Intelligence (AI) techniques aim at enabling intelligence in machines to solve a problem using human intelligence and thinking. By human intelligence, we mean that the ability of computer programs to perceive a situation, to reason about the problem, and to act accordingly [154]. AI techniques are mainly used in decision support systems, and one way to classify them is as follows [36, 110, 126, 138, 154]:

- Case-based reasoning,
- Fuzzy logic,
- Rule-based systems,
- Artificial neural networks,
- Multi-agent systems,
- Genetic algorithms,
- Swarm intelligence,
- Reinforcement learning.

In this thesis, we will briefly discuss on case-based reasoning, fuzzy logic methods, rule-based systems, artificial neural networks, and multi-agent systems. For rest of the AI techniques, the interested readers are suggested to refer the above mentioned references.

Case-based reasoning, as the name suggests, solves a problem using the knowledge that was gained from previously experienced similar situations (cases) [1, 126]. In this way, this technique learns the way a new problem is solved, tests the proposed solution using simulation methods, and stores the new solution in a database. A disadvantage of this approach is that it might not be clear what should be done for a case that is not yet present in the case base. However, new cases could be added on-line to deal with this problem.

Fuzzy logic systems, like humans, can handle situations where the available information about the system is vague or imprecise [86, 110]. To deal with such situations, fuzzy sets are used to qualify the variables of the system in a non-quantitative way. Fuzzy sets are characterised using membership functions (e.g. Gaussian, triangle, or normal) that take a value between 0 and 1, and that indicate to what degree a given element belongs to the set (e.g., a speed could be 60 % “high” and 40 % “medium”). The membership degrees can then be used to combine various rules and to derive conclusions. This process consists of three parts: fuzzification, inference, and defuzzification. Fuzzification involves the transformation of a value of a variable into a fuzzy value, by linking it a given fuzzy set and determining a value for degree of membership. Inference uses a set of rules based on expert opinions and system knowledge and combines them using fuzzy set operators such as complement, intersection, and union of sets. Defuzzification converts the fuzzy output of the inference step in to a crisp value using techniques such as maximum, mean-of-maxima, and centroid defuzzification. One of main difficulties of a fuzzy system can be the selection of appropriate membership functions for the input and output variables. Moreover, fuzzy systems are often combined with other AI techniques for their complete deployment.

Rule-based systems solve a problem using “if-then” rules [62, 129]. These rules are constructed using expert knowledge and stored in an inference engine. The inference engine has an internal memory that stores rules and information about the problem, a pattern matcher, and a rule applier. The pattern matcher searches through the memory to decide which rules are suitable for the problem, and next the rule applier chooses the rule to apply. These systems are suited to solve problems where experts can make confident decisions. However, this system works only with already created rules and in its basic implementation it does not involve learning.

Artificial neural networks try to mimic the way in which the human brain processes information [61, 157]. These systems are useful in solving nonlinear problems where the rules or the algorithm to find solutions are difficult to derive. The basic processing unit of a neural network is called neuron or node. Each node fires a new signal when it receives a sufficiently high input signal from the other connected nodes. These nodes are organised in layers (an input layer, an output layer, and a number of hidden layers) and are interconnected by links or synapses, each associated with weights. A disadvantage is that artificial neural networks are non-informative models, and do not provide an explanation for the outcomes or for any failure that may occur in the process.

An agent is an entity that can perceive its environment through sensors and act upon its environment through actuators in a such way that the performance criteria are met [78]. Multi-agent systems consist of a network of agents that are interacting among themselves to achieve specified goals. A high-level agent communication language is used by the agents

for communication and negotiation purposes. Multi-agent systems can be applied to model complex systems, but their dynamic nature and the interactions between agents may give rise to conflicting goals or resource allocation problems.

### **Control application for conventional traffic**

Of the AI methods discussed above case-based reasoning and fuzzy logic are most often described in literature for traffic control purposes. In practice, rule-based systems are also used very often for traffic management and control.

Many decision support systems have been proposed for traffic management purposes such as FRED (Freeway Real-Time Expert System Demonstration) [127, 159], the Santa Monica Smart Corridor Demonstration Project [82, 128], and TRYS [41, 94, 108].

In [41, 65, 73, 159] case-based systems are developed for a broad range of scenarios that may occur in a traffic network. This initial case-based system is in principle generated using historical data or off-line simulations. In [65, 73] each scenario in the case base is evaluated over a prediction horizon using simulation and can be characterised by “inputs” such as the current traffic state, the traffic control measure applied (e.g., ramp metering, lane closures, etc.), or expected incidents (duration of incidents), and by “outputs” such as predicted average values of the traffic states or predicted values for performance measures. Hence, the case-based system maintains an input-output relation for each considered scenario. Once the initial case base is constructed, the traffic control centre can use this case-based system as the basis for assessing a particular traffic problem and for determining the most appropriate control scenarios. In case the current situation has not been addressed before, a new, real-life scenario of the traffic system can be evaluated and added to the case-based system along with effect of the control measures. Hence, learning and updating the case-based system with newly encountered cases serves as the main advantage and motivation for the application of this system in the traffic control field.

Although this technique has been useful for controlling small networks, adding each possible case for a large-scale traffic network can be a difficult task. To solve this problem and to handle large-scale traffic networks, a possible solution has been proposed in [48]. The solution of [48] considers a large-scale network as small subnetworks and uses fuzzy logic to combine different cases in the case base. By using fuzzy logic, a precise match between the considered actual case and the relevant cases in the case-based system will not be required.

As indicated before fuzzy systems can be used when accurate information of the traffic model is difficult to obtain or is not available [25, 94]. A fuzzy logic controller for ramp metering with a description of the various steps (fuzzification, inference, and defuzzification) is presented in [152]. Several fuzzy sets that can relate a variable (input, output) to a particular situation can be defined such as fuzzy sets for local speed, local traffic flow, queue occupancy, metering rate and local occupancy. Using fuzzification input variables such as speeds, flows, occupancy levels in the vicinity of the fuzzy ramp meter controller, and output variables such as metering rates can be translated to fit the defined fuzzy sets and to obtain values for the degree of membership. Next, these values are fed to the inference engine, which is constructed using a set of rules based on the experience of traffic control centre operators and on off-line simulations. These rules are to be kept as simple and easy to understand and to modify. The result of the inference is then transformed into a crisp

*Table 2.1: Comparison of control design methods*

Control method	Computational complexity	Constraints (hard)	Future inputs	Model-based	Scalability
Static feedback	low	no	no	not explicitly	localised
Optimal control and MPC	high	yes	yes	model-based	system-wide
AI-based	medium	no	no	not explicitly	localised

value in the defuzzification step, after which the final result is applied to the traffic system or presented to the operator of the traffic control centre for further assessment.

Related work is described in [65, 106].

## 2.2.4 Comparison

We will now compare the control design methods discussed above based on the following directions:

- Computational complexity,
- Inclusion of hard constraints,
- Inclusion of future inputs,
- Model-based or not,
- Scalability.

The results of this analysis are shown in Table 2.1.

### Computational complexity

Although the control design methods discussed above provide suboptimal performance (except for optimal control in the error-free case), the computational complexity of these methods does vary depending on their application area. Static feedback control methods have low on-line computational requirements, but they are in practice mainly applicable to small-scale systems. The on-line computational time required for AI techniques is somewhat higher but still much less than that of optimal control and MPC since the latter use on-line optimisation. All three control techniques also require off-line computations for model identification and model parameter calibration as well as for parameter tuning, training, or rule generation. These off-line computational requirements are mainly determined by the model and its complexity (model order, degree of nonlinearity, etc.).

### Inclusion of hard constraints

Consideration of hard constraints is an inevitable aspect to be included when controlling and managing traffic flows. Hard constraints are those that are to be necessarily satisfied. In general, static feedback control does not consider the external constraints. This lack of inclusion of constraints also proves to be one of the major drawbacks of the static feedback



methods. Most AI methods also do not explicitly take (hard) constraints into account. Optimal control and MPC on the other hand use optimisation and as a consequence they can explicitly include hard constraints when determining the optimal control actions.

### **Inclusion of future inputs**

Static feedback controllers do not consider the external future inputs. Since optimal control and MPC are prediction-based, they can also take into account knowledge about future inputs (e.g., in traffic control one might detect traffic flows using detectors located upstream and use this information). In general, AI techniques may also use predictions.

### **Model-based or not**

All the control approaches explicitly or implicitly need a model to collect data and/or for tuning. However, by model-based we mean in particular whether the control method considers an explicit model in order to determine the optimal control actions. Although the parameters of a static feedback controller may be tuned using model-based simulations, the controller does not explicitly include a model of the traffic system inside it. In a similar way the rules or cases of an AI-based controller may also be determined using simulation models, but the models are not explicitly present in the controller. On the other hand, optimal controllers and MPC controllers in general use an explicit model of the (traffic) system as a prediction model when determining the optimal control actions.

### **Scalability**

Based on the control structure used, we can categorise control and management systems [4, 52, 134] as follows:

- Centralised,
- Distributed,
- Decentralised,
- Hierarchical.

In a centralised set-up all control measures are managed from one single point, the traffic control centre. However, in practice such an approach is not feasible due to the lack of scalability and robustness, the communication requirements, and the computational complexity. One way to deal with these issues is to use a distributed approach in which several local controllers each control their own region or set of measures and in which the local controllers communicate and coordinate their actions. If only communication is present but there is no coordination, we have a decentralised control structure, which is usually easier to implement but also less optimal. Hierarchical control frameworks combine features of centralised and distributed control: they use a hierarchical framework with several control layers in which lower-level controllers take care of the fast dynamics in a small region of influence, and higher-level controllers take care of the slower dynamics and the coordination over a larger region of influence.

In this context, static feedback controllers are mostly localised and are used to control small-scale systems (in a central way) or as lower-level controllers in a hierarchical control framework. Most AI-based control methods (except for multi-agent approaches) are also mostly localised and can also be used in a hierarchical control framework. MPC (and optimal control) determines the control actions based on the current and predicted future states of the system and thus allows for a system-wide coordination of the control actions. In principle, MPC can be used at all levels of a hierarchical control framework but due to its computational complexity it is less suited for the lower levels. In addition, recently some distributed MPC methods have been developed [31].

In this section we have presented a brief overview of applied traffic control methods that use currently existing traffic control measures. In the next sections we will consider future traffic control systems based on intelligent vehicles and automated highway systems.

## 2.3 Intelligent vehicles and traffic control

### 2.3.1 Intelligent vehicles

We can divide IV application areas into three categories depending on the level of support provided to the driver [22]:

- Advisory systems use a human machine interface (HMI) (e.g., optic or acoustic) to provide an advisory or warning to the drivers. Some examples include blind spot warning, parking assistance, lane departure warning systems, and drowsy driver monitoring.
- Semi-autonomous systems can take partial control of vehicle manoeuvres and often use haptic (meaning “based on the sense of touch”) measures to assist the driver and can take partial control of vehicle manoeuvres. Examples are an intelligent speed adaptation system using an active accelerator pedal that exerts a counterforce at speeds over the speed limit, or a forward collision mitigation or avoidance system that first warns the driver via seat vibrations and next starts to brake in case the driver does not react to the warning.
- Fully autonomous systems take complete control of vehicle operations and eliminate the driver from the control loop. Examples include fully automated adaptive cruise control and anti-lock braking systems.

IV-based control measures can be further classified based on their manoeuvres [23]:

- Lateral sensing and control systems assist drivers in controlling the lateral movements of the vehicle. Applications of such systems are lane departure warning systems, lane change assist systems, and parallel parking assist systems.
- Longitudinal sensing and control systems mainly help in controlling forward and backward movements of the vehicle, and maintaining safe speeds and safe inter-vehicle distances. Typical applications are adaptive cruise control, forward collision warning systems, and intelligent speed adaptation.

- Integrated lateral and longitudinal sensing and control systems provide both lateral and longitudinal support to the driver. Such applications can combine the functionalities of adaptive cruise control and lane keeping assistance systems to assist the driver in performing manoeuvres.

### 2.3.2 IV-based traffic management

The currently implemented traffic control and management systems are mainly using intelligence in the roadside infrastructure for controlling and managing the traffic system. However, such a system does not make use of the significant benefits offered by the intelligence — including the additional control, sensing, and communication capabilities — provided by IVs. So one way to improve performance and safety of the current transportation systems is by applying automation and advanced intelligent control methods to roadside infrastructure and vehicles. Based on the way a traffic control and management approach utilises the automated vehicles, we can categorise the ongoing research on automated driving for traffic management as follows:

- Automated platooning,
- Autonomous IVs.

We will now briefly explain these two approaches.

An interesting functionality that is allowed by full automation is to arrange the vehicles in closely spaced groups called “platoons” [28]. In a platoon, the first vehicle is called “leader” and the remaining vehicles are called “followers”. To avoid collisions, intra-platoon spacing (i.e., vehicle spacing within a platoon) is kept very small and the inter-platoon spacing is kept larger [97, 149]. In the literature, many control frameworks, mainly intended to study inter-vehicle communication technologies and to control the platoon manoeuvres in cooperation with the roadside infrastructure, have been developed and investigated [7, 63, 76, 124, 145]. Also, frameworks that allow communication between the (partially or fully) automated individual vehicles and with the nearby roadside infrastructure have been proposed [42].

The other main category of IV-based traffic management approaches involves individual autonomous IVs and is mainly based on self-organisation and distributed intelligence among the automated vehicles with no control from the roadside infrastructure. In this autonomous driving system, the vehicles try to aggregate themselves in groups, make their decisions based on local information only, and perform manoeuvres in a cooperative fashion. Modelling and analysis of such group behaviour mechanisms in traffic management is an active area of research [54, 96, 100, 119].

In this chapter, we focus on the first approach, as this approach most probably allows for a smoother transition from the current, exclusively roadside-based traffic control system towards a mixed IV-roadside traffic management system in which the intelligence between the roadside infrastructure and the intelligent vehicles are integrated so as to improve the traffic performance.

### 2.3.3 IV-based control measures

In this section, we discuss various IV technologies that support both roadside traffic control measures and automated platoons of IVs. The main functional areas of IV systems that support traffic management and control on freeways are:

- Adaptive Cruise Control (ACC),
- Intelligent Speed Adaptation (ISA),
- Dynamic route guidance.

#### Adaptive cruise control

Many vehicles have already been deployed with partial automation of the driving task by the use of the standard conventional cruise control [125].

An Adaptive Cruise Control (ACC) system is a radar-based system that extends the conventional cruise control system and that is designed to sense the vehicle immediately in front on the same lane, and to automatically adjust the speed of the equipped vehicle to that of the preceding vehicle so as to maintain a safe inter-vehicle distance [45, 47]. If there is no predecessor, then ACC retains the preset speed that was selected by driver.

Variations of ACC are available in the market, among which high-speed ACC and stop-and-go ACC are popular [120]. High-speed ACC uses a radar to measure speed and distance of the vehicle ahead so as to maintain a safe distance between the vehicles. This ACC system is designed to operate at medium (above 40 km/h) to higher speeds, because at this range of speeds, a sensor can discriminate between non-moving targets and moving vehicles. A radar can sense a low relative velocity with respect to a moving predecessor and a high relative velocity with respect to stationary objects. However, high-speed ACC systems have difficulties to operate at low speeds. Stop-and-go ACC is designed to work with low speeds (0 km/h to 40 km/h) that prevail on urban roads and in congested traffic. Basically, the stop-and-go ACC acts as a distance controller rather than a speed controller.

Although ACC vehicles are able to handle many traffic situations reliably [113], there exist some adverse traffic conditions such as short-headway cut-ins, or sudden and strong deceleration of the up-front vehicle that prevent the ACC from sensing and reacting to the situation immediately. Cooperative ACC is a further enhancement of ACC systems that utilises existing communication technologies (e.g., ad hoc wireless networks) to obtain real-time information about the speed, acceleration, braking, position, headway, and yaw rate of the preceding vehicle in order to maintain a safe but small headway, and to ensure smooth driving [147].

Using cooperative ACC, vehicles can travel at reduced headways. Hence, with reduced headways between vehicles, traffic flow can be improved.

#### Intelligent speed adaptation

A standard speed limiter is a system that restricts the speed of the vehicle when the driver tries to exceed the maximum allowed driving speed. When the speed limiter is incorporated with the intelligence to adjust the maximum driving speed to the speed limit specified by the roadside infrastructure or to the prevailing location-based legal speed limit, and to provide

feedback to the driver when that speed limit is exceeded, then we get the technology called Intelligent Speed Adaptation (ISA) [24, 39].

ISA systems can be characterised based on the extent to which the driver is made aware of the situation when the speed limit is exceeded. This results in advisory, voluntary, and mandatory systems [32, 141].

Another characterisation for ISA systems can be made based on the speed limit itself as a fixed or a dynamic speed limit. In the fixed case, the driver is informed about the speed limit, which could be obtained from a static database. A dynamic speed limit system takes current road conditions such as bad weather, slippery roads, or major incidents into account before prescribing the speed limit.

ISA can influence the traffic flow by limiting the maximum speed of the vehicles depending upon the actual traffic flow conditions. As such congestion can be delayed or alleviated by delaying vehicles so that by the time they arrive at a congested region the congestion has diminished or has already been dissolved.

### **Dynamic route guidance**

Nowadays, many individual vehicles are equipped with a route guidance or navigation system. Based on current or expected travel times, and delays due to congestion, a route guidance system advises a driver about the “best” route he can take to reach his requested destination [103, 131, 153]. This route recommendation mainly depends on the vehicle’s current location. Using a GPS, the position of the vehicle can be determined accurately. This information can be looked up on a digital road map and be used to determine the possible routes to reach the destination. These route recommendations may be calculated within the equipped vehicle or communicated to the vehicle from the local traffic centre.

A possible categorisation of route guidance systems is static versus dynamic route guidance. When the possible routes are computed based on the average traffic conditions (e.g, based on historical data or on the type of roads (such as urban corridor, highway, local road, etc.)), then this scheme is referred to as static route guidance system. If the current traffic conditions such as traffic jams, dynamic speed limits, and on-line predictions of travel times using real-time traffic data are taken into account while computing the route recommendations, then we have a dynamic route guidance system [84, 155].

Dynamic route guidance systems can improve the traffic flow, by providing route recommendations to the vehicles such that congestion is prevented from occurring or such that the effects of traffic jams are mitigated.

## **2.4 Control frameworks and architectures for IVHS**

Now we discuss the most important control architectures that have been developed for linking the roadside infrastructure and automated platoons. In particular, we consider the PATH, Dolphin, Auto21 CDS, CVIS, SafeSpot, and PReVENT frameworks.

### **2.4.1 PATH framework**

The PATH architecture [28, 74, 133, 149, 150] mainly focuses on the coordination of both roadside-vehicle and inter-vehicle activities.

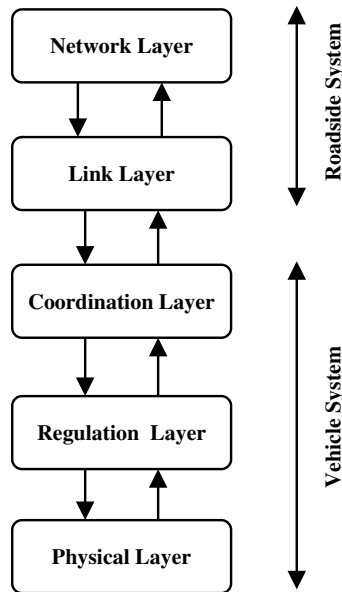


Figure 2.4: PATH architecture

The PATH framework considers a traffic network with many interconnected highways on which the vehicles are organised in platoons. The highways in the traffic network are considered to be divided into links (about 5 km long). A link is subdivided into segments (about 1 km long) with at least one exit or one entrance. A vehicle in the PATH framework is either considered as a leader, a follower, or a free agent (i.e., a one-vehicle platoon).

The PATH framework is a hierarchical structure in which the control of the automated highway system is distributed into five functional layers as shown in Figure 2.4:

- Physical layer,
- Regulation layer,
- Coordination layer,
- Link layer,
- Network layer.

The lower levels in this hierarchy deal with faster time scales (typically in the milliseconds range for the physical layer up to the seconds range for the coordination layer), whereas for the higher-level layers the frequency of updating can range from few times per minute (for the link layer) to once every few minutes (for the network layer). The controllers in the physical, regulation, and coordination layer reside inside the vehicles. The physical and regulation controllers govern single vehicles, whereas the coordination layer involves several vehicles. The link layer and the network layer controllers are located at the roadside, with the link layer controllers managing single freeway segments, and the network layer

controllers handling entire networks.

The same temporal and spatial division across different layers is also present in other frameworks that will be discussed in the next sections, such as the Dolphin framework and the Auto21 CDS framework.

Now we discuss each layer of the PATH framework in more detail, starting from the bottom of the hierarchy.

The physical layer of each vehicle involves the actual dynamics of the vehicle. This layer has controllers that perform the actuation of the steering wheel, throttle, and brakes. It also contains the sensors in the vehicle that collect information about the speed, the acceleration, and the engine state of the vehicle, and send it to the regulation layer.

The regulation layer controller executes the tasks specified by the coordination layer (such lane changes, and splits or merges of platoons) by converting them into throttle, steering, and braking inputs for the actuators of the vehicle. The regulation layer controller within each vehicle uses feedback control laws to execute the lateral and longitudinal manoeuvres and also notifies the coordination controller in case of any failures or unsafe outcomes of the manoeuvres.

Next in the hierarchy is the coordination layer with a controller residing inside each vehicle. This layer receives the commands from the link layer (such as set-points or profiles for the speeds, or platoon sizes). A coordination layer controller allows coordination with other neighbouring platoons using messages or communication protocols, and checks which manoeuvres (like lane changes, splits, or merges) have to be performed by a vehicle in order to achieve the platoon size or path trajectory specified by the link controller.

The next layer in the control hierarchy is the link layer. This layer has a controller located at each link and is mainly responsible for path and congestion control. Each link controller receives commands from the network layer (such as routes for the platoons) and based on these commands, the link controller calculates the maximum platoon size, and the optimum platoon velocity for each segment in the link it is managing. The link controller also sets the local path (which lane to follow) for each platoon.

The top layer in the hierarchy is called the network layer. At this layer, the controller computes control actions that optimise the entire network. Its task is to assign a route for each vehicle or platoon that enters the highway ensuring that the capacity of each potential route is utilised properly. There will be one network layer controller for the considered traffic network.

The PATH framework has also been used in other projects such as, e.g., the AHSRA (Advanced Cruise-Assist Highway System Research Association) project launched by the Japanese Ministry of Land, Infrastructure and Transport [2].

### 2.4.2 Dolphin framework

The Japanese Dolphin framework developed in [144, 145] is similar to the PATH architecture. The name of this framework is inspired by the way in which groups of dolphins move and which is mimicked by the automated vehicles in the platoons. The Dolphin framework considers vehicles to be arranged as platoons and develops inter-vehicle communication technologies to carry out cooperative driving for the purpose of smooth merging and lane changing.

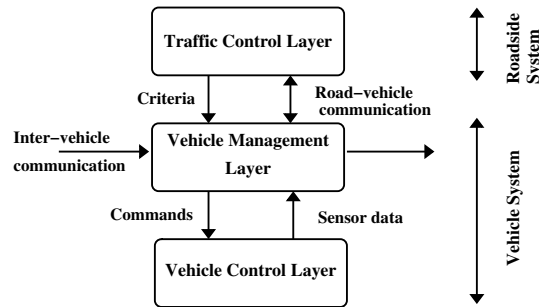


Figure 2.5: Dolphin architecture for cooperative driving

The Dolphin framework is composed of three layers as shown in Figure 2.5:

- Vehicle control layer,
- Vehicle management layer,
- Traffic control layer.

The vehicle controller within each vehicle senses the states and the conditions ahead of the vehicle such as vehicle speed and acceleration and sends this information to the vehicle management layer. The vehicle controller also receives commands for the vehicle's steering actions and determines the actions for the vehicle actuators.

The vehicle management controller, which resides in each vehicle, receives suggestions for the movements of the vehicle from the traffic controller via road-vehicle communication and also considers the messages from the neighbouring vehicles via inter-vehicle communication and the data received from the basic vehicle control layer. This controller determines the movements of the individual vehicle under platoon-based driving. Lateral and longitudinal control actions for the platoon leader or free agent are determined by this layer using, e.g., a localisation function by GPS and a digital map of the traffic network. The vehicle management controllers in the follower vehicles determine both the lateral and the longitudinal control commands for the vehicle they control.

The traffic control layer is the top layer and it includes roadside IVHS equipment like sign boards, conventional lane markers, and variable message signs, as well as logic such as laws, rules, and common sense. There is only one traffic control layer common to all the vehicles and it is part of the roadside infrastructure. The traffic layer consists of several distributed controllers, each of which determines advisory instructions for the vehicles in its own neighbourhood and sends these instructions to the vehicle management layer. For example, the instructions could be to maintain a small inter-vehicle distance between the followers in a platoon, and to support an autonomous driving for the leaders in a platoon.

### 2.4.3 Auto21 CDS framework

The Auto21 Collaborative Driving System (CDS) framework [7] is mainly inspired by the concepts of the PATH and Dolphin architectures. The CDS architecture considers platoons



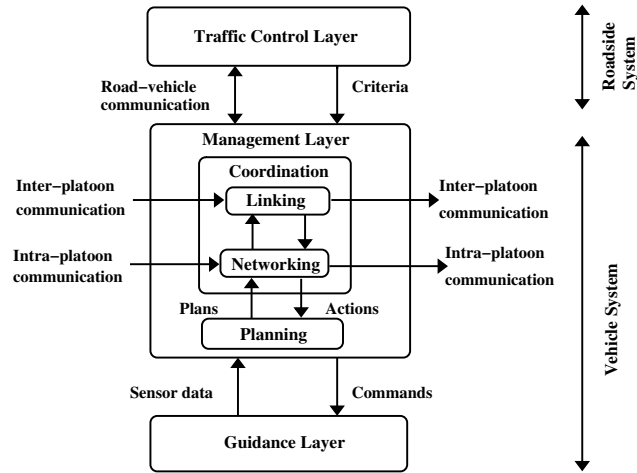


Figure 2.6: Auto21 CDS architecture for cooperative driving

of cars as autonomous agents and uses cooperative ACC technologies to support platoon-based driving. The CDS framework employs an inter-vehicle coordination system that can ensure coordination of vehicle activities during their merge and split operations from a platoon and that can maintain stability among the vehicles in a platoon. The hierarchical architecture of the Auto21 CDS framework consists of the following three layers as shown in Figure 2.6:

- Guidance layer,
- Management layer,
- Traffic control layer.

In every vehicle there is a guidance layer controller that senses the state of the vehicle and sends information on the position, speed, acceleration, etc. of the vehicle to the management layer. This controller also receives commands from the management layer, which are then translated into control actions for the throttle, braking, and steering.

The management layer is the main contribution of the CDS framework. The management layer contains several controllers, one for each vehicle. A management layer controller is responsible for determining the movements of the vehicle it manages using the information received from the traffic control layer, and makes sure that vehicle coordination constraints are satisfied through inter-vehicle communication. This layer is subdivided into coordination and planning sublayers that work cooperatively. The coordination sublayer has a linking module that communicates with the traffic control layer to receive suggestions for the lane change actions and that manages the inter-platoon activities. Once the linking module has made a choice on which the action to perform, the manoeuvres are executed (such as merge or split from a platoon) by a networking module, which coordinates the intra-platoon activities. The planning sublayer is responsible for making the plans to execute the manoeuvres inside the platoons. These plans are constructed in cooperation and coordination with the networking module.

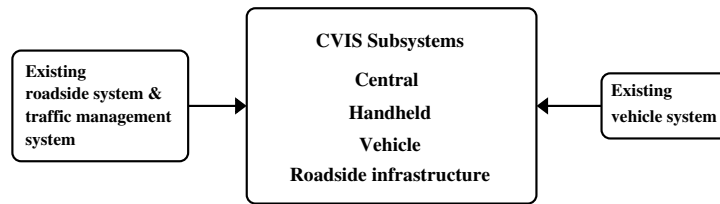


Figure 2.7: CVIS set-up

The traffic control layer is located in the roadside infrastructure and provides suggestions on desired speeds for segments, maximum platoon sizes, etc. The traffic control layer also supports the linking module (of the management layer) for making decisions on inter-platoon coordination actions [60].

#### 2.4.4 CVIS

CVIS (Cooperative Vehicle-Infrastructure Systems) [42, 146] is a European research and development project that aims to design, develop, and test technologies that allow communication between the cars and with the roadside infrastructure, which in turn will improve road safety and efficiency, and reduce environmental impact. Traffic systems with CVIS technologies select a suitable communication medium depending on user requirements and available media, and allow cars to communicate in a secure way using wireless technologies.

CVIS operates with existing traffic control and management centres, roadside infrastructures, and vehicle systems. The complete system can be considered as a single-level architecture with the existing systems and CVIS operating at the same level. Various networks and communication protocols have been developed within CVIS to enable communication between different subsystems. The time scale for this architecture ranges from minutes to hours.

A CVIS system is composed of four subsystems as shown in Figure 2.7: central, handheld, vehicle, and roadside subsystems. The central subsystem is a basically a service provider for the vehicle or the roadside infrastructure. Typical examples of central subsystems include traffic control and service centres, and authority databases. The handheld subsystem provides services such as pedestrian safety and remote management of other CVIS subsystems by allowing access to the CVIS system using PDAs and mobile phones. The vehicle subsystem is comprised of on-board systems and includes vehicle sensors and actuators, and equipment for vehicle-vehicle and vehicle-infrastructure communication. The roadside subsystem corresponds to the intelligent infrastructure that operates at the roadside and includes traffic signals, cameras, variable message signs, etc.

#### 2.4.5 SafeSpot

SafeSpot [130, 143] is a research project funded by the European 6th Framework Program on Information Society Technologies. The main objective of this project is to improve road safety using advanced driving assistance systems and intelligent roads. The safety margin assistant developed by the SafeSpot project uses advanced communication technologies to

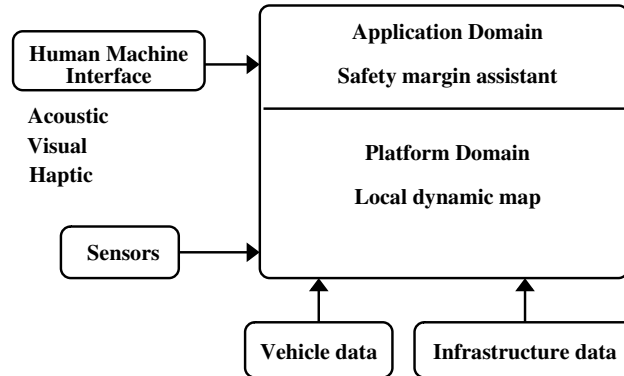


Figure 2.8: SafeSpot set-up

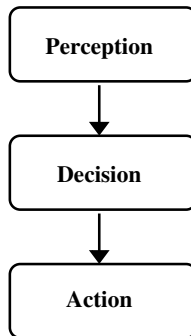


Figure 2.9: PReVENT set-up

obtain information about the surrounding vehicles and about the roadside infrastructure. This safety margin assistant can detect dangerous situations in advance and can make the driver aware of the surrounding environment using a human machine interface as shown in Figure 2.8. The time scale for this architecture ranges from seconds to minutes.

#### 2.4.6 PReVENT

PReVENT [121] is a European automotive industry activity co-funded by the European Commission. The main focus of the PReVENT project is to develop preventive applications and technologies that can improve the road safety. These safety applications use in-vehicle systems to maintain safe speeds and distances depending on the nature and severity of the obstacles, and to provide instructions and to assist the drivers in their driving tasks so as to avoid collisions and accidents.

The PReVENT architecture introduces a three-layer approach as shown in Figure 2.9 with the following layers:

- Perception layer,

- Decision layer,
- Action layer.

All these layers are located within the vehicle. From the perception layer upward to the action layer, the time complexity and update frequency of states typically ranges from milliseconds to seconds.

The perception layer uses on-board sensors (such as radar, cameras, and GPS receivers) in conjunction with digital maps and allows vehicle-to-vehicle and vehicle-to-infrastructure communication.

The decision layer assesses dangerous situations ahead of the vehicle and determines relevant actions that are needed to avoid such situations. The controller then passes this decision to the action layer.

The action layer then issues warnings to the driver about the severity of the situation through an appropriate human machine interface or through vehicle actuators such as the steering wheel or the brakes.

## 2.5 Comparison of the IVHS frameworks

In Section 2.4 we have discussed various existing frameworks and architectures for IVHS-based traffic management. Now, we will compare the frameworks based on the following features:

- Control objectives,
- Type of formation control (platoons or single cars),
- Intelligence at roadside and/or in vehicles,
- Presence and type of communication and/or coordination,
- Time scales involved.

The results of this analysis are shown in Table 2.2.

### Control objectives

For the PATH framework, the focus is primarily on designing and developing lateral and longitudinal controllers that allow automation of vehicles in a platoon and platoon manoeuvres. The PATH program has also developed coordination and communication techniques that are required during such manoeuvres. In addition, the roadside controllers in the PATH framework (i.e., the network and link layer controllers) determine the activities that need to be carried out in different freeway segments to avoid congestion. The Dolphin framework is vehicle-oriented, which means that the framework is mainly intended to study the inter-vehicle communication technologies that can be used to control a group of platoons that reside on neighbouring lanes and to support their manoeuvres. The AUTO21 CDS framework mainly aims at developing communication and coordination methodologies for merging and splitting actions within the platoon (intra-platoon manoeuvres). This

Table 2.2: Comparison of the various IVHS frameworks (in this table “veh” stands for “vehicles”, “V2V” means “vehicle-vehicle”, and “V2I” is short for “vehicle-infrastructure”)

Framework	Control objectives	Formation control	Intelligence	Communication & coordination	Time scale
PATH	platoon manoeuvres traffic flow efficiency	platoons	road/veh	V2V/V2I	ms-h
Dolphin	platoon manoeuvres	platoons	(road)/veh	V2V	ms-h
Auto21 CDS	intra-platoon manoeuvres	platoons	(road)/veh	V2V	ms-h
CVIS	traffic efficiency	cars	road/veh	V2V/V2I	min-h
SafeSpot	prevent/avoid road accidents	cars	(road)/veh	V2V/V2I	s-min
PReVENT	car safety	cars	veh	V2V	ms-s

framework uses an inter-platoon coordination model that was developed by the PATH project [8]. The objective of the CVIS project is to develop controllers and communication technologies for implementing a cooperative vehicle-highway system.

The PATH, Dolphin, AUTO21 CDS, and CVIS frameworks have developed control methodologies to be implemented in the roadside infrastructure to improve the traffic flow or in vehicles to allow automation of driving tasks. On the other hand, SafeSpot and PReVENT focus on improving the road safety by avoiding or preventing accidents, and they aim at integrated safety, with an emphasis on the potential of communication between vehicles and between vehicles and roadside systems.

### Type of formation control

The frameworks usually consider the vehicles to be controlled either as part of higher-level entities such as platoons, or as individual vehicles. The PATH, Dolphin, and Auto21 CDS frameworks allow platooning. On the other hand, SafeSpot, PReVENT, and CVIS do not use platoons.

### Intelligence at roadside and/or in vehicles

The PATH framework allows involvement of both roadside infrastructure and vehicles for improving traffic performance. Although the Dolphin and the Auto21 CDS frameworks consider distributed intelligence between roadside infrastructure and vehicles, the roadside infrastructure only provides suggestions and instructions to the vehicles. The platoons are not obliged to follow these suggestions. CVIS and SafeSpot incorporate intelligence in both vehicle and roadside infrastructure. PReVENT also includes distributed intelligence but with the main focus on vehicle intelligence.

## Presence and type of communication and/or coordination

Almost all the frameworks and projects have designed and developed technologies for inter-vehicle and roadside-vehicle communication for coordination of activities. Specifically, PATH has developed dedicated communication protocols [57, 76] and the Dolphin framework has developed a wireless local access network model for vehicle following, and inter-vehicle communication technologies for platoon manoeuvre coordinations [144]. For the coordination of tasks within the platoons, the AUTO21 CDS framework allows both a centralised set-up (i.e., the platoon leader instructs intra-platoon manoeuvres) or a decentralised set-up (i.e., all the members of the platoon are involved in the coordination)

SafeSpot, PREVENT, and CVIS focus on the issue of developing communication techniques that can be implemented in existing traffic networks and that can also be extended to AHS.

## Time scales involved

The time scales involved for the PATH, Dolphin, and Auto21 CDS frameworks vary from milliseconds to hours as one traverses from dynamics of the vehicle up to the roadside infrastructure levels. The time complexity and update frequency typically ranges from minutes to hours for CVIS, from seconds to minutes for SafeSpot, and from milliseconds to seconds for PREVENT.

## 2.6 Outlook

In this section we first sketch how the control design methods presented in Section 2.2 could eventually be applied within the various IVHS control architectures of Section 2.4. We also indicate the challenges and issues that still have to be addressed to get a full-fledged and complete IVHS-based traffic management and control system.

### 2.6.1 Application to IVHS

In relation to IVHS, longitudinal controllers for IVs inside platoons such as cooperative ACC controllers can be implemented using static feedback control. In particular, we now discuss how the accelerations for the follower vehicles within a platoon could be calculated using a static feedback controller. The follower vehicles in a platoon should use their on-board ACC system to maintain short intra-platoon distances. Hence, an ACC algorithm could consist of a combined speed and distance controller as described in [77]:

$$a_i(k) = K_2(h_{\text{ref},i}(k) - (x_{i+1}(k) - x_i(k))) + K_3(v_{i+1}(k) - v_i(k))$$

where  $a_i(k)$  is the acceleration for vehicle  $i$  at time  $t = kT$ ,  $v_i(k)$  is the speed of vehicle  $i$  at time  $t = kT$ ,  $x_i(k)$  is the longitudinal position of the rear of vehicle  $i$  at time  $t = kT$ ,  $h_{\text{ref},i}$  is the reference distance headway for vehicle  $i$ , and  $K_2$  and  $K_3$  are controller parameters. The reference distance headway is defined as follows:

$$h_{\text{ref},i}(k) = S_0 + v_i(k)T_{\text{head},i} + L_i \quad , \quad (2.1)$$

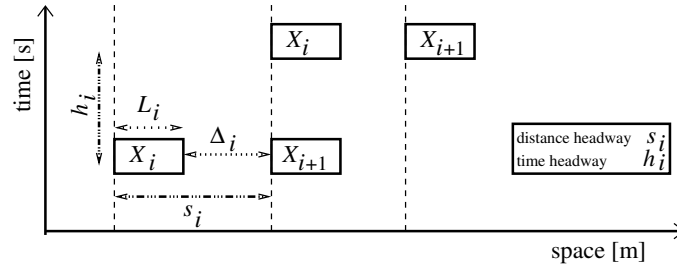


Figure 2.10: Time-space plot illustrating the definition of the time headway  $h_i$  and the distance headway  $s_i$  for vehicle  $i$

where  $S_0$  is the minimum safe distance headway that is to be maintained at zero speed,  $T_{\text{head},i}$  is the desired time headway for vehicle  $i$ , and  $L_i$  is the length of vehicle  $i$ . These definitions are illustrated using two vehicles  $i$  and  $i+1$  travelling on a single lane in a time-space coordination system as shown in Figure 2.10. The time and distance directions are shown in vertical and horizontal axes respectively. The time headway is more often considered rather than distance headway because of its simplicity and ease of measuring. In practice, the distance headway can be obtained from photos.

Optimal control and in particular MPC can be applied either in vehicle controllers or in roadside controllers. For a low-level control tasks involving manoeuvres of individual vehicles such as ACC or ISA, which need frequent updates of the control signal (e.g., the acceleration), these methodologies may prove to be very slow due to the computational complexity involved. However, MPC is excellently suited for higher-level IV-based traffic control and management purposes and to determine optimal speeds, lane allocations, and on-ramp release times for the platoons. MPC offers a strategy that can determine the control actions based on the current and predicted future states of the traffic network, and that can optimise the performance objective assigned to roadside infrastructure over a given time horizon subject to the operational constraints and the constrained imposed on them. Moreover, the feedback and the receding horizon approach of MPC allow to reduce the effects of possible mismatch errors between the actual real-world traffic flows and the predicted traffic flows. Some results in this direction are described in [13, 14]. In addition, since MPC is an optimisation-based approach, it is able to deal with the many different, often conflicting objectives (e.g., safety versus efficiency) that play a role in IVHS through the use of multi-criteria optimisation [107].

Fuzzy controllers can be used to implement an ACC controller using the deviations in speeds and distances as input variables. A typical example of such an application is dealt with in [71]. Moreover, fuzzy control can also be used to assist in making decisions for traffic management purposes. On the other hand, case-based reasoning will be more suitable for traffic planning and management purposes than for individual vehicle controllers.

## 2.6.2 Challenges and open issues

Now we discuss the main technological, economical, and societal challenges that will have to be addressed when actually implementing an IVHS systems.

Although we have indicated above how control design methods such as static feedback control, MPC, and AI-based control could be used for IVHS and traffic management and control systems based on intelligent vehicles and platoons, real integration of these methods is still lacking. This is one of the challenges that still have to be addressed. However, the deployment of these control design methodologies does not ensure stability and robustness of the traffic system since IVHS and IVs are non-linear and often even hybrid (i.e., they exhibit both continuous dynamics and discrete-event behaviour (switching)).

The new control methods developed within PATH and the other frameworks have mainly focused on the lower layers, i.e., the coordination between the vehicles and the platoons, and have not really focused on higher-level traffic management aspects beyond the single free-way level such as route guidance and network-wide coordination. Moreover, the segment concepts used in PATH may lead to difficulties when dealing with very long platoons. Indeed, when the platoon size is very large, then a platoon might cover several segments, and then it might be difficult for the roadside controllers involved to assign the appropriate activities for the platoon, and also for the involved vehicle controllers to coordinate their activities. This raises two important open problems: platoon formation and scalability.

In literature there are no strict rules available on how to form platoons and on how many vehicles to include in a platoon. This can either be specific to a given road or to a destination. There are few articles that deal with vehicle sorting with respect to platoon sizes and platoon formation time, and also on the design of platoon manoeuvre protocols [59, 75].

Some of IVHS frameworks such as the PATH, Auto21 CDS, and Dolphin frameworks are by nature hierarchical and offer thus a certain degree of scalability with regard to network size. Other frameworks such PREVENT, SafeSpot, and CVIS are not explicitly hierarchical and are thus not inherently scalable with regard to network size. However, none of the frameworks *explicitly* addresses scalability and the scalability of the frameworks has not yet been investigated in detail in literature. So this is also a topic for future research. In addition, scalability with regard to platoon sizes is also an open issue.

Hence, one of the major requirements is the development of a (new) integrated hierarchical traffic management framework for IVHS and the design and implementation of appropriate control methods for such a framework. Moreover, the full automation present in IVHS may also lead to new traffic control measures that can currently not be implemented, such as “real-valued” speed limits for more accurate control, smoother merging at on-ramps through speed and spacing control, etc. These new measures should also be accommodated in the control framework.

The technical issues outlined above are still open and need to be addressed. Moreover, the IVHS approach requires major investments to be made by both the government (or the body that manages the highway system) and the constructors and owners of the vehicles. Since few decisions are left to the driver and since the AHS assumes almost complete control over the vehicles, which drive at high speeds and at short distances from each other, a strong psychological resistance to this traffic congestion policy is to be expected. In addition, the fact that vehicles can be tracked through the entire network may raise some concerns regarding privacy and liability issues.

Another important question is how the transition of the current highway system to an AHS-based system should occur and — once it has been installed — what has to be done with vehicles that are not yet equipped for IVHS. As an intermediate step towards IVHS the current highway management system could start to communicate with the IVs and use



information obtained from, e.g., the route guidance system of the IVs in order to more accurately forecast traffic loads. Other transition issues that have to be taken into account are [50]: How will the system be funded? What types of accidents can be expected to occur in AHS, in what numbers, and with what consequences? Will bi-directional communication and transfer of information be allowed between IVs and roadside infrastructure? What are the legal implications of an accident, especially if it were caused by system error or a system oversight? How will an AHS implementation be coordinated on an international level? etc.

## **2.7 Summary**

We have presented an overview of traffic management and control frameworks for IVHS. First, we have given a short survey of the main control design methods currently used for freeway traffic control. Next, we have discussed various traffic management architectures for IVHS such as PATH, Dolphin, Auto21 CDS, CVIS, SafeSpot, and PReVENT. The frameworks have been compared in a qualitative way and we have sketched how the current traffic control methodologies could fit in an IVHS-based traffic control set-up. Finally, we have identified some open issues and future challenges in the further implementation and actual deployment of IVHS traffic management systems.



## **Chapter 3**

# **A New Traffic Management Framework for IVHS**

### **3.1 Introduction**

Based on an extensive survey of existing IV-based traffic management approaches and frameworks, the main components and results of which are discussed in Chapter 2, we now propose a new IV-based framework that combines several of the strong points of existing architectures and that further enhances and extends them in several directions. The objective of the framework is to integrate the intelligence of roadside infrastructure and IV systems with automation. We propose a framework that distributes the intelligence between roadside infrastructure and vehicles, that assigns the traffic control actions based on platoons rather than on a segment concept, and that uses the IV-based control measures to improve the traffic performance in terms of safety, throughput, and environment. We want also to include the advantages of other frameworks such as CarTalk, SafeSpot, Prevent, CVIS that design and develop technologies for inter-vehicle and roadside/vehicle communication. In this chapter, we present the new hierarchical IV-based traffic control framework in more detail.

### **3.2 IV-based control framework**

The goal of the proposed framework is to use the additional measures and control handles offered by in-vehicle telematics that support platooning and to implement them in a hierarchical roadside/vehicle traffic management structure to substantially improve traffic performance in terms of safety, throughput, reliability, environment, and robustness. The framework mainly aims at a multi-level control structure with local controllers at the lowest level and one or more higher supervisory control levels. Moreover, it will use a combination and integration of techniques from computer science and control engineering in order to obtain coordination at and across all control levels.

Our framework is inspired by the AHS platoon concept and uses IV-based control measures to implement a next-level of traffic control and management, which shifts away from

the rather global road-side traffic management to a more vehicle-oriented traffic management. We will consider both roadside-vehicle traffic management and interaction, and inter-vehicle traffic management and coordination. Although the PATH framework [74, 149, 150] includes both roadside infrastructure and vehicles, much of the research work was carried on the vehicle control side. The roadside controllers determine the activities that need to be carried out in different segments. However, when the platoon size is allowed to be long enough, then it might be difficult for the roadside controller to assign the activities as the platoon resides in between two segments, and also for the vehicle controller to complete the activity within the specified space. For this reason, we use platoon-based roadside controllers (so without segments). In our framework, vehicles will travel as platoons. Every platoon will then have a leader vehicle and the rest of the vehicles, if any, as followers. The platoon leader uses ISA and the followers will be equipped with ACCs. In addition, in order to be able to deal with large-scale traffic network we will consider a hierarchical control structure with several layers each of which consist of several controllers (see Figure 3.1). The reasons for selecting such a distributed, hierarchical control structure for traffic management are:

- Scalability: Adding, altering, or removing layers can be easily executed without altering other layers in the architecture, i.e., a hierarchical structure is more flexible for modifications in problem size.
- Robustness: In case of any unexpected failures in one of the components, the framework in general, will still be able to complete its execution of planned tasks (graceful degradation).

### 3.2.1 Proposed control framework

A general structure of the proposed control architecture is shown in Figure 3.1. The architecture consists of several layers, which can be characterised as follows:

#### Higher-level controllers

Higher-level controllers (area, regional, and supraregional) provide network-wide coordination of the lower-level and middle-level controllers. For example, there could be an area controller to control/supervise the activities of a collection of roadside controllers. In turn, a group of area controllers could be supervised by regional controllers, etc.

#### Roadside controller

The roadside controller in the hierarchy uses IV-based control measures to improve the traffic flow. The controller assigns desired speeds for each platoon (ISA), safe distances between platoons (Cooperative ACC), metering values on the on-ramps and off-ramps (ramp metering), desired platoon sizes, provides dynamic route guidance for the platoons, and also instructs for merges, splits, and lane changes of platoons. This layer may control a part of a highway, an entire highway, or a collection of highways.

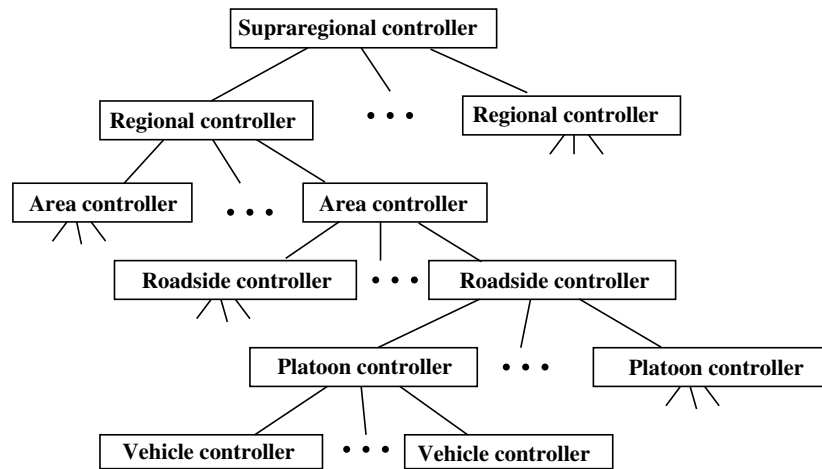


Figure 3.1: IV-based traffic control framework

### Platoon controller

The platoon controller is responsible for control and coordination of each intelligent vehicle in the platoon. The platoon controller receives commands from the roadside controller and is mainly concerned with actually executing the inter-platoon manoeuvres such as merges with other platoons, splits, and lane changes and intra-platoon activities such as maintaining safe inter-vehicle distances and acceleration for accomplishing the tasks assigned by the roadside controller.

### Vehicle controller

The vehicle controllers present in each vehicle are the low-level controllers of the architecture. They receive commands from the platoon controllers (e.g. set points or reference trajectories for speeds, paths, headways) and translate these commands into control signals for the vehicle actuators such as throttle, braking, and steering actions.

### Time scales involved

The time scales involved for our hierarchical framework vary from milliseconds to hours as one traverses from dynamics of the vehicle up to the roadside infrastructure levels. The time complexity and update frequency typically ranges from milliseconds to seconds for the vehicle controllers, from seconds to minutes for the platoon controllers, from minutes to quarters of an hour for roadside controllers, and hours for the area controllers.

## 3.3 Main improvements and extensions

The main improvements and extensions of the new framework are:

- We allow vehicles to communicate both with each other and with the roadside infrastructure in an integrated way.
- We consider both in-vehicle IV-based traffic control measures (such as ISA and ACC) and roadside traffic control measures (such as ramp metering, variable speed limits, dynamic route guidance, shoulder lanes openings, etc.) and apply them in an integrated way.
- The platoon size is one of the variables that are optimised and it can range from 1 to a very large number, depending on the current traffic conditions and the traffic network and on what is best for the given traffic performance criterion.
- The framework is integrated with a model-based predictive control strategy that determines optimal traffic control measures in a receding horizon approach.
- The framework is also suited for inter-urban and even urban traffic networks.

### 3.3.1 Integrated in-vehicle and roadside control measures

In particular, the proposed framework incorporates the following IV-based control measures:

- **ISA:** ISA can influence the traffic flow by externally controlling the speed of the vehicles by limiting the maximum speed depending upon the actual traffic flow conditions. In our approach, we will control the speed limit, which are determined by the roadside controller based on the current and future predicted traffic conditions.
- **Cooperative ACC:** Once the vehicle is equipped with an automatic speed control device, we can integrate other possible options in it and to take advantage of cooperative ACCs. With cooperative ACC, vehicles in a platoon can exchange information (e.g., destination, speed, acceleration, braking and so on) and can react to and handle unexpected manoeuvres safely by maintaining safe and short inter-vehicle distances, and high speeds.
- **Dynamic route guidance:** In our approach, we also consider the interaction of the roadside infrastructure with the dynamic route planning system: e.g., if there are several alternative routes to a destination for a platoon, then the roadside controller could suggest or impose routes in such a way that the overall traffic performance is improved.

In addition, we also combine these IV-based traffic control measures with roadside traffic control measures such as ramp metering, traffic signals, lanes closures, shoulder lane openings, etc.

The actual control strategy could then make use of a model-based predictive control approach (MPC) [30, 101]. MPC has originated in the process industry and has also been extended to conventional roadside-based non-IV traffic management [18, 66, 67]. This approach can also be extended to the proposed new IV-based traffic management framework as will be presented in the next chapters.

Note that we also do not necessarily limit the proposed framework to freeway networks only, but also can include inter-urban and even urban arterials and networks. Of course, due to more fragmented nature of the urban and inter-urban traffic network, optimal platoon sizes in these latter environments will be much smaller. As the platoon size is also one of the control parameters in our framework this can be taken into account.

### 3.4 Contributions of our approach

The main contributions of the proposed framework are:

- Our approach integrates the roadside infrastructure and the platoon of vehicles as is also done in the AHS framework, but the roadside controller will calculate optimal control actions or set points for each platoon rather than to each vehicle in a segment, and will use IV-based measures to obtain an improved system optimum traffic management.
- Our framework can determine and assign optimal platoon sizes. So the platoons can vary from one-sized to a very large number depending mainly on the instructions given by the roadside controller.
- The hierarchical structure with one or more supervisors added to the existing AHS architecture could prove to be an advantage of our framework. Large-scale traffic networks can be considered and also coordination of the mid-level and low-level controllers for such a wide network can be obtained by our framework.
- AHS and other architectures focused their implementation either on highways or on urban roads. Both of them were not treated as a single system. This could create problems when the vehicles exit from the automated highway and enter into a congested and manual controlled urban network, or vice versa. In this case, the automation concept may provide a drastic change to the driver behaviour and this might prove to be a disadvantage. Therefore, we propose to implement our framework on both the highways and urban roads, and apply the IV-based control measures and autonomous platooning (with small-sized platoons in the urban network).

### 3.5 Open problems

Our framework has plenty of opportunities and open problems left to explore. Some of them are listed below:

- We will develop efficient control methods and algorithms that will help in achieving cooperation and coordination at and across the hierarchical levels. The performance and complexity of the control methods and algorithms at each levels will be studied and analysed. This is the topic of the next chapters.
- Modelling of platoons from the roadside controller point of view is an open issue. Lane-changing manoeuvres for a large-sized platoon must be analysed. We will use a big-car model for modelling platoon behaviours and is discussed in Section 4.4.5.

- We will analyse, verify, and investigate the flexibility of the proposed framework for different case studies and scenarios of the traffic network. First of all, we will implement this multi-level framework for a small-scale system and observe its effects on the traffic performance. As a next step, we will implement this framework for a large-scale traffic network. The study can include investigating what type of traffic models is to be used and what the next steps should be towards an actual implementation of the framework in practice.

### **3.6 Summary**

In this chapter, we have proposed a framework that combines IV-based measures and roadside infrastructure for implementing a next-level, next generation traffic management. Topics for next chapters include: developing control design methods, determining appropriate traffic models for this IV-based framework, defining the case studies, implementing this framework on a small-scale setup, developing software tools, and analysing the trade-offs between computational complexity and efficiency for this framework.



## Chapter 4

# Roadside Controller

The overall control framework we use is the hierarchical framework that we have presented in Chapter 3. In this chapter we present a variant of IVHS in which the monitoring and control capabilities offered by automated intelligent vehicles (IVs) are combined with those of the roadside infrastructure. All vehicles are assumed to be fully automated and the automation of driving tasks allows to arrange the IVs in platoons. Thus the proposed control IVHS approach consists of interacting roadside controllers and platoons of IVs, and provides a platoon-specific management. In this chapter, we focus on the control layer that manages the different platoons on the individual freeway stretch as well as on the access points to the IVHS from the non-automated part of the traffic network. In particular, we combine dynamic speed limits and lane allocation for the platoons on the IVHS highways with access control for the on-ramps using ramp metering, and we propose a model-based predictive control approach to determine optimal speed limits and lane allocations as well as optimal release times for the platoons at the on-ramps. In order to illustrate the potential of the proposed traffic control method, we apply it to a simple simulation example in which the aim is to minimise the total time all vehicles spend in the network by optimally assigning dynamic speed limits, lane allocations, and on-ramp release times to the platoons compared to the situation with controlled human drivers.

### 4.1 Introduction

There are many ways to reduce the frequency and impact of traffic jams (such as building new roads, introducing road pricing, stimulating modal shift, promoting public transportation, etc.). On the longer term one of the most promising approaches is the use of advanced traffic management and control methods in which control measures such as traffic signals, dynamic route information panels, ramp metering installations, dynamic speed limits, etc. are used to control the traffic flows and to prevent or to reduce traffic jams, or more generally to improve the performance of the traffic system. As a next step in this direction, advanced control methods and advanced communication and information technologies are currently being combined with the existing transportation infrastructure and equipment. This will result in integrated traffic management and control systems that incorporate intelligence in both the roadside infrastructure and in the vehicles, such as Intelligent

Vehicle Highway Systems (IVHS) [150], Intelligent Transportation Systems [136], Automated Highway Systems [74], or Cooperative Vehicle Infrastructure Systems [42]. In this chapter just as in the rest of the thesis we will use IVHS as a generic word to indicate (a mixture of) these systems.

In IVHS every vehicle contains an automated system that can take over the driver's tasks in steering, braking, and throttle control. This complete automation of driving tasks allows to arrange the vehicles in closely spaced groups called platoons. In the platooning approach cars travel on the highway in platoons with minimum small distances (e.g., 2 m) between vehicles within the platoon, and much larger distances (e.g., 30–60 m) between different platoons. High speeds and short intraplatoon spacings allow more vehicles to be accommodated on the network, which substantially increases the maximal traffic flows [149]. Moreover, compared to the situation with human drivers, the full automation present in IVHS also has a positive effect on delays and reaction times. In practice, traffic congestion results in a phenomenon called capacity drop [58], which causes the expected maximum outflow from the jammed traffic to be less than in the case of free-flow traffic. This is mainly due to the delay in reaction time and due to the disturbance caused by lane changes (if any lanes are available) and the increased intervehicle distance when vehicles start to exit from a traffic jam [95]. For human drivers the capacity drop is typically of the order of 2–7%. With fully automated vehicles the capacity drop can be reduced to almost 0%, which results in an even more efficient use of the available infrastructure.

In the proposed approach platooning is integrated with conventional traffic control measures such as dynamic speed limits, route guidance, ramp metering, lane closures, etc. More specifically, we will consider MPC approach (Model Predictive Control) to determine appropriate speed limits and lane allocations for the platoons within the IVHS and appropriate release times of vehicles or platoons that enter the IVHS through on-ramps so as to optimise the performance of the traffic system. Possible performance measures in this context are throughput, travel times, safety, fuel consumption, robustness, etc.

The chapter is organised as follows. In Section 4.2 we briefly recapitulate<sup>1</sup> the hierarchical IV-based traffic control framework of Chapter 3. Section 4.3 describes the general principles of MPC. Next, we explain in Section 4.4 how MPC can be adapted for traffic management and control in IVHS. In Section 4.5 we apply the proposed approach to a case study based on simulations and we illustrate the potential effects of the proposed approach on the traffic flow performance of an IVHS. Section 4.6 concludes the chapter.

## 4.2 A hierarchical framework for IV-based traffic management

First we briefly present the hierarchical control framework for IVHS we have proposed in Chapter 3 and which is closely related to the PATH framework [132]. The framework of Chapter 3 distributes the intelligence between roadside infrastructure and vehicles, and uses IV-based control measures to prevent congestion and/or to improve the performance of the traffic network.

---

<sup>1</sup>The reason for including this recapitulation is that this allows the chapters of the PhD thesis to be read independently from each other.

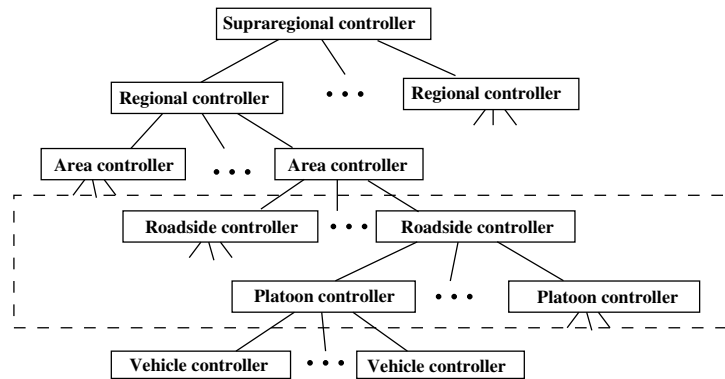


Figure 4.1: IV-based framework of Chapter 3. The focus of the current chapter is indicated by the dashed box.

The control architecture of Chapter 3 is based on the platoon concept and consists of a multi-level control structure with local controllers at the lowest level and one or more higher supervisory control levels as shown in Figure 4.1. The layers of the framework can be characterised as follows:

- *Higher-level controllers* (such as area, regional, and supraregional controllers) provide network-wide coordination of the lower-level and middle-level controllers as well as long-distance route assignment and route planning. E.g., the activities of a group of roadside controllers could be supervised by an area controller. In turn, a group of area controllers could be supervised or controlled by regional controllers, and so on.
- *Roadside controllers* use IV-based control measures to improve the traffic flow. A roadside controller may control a part of a highway, an entire highway, or a collection of highways. The main tasks of the roadside controllers are to assign desired speeds and lanes for each platoon, safe distances to avoid collisions between platoons, desired platoon sizes depending on the traffic conditions, to provide dynamic route guidance for the platoons (within the region controlled by the roadside controller), and to instruct for merges, splits, and lane changes of platoons.
- *Platoon controllers* receive commands from the roadside controllers and are responsible for control and coordination of the vehicles inside the platoon. The platoon controllers are mainly concerned with actually executing the interplatoon manoeuvres (e.g., merges, splits, and lane changes) and with intraplatoon activities (e.g., maintaining safe intervehicle distances).
- *Vehicle controllers* present in each vehicle receive commands from the platoon controllers (e.g., set-points or reference trajectories for speeds, headways, and paths) and they translate these commands into control signals for the vehicle actuators (e.g., throttle, braking, and steering actions).

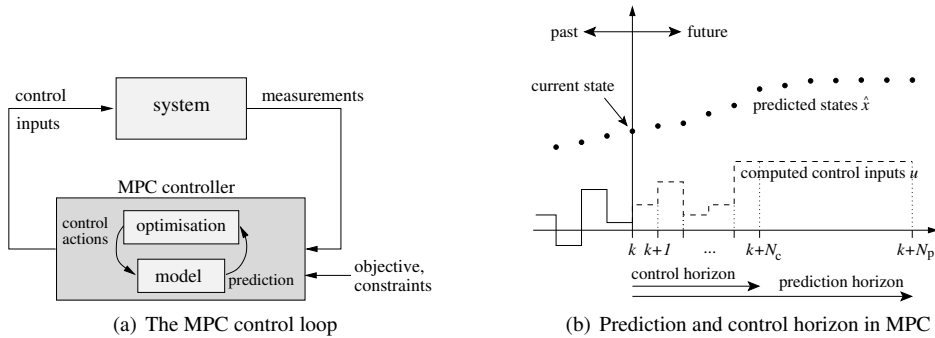


Figure 4.2: Schematic representation of MPC

For a more extensive description of the framework and its main advantages and extensions with respect to the state of the art, the interested reader is referred to Chapter 3.

In the remainder of the chapter we will focus on the roadside controllers and on their interaction with the platoons and the platoon controllers. Note that the roadside controller considers each platoon in the highway network as one single entity. This significantly reduces the complexity of the control problem compared to the case where each individual vehicle would be controlled by the roadside controller. As a consequence, the whole traffic network can be managed more efficiently.

In this chapter we also consider the interface between the IVHS network (i.e., the fully automated road network), and the non-automated road network, where drivers still have full manual control over their vehicle. The interface consists of on-ramps, at which the IVHS control architecture will take over control of the vehicles and arrange them in platoons. The roadside controllers of the IVHS control structure then determine the release times of these platoons into the IVHS network.

### 4.3 Model Predictive Control (MPC)

In this section we briefly present the general principles of Model Predictive Control (MPC) [101] (see Figure 4.2).

MPC is an on-line, sampling-based, discrete-time receding horizon control approach that uses (numerical) optimisation and an explicit prediction model to determine the optimal values for the control measures over a given prediction period. One of the main advantages of MPC is that it can handle various hard constraints on the inputs and states of the system. In addition, MPC has a built-in feedback mechanism due to the use of a receding horizon approach, and it is easy to tune.

MPC works as follows. Let  $T_{\text{ctrl}}$  be the control time step, i.e., the time interval between two updates of the control signal settings. At each control step  $k$  (corresponding to the time instant  $t = kT_{\text{ctrl}}$ ), the roadside controller first measures or determines the current state  $x(k)$  of the system. Next, the controller uses an optimisation algorithm in combination with a model of the system to determine the control sequence  $u(k), \dots, u(k+N_p-1)$  that optimises a given performance criterion  $J_{\text{perf}}(k)$  over a time interval  $[kT_{\text{ctrl}}, (k+N_p)T_{\text{ctrl}}]$  subject to the

operational constraints. Here  $N_p$  denotes the prediction horizon. In order to reduce the computational complexity, one often introduces a constraint of the form  $u(k+j) = u(k+j-1)$  for  $j = N_c, \dots, N_p-1$ , where  $N_c (< N_p)$  is called the control horizon.

The optimal control inputs are then applied to the system in a receding horizon approach as follows. At each control step  $k$  only the first control sample  $u^*(k)$  of the optimal control sequence  $u^*(k), \dots, u^*(k+N_c-1)$  is applied to the system. Next, the prediction horizon is shifted one step forward, and the prediction and optimisation procedure over the shifted horizon are repeated using new measurements. This receding horizon approach introduces a feedback mechanism, which allows to reduce the effects of possible disturbances and mismatch errors.

## 4.4 MPC for IVHS

In this section we explain in detail how MPC can be used for traffic management and control of IVHS. We focus on the roadside controller, and in particular on how MPC can be applied for speed control, lane allocation, and on-ramp control in IVHS.

### 4.4.1 States and control inputs

Recall that at every control step the MPC controller measures or estimates the current state of the traffic network. Since the roadside controllers work with platoons as basic entities, in our case the state of the system includes the positions, lanes, and speeds of the platoon leaders and the lengths of the platoons, as well as the number of platoons waiting at the mainstream origins and at the on-ramps of the IVHS network.

The control signal consists of the speed limits for the platoon leaders, lane allocations for the platoons, on-ramp release times, etc. All these control inputs will be updated at every control step. Note that in principle the platoon size (and the resulting split or merge decisions for platoons) could also be a decision variable. However, to reduce the computational complexity, we may update the platoon sizes at a slower rate than the other control variables. Alternatively, we could assume that the platoon sizes can only change at the boundaries of the region controlled by a roadside controller and are thus fixed for platoons already in the network.

### 4.4.2 Performance criterion and constraints

Possible performance criteria  $J_{\text{perf}}(k)$  for MPC for IVHS are the total time spent in a traffic network, the total throughput, the total fuel consumption, safety, or a combination of these, all evaluated over the time period  $[kT_{\text{ctrl}}, (k+N_p)T_{\text{ctrl}}]$ .

Moreover, in order to prevent oscillations and frequent shifting in the control signals, one often adds a penalty on variations in the control signal  $u$ , which results in the total performance function

$$J_{\text{tot}}(k) = J_{\text{perf}}(k) + \alpha \sum_{j=0}^{N_c-1} \|u(k+j) - u(k+j-1)\|_2, \quad (4.1)$$

at control step  $k$ , where  $\alpha > 0$  is a weighting factor.

The MPC controller can also explicitly take into account operational constraints such as minimum separation between the platoons, minimum and maximum speeds, minimum headways, or in case a vehicle has to take an exit, a constraint that it should have moved in the rightmost lane  $x$  km before the exit, etc.

### 4.4.3 Optimisation methods

Solving the MPC optimisation problem (i.e., computing the optimal control actions) is the most demanding operation in the MPC approach. In our case the MPC approach gives rise to nonlinear nonconvex optimisation problems that have to be solved on-line. Moreover, in general there will be continuous variables (dynamic speed limits, metering rates, release times, etc.) as well as integer variables (lane allocation, platoon size, etc.). Hence, a proper choice of optimisation techniques that suit the nature of the problem has to be made. In our case global or multi-start local optimisation methods are required such as multi-start sequential quadratic programming [118, Chapter 5] or multi-start pattern search [6] in case there are only continuous variables, or branch-and-bound algorithms [40], genetic algorithms [46], or simulated annealing [49] in the mixed integer case.

### 4.4.4 Prediction models for IVHS

An important factor that determines the choice of the model to be used in MPC is the trade-off between accuracy and computational complexity since at each control step  $k$  the model will be simulated repeatedly within the on-line optimisation algorithm. As a consequence, very detailed microscopic traffic simulation models are usually not suited as MPC prediction model. Instead, simplified or more aggregate models are used.

In this section we describe some (simplified) traffic models that could be used as (part of the) prediction model within the MPC-based roadside controller. Note, however, that the proposed MPC approach is generic and modular, so that in case a given prediction model does not perform well, it can easily be replaced by another, more complex prediction model.

Since in the case study of Section 4.5 we will compare the platoon-based approach with human drivers, we will discuss models both for human drivers and for intelligent vehicles and platoons.

#### Vehicle models

We use general kinematic equations to describe the behaviour of the vehicles, which, after discretisation leads to:

$$x_i(\ell+1) = x_i(\ell) + v_i(\ell)T_{\text{sim}} + 0.5a_i(\ell)T_{\text{sim}}^2 \quad (4.2)$$

$$v_i(\ell+1) = v_i(\ell) + a_i(\ell)T_{\text{sim}} \quad (4.3)$$

where  $\ell$  is the simulation step counter,  $T_{\text{sim}}$  the simulation time step,  $x_i(\ell)$  the longitudinal position of the rear of vehicle  $i$  at time  $t = \ell T_{\text{sim}}$ ,  $v_i(\ell)$  the speed of vehicle  $i$  at time  $t = \ell T_{\text{sim}}$ , and  $a_i(\ell)$  the acceleration for vehicle  $i$  at time  $t = \ell T_{\text{sim}}$ . The acceleration used in (4.2)–(4.3) is calculated according to the current driving situation as will be explained below. In addition, the acceleration is limited between a maximum acceleration  $a_{\text{acc,max}}$  and a maximum

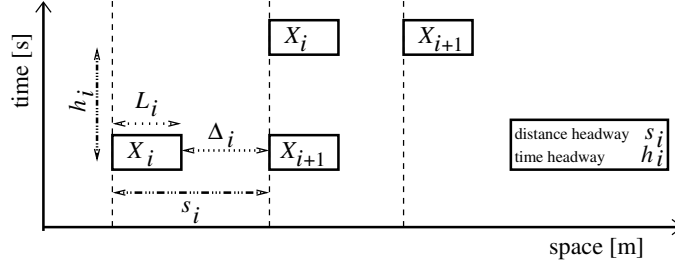


Figure 4.3: Time-space plot illustrating the definition of the time headway  $h_i$  and the distance headway  $s_i$  for vehicle  $i$

(in absolute value) comfortable deceleration  $a_{\text{dec,max}}$ .

$$a_i(\ell) = \text{sat}(a_{\text{target},i}(\ell), a_{\text{dec,max}}, a_{\text{acc,max}}) \quad (4.4)$$

where  $\text{sat}$  is the saturation function<sup>2</sup> and  $a_{\text{target},i}(\ell)$  is the reference acceleration computed via the formulas given in the next subsections. Also, it is assumed that the vehicle length ( $L_i$ ) is the distance from the rear to the front of the vehicle. The distance between the front of the following vehicle and the rear of the vehicle in front is called the following distance  $\Delta_i$ . The time headway  $h_i$  of a vehicle is defined as the time difference between the passing of the rear ends of the vehicle's predecessor and the vehicle itself at a certain location. When the speeds of the vehicles are considered to be constant, then the time headway is the amount of time necessary for vehicle  $i$  to reach the current position of vehicle  $i+1$ . The distance headway of a vehicle  $s_i$  is defined as the distance between the rear of the vehicle and the rear of its predecessor vehicle at a specific instant of time. The distance headway  $s_i$  is equal to the sum of the vehicle length  $L_i$  and its following distance  $\Delta_i$ .

These definitions are illustrated using two vehicles  $i$  and  $i+1$  travelling on a single lane in a time-space coordination system as shown in Figure 4.3. The time and distance directions are shown in vertical and horizontal axes respectively. The time headway is more often considered rather than distance headway because of its simplicity and ease of measuring. In practice, the distance headway can be obtained from photos.

### Longitudinal models for human drivers

Now we describe the longitudinal behaviour of the vehicles. First we consider models for human drivers, and in the next subsection we discuss models for the IVs and for the platoons.

For human-driver models we distinguish between free-flow and car-following behaviour:

- *Free-flow model*: The acceleration for free-flow driving conditions is determined by the delayed difference between the current speed and the reference speed:

$$a_{\text{target},i}(\ell) = K(v_{\text{ref},i}(\ell - \sigma) - v_i(\ell - \sigma)) \quad (4.5)$$

<sup>2</sup>The saturation function  $\text{sat}$  is defined as follows  $\text{sat}(x, U, L)$  equal to  $x$  if  $L \leq x \leq U$ , equal to  $U$  if  $x > U$  and equal to  $L$  if  $x < L$ .

where  $K$  is a model parameter,  $v_{\text{ref},i}$  is the reference speed, and  $\sigma$  is the reaction delay. The reference speed  $v_{\text{ref},i}$  can either be issued by the roadside controller or it can be the driver's desired speed or the legal maximum speed for the road the vehicle is currently on. In connection with the reaction delay  $\sigma$  we assume that the corresponding reaction time  $T_{\text{react}}$ , which typically has a value of 1–1.2 s, is an integer multiple of the simulation time step  $T_{\text{sim}}$ . As a result,  $\sigma = \frac{T_{\text{react}}}{T_{\text{sim}}}$  is an integer.

- *Car-following model:* As described in [27] there exist various types of car-following models such as stimulus response models [102], collision avoidance models [87], psychophysical models [105], and cellular automata models [109].

We will use a stimulus response model to describe the behaviour of human drivers. Stimulus response models are based on the hypothesis that each vehicle accelerates or decelerates as a function of the relative speed and distance between the vehicle and its predecessor. In particular, the Gazis-Herman-Rothery (GHR) model [55] states that after a reaction delay, the follower vehicle  $i$  accelerates or decelerates in proportion to the speed of the vehicle itself, to the relative speed with respect to its predecessor (vehicle  $i+1$ ), and to the inverse of distance headway between them. The acceleration is thus given by

$$a_{\text{target},i}(\ell) = C v_i^\beta(\ell) \frac{(v_{i+1}(\ell-d) - v_i(\ell-d))}{(x_{i+1}(\ell-d) - x_i(\ell-d))^\gamma}, \quad (4.6)$$

where  $C$ ,  $\beta$ , and  $\gamma$  are the model parameters (possibly with different values depending on whether the vehicle is accelerating or decelerating), and  $d$  is the driver delay. Here we assume again that  $T_{\text{delay}}$  the delay time, which typically has a value of 1–1.2 s, is an integer multiple of  $T_{\text{sim}}$ . So,  $d = \frac{T_{\text{delay}}}{T_{\text{sim}}}$  is an integer. We have selected this model for our research mainly due to the simple formulation used by the model to describe the car-following behaviour.

### Longitudinal models for platoons in IVHS

In our approach, the intelligent vehicles within the platoons use adaptive cruise control (ACC) and intelligent speed adaptation (ISA) measures and are arranged in platoons. We now discuss how the accelerations for the platoon leaders and for the follower vehicles within a platoon are calculated:

- *Platoon leader model:* Platoon leaders have an enforced-ISA system and the calculation of the acceleration for the platoon leader is based on a simple proportional controller:

$$a_i(\ell) = K_1 (v_{\text{ISA}}(\ell) - v_i(\ell)), \quad (4.7)$$

where  $K_1$  is the proportional constant, and  $v_{\text{ISA}}$  is the reference ISA speed provided by the roadside controller.

- *Follower vehicle model:* The follower vehicles in a platoon will use their on-board ACC system to maintain short intraplatoon distances. The cooperative ACC algorithm



consists of a combined speed and distance controller:

$$a_i(\ell) = K_2(h_{\text{ref},i}(\ell) - (x_{i+1}(\ell) - x_i(\ell))) + K_3(v_{\text{platoonleader}}(\ell) - v_i(\ell)) \quad , \quad (4.8)$$

where  $K_2$  and  $K_3$  are constants, and  $h_{\text{ref},i}$  is the reference distance headway for vehicle  $i$ . Note that the speed controller is based on the same principle as the one used in the platoon leader model, but with the speed of the platoon leader as the reference speed. The distance controller calculates the safe distance headway as follows:

$$h_{\text{ref},i}(\ell) = S_0 + v_i(\ell)T_{\text{head},i} + L_i \quad , \quad (4.9)$$

where  $S_0$  is the minimum safe distance headway that is to be maintained at zero speed,  $T_{\text{head},i}$  is the desired time headway for vehicle  $i$ , and  $L_i$  is the length of vehicle  $i$ .

#### 4.4.5 Platoon-based prediction model

At a more aggregate level, we can also consider a platoon of vehicles as a single entity without taking the detailed interactions among the individual vehicles within a platoon into account. So, essentially we consider a platoon as one vehicle with a length that is a function of the speed of the platoon (due to the dependence of the intervehicle spacing managed by the cooperative ACC on the speed (cf. (4.9)), and of the number and lengths of the vehicles in the platoon. The dynamics equations for the speed and position of the platoon are the same as those of a platoon leader presented above. Consider platoon  $p$  and assume for the sake of simplicity that the vehicles in the platoon are numbered 1 (last vehicle), 2 (one but last vehicle),  $\dots$ ,  $n_p$  (platoon leader). The speed dependent length  $L_{\text{platoon},p}(\ell)$  of platoon  $p$  is then given by

$$L_{\text{platoon},p} = (n_p - 1)S_0 + \sum_{i=1}^{n_p-1} T_{\text{head},i}v_{n_p}(\ell) + \sum_{i=1}^{n_p} L_i \quad , \quad (4.10)$$

where  $S_0$  the minimum safe distance that is to be maintained at zero speed,  $T_{\text{head},i}$  is the desired time headway for vehicle  $i$ ,  $v_{n_p}$  the speed of the platoon (leader), and  $L_i$  the length of vehicle  $i$ .

#### Merging at on-ramps and lane changing for human drivers

In order to model the merging and lane changing behaviour of vehicles, we could — in the interest of simulation speed and efficiency — use the following simplified models.

For individual human-driven vehicles (cf. the case study of Section 4.5 below) we assume that a vehicle on an on-ramp can join the mainstream lane provided that there is a sufficient large gap and that no collision is imminent. If both conditions are satisfied then the vehicle joins the mainstream line with a speed that is equal to that of the immediate predecessor (if present) or equal to the (ISA or legal) speed limit otherwise.

Lane changes can be modelled similarly: if there is a slower vehicle ahead and if the speed of the vehicles in an adjacent lane is higher than that of the vehicle's predecessor in the current lane, the vehicle can join the other lane provided that there is a sufficient large gap and that no collision is imminent. In this case the vehicle's speed should not be modified.

### Merging at on-ramps and lane changing for platoons

In order to model the merging behaviour of platoons at on-ramps and the lane changing behaviour of platoons, we could use a similar simplified model that operates at the platoon level.

We consider each platoon at an on-ramp as one entity that will join the mainstream lane as soon as there is a sufficient large gap (including safety distances) available between the platoons on the mainstream lane and provided that the merging will not result in a collision in the next time steps. If both conditions are satisfied, then the platoon joins the mainstream line (with a speed that is imposed by the roadside controller).

Likewise, if a lane change is imposed on a platoon by the roadside controller, we assume that the platoon moves to the assigned lane as one entity. Note that in this case the roadside controller is responsible for taking care that there is a sufficiently large gap (including safety distances) available between the platoons on the other lane and that the lane change will not result in a collision in the next time steps.

## 4.5 Case study

Now we present a simple case study in which the MPC control strategy for the roadside controller layer that has been described in Section 4.3 is applied. First, we will describe the set-up and the scenario used to evaluate the performance of the proposed approach, the prediction and simulation models used, as well as other implementation details. Next, we will discuss and analyse the results obtained from the simulations.

### 4.5.1 Set-up

To illustrate the proposed MPC approach for the roadside controller we use a basic set-up consisting of a 7 km two-lane highway stretch with one mainstream origin, two on-ramps (located at position  $x = 2000$  km and  $x = 4000$  km), and one destination (see Figure 4.4). We compare three different situations:

- uncontrolled traffic with human drivers,
- controlled traffic with human drivers and with ISA and (conventional) ramp metering as control measures,
- IV-based traffic control with platoons and with dynamic speeds, on-ramp release times, and lane allocations for the platoons as control measures.

For the sake of simplicity all vehicles are assumed to be of the same length ( $L_i = 4$  m for  $i=1,2,\dots,n$ ).

For the controlled situation with human drivers we assume that ISA limits the speed in a hard way and that human drivers cannot surpass the imposed speed limit. Similarly, we assume that the imposed ramp metering rate is adhered to.

In the IV-based case with platoons we assume that all the vehicles are fully automated IVs equipped with advanced communication and detection technologies such as in-vehicle computers and sensors, and with on-board ACC and ISA controllers.

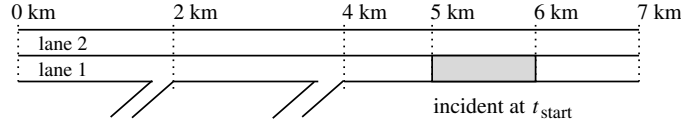


Figure 4.4: Set-up of the case study

### 4.5.2 Scenario

We simulate a period of 9 min starting at time  $t_{\text{start}} = 7:00$  A.M and ending at time  $t_{\text{end}} = 7:09$  A.M. The demand of vehicles is taken to be constant during the simulation period, and equals 1365 veh/h/lane (lane 1) and 1512 veh/h/lane (lane 2) for the mainstream origin and 404 veh/h (on-ramp 1) and 422 veh/h (on-ramp 2) for the on-ramps.

For the proposed scenario the initial state of the network is as follows. We assume that before time  $t_{\text{start}}$  an incident has occurred at position  $x = 5$  km in lane 1, resulting in a blockage in lane 1 from position  $x = 5$  km up to position  $x = 7$  km at time  $t_{\text{start}}$ . This incident blockage will serve as the main bottle-neck in our set-up.

In the upstream sections 3 and 4 of lane 2 (i.e., from position  $x = 2$  km up to  $x = 4$  km) the initial density is 50 veh/km/lane, and in the other sections there are no vehicles. Similarly, in the upstream section 4 of lane 1, the initial density is 25 veh/km/lane. Moreover, at time  $t_{\text{start}}$  the on-ramp and mainstream origin queues are empty. The incident situation continues for the entire simulation period  $[t_{\text{start}}, t_{\text{end}}]$ . During this interval, there is no outflow from the incident region.

The demand profile considered for this particular layout is shown in Figure 4.5.2. Both the mainstream demands raise to a relatively higher level of demand (i.e., more than the capacity of the network) for first 20 min, maintain almost the same higher level of demand for the next 30 min and drop to a low value in the next 10 min. During the first 20 min, the demand on the on-ramps which started at a higher level drops to a reduced level, and maintain a low level of demand for the rest of the simulation period. The network considered has a capacity of 1600 veh/h/lane [83].

### 4.5.3 Models

As indicated above, we are interested in comparing the simulation results obtained for the given scenario using human driving (both without and with control) and using our platoon-based hierarchical approach. For this purpose, we have developed simulation models in Matlab for human driving and platoon driving. For the sake of simplicity and to avoid calibration, we have used the same models for both simulation and prediction purposes in this simulation study.

For the vehicle models we have used the models of Section 4.4.4. In particular, we have used (4.2)–(4.3) with the accelerations given by respectively (4.5)–(4.6) (with  $v_{\text{ref},i}(\ell)$  equal to the legal speed limit of 120 km/h) for uncontrolled human drivers, (4.5)–(4.6) (with  $v_{\text{ref},i}(\ell)$  equal to the ISA speed limit) for human drivers with ISA, and (4.7)–(4.9) for platoons of intelligent vehicles. If we express distances in m, times in s, speeds in m/s, accelerations in  $\text{m/s}^2$ , etc., then the various parameters in these models have the following values (these values inspired by the MITSIM model [156]): For the car-following model (4.6) we

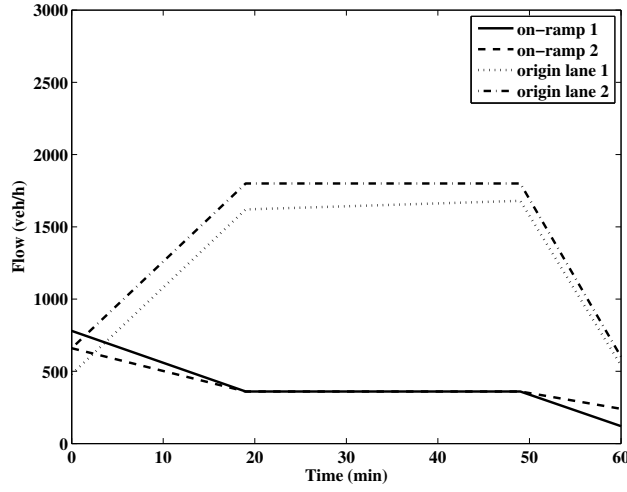


Figure 4.5: The demand scenario considered

have  $C = 1.55$ ,  $\beta = 1.08$ , and  $\gamma = 1.65$  for deceleration, and  $C = 2.55$ ,  $\beta = -1.67$ , and  $\gamma = -0.89$  for acceleration. Furthermore, we have selected  $d = 1$ ,  $\sigma = 1$ ,  $K = 0.01$ , and  $K_1 = 0.4$ . For the follower vehicle model (4.8)–(4.9) we have  $K_2 = 0.6$ ,  $K_3 = 1.2$ ,  $K_4 = 1$ ,  $S_0 = 3$ , and  $T_{\text{head}} = 0.5$  for all vehicles.

For the platoon model (4.10) we have selected  $S_1 = 0.5$ . Moreover,  $a_{\text{acc,max}} = 3$  and  $a_{\text{dec,max}} = -3$  for all models.

If there is a congestion in a segment of the highway, then the maximum outflow from this congested segment will become less when compared to free-flow traffic due to the capacity drop. The value of the capacity drop due to congestion in our case is around 7 % for human drivers (both in the controlled and the uncontrolled case) and almost 0 % for platoons (due to the full automation). For human drivers the capacity drop is included by setting the reaction delay  $d$  in the car-following model (4.6) equal to  $d = 4$  for the first vehicle that leaves the situation, and by reducing this reaction delay every simulation step with 1, until it gets back to the regular value of  $d = 1$ . The threshold speeds for determining whether or not a given vehicle is in a congested or uncongested situation are 30 km/h and 50 km/h respectively (in between the previous congestion state is preserved; so the capacity drop model contains hysteresis).

The time step  $T_{\text{sim}}$  for the simulations is set to 1 s.

#### 4.5.4 Control problem

##### Objective function

The goal of our traffic controller is to improve the traffic performance. The objective that we consider is minimisation of the total time spent (TTS) by all the vehicles in the network using dynamic speed limits, lane allocations (for the platoons), and on-ramp metering as the

control handles. The TTS for the entire simulation period can be expressed as

$$J_{\text{TTS, sim}} = \sum_{\ell=0}^{N_{\text{sim}}} (n_{\text{veh}}(\ell) + q_{\text{main}}(\ell) + q_{\text{on}}(\ell)) T_{\text{sim}} \quad , \quad (4.11)$$

where  $N_{\text{sim}} = 540$  is the total number of simulation steps (of length  $T_{\text{sim}} = 1$  s) within the entire simulation period of 9 min,  $n_{\text{veh}}(\ell)$  is the number of vehicles that are present within the network at time  $t = \ell T_{\text{sim}}$ ,  $q_{\text{main}}(\ell)$  is the number of vehicles in the queue at the mainstream origin at time  $t = \ell T_{\text{sim}}$ , and  $q_{\text{on}}(\ell)$  is the number of vehicles present in the on-ramp queues at time  $t = \ell T_{\text{sim}}$ .

The corresponding performance function  $J_{\text{perf}}(k)$  used in the MPC approach at control step  $k$  is then given by

$$J_{\text{perf}}(k) = \sum_{\ell=kM}^{(k+N_p)K-1} (n_{\text{veh}}(\ell) + q_{\text{main}}(\ell) + q_{\text{on}}(\ell)) T_{\text{sim}} \quad ,$$

with  $M = \frac{T_{\text{ctrl}}}{T_{\text{sim}}}$  (note that as we will select the control time step  $T_{\text{ctrl}}$  to be an integer multiple of the simulation time step  $T_{\text{sim}}$ ,  $M$  will be an integer). In the total MPC objective function we also include a penalty term with  $\alpha = 0.01$  (cf. (4.1)).

### Control inputs

For the controlled human situation the applied control measures are (conventional) on-ramp metering and ISA (with one speed limit for each section of 1 km length between position  $x = 0$  km and position  $x = 5$  km).<sup>3</sup>

So in this case the control signal  $u$  for the MPC problem of control step  $k$  includes the ISA speed limits for the first 5 sections and the ramp metering rates for two on-ramps (expressed as a number between 0 and 1) at control steps  $k$  up to  $k + N_c - 1$ , i.e., we have  $7N_c$  variables in total.

For the platoon-based approach the control signal  $u$  for the MPC problem of control step  $k$  includes speed limits and lane allocations for all platoons that are or will be present in the network during the control horizon period as well as the on-ramp release times for the platoons at the on-ramp during the control horizon period.

So if  $P_k$  is the number of platoons that are present in the network at time  $t = kT_{\text{ctrl}}$  or that could enter the network via the mainstream origin between time  $t = kT_{\text{ctrl}}$  and up to time  $t = (k + N_c)T_{\text{ctrl}}$ , and if  $Q_k$  is the number of platoons that could enter the network via the on-ramp between  $t = kT_{\text{ctrl}}$  and  $t = (k + N_c)T_{\text{ctrl}}$ , we have  $2P_k N_c + 2Q_k (N_c - 1) + Q_k$  variables in total. To keep the number of optimisation variables fixed during the optimisation process and to take the future demand of platoons into account, we use autonomous control for vehicles entering between  $t = (k + N_c)T_{\text{ctrl}}$  and  $t = (k + N_p)T_{\text{ctrl}}$ . By autonomous control, we mean that the platoons on the highways would be assigned with the speed of its immediate predecessor (platoon). If no predecessor platoon exists then the free-flow speed will be assigned to the uncontrolled autonomous platoon. Each platoon will be assigned the same lane in which it originated, until it is forced to perform a lane change due to any incident

<sup>3</sup>Note that considering speed limits in the remaining sections is not necessary in the proposed scenario as for these sections setting the speed limits equal to the legal speed limit of 120 km/h yields an optimal solution.

blockages. The platoons from on-ramps are allowed to enter the highway once the highway–on-ramp merging location can accommodate the entire platoon.

As in this case study we focus on dynamic speed limits and lane allocations for each platoon and on on-ramp metering, the platoon size is not yet considered to be a control variable, but it is kept fixed at 25 for all platoons.

### Constraints

In the platoon-based approach the roadside controller has to take care of maintaining safe interplatoon distances. This condition is included as a constraint in the MPC optimisation problem. In particular, the minimal safe distance between the front end of a platoon  $p_1$  and the rear end of its immediate predecessor platoon  $p_2$  in the same lane is given by:

$$S_{0,\text{platoon}} + T_{\text{head,platoon}} v_{\text{platoon},p_1} ,$$

where  $v_{\text{platoon},p_1}$  is the speed of platoon  $p_1$ . For the case study we have selected  $S_{0,\text{platoon}} = 20$  m and  $T_{\text{head,platoon}} = 1.2$  s. Moreover, we consider a maximum speed of 120 km/h for both the human drivers and the platoon leaders.

### Horizons

The control time step  $T_{\text{ctrl}}$  is set at 1 min. The prediction horizon  $N_p$  is assumed to be the average time required for a vehicle to travel the entire network using the average speed. So for our case study we have taken a value that corresponds to 7 min ( $N_p = 7$  km/60 km/h = 0.1 h = 7 min). For the control horizon  $N_c$  we have selected a value that corresponds to 3 min so as to limit the number of optimisation variables.

### Optimisation algorithms

As we consider dynamic speed limits, on-ramp metering, and lane allocation as control measures there will be both continuous and integer variables in the MPC optimisation problem. For the optimisation we have used the `patternsearch` command incorporated in the Genetic Algorithm and Direct Search Toolbox of Matlab for the continuous optimisation problems (i.e., the determination of the speeds and on-ramp metering rates for the controlled human case) and the `glcFast` command of the Matlab/Tomlab toolbox [142] for the mixed integer optimisation problems (i.e., the determination of the speeds, on-ramp release times, and lane allocation for the platoon case). The `patternsearch` command implements a pattern search algorithm [6] and the `glcFast` command implements an extended version of the DIRECT algorithm of [79]. Both methods have been executed multiple times with different initial starting points so as to get a good approximation of the optimal solution. For the controlled human driver scenario we have used 20 multi-start points. For the platoon-based approach we have used bilevel optimisation with binary optimal on lane allocation (since limited number of variables, brute force enumeration can be used) and with real-valued optimal on speeds and on-ramp release times. In particular, for the real-valued variables, we have used 30 multi-start points. We have opted to use these particular optimisation algorithms because our simulation experiments have shown that they provide a good trade-off between optimality and speed.

Table 4.1: Results for the three approaches

Case	TTS (veh.h)	Relative improvement	no of control variables for $N_p=7$ and $N_c=3$
uncontrolled case	27.44	0 %	0
controlled (human drivers)	24.72	9.91 %	$7N_c$
controlled (platoons)	20.67	24.67 %	$2P_kN_c + 2Q_k(N_c - 1) + Q_k$

#### 4.5.5 Results and analysis

For the scenario presented above, closed-loop MPC simulations have been carried out. The results of the simulations are reported in Table 4.1. In particular, we indicate the total time spent by all vehicles in the network during the entire simulation period of 9 min.

We will analyse the scenario for the reference case (uncontrolled case) as follows. The total capacity of the considered network is 3200 veh/h. The total demand equals 3703 veh/h (1365 veh/h+1512 veh/h+404 veh/h+422 veh/h), which is higher than the total capacity. In the uncontrolled case, the congestion will mainly occur on the mainstream, as the vehicles from on-ramps will always try to enter the mainstream. The capacity available for the mainstream demands will be  $3200-422 = 2778$  veh/h (allowed capacity - demand on on-ramp 2), which is less for the original demand (3281 veh/h (i.e., demand from mainstream origin and from on-ramp 1)). This exceeding demand situation, when not controlled properly will lead to traffic jams.

The considered set-up and scenario can be analysed in three categories as follows: segments upstream the bottleneck, the bottleneck segment, and segments downstream the bottleneck. The segments far upstream the bottleneck i.e., segments from 1 to 4 allow vehicles to travel at their free-flow speed. However, segment 3 may get congested if the demand from on-ramp 1 exceeds the allowed capacity. Due to the constant, high demand from on-ramp 2 and from the mainstream origins, and due to the bottleneck caused by the incident, congestion is expected to occur on segment 5. In the bottleneck segment 6, the available capacity can be utilised fully by the vehicles, but at the price of low mean speeds. The outflow capacity from segment 6 cannot reach the maximum flow, as more vehicles are not allowed to travel through the bottleneck.

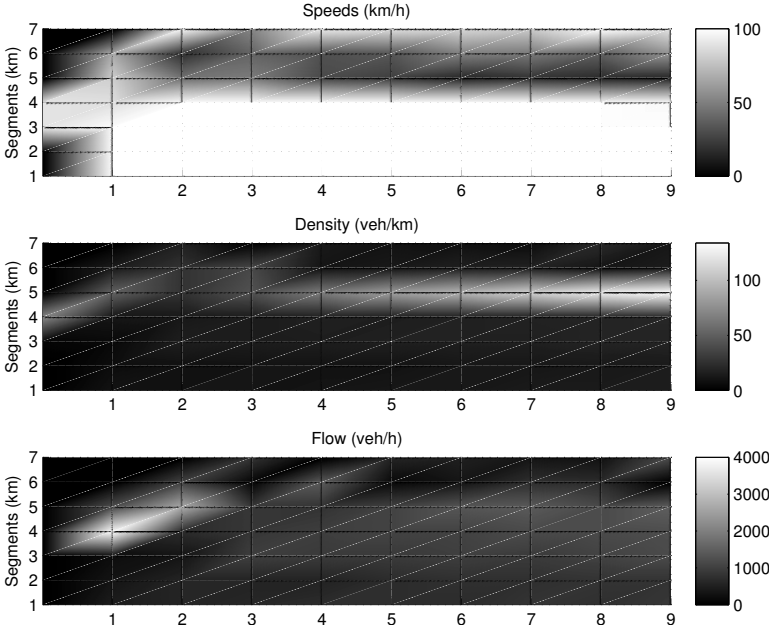
Figure 4.6 show the simulation results for the uncontrolled case in two different views for a better understanding. In Figure 4.6(a), the traffic flow direction is from the bottom to the top, and since all segments have the same length, this axis should be interpreted as the distance travelled by the vehicles. During the entire simulation, the speed limits for all the segments are set to 120 km/h. The first plot in Figure 4.6(a) shows the mean speeds at the segments. Light colours represent high mean speeds. The dark area starting at the 5<sup>th</sup> segment after 1 min denotes low speeds and the reason for the decreased speed (i.e., increased density) can be interpreted as follows. When a driver is confronted with an incident on segment 6 on the same lane (lane 1), he starts to decelerate in order to avoid a collision. In case there is no space in lane 2 or in case the speed on lane 2 is almost the same as on lane 1, the driver waits and stays on lane 1 until the incident eventually gets cleared. However, once there is a possibility to perform a safe lane change manoeuvre, the driver moves to lane 2. In the uncontrolled case there is no ramp metering action that can

prevent or delay an extra flow of vehicles from entering the mainstream highway. However, the increasing density in lane 2 due to the effects of the incident in lane 1 causes congestion, which in its turn leads to a capacity drop for vehicles leaving the traffic jam. Next, the flow plot also shows the traffic jam and is visible as dark area in segment 5 indicating lower flow. In the density plot, there are vehicles in the upstream segments. Since they are less in number than the congested number of vehicles, they are not clearly visible. In Figure 4.6(b) we can visualise the increase in density in the upstream segments. Also, the irregularities seen in the mean-speeds of segments 6 and 7 in Figure 4.6(b) can be explained as a consequence of stop-and-go behaviour of the vehicles leaving the congested area. Once the traffic congestion sets in, both the mainstream vehicles and the on-ramp vehicles drive on and have to wait in a queue until the traffic jam dissolves. All this results in a large time spent in the network for the vehicles, and thus also in a higher value of the TTS for the entire simulation period.

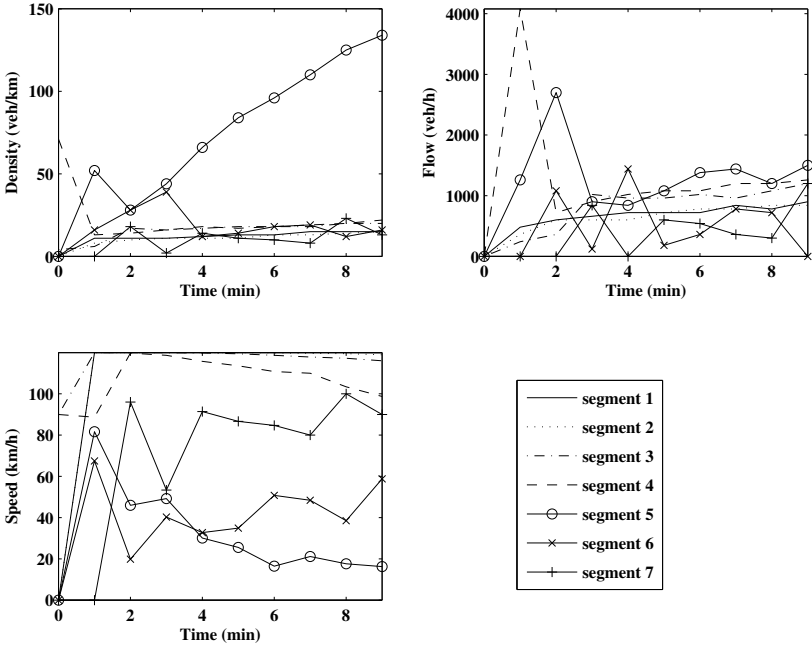
The MPC-human controlled case for the same situation considered above, the speed limit becomes active and reduce the inflow from the mainstream, and ramp metering gradually switches on and keeps the total outflow high as shown in Figure 4.7. For this case, the MPC approach can predict the presence of an incident and prevent it or diminish its negative impact by slowing down vehicles (using speed limits) or delaying vehicles (via on-ramp metering) before they reach the incident. In Figure 4.7(a), we can see that the speed limit control in segments 4 and 5 has delayed the vehicles from getting nearer to the bottleneck area (since the congestion will then be dissolved or at least less severe by the time the vehicles then reach the congested area) and has provided entry space for the on-ramp vehicles. Since the traffic maintains a homogenous traffic flow, the density plot and the flow plot in Figure 4.7 are not showing the details in precise. In Figure 4.7(b), we can clearly see that the ramp metering signals for on-ramp 2 has been regulated such that the traffic flow entering via the on-ramp 2 does not cause congestion in segment 5. This controlled approach with human drivers, ISA control, and ramp metering yields an improvement in TTS over the uncontrolled case of about 10%. Thus in the MPC-human controlled case, the segments where the speed limits can influence the traffic flow are segments 1 to 5. The speed limits and the ramp metering rates provided by the MPC-based roadside controller for the human drivers are shown in Figure 4.8.

As the platoon controlled case is not a segment based approach, the plots in Figure 4.9(b) do not provide indepth details of the traffic situation. Thus we will consider the plots in Figure 4.9(a) for explanation of the platoon controlled traffic situation. From the speed plot in Figure 4.9(a), we can see that the platoons are allowed to travel at higher speeds through the segments. The main idea behind speed limit control and on-ramp release time control for platoons is the same as for human controlled approach. Moreover, the full automation for IVHS allows to maintain small intervehicle distances (so that more cars are allowed to traverse the network more quickly) even in the case of possible congestion in segment 5 (due to the incident and the on-ramp 2 demand) and it results in an almost 0% capacity drop. The additional performance improvement obtained by our approach is caused by the optimal lane allocation and the full automation in addition to speed limits and ramp metering. The lane allocation control measure also helps to better react to the incident and to allow for lane changes for platoons that would otherwise be blocked in front of the congested region. IV-based traffic with platoons results in the best performance with an improvement of about 24.6% with respect to the uncontrolled case and of about 16.39%



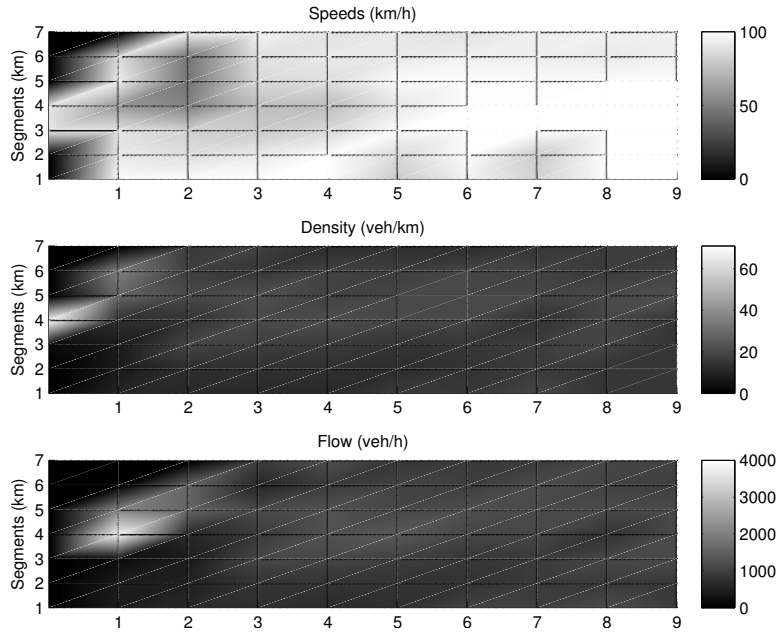


(a) Simulations for uncontrolled case

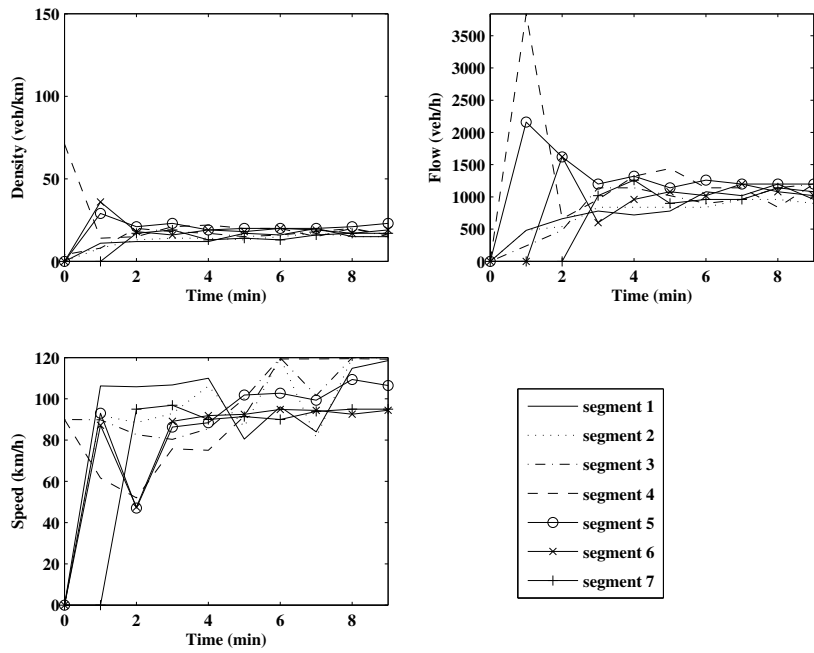


(b) Simulations for uncontrolled case (in numbers)

Figure 4.6: Simulations for human drivers uncontrolled case



(a) Simulations for human controlled case



(b) Simulations for human controlled case (in numbers)

Figure 4.7: Simulations for human drivers controlled case

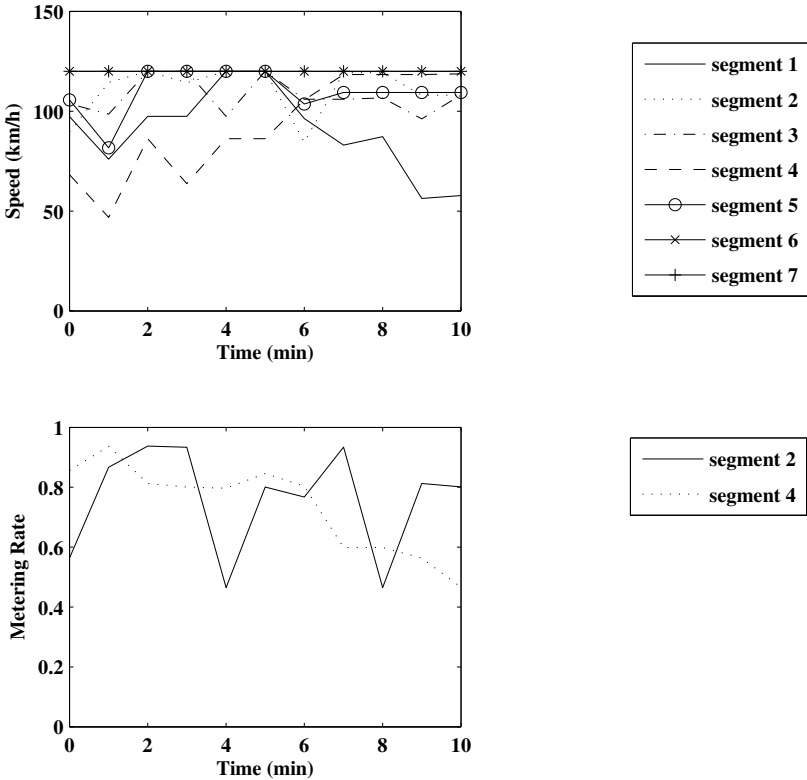
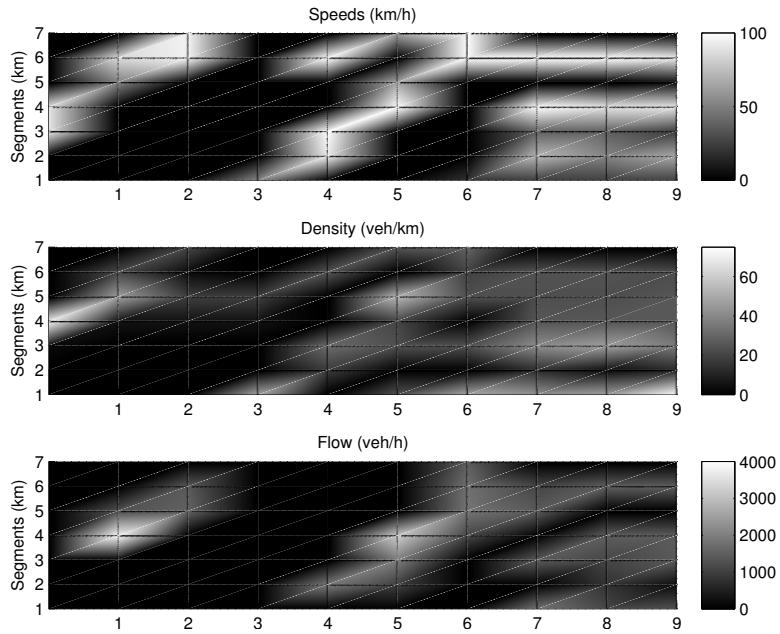
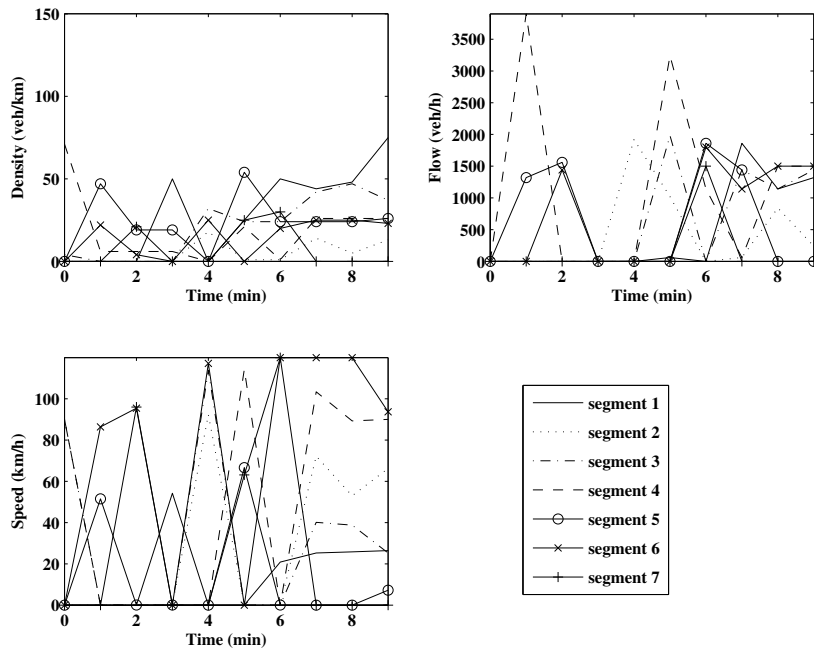


Figure 4.8: The speed limits and ramp metering rates for controlled case



(a) Simulations for platoon controlled case



(b) Simulations for platoon controlled case (in numbers)

Figure 4.9: Simulations for platoon controlled case

with respect to the controlled human case.

## 4.6 Summary

We have presented how model predictive control (MPC) can be used to determine optimal platoons speeds, lane allocations, platoon sizes, platoon release times at on-ramps, etc. in Intelligent Vehicle Highway Systems (IVHS). The proposed approach has been illustrated using a case study based on simulations and with dynamic speed limits, lane allocation, and on-ramp metering as control measures. The results of the case study highlight the potential benefits and improvements that can be obtained by using MPC for intelligent speed adaptation in IVHS.

Future research topics include: additional and more extensive case studies, inclusion of additional control measures, development of efficient algorithms, assessment of the effects of model mismatches, explicit consideration of the other levels in the IVHS control hierarchy of [10], and extension to larger networks.



## Chapter 5

# Area Controller

We are considering Intelligent Vehicles Highway Systems (IVHS) consisting of automated highway systems in which intelligent vehicles (IVs) organised in platoons drive to their destination, controlled by a hierarchical control framework. In this chapter, we present a route guidance approach that can be used by the area controllers in the IVHS control framework of Chapter 3. In general, the optimal route choice control problem is a nonlinear integer optimisation problem with high computational requirements, which makes the problem intractable in practice. Therefore, we propose two simplified but fast simulation models to describe the flows of platoons in the network.

For the first model, which only considers flows and queues lengths, we show that the optimal route choice control problem can be approximated by a linear or a mixed-integer linear problem. With a simple case study we illustrate that this results in a balanced trade-off between optimality and computational efficiency.

The second model is a macroscopic model based on an existing traffic flow model for human drivers (METANET), that we adapt to fit the platoon framework. For this model, the optimal route choice control problem results in an optimisation problem with only real-valued optimisation variables. This approach will also be illustrated with a simple case study.

### 5.1 Introduction

The recurring traffic congestion problems and their related costs have resulted in various solution approaches. One of these involves the combination of the existing transportation infrastructure and equipment with advanced technologies from the field of control theory, communication, and information technology. This results in integrated traffic management and control systems, called Intelligent Vehicle Highway Systems (IVHS), that incorporate intelligence in both the roadside infrastructure and in the vehicles. Although this step is considered to be a long-term solution, this approach is capable of offering significant increases in the performance of the traffic system [50, 80, 136].

In IVHS all vehicles are assumed to be fully automated with throttle, braking, and steering commands being determined by automated on-board controllers. Such complete automation of the driving tasks allows to organise the traffic in platoons, i.e., a closely spaced

group of vehicles travelling together with short intervehicle distances [132, 149]. Platoons can travel at high speeds and to avoid collisions between platoons at these high speeds, a safe interplatoon distance of about 20–60 m should be maintained. Also, the vehicles in each platoon travel with small intraplatoon distances of about 2–5 m, which are maintained by the automated on-board speed and distance controllers. By travelling at high speeds and by maintaining short intraplatoon distances, the platoon approach allows more vehicles to travel on the network, which improves the traffic throughput [28, 97].

In Chapter 3 we have proposed a hierarchical traffic management and control framework for IVHS that builds upon earlier research in this field such as the PATH framework [132]. The control architecture of Chapter 3 consists of a multi-level control structure with local controllers at the lowest level and one or more higher supervisory control levels (see also Figure 5.1). In this framework there are roadside controllers that provide speed and lane allocation instructions to the platoons. These roadside controllers typically manage single stretches of highways. A collection of highways is then supervised by so-called area controllers that mainly take care of the route guidance instructions for the platoons and that also coordinate the various roadside controllers in their area. In this chapter, we will concentrate on how the area controllers can determine optimal routes for the platoons using optimal control. Traffic flow models can be used for prediction, simulation, estimation, and for control related tasks. In general, traffic flow models can be categorised as:

- Microscopic flow models,
- Mesoscopic flow models,
- Macroscopic flow models.

Microscopic models represent every vehicle as an autonomous entity and model the interaction among individual vehicles [27, 34, 102]. Microscopic models use sub-models to represent acceleration, lane changing decisions, and gap acceptance behaviour of each vehicle.

On a mesoscopic scale, the behavioural rules for individual vehicles are replaced by principles from statistical mechanics or gas kinetics. This model represents the traffic stream as a packet of vehicles [68, 72, 122].

Macroscopic models consider the traffic flow as a continuum, i.e., a fluid or a gas with specific characteristics [43, 44, 98]. The aggregated variables that are used to describe the dynamics of traffic flow are mean speed, flow, and density.

The choice of the appropriate model depends on the level of detail required and also on the computational requirements. Microscopic traffic flow models become mathematically intractable for large-scale traffic systems due to the high computational requirements. They are in fact more suitable for simulation purposes and for off-line evaluation of developed control strategies than for on-line model-based control. On the other hand, macroscopic models are less detailed and thus less accurate than microscopic traffic flow models, but due to their fast execution they are well-suited for on-line model-based control.

If we would optimise the routes for each platoon using a microscopic traffic flow model, the optimal route choice problem would lead to a mixed-integer optimisation over the routes (integer variables) and e.g., speeds (real variables). Therefore, we will develop a macroscopic traffic model for IVHS in this chapter.



The chapter is organised as follows. In Section 5.2 we briefly recapitulate<sup>1</sup> the hierarchical traffic management and control framework of Chapter 3. Next, we focus on the route guidance tasks of the area controllers and we present a simplified flow model and the corresponding optimal route guidance problem in Section 5.3. We consider both the static (constant demands) and the dynamic case (time-varying demands). In general, the dynamic case leads to a nonlinear nonconvex optimisation problem, but we show that this problem can be approximated using mixed-integer linear programming (MILP). We present a simple example that illustrates that the MILP approximation provides a good trade-off between optimality and computational efficiency. In Section 5.4 we propose an alternative approach based on a macroscopic traffic flow model for human drivers (METANET). We present the basic METANET model for human drivers and we explain how it can be adapted to platoons. Next, we use this model to determine optimal splitting rates at the network nodes. This yields an optimisation problem with real-valued variables only, which thus results in a computation complexity that is much lower than the original mixed-integer optimisation problem. This approach is also illustrated using a case study. Section 5.6 concludes the chapter.

## 5.2 Intelligent vehicle highway systems (IVHS)

We now briefly recapitulate the hierarchical control framework for IVHS we have proposed in Chapter 3. This framework is based on the platoon concept and it distributes the intelligence between the roadside infrastructure and the vehicles using control measures such as intelligent speed adaption, adaptive cruise control, lane allocation, on-ramp access control, route guidance, etc. to prevent congestion and to improve the performance of the traffic network. The control architecture of Chapter 3 consists of a multi-level control structure with local controllers at the lowest level and one or more higher supervisory control levels as shown in Figure 5.1. The layers of the framework can be characterised as follows:

- The *vehicle controllers* present in each vehicle receive commands from the platoon controllers (e.g., set-points or reference trajectories for speeds (for intelligent speed adaption), headways (for adaptive cruise control), and paths) and they translate these commands into control signals for the vehicle actuators such as throttle, braking, and steering actions.
- The *platoon controllers* receive commands from the roadside controllers and are responsible for control and coordination of each vehicle inside the platoon. The platoon controllers are mainly concerned with actually executing the interplatoon manoeuvres (such as merges with other platoons, splits, and lane changes) and intraplatoon activities (such as maintaining safe intervehicle distances).
- The *roadside controllers* may control a part of a highway or an entire highway. The main tasks of the roadside controllers are to assign speeds for each platoon, safe distances to avoid collisions between platoons, appropriate platoon sizes, and ramp metering values at the on-ramps. The roadside controllers give instructions for merging, splitting, and lane changes to the platoons.

<sup>1</sup>The reason for including this recapitulation is that this allows the chapters of the PhD thesis to be read independently from each other.

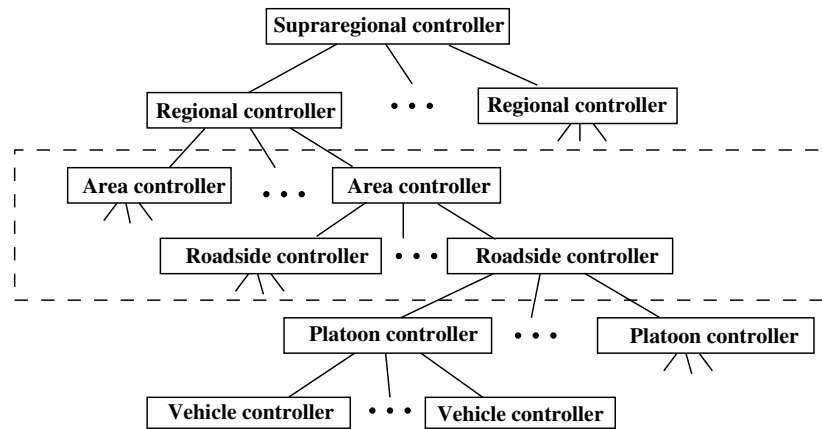


Figure 5.1: The hierarchical control framework of Chapter 3 for IVHS

- The *higher-level controllers* (such as area, regional, and supraregional controllers) provide network-wide coordination of the lower-level and middle-level controllers. In particular, the area controllers provide area-wide dynamic route guidance for the platoons, and they supervise and coordinate the activities of the roadside controllers in their area by providing set-points and control targets. In turn, a group of area controllers could be supervised or controlled by a regional controller, and so on.

The lower levels in this hierarchy deal with faster time scales (typically in the milliseconds range for the vehicle controllers up to the seconds range for the roadside controllers), whereas for the higher-level layers the frequency of updating can range from few times per minute (for the area controllers) to a few times per hour (for the supraregional controllers).

In Chapter 4 and [13–15] we have proposed model predictive control methods for the roadside controllers to determine optimal speeds, lane allocations, and on-ramp release times for the platoons. In the remainder of the chapter we will focus on the area controllers and in particular on how optimal routes can be determined for the platoons.

## 5.3 Optimal route choice control in IVHS using mixed-integer linear programming

### 5.3.1 Approach

In principle, the optimal route choice control problem in IVHS consists in assigning an optimal route to each individual platoon in the network. However, this results in a huge non-linear integer optimisation problem with high computational complexity and requirements, making the problem intractable in practice. So, since considering each individual platoon is too computationally intensive, we will consider streams of platoons instead (characterised by (real-valued) demands and flows expressed in vehicles per hour). The routing problem will be recast as the problem of determining the flows on each link.

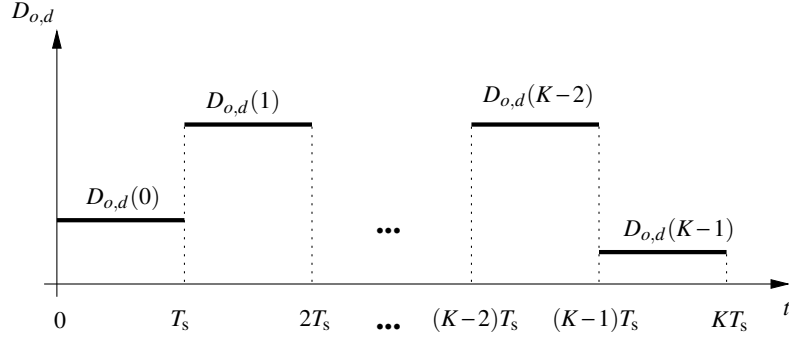


Figure 5.2: Piecewise constant time-varying demand profile  $D_{o,d}$  for the dynamic case

### 5.3.2 Set-up

We consider the following set-up. The highways in the traffic network are considered to be divided into links. We have a transportation network with a set of origin nodes  $\mathcal{O}$ , a set of destination nodes  $\mathcal{D}$ , and a set of internal nodes  $\mathcal{I}$ . Define the set of all nodes as  $\mathcal{V} = \mathcal{O} \cup \mathcal{I} \cup \mathcal{D}$ . Nodes can be connected by one or more (unidirectional) links. The links correspond to freeway stretches. The set of all links is denoted by  $\mathcal{L}$ .

For each origin-destination pair  $(o, d) \in \mathcal{O} \times \mathcal{D}$  we define the set  $L_{o,d} \subseteq \mathcal{L}$  of links that belong to some route going from  $o$  to  $d$ . For every link  $l \in \mathcal{L}$  we define the set  $S_{o,d,l}$  of origin-destination pairs  $(o, d) \in \mathcal{O} \times \mathcal{D}$  such that  $l$  belongs to some route going from  $o$  to  $d$ .

For each pair  $(o, d) \in \mathcal{O} \times \mathcal{D}$ , there is a constant demand  $D_{o,d}$  (in the static case) or a dynamic, piecewise constant demand pattern  $D_{o,d}(\cdot)$  as shown in Figure 5.2 with  $D_{o,d}(k)$  the demand of vehicles at origin  $o$  with destination  $d$  in the time interval  $[kT_s, (k+1)T_s)$  for  $k = 0, \dots, K-1$  with  $K$  the simulation horizon and  $T_s$  the simulation time step (we assume that beyond  $T = KT_s$  the demand is 0).

For each link  $l \in \mathcal{L}$  in the network<sup>2</sup> there is a maximal capacity  $C_l$ . We assume that there is a fixed average speed  $v_l$  on each link  $l$ . Let  $\tau_l$  denote the travel time on link  $l$ :  $\tau_l = \frac{\ell_l}{v_l}$  where  $\ell_l$  is the length of link  $l$ . We denote the set of incoming links for node  $v \in \mathcal{V}$  by  $L_v^{\text{in}}$ , and the set of outgoing links by  $L_v^{\text{out}}$ . Note that for origins  $o \in \mathcal{O}$  we have  $L_o^{\text{in}} = \emptyset$  and for destinations  $d \in \mathcal{D}$  we have  $L_d^{\text{out}} = \emptyset$ .

The aim is now to assign actual (real-valued) flows  $x_{l,o,d}$  (in the static case) or  $x_{l,o,d}(k)$  (in the dynamic case) for every pair  $(o, d) \in \mathcal{O} \times \mathcal{D}$  and every  $l \in L_{o,d}$ , in such a way that the given performance criterion (e.g., total time spent on the network) is minimised with constraints maintained (e.g., the capacity of the links is not exceeded). In the dynamic case  $x_{l,o,d}(k)$  denotes the flow of vehicles from origin  $o$  to destination  $d$  that enter link  $l$  in the time interval  $[kT_s, (k+1)T_s)$ .

For the optimal route choice problem we now consider four cases with a gradually increasing complexity:

- Static case with sufficient network capacity,
- Static case with queues at the boundaries of the network only,

<sup>2</sup>This approach can easily be extended to the case where also the internal nodes  $v \in \mathcal{I}$  have a finite capacity.

- Dynamic case with queues at the boundaries of the network only,
- Dynamic case with queues inside the network.

For the dynamic cases we will focus on optimal control for the sake of simplicity of the expositions, but the proposed approach can also be included in a model predictive control framework.

### 5.3.3 Static case with sufficient network capacity

Here we assume that there is a constant demand for each origin-destination pair and that the total network capacity is such that the entire demand can be processed, so that there will be no queues at the boundaries or inside the network. Let us now describe the equations to model this situation.

For every origin node  $o \in O$  we have:

$$\sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d} = D_{o,d} \quad \text{for each } d \in \mathcal{D}. \quad (5.1)$$

For every internal node  $v \in I$  and for every pair  $(o, d) \in O \times \mathcal{D}$  we have

$$\sum_{l \in L_v^{\text{in}} \cap L_{o,d}} x_{l,o,d} = \sum_{l \in L_v^{\text{out}} \cap L_{o,d}} x_{l,o,d} . \quad (5.2)$$

We also have the following condition for every link  $l$ :

$$\sum_{(o,d) \in S_{od,l}} x_{l,o,d} \leq C_l . \quad (5.3)$$

Finally, the objective function is given as follows<sup>3</sup>:

$$J_{\text{links}} = \sum_{(o,d) \in O \times \mathcal{D}} \sum_{l \in L_{o,d}} x_{l,o,d} \tau_l T , \quad (5.4)$$

which is a measure for the total time the vehicles or platoons spend in the network. In order to minimise  $J_{\text{links}}$  we have to solve the following optimisation problem:

$$\min J_{\text{links}} \quad \text{s.t. (5.1)–(5.3)} \quad (5.5)$$

Clearly, this is a linear programming problem. The linear programming problems are efficiently solvable using (a variant of) the simplex method or an interior-point method [118, Chapter 1].

### 5.3.4 Static case with queues at the boundaries of the network only

In case the capacity of the network is less than the demand, then problem (5.5) will not be feasible. The bottlenecks represent the places on network where capacity is restricted, which in turn may develop queues at origins. In order to be able to determine the optimal

<sup>3</sup>Recall that  $T = KT_s$  is the length of the simulation period.

routing in this particular case, we have to take into account that queues might appear at the origin of the network.

Let us first write down the equations for the flows inside the network.

For every origin node  $o \in O$  we have:

$$\sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d} \leq D_{o,d} \quad \text{for each } d \in \mathcal{D}. \quad (5.6)$$

Equations (5.2) and (5.3) also hold in this case.

Let us now describe the behaviour of the queues. Since the actual flow out of origin node  $o$  for destination  $d$  is given by

$$F_{o,d}^{\text{out}} = \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d} ,$$

the queue length at the origin  $o$  for vehicles or platoons going to destination  $d$  will increase linearly with a rate  $D_{o,d} - F_{o,d}^{\text{out}}$  (note that by (5.6) this rate is always nonnegative). At the end of the simulation period (which has length  $T$ ) the queue length will be  $(D_{o,d} - F_{o,d}^{\text{out}})T$ , and hence the average queue length is  $\frac{1}{2}(D_{o,d} - F_{o,d}^{\text{out}})T$ . So the total time spent in the origin queues is

$$\begin{aligned} J_{\text{queue}} &= \sum_{(o,d) \in O \times \mathcal{D}} \frac{1}{2} (D_{o,d} - F_{o,d}^{\text{out}}) T^2 \\ &= \sum_{(o,d) \in O \times \mathcal{D}} \frac{1}{2} \left( D_{o,d} - \sum_{l \in L_o^{\text{out}}} x_{l,o,d} \right) T^2 . \end{aligned}$$

In order to minimise the total time spent we have to solve the following optimisation problem:

$$\min (J_{\text{links}} + J_{\text{queue}}) \quad \text{s.t. (5.2), (5.3), and (5.6).} \quad (5.7)$$

This is also a linear programming problem.

### 5.3.5 Dynamic case with queues at the boundaries of the network only

Now we consider a piecewise constant demand pattern for every origin-destination pair. Moreover, we assume that the travel time  $\tau_l$  on link  $l$  is an integer multiple of  $T_s$ , say

$$\tau_l = \kappa_l T_s \quad \text{with } \kappa_l \text{ an integer.} \quad (5.8)$$

Let  $q_{o,d}(k)$  denote the partial queue length of vehicles at origin  $o$  going to destination  $d$  at time instant  $t = kT_s$ . In principle, the queue lengths should be integers as their unit is “number of vehicles”, but we will approximate them using reals.

For every origin node  $o \in O$  we now have:

$$\sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k) \leq D_{o,d}(k) + \frac{q_{o,d}(k)}{T_s} \quad \text{for each } d \in \mathcal{D}, \quad (5.9)$$

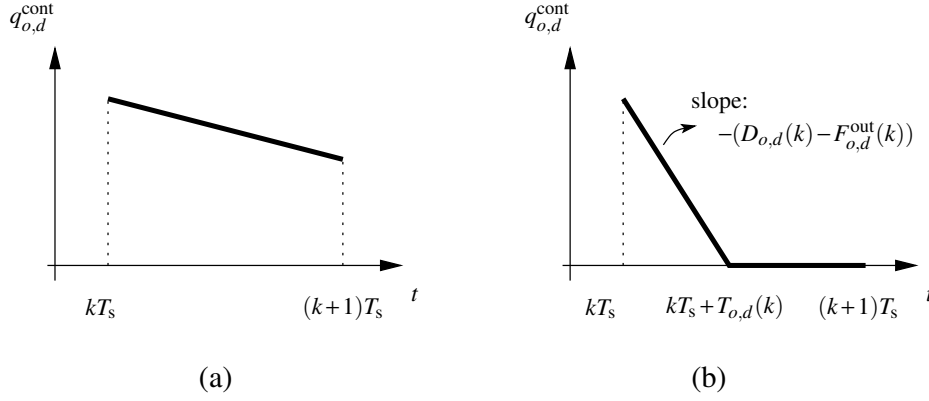


Figure 5.3: Two possible cases for the evolution of the continuous-time queue length  $q_{o,d}^{\text{cont}}$  in the time interval  $[kT_s, (k+1)T_s]$

with by definition  $D_{o,d}(k) = 0$  for  $k \geq K$ ,  $q_{o,d}(k) = 0$  for  $k \leq 0$ . Note that the term  $\frac{q_{o,d}(k)}{T_s}$  in (5.9) is due to the assumption that whenever possible and feasible the queue is emptied in the next sample period, with length  $T_s$ .

Taking into account that every flow on link  $l$  has a delay of  $\kappa_l$  time steps before it reaches the end of the link, we have

$$\sum_{l \in L_v^{\text{in}} \cap L_{o,d}} x_{l,o,d}(k - \tau_l) = \sum_{l \in L_v^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k) \quad (5.10)$$

for every internal node  $v \in I$  and for every pair  $(o,d) \in O \times \mathcal{D}$ , with  $x_{l,o,d}(k) = 0$  for  $k \leq 0$ . We also have the following condition for every link  $l$ :

$$\sum_{(o,d) \in S_{od,l}} x_{l,o,d}(k) \leq C_l \quad (5.11)$$

Let us now describe the behaviour of the queues. Since the actual flow out of origin node  $o$  for destination  $d$  in the time interval  $[kT_s, (k+1)T_s]$  is given by

$$F_{o,d}^{\text{out}}(k) = \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k) \quad (5.12)$$

the queue length at the origin  $o$  for vehicles going to destination  $d$  will increase linearly with a rate<sup>4</sup>  $D_{o,d}(k) - F_{o,d}^{\text{out}}(k)$  in the time interval  $[kT_s, (k+1)T_s]$ . Hence,

$$q_{o,d}(k+1) = \max(0, q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s) \quad (5.13)$$

In order to determine the time  $J_{\text{queue},o,d}(k)$  spent in the queue at origin  $o$  in the time interval  $[kT_s, (k+1)T_s]$  for traffic going to destination  $d$ , we have to distinguish between two cases depending on whether or not the continuous-time queue length  $q_{o,d}^{\text{cont}}$  becomes equal

<sup>4</sup>In contrast to Section 5.3.4 this rate can now also be negative.

to zero *inside*<sup>5</sup> the interval  $[kT_s, (k+1)T_s]$  (see Cases (a) and (b) of Figure 5.3). For Case (b) we define

$$T_{o,d}(k) = \frac{q_{o,d}(k)}{F_{o,d}^{\text{out}}(k) - D_{o,d}(k)} \quad (5.14)$$

as the time offset after  $kT_s$  at which the queue length becomes zero. Then we have

$$J_{\text{queue},o,d}(k) = \begin{cases} \frac{1}{2}(q_{o,d}(k) + q_{o,d}(k+1))T_s & \text{for Case (a),} \\ \frac{1}{2}q_{o,d}(k)T_{o,d}(k) & \text{for Case (b).} \end{cases}$$

Due to the denominator term in (5.14)  $J_{\text{queue},o,d}(k)$  is in general a nonlinear function. Now assume that we simulate the network until time step  $K_{\text{end}} \geq K$  (e.g., until all queues and all flows have become<sup>6</sup> equal to zero). Then we have

$$J_{\text{queue}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d) \in O \times \mathcal{D}} J_{\text{queue},o,d}(k) .$$

The time spent in the links is now given by

$$J_{\text{links}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d) \in O \times \mathcal{D}} \sum_{l \in L_{o,d}} x_{l,o,d}(k) \kappa_l T_s^2 \quad (5.15)$$

In order to minimise the total time spent we have to solve the following optimisation problem with  $J_{\text{links}}$  still defined by 5.15:

$$\min (J_{\text{links}} + J_{\text{queue}}) \quad \text{s.t. (5.9)–(5.13).} \quad (5.16)$$

Due to the presence of constraint (5.13) and the nonlinear expression for  $J_{\text{queue},o,d}(k)$  in Case (b) this is a nonlinear, nonconvex, and nonsmooth optimisation problem. In general, these problems are difficult to solve and require multi-start local optimisation methods (such as Sequential Quadratic Programming (SQP)) or global optimisation methods (such as genetic algorithms, simulated annealing, or pattern search) [118]. However, in Section 5.3.7 we will propose an alternative approximate solution approach based on mixed-integer linear programming.

In case of unpredictable congestion caused by incidents or accidents on the road, the roadside controller will inform the area controller about the congestion in terms of reduced capacity of the link.

### 5.3.6 Dynamic case with queues inside the network

Now we consider the case with queues inside the network. If there are queues formed, we assume that they are formed at the end of the links and that the queues are vertical. In fact,

<sup>5</sup>So we are only Case (b) if  $q_{o,d}^{\text{cont}}$  becomes equal to zero for some time  $t$  with  $kT_s < t < (k+1)T_s$ , i.e., if  $q_{o,d}(k) > 0$  and  $q_{o,d}(k) + (D_{o,d}(k) - F_{o,d}^{\text{out}}(k))T_s < 0$ . All other situations belong to Case (a).

<sup>6</sup>If this is not the case we have to add an end-point penalty on the queue lengths and flows at time step  $K_{\text{end}}$ .

for the sake of simplicity and in order to obtain linear equations, we assign the queues to the nodes instead of the links.

This case is similar to the previous case, the difference being that (5.10) is now replaced by (cf. also (5.9)):

$$\sum_{l \in L_v^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k) \leq \left( \sum_{l \in L_v^{\text{in}} \cap L_{o,d}} x_{l,o,d}(k - \tau_l) \right) + \frac{q_{v,o,d}(k)}{T_s}, \quad (5.17)$$

where  $q_{v,o,d}(k)$  is the partial queue length at node  $v$  for vehicles or platoons going from origin  $o$  to destination  $d$  at the time instant  $t = kT_s$ . Moreover,

$$q_{v,o,d}(k+1) = \max(0, q_{v,o,d}(k) + (F_{v,o,d}^{\text{in}}(k) - F_{v,o,d}^{\text{out}}(k))T_s)$$

with the flow into and out of the queue being given by

$$F_{v,o,d}^{\text{in}}(k) = \sum_{l \in L_v^{\text{in}} \cap L_{o,d}} x_{l,o,d}(k - \tau_l) \quad (5.18)$$

$$F_{v,o,d}^{\text{out}}(k) = \sum_{l \in L_v^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k). \quad (5.19)$$

Similar to  $J_{\text{queue},o,d}(k)$  we also define the time  $J_{\text{queue},v,o,d}(k)$  spent in the queue at node  $v$  in the time interval  $[kT_s, (k+1)T_s)$  for traffic going from origin  $o$  to destination  $d$ , and we extend the definition of  $J_{\text{queue}}$  into

$$J_{\text{queue}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d) \in O \times D} \left( J_{\text{queue},o,d}(k) + \sum_{v \in I} J_{\text{queue},v,o,d}(k) \right) T_s.$$

with  $J_{\text{queue},v,o,d}(k)$  defined as follows:

$$J_{\text{queue},v,o,d}(k) = \begin{cases} \frac{1}{2}(q_{v,o,d}(k) + q_{v,o,d}(k+1))T_s & \text{for Case (a),} \\ \frac{1}{2}q_{v,o,d}(k)T_{v,o,d}(k) & \text{for Case (b).} \end{cases}$$

For Case (b) we define

$$T_{v,o,d}(k) = \frac{q_{v,o,d}(k)}{F_{v,o,d}^{\text{out}}(k) - F_{v,o,d}^{\text{in}}(k)} \quad (5.20)$$

as the time offset after  $kT_s$  at which the queue length becomes zero. This also results in a nonlinear, nonconvex, and nonsmooth optimisation problem. However, in the next section we will show that this problem can also be approximated using mixed-integer linear programming.

### Remarks

For the dynamic case with queues at the boundaries of the network only, the area controllers will be designed to provide optimal flows in such a way that once the vehicles enter the network, they will travel at their free-flow speed without encountering traffic jams. When the



traffic demand exceeds the capacity of the network, queues will be created at one or more origins. Moreover, the congestion-free network comes at the price of increased queue lengths at the origins. However, when the considered network is larger, then it would be a better option for the model to allow the vehicles to enter the network rather than to create queues at the origin.

Now, we will consider the dynamic case with queues inside the network. We will use vertical queues to account for spillbacks and blockages on adjacent links. There exist two possibilities to create queues inside the network. It can be either at the nodes or at the end of the links. In case of the latter option, the queue at the end of each link will be created by considering its own dynamics and also the behaviour of flows on the adjacent links. However, queues on every link lead to nonlinear equations and a nonlinear optimisation problem (see, e.g., [148] where a similar routing problem is considered for human drivers). Therefore, we will consider the queue formations at nodes rather than at links. The queues at the nodes, to a certain extent, can include the spillback effect by using capacity constraints on links. In addition, the model can also include restrictions on the number of vehicles waiting in the queues, by imposing a maximum allowed queue length.

### 5.3.7 Approximation based on mixed-integer linear programming

Recall that the dynamic optimal route guidance problems of Sections 5.3.5 and 5.3.6 are nonlinear, nonconvex, and nonsmooth. Now we will show that by introducing an approximation these problems can be transformed into mixed-integer linear programming (MILP) problems, for which efficient solvers have been developed [51].

First we consider the case with queues at the origins only, i.e., we consider the optimisation problem (5.16). Apart from (5.13) this problem is a linear optimisation problem.

Now we explain how we can transform (5.13) into a system of linear equations by introducing some auxiliary boolean variables  $\delta$ . To this aim we use the following properties [20], where  $\delta$  represents a binary-valued scalar variable,  $y$  a real-valued scalar variable, and  $f$  a function defined on a bounded set  $X$  with upper and lower bounds  $M$  and  $m$  for the function values:

**P1:**  $[f \leq 0] \iff [\delta = 1]$  is true if and only if

$$\begin{cases} f \leq M(1-\delta) \\ f \geq \epsilon + (m-\epsilon)\delta \end{cases},$$

where  $\epsilon$  is a small positive number<sup>7</sup> (typically the machine precision),

**P2:**  $y = \delta f$  is equivalent to

$$\begin{cases} y \leq M\delta \\ y \geq m\delta \\ y \leq f - m(1-\delta) \\ y \geq f - M(1-\delta) \end{cases}.$$

<sup>7</sup>We need this construction to transform a constraint of the form  $y > 0$  into  $y \geq \epsilon$ , as in (mixed-integer) linear programming problems only non-strict inequalities are allowed.

Depending on the order in which these properties are applied and in which additional auxiliary variables are introduced, we may end up with more or less binary and real variables in the final MILP problem. The number of binary variables — and to a lesser extent the number of real variables — should be kept as small as possible since this number has a direct impact of the computational complexity of the final MILP problem.

To reduce the number of real variables in the final MILP problem, we first eliminate  $F_{o,d}^{\text{out}}(k)$  and we write (5.13) as

$$q_{o,d}(k+1) = \max \left( 0, q_{o,d}(k) + (D_{o,d}(k) - \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k)) T_s \right). \quad (5.21)$$

Note that this is a nonlinear equation and thus it does not fit the MILP framework. Let  $D_{\max,o,d} = \max_k D_{o,d}(k)$  be the maximal demand for origin-destination pair  $(o,d)$ , let the  $F_{\max,o,d} = \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} C_l$  be the maximal possible flow out of origin node  $o$  towards destination  $d$ , and let  $q_{\max,o,d} = D_{\max,o,d} T_s K_{\text{end}}$  be the maximal origin queue length at origin  $o$  for traffic going to destination  $d$ . If we define  $m_{o,d}^{\text{low}} = -F_{\max,o,d} T_s$  and  $m_{o,d}^{\text{upp}} = q_{\max,o,d} + D_{\max,o,d} T_s$ , then we always have

$$m_{o,d}^{\text{low}} \leq q_{o,d}(k) + (D_{o,d}(k) - \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k)) T_s \leq m_{o,d}^{\text{upp}}.$$

Next, we introduce binary variables  $\delta_{o,d}(k)$  such that

$$\delta_{o,d}(k) = 1 \text{ if and only if } q_{o,d}(k) + (D_{o,d}(k) - \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k)) T_s \geq 0.$$

Using Property **P1** with the bounds  $m_{o,d}^{\text{low}}$  and  $m_{o,d}^{\text{upp}}$  this condition can be transformed into a system of linear inequalities. Now we have (cf. (5.21))

$$q_{o,d}(k+1) = \delta_{o,d}(k) \left( q_{o,d}(k) + (D_{o,d}(k) - \sum_{l \in L_o^{\text{out}} \cap L_{o,d}} x_{l,o,d}(k)) T_s \right). \quad (5.22)$$

This expression is still nonlinear since it contains a multiplication of a binary variable  $\delta_{o,d}(k)$  with a real-valued (linear) function. However, by using Property **P2** this equation can be transformed into a system of linear inequalities.

So by introducing some auxiliary variables  $\delta_{o,d}(k)$  we can transform the original nonlinear equation (5.13) into a system of additional linear equations and inequalities.

Recall that  $J_{\text{queue},o,d}(k)$  is in general a nonlinear function due to the occurrence of Case (b) of Figure 5.3. However, if we also use the expression of Case (a) for Case (b), then we can approximate  $J_{\text{queue},o,d}(k)$  as<sup>8</sup>

$$J_{\text{queue},o,d}(k) = \frac{1}{2} (q_{o,d}(k) + q_{o,d}(k+1)) T_s,$$

which is a linear expression. This implies that the overall objective function  $J_{\text{links}} + J_{\text{queue}}$  is now linear. So the problem (5.16) can be approximated by an MILP problem.

<sup>8</sup>This is exact for Case (a) and an approximation for Case (b). However, especially if  $T_s$  is small enough, the error we then make is negligible.

Several efficient branch-and-bound MILP solvers [51] are available for MILP problems. Moreover, there exist several commercial and free solvers for MILP problems such as, e.g., CPLEX, Xpress-MP, GLPK, or `lp_solve` (see [5, 99] for an overview). In principle, — i.e., when the algorithm is not terminated prematurely due to time or memory limitations, — these algorithms guarantee to find the global optimum. This global optimisation feature is not present in the other optimisation methods that can be used to solve the original nonlinear, nonconvex, nonsmooth optimisation problem (5.16). Moreover, if the computation time is limited (as is often the case in on-line real-time traffic control), then it might occur that the MILP solution can be found within the allotted time whereas the global and multi-start local optimisation algorithm still did not converge to a good solution. As a result, the MILP solution — even although it solves an approximated problem — might even perform better than the solution returned by the prematurely terminated global and multi-start local optimisation method. In general, we can say that the MILP solution often provides a good trade-off between optimality and computational efficiency, as will be illustrated in the case study of Section 5.3.8.

Using a similar reasoning as above we can also transform the routing problem with queues inside the network of Section 5.3.6 into an MILP problem. Note however that in this case the number of binary variables may become quite large.

Some ways to reduce the complexity of the above approach are to embed it in a model predictive control framework (because then the optimisation horizon is much shorter than the full simulation horizon currently used due to the optimal control setting) or to reduce the number of partial queues by only considering a limited set of possible/allowed routes between each origin-destination pair. Moreover, if there are no origin-destination dependent performance functions or constraints, we can combine the flows and queues for the various origins at each link and node. This will considerably reduce the number of variables.

### 5.3.8 Case study

In this section we present a simple case study involving a basic set-up to illustrate the area-level control approach for IVHS proposed in this chapter. In particular, we will consider the dynamic case with queues at the origins of the network only (cf. Section 5.3.5). First, we will describe the set-up and the details of the scenario used for our simulation and thus solve problem (5.16). Next, we will discuss and analyse the obtained results.

#### Scenario

We consider a simple network of highways with one origin  $o_1$  and two destinations  $d_1, d_2$ , and three internal nodes  $v_1, v_2$ , and  $v_3$  (see Figure 5.4). The network consists of three high-capacity links connecting  $o_1$  to  $v_1, v_2$  to  $d_1$ , and  $v_3$  to  $d_2$ , as well as six links connecting the internal nodes, allowing four possible routes to each destination (e.g.,  $d_1$  can be reached via  $l_1, l_2, l_3+l_5$ , and  $l_4+l_5$ ).

We simulate a period of 60 min. The simulation time step  $T_s$  is set to 1 min. The demand pattern is piecewise constant during the simulation period and is given in Table 5.1. The demand to be processed in the period [10,30) higher than the capacity of the network, giving rise to an origin queue for each destination. The capacities on the links directly connected to the origin and destination nodes are assumed to be high enough so that no queues are

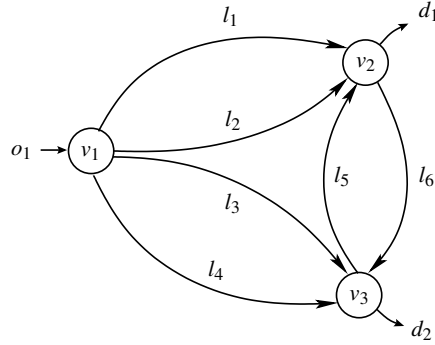


Figure 5.4: Set-up of case study network

Period (min)	0–10	10–30	30–40	40–60
$D_{o_1,d_1}$ (veh/h)	5000	8000	2500	0
$D_{o_1,d_2}$ (veh/h)	1000	2000	1000	0

Table 5.1: Demand profiles used in the case study

formed on them, and the travel time on these links is assumed to be negligible. The maximum capacities associated with the links between the internal nodes are  $C_1=1900$  veh/h,  $C_2=2000$  veh/h,  $C_3=1800$  veh/h,  $C_4=1600$  veh/h,  $C_5=1000$  veh/h, and  $C_6=1000$  veh/h. Depending on the speed and length of each link, different travel times can be obtained, which are characterised by (cf. (5.8))  $\kappa_1=10$ ,  $\kappa_2=9$ ,  $\kappa_3=6$ ,  $\kappa_4=7$ ,  $\kappa_5=2$ , and  $\kappa_6=2$ . For the proposed scenario the initial state of the network is taken to be empty.

We consider three different cases:

- Case A: no control,
- Case B: controlled using the MILP solution,
- Case C: controlled using the exact solution.

### Results and analysis

We have used Matlab to compute the optimal route choice solutions in Cases B and C. More specifically, the MILP problem of Case B has been solved using CPLEX, implemented through the `cplex` interface function of the Matlab Tomlab toolbox. For Case C we have used the SQP function SNOPT, implemented via the function `snopt` of the Matlab Tomlab toolbox. For Case C we have considered three different choices for the starting points: 5 random initial points, 50 random initial points, and the MILP solution as the initial point. The results of the numerical experiments are listed in Table 5.2.

In case of no control (Case A), the capacities of the direct links  $l_1$ ,  $l_2$ ,  $l_3$ , and  $l_4$  are consumed up to their maximum while the links  $l_5$  and  $l_6$  are not used due to the fact that

<sup>9</sup>On a 1 GHz Athlon 64 X2 Dual Core 3800+ processor with 3 GB of RAM.

Case	$J_{\text{queue}}$ (veh.h)	improvement	CPU time <sup>9</sup> (s)
no control	1434	0 %	–
MILP	1081	24.6 %	0.27
SQP (5 initial points)	1067	25.6 %	90.0
SQP (50 initial points)	1064	25.8 %	983
SQP (with MILP solution as initial point)	1064	25.8 %	1.29

Table 5.2: Results for the case study of Section 5.3.8. The improvement is expressed with respect to the no-control case

all vehicles and platoons want to take the shortest routes. At the point when the demand exceeds the maximum capacity of the links (i.e., during 10 to 30 minutes, the total demand 10000 veh/h exceeds the maximum capacity 4000 veh/h) origin queues are formed. As the simulation advances further, the queue length also increases linearly with time, thus leading to a large total time spent of 1434 veh.h.

When control is applied, the area controller assigns the routes to the platoons in a system optimum manner. By system optimum, we mean that some of the platoons and vehicles can even be assigned a longer route rather than the direct or shortest routes, if this leads to an improvement of the total traffic performance. This results in a performance improvement of 24.6 % for the MILP solution (Case B), and — depending also on the number of initial points considered — in a performance improvement of up to 25.8 % for the exact solution (Case C). Note that, for this case study using the MILP solution as the starting point for SQP yields the optimal solution at very low computational costs (1.29 s).

Although the exact solution performs better than the MILP solution, this comes at the cost of an increased computation time due to the multi-start SQP, which results in a total computation time that can be much larger than  $T_s$  (1 min). In practice, where the approach will typically be applied on-line in a moving horizon approach, this excessive computation time makes the multi-start SQP approach infeasible, whereas the MILP solution can be computed within the sampling time interval  $T_s$  while having almost the same performance as the multi-start SQP solution.

## 5.4 Optimal routing for IVHS using a macroscopic traffic flow model

In Section 5.3.2, we have used a rough approximation of the real network traffic dynamics. An alternative but somewhat more refined way to obtain a simplified model to describe the traffic flows in IVHS is to use some of the existing traffic flow models for human drivers and to adapt them to fit the IVHS framework.

### 5.4.1 Macroscopic traffic flow characteristics

Consider a traffic network consisting a several links, each of which is divided in one or more segments. Let  $T_s$  be the sampling or simulation time step for the macroscopic network model. The flow in a each segment  $m$  of the traffic network can characterised by three macroscopic variables:

- Mean speed  $v_m(k)$ ,
- Traffic density  $\rho_m(k)$ ,
- Traffic flow, intensity or volume  $q_m(k)$ .

where  $v_m(k)$  is the mean speed of the vehicles in segment  $m$  at time  $kT_s$ ,  $\rho_m(k)$  is the number of vehicles in segment  $m$  at time  $kT_s$ ,  $q_m(k)$  is the number of vehicles leaving segment  $m$  during time period  $[kT_s, (k+1)T_s]$ , where the counter  $k$  corresponds to the time instant  $t = kT_s$ .

The traffic density  $\rho$  is the number of vehicles per kilometre i.e., it characterises at a specific point of time, how crowded a particular segment of a road is. In relation to the microscopic variables, the traffic density can be derived using the average space headway ( $\bar{s}$ ), and number of vehicles  $N_{\text{veh}}$  in that segment as

$$\rho = \frac{N_{\text{veh}}}{L_{\text{seg}}} \approx \frac{N_{\text{veh}}}{\sum_{i=1}^{N_{\text{veh}}} (s_i)} = \frac{1}{\bar{s}} \quad (5.23)$$

where  $L_{\text{seg}}$  is the length of the given road segment and  $s_i$  is the space headway for vehicle  $i$  (i.e., the distance difference in position between the rear of vehicle  $i$  and the rear of its predecessor).

The traffic flow or volume  $q$  is the number of vehicles passing through a freeway location (marked by, e.g., a detector) per unit time  $N_{\text{veh1}}$ . In relation to the microscopic variables, the traffic flow can be defined as a reciprocal of the average time headway ( $\bar{h}$ ). Assuming a certain time period  $\Delta T$ , the flow can expressed as:

$$q = \frac{N_{\text{veh1}}}{\Delta T} \approx \frac{N_{\text{veh1}}}{\sum_{i=1}^{N_{\text{veh1}}} (h_i)} = \frac{1}{\bar{h}} \quad (5.24)$$

where the time headway  $h_i$  of vehicle  $i$  is the amount of time necessary for the rear of vehicle  $i$  to reach the current position of the rear of its predecessor.

The three basic macroscopic variables are related to each other by the fundamental relation

$$q = \rho v \quad (5.25)$$

This means that out of these three variables only two are independent. In the sequel we will consider  $v$  and  $\rho$  to be the independent variables.

Let us now consider the (equilibrium) relation between the speed  $v$  and the density  $\rho$ . When the density on the road is very low and the average distance headway is large, the drivers travel at their desired speed. This is called free-flow driving. As the density starts to increase due to the increasing demand, the vehicles will start to reduce their speed slightly and follow their predecessor while maintaining a safe time headway. Once the critical density (i.e., the density at which the capacity of the network is being utilised at its

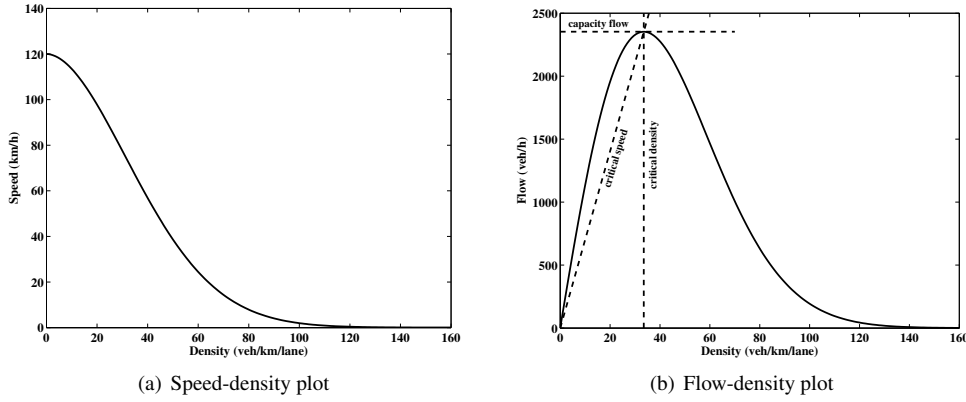


Figure 5.5: Fundamental diagram for human drivers

maximum) is reached, the speed starts to decrease significantly resulting in a traffic jam. When the density is at its maximum ( $\rho_{\max}$ ), the vehicle speed drops to almost zero. The (equilibrium) relation between the speed  $v$  and the density  $\rho$  can be modelled as [102]:

$$V(\rho) = v_{\text{free}} \exp \left[ -\frac{1}{a} \left( \frac{\rho}{\rho_{\text{crit}}} \right)^a \right] \quad (5.26)$$

where  $\rho_{\text{crit}}$  is the critical density,  $v_{\text{free}}$  is the free-flow speed, and  $a$  is a model parameter. Typical values for these parameters are  $v_{\text{free}}=120$  km/h,  $\rho_{\text{crit}}=33.5$  veh/km/lane,  $a=1.867$ , and  $\rho_{\max}=180$  veh/km/lane [90]. The fundamental relation given in (5.26) can be depicted using the so-called fundamental diagram as shown in Figure 5.5 for a single lane. This figure shows the maximum flow  $q_{\max}$ , and the critical density  $\rho_{\text{crit}}$ .

### 5.4.2 Macroscopic traffic flow for IVs

When semi-automatic or intelligent vehicles are used on the road, the macroscopic traffic flow will change. An example of such a change is given by Bose et al. [26], where Adaptive Cruise Control (ACC) is considered with a constant time headway policy. The constant time headway policy is the control form most often used for ACC [26, 139, 158]. The spacing is given in [158] as

$$s_i = h_{\text{des}} v_i + L_{\text{veh}} \quad (5.27)$$

where  $h_{\text{des}}$  is the desired time headway,  $v_i$  is the velocity of the vehicle of interest (the following vehicle), and  $L_{\text{veh}}$  is the length of the vehicle, which, for the sake of simplicity, is assumed to be the same for all vehicles (if this is not the case, the average vehicle length  $\bar{L}_{\text{veh}}$  should be used in the equations below).

Using (5.23), for a given speed  $v$  and (average) space headway  $s$  the maximal density with IVs can be expressed as the reciprocal of the inter-vehicle spacing:

$$\rho_{\text{ACC}} = \frac{1}{s} = \frac{1}{h_{\text{des}} v + L_{\text{veh}}} \quad (5.28)$$

Rewriting (5.28) gives an expression for the (maximally possible) speed as

$$v = \frac{1}{h_{\text{des}}} \left( \frac{1}{\rho_{\text{ACC}}} - L_{\text{veh}} \right)$$

Now taking into account that the speed cannot exceed the free-flow speed  $v_{\text{free}}$ , the expression for the desired speed using 100% ACC-equipped vehicles becomes

$$v_{\text{ACC}} = \begin{cases} v_{\text{free}}, & \text{if } \rho \leq \rho_{\text{ACC,crit}} \\ \frac{1}{h_{\text{des}}} \left( \frac{1}{\rho} - L_{\text{veh}} \right), & \text{if } \rho > \rho_{\text{ACC,crit}} \end{cases} \quad (5.29)$$

For a situation with 100% ACC-equipped vehicles the critical density  $\rho_{\text{ACC,crit}}$  at which the maximal flow is obtained, is thus given by:

$$\rho_{\text{ACC,crit}} = \frac{1}{h_{\text{des}} v_{\text{free}} + L_{\text{veh}}}$$

Using (5.29) and (5.25) the relation between the flow and density becomes

$$q_{\text{ACC}} = \begin{cases} \rho v_{\text{free}}, & \text{if } \rho \leq \rho_{\text{ACC,crit}} \\ \frac{1}{h_{\text{des}}} (1 - \rho L_{\text{veh}}), & \text{if } \rho > \rho_{\text{ACC,crit}} \end{cases} \quad (5.30)$$

For typical values of  $h_{\text{des}} = 0.5$  s,  $L_{\text{veh}} = 4$  m, and  $v_{\text{free}} = 120$  km/h, we obtain  $\rho_{\text{ACC,crit}} = 48.39$  veh/km and the speed-density and flow-density curves shown in Figure 5.6. The flow-density curve illustrates that traffic with IVs will always yield a better performance than that of human drivers, and it also shows that the maximum flow is more than doubled. Remark: The value of  $\rho_{\text{ACC,crit}}$  was determined without considering the inter-platoon separations. If we consider this factor and denote as  $\gamma$ , then the critical density  $\rho_{\text{ACC,crit}}$  becomes:

$$\rho_{\text{ACC,crit}} = \frac{\gamma}{h_{\text{des}} v_{\text{free}} + L_{\text{veh}}}$$

Using (5.29) and (5.25) the relation between the flow and density becomes

$$q_{\text{ACC}} = \begin{cases} \rho v_{\text{free}}, & \text{if } \rho \leq \rho_{\text{ACC,crit}} \\ \frac{\gamma}{h_{\text{des}}} (1 - \rho L_{\text{veh}}), & \text{if } \rho > \rho_{\text{ACC,crit}} \end{cases} \quad (5.31)$$

A typical value for  $\gamma$  would be taken as 0.95.

### 5.4.3 A METANET-like macroscopic model for IVs

The METANET model is a second-order macroscopic traffic flow model that has been proposed by Papageorgiou and his co-workers [90, 91, 104, 115]. The METANET model is discretised in time and space. As it deals with macroscopic variables rather than the variables or states of individual vehicles, it is suited for on-line computational purposes. The basic METANET model consists of link equations and node equations (nodes can represent a junction or a bifurcation point). The link model describes the behaviour of the traffic in the highway stretches, and the node model describes the behaviour of the traffic at the nodes in the network. The METANET model can be classified as destination-oriented or destination-independent. Since we will use the METANET model for solving routing problems, we will describe the destination-oriented model.



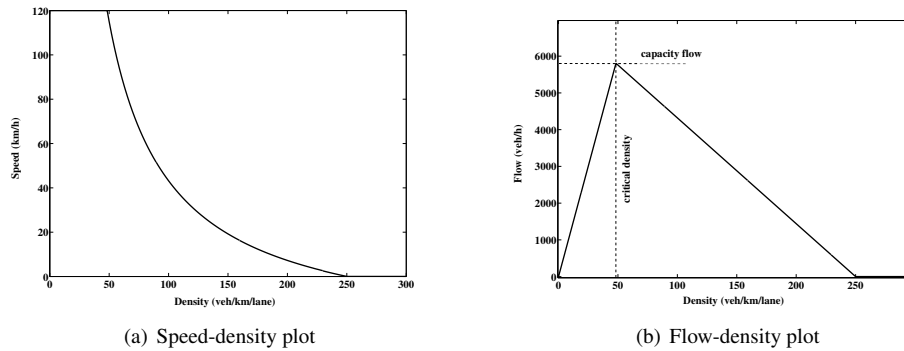


Figure 5.6: Fundamental diagram for IVs

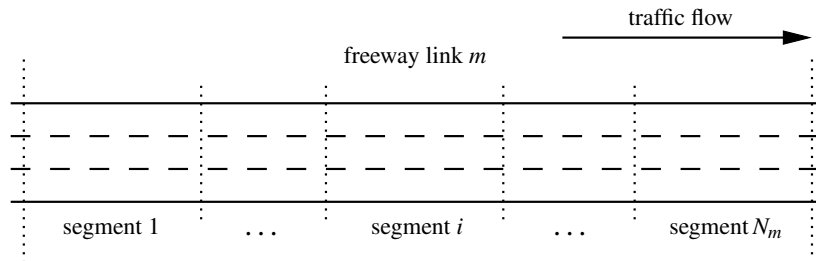


Figure 5.7: Freeway link in METANET

### Destination-oriented model

In this section, we describe destination-oriented METANET model, which explicitly models the traffic flow with routing choices for multiple origin and destinations. At each node with two or more outgoing links, this model associates splitting rates to each reachable destination from that node. These splitting rates describe the how the traffic flow at the node destined to a particular destination must be distributed among the set of leaving links.

In the case of human drivers the splitting rates are determined by an autonomous process called traffic assignment. However, in the case of IVHS the splitting rates can be considered to be a controllable input, since there the traffic management system has full control over the IVs.

### Link model

The METANET model represents a network as a directed graph with the links corresponding to freeway stretches as shown in Figure 5.7. Each freeway link has uniform characteristics, i.e., no on-ramps or off-ramps, and no major changes in geometry. Where major changes occur in the characteristics of the link or in the road geometry (e.g., on-ramp or an off-ramp), a node is placed.

In the METANET each link  $m$  is divided into  $N_m$  segments with length  $L_m$ . The number

of lanes on link  $m$  is denoted by  $\lambda_m$ . The traffic flow in segment  $i$  of link  $m$  destined to a destination  $j$  is characterised by three macroscopic variables:

- Mean speed  $v_{m,i}(k)$  [km/h]
- Partial density  $\rho_{m,i,j}(k)$  [veh/km/lane]
- Traffic flow  $q_{m,i}(k)$  [veh/h]

where  $k$  is the discrete time instant  $t = kT_s$  where  $T_s$  is the simulation time step (typically round 10 seconds). At time step  $k$ , the partial density  $\rho_{m,i,j}(k)$  describes the density in the segment  $i$  of link  $m$  that is travelling to destination  $j$ . If destination  $j$  is not reachable from link  $m$  we set  $\rho_{m,i,j}(k) = 0$  for all  $i, k$ .

Note that we use partial densities  $\rho_{m,i,j}(k)$  to distinguish the traffic with different destinations. However, the same mean speed  $v_{m,i}(k)$  is assigned to all the vehicles travelling in a segment irrespective of their destination.

The segment length  $L_m$  is usually in the range of 500 to 1000 m. For stability reasons, a vehicle travelling in a segment at its free speed is not allowed to pass the segment in one simulation time step. So, the following condition should be satisfied:

$$L_m > v_{\text{free}} T_s \quad (5.32)$$

Note that for the values of  $L_m$ ,  $v_{\text{free}}$ , and  $T_s$  given above this condition is satisfied.

For each segment in a link, for all possible destinations reachable via the link, the conservation of vehicles in a segment can be expressed as

$$\rho_{m,i,j}(k+1) = \rho_{m,i,j}(k) + \frac{T}{L_m \lambda_m} (\gamma_{m,i-1,j}(k) q_{m,i-1}(k) - \gamma_{m,i,j}(k) q_{m,i}(k)) \quad (5.33)$$

where  $q_{m,i-1}(k)$  is the traffic flow that flows out of segment  $i-1$  of link  $m$  into segment  $i$  for simulation time step  $k$ ,  $q_{m,i}(k)$  is the flow out of segment  $i$  of link  $m$ , and  $\gamma_{m,i,j}(k)$  is the composition rate for the traffic flow in segment  $i$  of link  $m$  with destination as  $j$  at simulation time step  $k$ , defined as:

$$\gamma_{m,i,j}(k) = \frac{\rho_{m,i,j}(k)}{\rho_{m,i}(k)} \quad (5.34)$$

where the total density  $\rho_{m,i}(k)$  in segment  $i$  of link  $m$  is defined as

$$\rho_{m,i}(k) = \sum_{j \in J_m} \rho_{m,i,j}(k)$$

where  $J_m$  is the set of destinations reachable via link  $m$ .

The update of the mean speed is calculated based on a convection term, a relaxation term, and an anticipation term. More specifically, the mean speed in segment  $i$  of link  $m$  at the next discrete time step  $k+1$  is given by

$$v_{m,i}(k+1) = v_{m,i}(k) + v_{\text{rel},m,i}(k) + v_{\text{con},m,i}(k) + v_{\text{anticip},m,i}(k) \quad (5.35)$$

The relaxation term  $v_{\text{rel}}$  expresses that the mean speed is dependent on the density of the segment and states that the drivers in the segment accelerate or decelerate to achieve

a desired equilibrium speed  $V(\rho_{m,i}(k))$ . The relaxation term is directly proportional to the difference between the actual mean speed and  $V(\rho_{m,i}(k))$  and can be expressed as:

$$v_{\text{rel},m,i}(k) = \frac{T_s}{\tau} (V(\rho_{m,i}(k)) - v_{m,i}(k)) \quad (5.36)$$

where  $\tau$  is related to the driver's response time and where  $V(\rho_{m,i}(k))$  can be derived either for human drivers or for IVs. For human drivers a typical value for  $\tau$  is 18 s [90]. For IVs this value will be much lower, e.g., 8 s. For human drivers,  $V(\rho_{m,i}(k))$  is given by (cf. 5.26)

$$V(\rho_{m,i}(k)) = v_{\text{free},m} \exp \left[ -\frac{1}{a_m} \left( \frac{\rho_{m,i}(k)}{\rho_{\text{crit},m}} \right)^{a_m} \right] \quad (5.37)$$

where  $a_m$  is a model parameter for the specific link  $m$ ,  $v_{\text{free},m}$  is the free-flow speed for link  $m$ , and  $\rho_{\text{crit},m}$  is the critical density at which the traffic flow becomes maximal.

The expression of  $V(\rho_{m,i}(k))$  for IVs is given by (cf. (5.29)):

$$V(\rho_{m,i}(k)) = \begin{cases} v_{\text{free}}, & \text{if } \rho_{m,i}(k) \leq \rho_{\text{ACC,crit},m} \\ \frac{1}{h_{\text{des}}} \left( \frac{1}{\rho_{m,i}(k)} - L_{\text{veh}} \right), & \text{if } \rho_{m,i}(k) > \rho_{\text{ACC,crit},m} \end{cases} \quad (5.38)$$

The convection term  $v_{\text{con}}$  is given by:

$$v_{\text{con},m,i}(k) = \frac{T_s}{L_m} v_{m,i}(k) (v_{m,i-1}(k) - v_{m,i}(k)) \quad (5.39)$$

and accounts for the effect caused by the vehicles entering the segment  $i$  from upstream segment  $i-1$  with different mean speed. The larger the deviation, the larger is the effect of convection term on the mean speed.

The anticipation term  $v_{\text{anticip}}$  expresses the change in speed induced by the drivers' experience of (total) density that prevails in the downstream segment and is given by:

$$v_{\text{anticip},m,i}(k) = -\frac{\eta T_s}{\tau L_m} \frac{\rho_{m,i+1}(k) - \rho_{m,i}(k)}{\rho_{m,i}(k) + \kappa} \quad (5.40)$$

where  $\kappa$  and  $\eta$  are model parameters. Typical values are  $\kappa=40$  veh/km/lane and  $\eta=60$  km<sup>2</sup>/h [90].

The partial traffic flow for each segment can be described as follows:

$$q_{m,i,j}(k) = \rho_{m,i,j}(k) v_{m,i}(k) \lambda_m \quad (5.41)$$

### Origin model

Origins receive traffic demand and forward it to the freeway. Origins are modelled using a simple queue model. A queue is formed at origin  $o$  when the traffic demand  $d_o(k)$  exceeds the service rate  $q_o(k)$  of the origin. The queue length  $w_{o,j}(k+1)$  destined to destination  $j$  at origin  $o$  can be determined from the previous queue length and the total demand  $d_o(k)$  at time step  $k$  as follows:

$$w_{o,j}(k+1) = w_{o,j}(k) + T_s \gamma_{o,j}(k) (d_o(k) - q_o(k)) \quad (5.42)$$

with  $\gamma_{o,j}(k)$  is the fraction of the demand travelling to destination  $j$  from origin  $o$ . The outflow or service rate at origin  $q_o(k)$  is taken to be the minimum of the number of vehicles that want to enter (i.e., total demand at time instant  $k$  plus the total number of vehicles in the partial queues) and the number of vehicles that are actually allowed to enter due to the availability of space. The outflow can be expressed as:

$$q_o(k) = \min \left[ d_o(k) + \frac{w_o(k)}{T_s}, \quad Q_{\text{cap},o} \min \left( 1, \frac{\rho_{\text{max}} - \rho_{\mu,1}(k)}{\rho_{\text{max}} - \rho_{\text{crit},\mu}} \right) \right] \quad (5.43)$$

where  $Q_{\text{cap},o}$  is the capacity (veh/h) of the origin  $o$  under free-flow conditions,  $\rho_{\text{max}}$  is the maximum density of a segment, and  $\mu$  is the index of the link to which the origin is connected.

### Node model

The node model describes how the traffic should be routed among the set of entering and leaving links of a node. For a given node  $n$ , let  $I_n$  denote the set of input links, and let  $O_n$  denote the set of output links. The traffic flow  $Q_{n,j}(k)$  with destination  $j$  that enters the node  $n$  at simulation step  $k$  is distributed to the output links according to

$$Q_{n,j}(k) = \sum_{\mu \in I_n} q_{\mu,N_\mu}(k) \gamma_{\mu,N_\mu,j}(k) \quad (5.44)$$

$$q_{n,m,\text{out}}(k) = \sum_{j \in J_m} \beta_{n,m,j}(k) Q_{n,j}(k) \quad (5.45)$$

where  $q_{\mu,N_\mu}(k)$  is the flow leaving the last segment of link  $\mu$ ,  $\beta_{n,m,j}(k)$  is the splitting rate in node  $n$  that is defined as the fraction of the traffic flow heading towards destination  $j$  that leaves node  $n$  via output link  $m$ ,  $J_m$  is the set of destinations that are reachable through link  $m$ , and  $q_{n,m,\text{out}}(k)$  is the total traffic flow that leaves node  $n$  via output link  $m$  at simulation step  $k$ .

The composition rate  $\gamma_{n,m,\text{out},j}(k)$  of the traffic flow out of node  $n$  into link  $m$  is given by:

$$\gamma_{n,m,\text{out},j}(k) = \frac{\beta_{n,m,j}(k) Q_{n,j}(k)}{q_{m,\text{out}}(k)} \quad (5.46)$$

### Downstream density

Consider a node  $n$  with input link  $m \in I_n$ . Note that the anticipation term (5.40) of the speed update equation for segment  $i$  of link  $m$  contains the downstream density  $\rho_{m,i+1}(k)$ . Hence, we also need an expression for the downstream density for the last segment (segment  $N_m$ ) of link  $m$ . To this aim we introduce a virtual segment  $N_m + 1$  at the end of link  $m$  as shown in Figure 5.8, and we capture the effect of the downstream density of the output links leaving node  $n$  by the following expression:

$$\rho_{m,N_{m+1}}(k) = \frac{\sum_{\mu \in O_n} \rho_{\mu,1}^2(k)}{\sum_{\mu \in O_n} \rho_{\mu,1}(k)} \quad (5.47)$$

where  $\rho_{\mu,1}(k)$  is the density of the first segment of output link  $\mu$ .

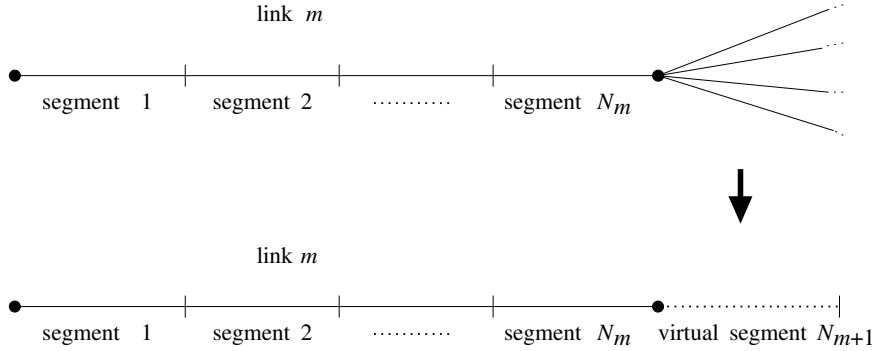


Figure 5.8: The first segments of the outgoing links from a node are aggregated into a virtual segment to provide a virtual downstream density  $\rho_{m,N_{m+1}}(k)$

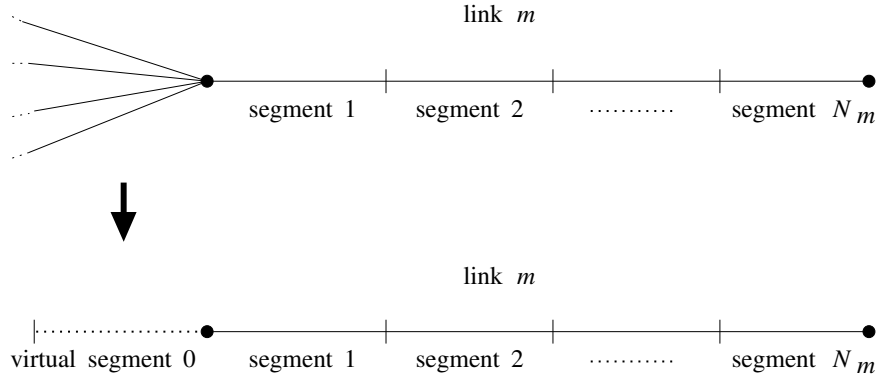


Figure 5.9: The last segments of the incoming links are aggregated into a virtual segment to provide a virtual upstream speed  $v_{m,0}(k)$

### Upstream speed

Similarly as above, when a node  $n$  has many input links (junction node) as shown in Figure 5.9, then the downstream speed for the first segment  $i = 1$  of an outgoing link required in the convection term (5.39) is captured by adding a virtual segment at the beginning of the link and by setting

$$v_{m,0}(k) = \frac{\sum_{\mu \in I_n} v_{\mu, N_{\mu}}(k) q_{\mu, N_{\mu}}(k)}{\sum_{\mu \in I_n} q_{\mu, N_{\mu}}(k)} \quad (5.48)$$

where  $N_{\mu}$  is the index of the last of last segment of link  $\mu$ .

### Extensions

Hegyi et al. [64, 67] have extended the METANET model to include traffic control measures such as dynamic speed limits, and mainstream metering, to account for anticipation behaviour of drivers to varying downstream densities, and to model main-stream origins.

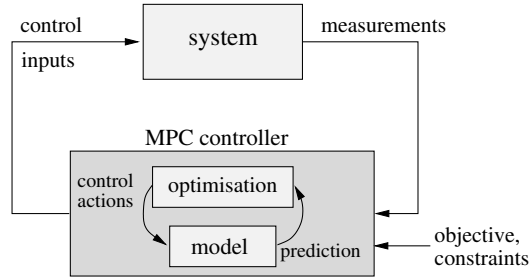


Figure 5.10: Schematic view of the MPC structure.

These extensions can also be integrated in the proposed METANET-like traffic flow model for IVHS.

#### 5.4.4 Model predictive route choice control

Now we use the model of the previous subsection to derive a model-based predictive control approach that can be used by the area controllers to determine the optimal splitting rates.

More specifically, we adopt the model predictive control (MPC) scheme [30, 101] (see Figure 5.10). In the MPC control scheme a discrete-time model is used to predict the future behaviour of the process, and the MPC controller uses (numerical) optimisation to determine the control signals that result in an optimal process behaviour over a given prediction horizon. The resulting optimal control inputs are applied using a rolling horizon scheme. At each control step  $k$  the state of the traffic system is measured or estimated, and an optimisation is performed over the prediction horizon  $[kT, (k+N_p)T]$  to determine the optimal control inputs, where  $N_p$  is the prediction horizon. Only the first value of the resulting control signal (the control signal for time step  $k$ ) is then applied to the process. At the next control step  $k+1$  this procedure is repeated.

To reduce the computational complexity and to improve stability often a control horizon  $N_c$  ( $\leq N_p$ ) is introduced in MPC, and after the control horizon has been passed the control signal is taken to be constant. So there are two loops: the rolling horizon loop and the optimisation loop inside the controller. The loop inside the controller of Figure 5.10 is executed as many times as needed to find the optimal control signals at control step  $k$ , for the given  $N_p$ ,  $N_c$ , traffic state, and expected demands. The loop connecting the controller and the traffic system is performed once for each control step  $k$  and provides the state feedback to the controller. This feedback is necessary to correct for (the ever present) prediction errors, and to provide disturbance rejection (compensation for unexpected traffic demand variations). The advantage of this rolling horizon approach is that it results in an on-line adaptive control scheme that allows us to take changes in the system or in the system parameters into account by regularly updating the model of the system.

For our case the control variables in this set-up are the splitting rates at the nodes with more than one outgoing link (and if speed limits are included, also these speed limits). The optimisation signals include the control variables as well as the state variables of the macroscopic METANET-like traffic flow model for IVHS derived above.

One can also include various constraints such as maximal flows on certain links, max-

imal speeds (in case dynamic speed limits are used), maximal travel times for selected origin-destination pairs, etc. Moreover, there is a physical constraint that for a given node the sum of the splitting rates should be equal to 1:

$$\sum_{m \in O_n} \sum_{j \in J_m} \beta_{n,m,j}(k) = 1 \quad (5.49)$$

for all  $n, k$ .

A typical objective function to be used is the total time spent (TTS) by all the vehicles in the network. This includes both the time spent travelling through the network and the time spent waiting in the queues, if any. So at time step  $k$  we get the following expression for the total time spent in the period  $[kT, (k+N_p)T]$ :

$$J_{\text{TTS}}(k) = \sum_{j=0}^{N_p-1} \left( \sum_{(m,i) \in \mathcal{L}_{\text{ls}}} \rho_{m,i}(k+j) L_m \lambda_m + \sum_{(o,j) \in O_{\text{od}}} w_{o,j}(k+j) \right) T_s \quad (5.50)$$

where  $\mathcal{L}_{\text{ls}}$  is the set of all link-segment index pairs  $(m, i)$  of the network, and  $O_{\text{od}}$  the set of all origin-destination pairs  $(o, j)$ . Note that in (5.50) the expression between the brackets gives the total number of vehicles present in the links and queues of the network at time step  $k+j$ . This value is multiplied by the simulation time step  $T_s$  and summed over the discrete time steps to obtain the total time spent.

Minimising  $J_{\text{TTS}}$  then results in a nonlinear nonconvex optimisation problem with real-valued variables. In general this is an NP-hard problem [53] just like the original route choice problem. In fact with 5.50, we now have an optimisation problem with real-valued variables, which offers a major advantage compared to the mixed-integer optimisation of the original route choice problem, since due to the smoothness of the objective function, the real-valued optimisation will require a lower computational effort to find (sub)optimal solutions.

To solve the nonlinear optimisation problem we can use a global or a multi-start local optimisation method [118] such as multi-start sequential quadratic programming, pattern search, genetic algorithms, or simulated annealing.

### 5.4.5 Case study

In this subsection we present a simple case study involving a basic set-up to illustrate the area-level control approach for IVHS proposed in this section. First, we will describe the set-up and the details of the scenario used for our simulations. Next, we will discuss and analyse the obtained results.

#### Scenario

We consider a simple network of highways with one origin  $o_1$  and two destinations  $d_1, d_2$ , and three internal nodes  $v_1, v_2$ , and  $v_3$  (see Figure 5.11). Note that this example is similar to the one of Section 5.3.8. However, due to the different type of model used, the way the case study is specified will differ. The network of Figure 5.11 consists of three links<sup>10</sup>

<sup>10</sup>In contrast to the example of Section 5.3.8 we now explicitly consider these origin and destination links in order to account for the effect of the boundary conditions.

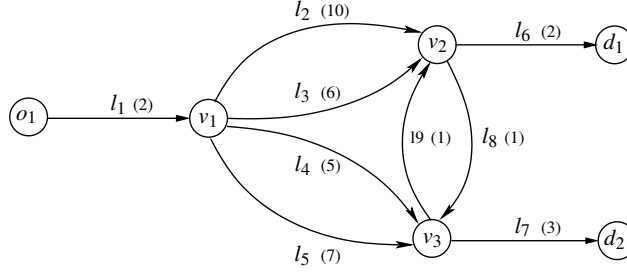


Figure 5.11: Set-up of case study network

Period (min)	0–10	10–30	30–40	40–60
$D_{o_1,d_1}$ (veh/h)	5000	8000	2500	0
$D_{o_1,d_2}$ (veh/h)	1000	2000	1000	0

Table 5.3: Demand profiles used in the case study

connecting  $o_1$  to  $v_1$ ,  $v_2$  to  $d_1$ , and  $v_3$  to  $d_2$ , as well as six links connecting the internal nodes allowing four possible routes to each destination (e.g.,  $d_1$  can be reached via  $l_2$ ,  $l_3$ ,  $l_4+l_9$ , and  $l_5+l_9$ ). In Figure 5.11, the values within brackets indicates the number of segments ( $N_m$ ) in the particular link. The length of a segment ( $L_m$ ) in any link is taken to be 1 km.

We consider four different cases (due to the use of two fundamental diagrams):

- Case A: no control case with human drivers,
- Case B: controlled case with humans drivers,
- Case C: controlled case with platoons.

For all the links we use the following values for the parameters of the METANET(-like) model (see also [90] and the previous subsections):  $v_{\text{free}}=120$  km/h,  $a = 1.867$ ,  $\kappa = 40$  veh/km/lane and  $\eta=60$  km<sup>2</sup>/h.

For the case if human drivers we use  $\rho_{\text{crit}}=33.5$  veh/km/lane,  $\tau=18$  s, and the fundamental  $V-\rho$  relation (5.37), while for the IV case we use  $\rho_{\text{crit}}=48.39$  veh/km/lane,  $\tau=8$  s, and the fundamental  $V-\rho$  relation (5.38).

We simulate a period of 60 min. The simulation time step  $T_s$  is set to 20 s. The demand pattern is piecewise constant during the simulation period and is given in Table 5.3. The demand to be processed in the period [10,30) higher than the capacity of the network, giving rise to an origin queue for each destination. For the proposed scenario the initial state of the network is taken to be empty. We choose  $N_p = 20$  and  $N_c = 6$ . For the sake of simplicity we take the simulation model to be equal to the prediction model.

### Control problem

The control variables considered for this case study are the splitting rates  $\beta_{n,m,j}(k)$  associated with all reachable destinations via outgoing links for each internal node for  $k =$



$0, 1, \dots, N_{\text{sim}} - 1$  where  $N_{\text{sim}} = 180$  is the total number of simulation steps (of length  $T_s = 20$  s) within the entire simulation period of 60 min.

Since it makes no sense to send vehicles reaching node  $v_2$  and going to destination 1, towards link  $l_8$  we set  $\beta_{v_2, l_6, 1}(k) = 1$  and  $\beta_{v_2, l_8, 1}(k) = 0$  for all  $k$ . Likewise, we set  $\beta_{v_2, l_6, 2}(k) = 0$  and  $\beta_{v_2, l_8, 2}(k) = 1$  for all  $k$ . For node  $v_3$  we have similar expressions:  $\beta_{v_3, l_7, 1}(k) = 0$  and  $\beta_{v_3, l_9, 1}(k) = 1$ ,  $\beta_{v_3, l_7, 2}(k) = 1$ ,  $\beta_{v_3, l_9, 2}(k) = 0$  for all  $k$ . So in fact the optimisation variables are  $\beta_{v_1, m, j}(k)$  for  $m = l_2, l_3, l_4, l_5$  and  $j = 1, 2$ .

We have the following constraints (cf. (5.49)):

$$\beta_{v_1, l_2, j}(k) + \beta_{v_1, l_3, j}(k) + \beta_{v_1, l_4, j}(k) + \beta_{v_1, l_5, j}(k) = 1$$

for  $j = 1, 2$  and for all  $k$ .

The goal of our area controller is to improve the traffic performance. The objective that we consider for our case study is minimisation of the total time spent (TTS) by all the vehicles in the network using routing as the control measure. The TTS for the entire simulation period can be expressed as (cf. (5.50)):

$$J_{\text{TTS, sim}} = \sum_{k=0}^{N_{\text{sim}}-1} \left( \sum_{(m,i) \in \mathcal{L}_{\text{ls}}} \rho_{m,i}(k) L_m \lambda_m + \sum_{(o,j) \in \mathcal{O}_{\text{od}}} w_{o,j}(k) \right) T_s \quad (5.51)$$

where  $\mathcal{L}_{\text{ls}}$  is the set of all link-segment index pairs  $(m, i)$  of the network, and  $\mathcal{O}_{\text{od}}$  the set of all origin-destination pairs  $(o, j)$ .

## Results and analysis

The results of the numerical experiments are listed in Table 5.4.

In case of no control (Cases A and B), the capacities of the direct links  $l_1, l_2, l_3$ , and  $l_4$  are consumed up to their maximum while the links  $l_8$  and  $l_9$  are not used due to the fact that all vehicles and platoons want to take the shortest routes. At the point when the demand exceeds the maximum capacity of the links, origin queues are formed. As the simulation advances, the queue length increases with time, thus leading to a huge total time spent.

For the controlled cases (Cases B and C) the area controller assigns the splitting rates at the internal node  $v_1$  and routes the traffic flow (human drivers or platoons) in a system-optimum manner such that the traffic performance is improved. When platoons of ACC-equipped vehicles are deployed in the traffic system, the traffic performance is improved more than the human drivers case. For these cases we have used the SQP function SNOPT, implemented via the function `snopt` of the Matlab Tomlab toolbox, to compute the optimal splitting rates. Compared to Case A this results in a performance improvement of about 3 % for Case B and of about 46 % for Case C.

## 5.5 Interface between area and roadside controllers

The area controllers will provide flow targets to the roadside controllers (e.g., using the approaches proposed in Sections 5.3, and 5.4), which then have to control the platoons that are under their supervision in such a way that these targets are met as well as possible. In our

<sup>10</sup>On a 1 GHz Athlon 64 X2 Dual Core 3800+ processor with 3 GB of RAM.

Case	$J_{\text{TTS, sim}}$ (veh.h)	improvement (%)
no control (A)	686.28	0 %
controlled human	670.91	3 %
controlled IVs	366.07	46 %

Table 5.4: Results for the case study. The improvement is expressed with respect to the no-control case with human drivers.

IVHS framework, the area and roadside controllers use a different level of representation of the traffic. However, consistency between the macroscopic traffic characteristics (such as flows, splitting rates) and the platoons has to be established to ensure efficiency of the approach. Hence, our approach requires an interface between these controllers, which is implemented as follows:

Once the optimal target flows are determined by the area controller, the roadside controller has to realise the target flows on each link as well as possible by using the available control measures such as speed limits, ramp metering, route guidance, and lane control. In order to achieve the specified optimal flows on the links, the roadside controller will use a performance criterion based on the minimisation of the difference between the reference flows and the actual flows. The roadside controller is aware of the complete information of the IVs (i.e., position, speed, lane, origin, destination) and hence can determine the actual flow  $x_{l,o,d}(k)$  on each link  $l$  for all  $(o,d) \in \mathcal{O} \times \mathcal{D}$ . At each control step  $k$ , the roadside controller will be updated with this flow information. The corresponding performance function  $J_{\text{perf}}$  used by the roadside controller is then given by

$$J_{\text{perf}} = \sum_{k=0}^{K_{\text{end}}-1} \sum_{(o,d) \in \mathcal{O} \times \mathcal{D}} \sum_{l \in L_{o,d}} (x_{l,o,d}^{\text{opt}}(k) - x_{l,o,d}(k))^2 \quad (5.52)$$

where  $x_{l,o,d}^{\text{opt}}(k)$  is the optimal flow specified by the area controller for link  $l$  in  $(o,d) \in \mathcal{O} \times \mathcal{D}$  for all  $k$ .

At the nodes, the roadside controller will additionally provide routing instructions for every platoon on the stretch under its supervision. The roadside controller will give these routing instructions by taking account of platoons' destinations and also the target flows on the adjacent highways. If necessary, the node controller can provide commands for platoon splits and merges, and determine new platoon compositions and platoon lengths.

## 5.6 Summary

We have considered the optimal route guidance problem for IVHS. Since the resulting optimisation problem is a nonlinear mixed-integer optimisation problem that in general is too involved for on-line, real-time implementation, we have explored approximations resulting in simplified but fast simulation models that yield mixed-integer *linear* or nonlinear *real-valued* optimisation problems, for both of which efficient solvers exist.

The first approach was based on a simplified model that describes the movement of the platoons in the network via flows. We have shown that using this model the optimal route choice control problem can be approximated by a linear or a mixed-integer linear problem. With a case study we have illustrated that the resulting approach can offer a balanced trade-off between computational efficiency and optimality.

In the second approach we have developed a new model to describe the flow of platoons in the IVHS based on the macroscopic METANET traffic flow model, which has been adapted to fit the case of platoons of intelligent vehicles equipped with Adaptive Cruise Control (ACC). This model has subsequently been used in a model predictive control approach for determining optimal splitting rates of the platoon flows at the nodes in the network. Lower-level roadside controller can then translate these splitting rates into actual route instructions for the platoons. The proposed approach has also been illustrated via a simple case study.

In our future research, we will compare these two approaches (with a microscopic model as simulation model), consider additional case studies, and assess the performance improvement of the proposed approaches with respect to an approach based on mixed-integer optimisation for the original route choice problem. We will also investigate the coordination and mutual interaction between various area controllers and between the area controllers and the roadside controllers.



## Chapter 6

# Traffic Management with Semi-Autonomous Intelligent Vehicles<sup>1</sup>

In this chapter, we deal with traffic congestion problems by using an intermediate approach which deploys a system that can be conceptually placed between the conventional traffic systems and fully automated IVHS that was discussed in the previous chapters. More specifically, human-driven vehicles in the conventional traffic systems are equipped with advanced driver assistance systems that can improve the traffic flow and that can also support the existing traffic control measures. We will in particular focus on how an infrastructure-based traffic controller can determine appropriate dynamic speed limits for such an intermediate traffic systems.

### 6.1 Introduction

The ever-increasing demand for mobility and transportation results in a growing traffic congestion problem. One of the promising approaches to reduce the frequency and impact of traffic jams is the optimal and efficient usage of the existing system. This approach will result in integrated traffic management and control systems that incorporate intelligence in both the roadside infrastructure and in the vehicles, and that are commonly called Intelligent Vehicle Highway Systems (IVHS). Vehicles in an IVHS are arranged in closely spaced groups called platoons. In the platooning approach cars travel on the highway in platoons with small distances (e.g., 2 m) between vehicles within the platoon, and much larger distances (e.g., 30–60 m) between different platoons. High speeds and short intraplatoon spacings allow more vehicles to be accommodated on the network, which substantially increases the maximal traffic flows [149].

In this chapter we discuss a traffic control approach that is situated in between the current situation with human drivers and no direct interaction between the roadside control

---

<sup>1</sup>This chapter is based on the MSc thesis of Mernout Burger [29].

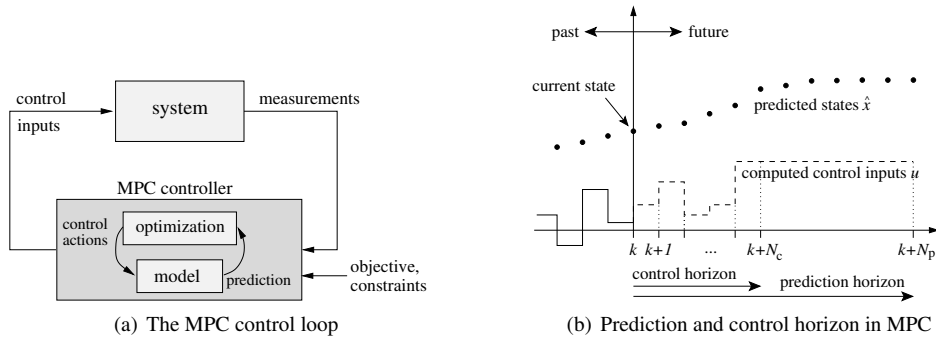


Figure 6.1: Schematic representation of MPC

system and the vehicles, and the future IVHS-based situation with fully controlled intelligent vehicles. In particular, we consider a situation in which intelligent vehicles driven by human drivers and equipped with ISA interact with the roadside controller (advisory ISA). Therefore, we call this type of vehicles semi-autonomous, as the human driver is still in control. We will investigate the possibility of using the semi-autonomous vehicles for traffic control, and design a traffic control method for this traffic system.

In Section 6.2.1, we present the strategy behind the model-based predictive control (MPC) approach. Later in Section 6.3, we propose an MPC approach to determine appropriate dynamic speed limits for the semi-autonomous traffic system. In Section 6.4, we discuss a model that could be used for simulation purposes. We also present the details of plugins that will be used as an interface between the simulation package Paramics and the Matlab engine where the traffic controller will be implemented. In Section 6.6, we apply the proposed approach to a case study based on simulations and we highlight the potential effects of the semi-autonomous systems on the traffic performance.

## 6.2 Preliminaries

### 6.2.1 Model Predictive Control (MPC)

Model Predictive Control (MPC) [30, 101] has originated in the process industry and it has already been successfully implemented in many industrial applications. MPC makes use of discrete-time models. Let  $T_{\text{ctrl}}$  be the control sampling interval, i.e., the time interval between two updates of the control signal settings. At each time step  $k$  (corresponding to time  $t = kT_{\text{ctrl}}$ ), the MPC controller first measures or determines the current state  $x(k)$  of the system. Next, the controller uses (on-line) optimisation and an explicit prediction model to determine the optimal values for the control measures over a given prediction period determined by the control horizon  $N_p$  (see Figure 6.1(b)). In order to reduce the computational complexity of the problem, one often introduces a constraint of the form  $u(k+j) = u(k+j-1)$  for  $j = N_c, \dots, N_p - 1$ , where  $N_c$  is called the control horizon.

The optimal control inputs are then applied to the system in a receding horizon approach as follows. At each control step  $k$  only the first control sample  $u^*(k)$  of the optimal control

sequence  $u^*(k), \dots, u^*(k+N_c-1)$  is applied to the system. Next, the prediction horizon is shifted one step forward, and the prediction and optimisation procedures over the shifted horizon are repeated using new system measurements. This receding horizon approach introduces a feedback mechanism, which allows to reduce the effects of possible disturbances and mismatch errors.

### 6.2.2 In-vehicle measures

We briefly discuss various in-vehicle technologies that serve as an intermediate in between the current traffic management system and IVHS-based approach. We will consider a traffic situation where the human-driven vehicles are equipped with ISA.

A standard speed limiter is a system that restricts the speed of the vehicle when the driver tries to exceed the maximum allowed driving speed. When the speed limiter is incorporated with the intelligence to adjust the maximum driving speed to the speed limit specified by the roadside infrastructure or to the prevailing location-based legal speed limit, and to provide feedback to the driver when that speed limit is exceeded, then we get the technology called Intelligent Speed Adaptation (ISA) [24, 39]. Based on the way the system reacts on exceeding the legal speed limit, advisory systems only give a warning (visually or via sound). Systems that decrease the possibility of speeding by increasing the resistance of the accelerator pedal, called an active acceleration pedal, and systems that do not allow a driver to exceed the speed limit (possibly also by limiting the throttle) can either be voluntary or mandatory (also called optional and obligatory). The voluntary system can be turned on and off by the user, the mandatory system is always on.

## 6.3 MPC for semi-autonomous traffic

Now, we focus on how MPC can be applied for speed control of the semi-autonomous traffic systems. In this section, we will use a macroscopic model (METANET) for predictions and will use microscopic model Paramics to represent the real-life traffic situation.

### 6.3.1 Prediction model

There exists a wide range of traffic models [43]. Microscopic models are more suitable for simulation purposes and, for off-line evaluation of developed control strategies than for on-line model-based control. In principle, microscopic traffic simulation models are usually not suited as MPC prediction model. An important factor that determines the choice of the prediction model to be used in MPC is the trade-off between accuracy and computational complexity. The prediction model needs to be able to simulate the entire network much faster than real-time in order to be able to predict the future states of the network in time. For the prediction of the future states of the traffic network with semi-autonomous vehicles, we use the macroscopic traffic flow model METANET [90, 91, 104, 115].

In the METANET model, the traffic network is represented as a directed graph with the links corresponding to the highway stretches. In this section, we will consider a network with one origin and one destination. Hence, we can use a destination-independent METANET model. The basic METANET model consists of link equations and node equations. A detailed description of destination-oriented METANET model was discussed in

$\rho_{\text{up}}(k)$	$\rho_1(k)$	$\rho_{i-1}(k)$	$\rho_i(k)$	$\rho_{i+1}(k)$	$\rho_{18}(k)$	$\rho_{\text{down}}(k)$
$v_{\text{up}}(k)$	$v_1(k)$	$v_{i-1}(k)$	$v_i(k)$	$v_{i+1}(k)$	$v_{18}(k)$	$v_{\text{down}}(k)$
upstream segment	direction of traffic flow $\rightarrow$				downstream segment	

Figure 6.2: State variables of the segmented road. Since we will work with a link, the link variable  $m$  is left out.

Chapter 5 in Section 5.4.3. Each highway link  $m$  is divided into  $N_{s,m}$  segments of equal length  $L_m$ , with  $\lambda_m$  the number of lanes on link  $m$ . For illustration, we consider a 10 kilometres stretch of a highway as shown in Figure 6.2. The highway is divided into 20 segments of 500 metres length, with two lanes at each segment. On all segments, the density and mean speed are measured. The first and last segments are used to describe boundary values, and the other 18 segments are used to control the traffic. In Figure 6.2 the uncontrolled parts are shown in grey.

Each segment  $i$  of highway link  $m$  is now characterised by

- Mean speed  $v_{m,i}(k)$  [km/h]
- Traffic density  $\rho_{m,i}(k)$  [veh/km/lane]
- Traffic flow  $q_{m,i}(k)$  [veh/h]

where  $k$  is the discrete time instant  $t = kT$ , where  $T$  is the simulation time step, typically 10 seconds for a segment length  $L_m$  of 500 metres.

Since we are working with only one link, also the link variable  $m$  can be left out of the equations. The extension for variable speed limits is used to model the controlled segments. The update equations for the state variables for a road segment then become

$$\begin{aligned}
\rho_i(k+1) &= \rho_i(k) + \frac{T}{\lambda L} (q_{i-1}(k) - q_i(k)) \\
&= \rho_i(k) + \frac{T}{L} (\rho_{i-1}(k)v_{i-1}(k) - \rho_i(k)v_i(k)) \\
v_i(k+1) &= \max \left[ v_i(k) + \frac{T}{\tau} (v_{\text{des},i}(k) - v_i(k)) \right. \\
&\quad \left. + \frac{T}{L} v_i(k) (v_{i-1}(k) - v_i(k)) - \frac{\eta_i(k)T}{\tau L} \frac{\rho_{i+1}(k) - \rho_i(k)}{\rho_i(k) + \kappa}, \quad v_{\min} \right] \quad (6.1)
\end{aligned}$$

where  $\kappa$  and  $\eta$  are model parameters,  $v_{\min}$  is minimum allowed mean speed (typical value is 4 km/h) and  $v_{\text{des},i}(k)$  is the desired mean speed of the vehicles. Let us now consider the (equilibrium) relation between the speed  $v$  and the density  $\rho$ . When the density on the road is very low and the average distance headway is large, the drivers travel at their desired speed. This is called free-flow driving. As the density starts to increase due to the increasing demand, the vehicles will start to reduce their speed slightly and follow their predecessor while maintaining a safe time headway. Once the critical density (i.e., the density at which the capacity of the network is being utilised at its maximum) is reached, the speed starts to decrease significantly resulting in a traffic jam. When the density is at its maximum ( $\rho_{\max}$ ),



the vehicle speed drops to almost zero. The (equilibrium) relation between the speed  $v$  and the density  $\rho$  can be modelled as [102]:

$$v_{des,i}(k) = v_{free} \exp \left[ -\frac{1}{a} \left( \frac{\rho_i(k)}{\rho_{crit}} \right)^a \right] \quad (6.2)$$

where  $\rho_{crit}$  is the critical density,  $v_{free}$  is the free-flow speed, and  $a$  is a model parameter.

Hegyí *et al.* [66] extended this model in order to include the effects of variable speed limits and to account for different reactions of drivers to varying downstream densities. The effect of variable speed limits is included by adjusting the desired speed equation  $v_{des,i}(k)$  to take the minimum of the density based free-flow speed, or the given variable speed limit  $v_{ctrl,i}(k)$  adjusted by a compliance factor  $1 + \alpha$  to indicate the noncompliance to the speed limit. Equation (6.2) then becomes

$$v_{des,i}(k) = \min \left( v_{free} \exp \left[ -\frac{1}{a} \left( \frac{\rho_i(k)}{\rho_{crit}} \right)^a \right], (1 + \alpha)v_{ctrl,i}(k) \right) \quad (6.3)$$

In order to include the different reactions of drivers to the downstream density the global variable  $\eta$  in (6.1) should be replaced by the segment, and time dependent variable

$$\eta_i(k) = \begin{cases} \eta_{high} & \text{if } \rho_{i+1}(k) \geq \rho_i(k) \\ \eta_{low} & \text{if } \rho_{i+1}(k) < \rho_i(k) \end{cases}$$

Origins are modelled using queues. The length of the queue  $w_o(k)$  is given by

$$w_o(k+1) = w_o(k) + T(d_o(k) - q_o(k)) \quad (6.4)$$

where  $d_o(k)$  is the demand and  $q_o(k)$  is the outflow at time  $k$ . The outflow is the minimum of the demand and the maximal flow that can enter the motorway:

$$q_o(k) = \min \left[ d_o(k) + \frac{w_o(k)}{T}, Q_o \frac{\rho_{max} - \rho_{\mu,1}(k)}{\rho_{max} - \rho_{crit,\mu}} \right] \quad (6.5)$$

where  $Q_o$  is the on-ramp capacity under free-flow conditions,  $\rho_{max}$  is the maximum density on the motorway, and  $\mu$  the index of the link under consideration.

The boundary states are also updated, using the state of the boundary segment of interest for the unknown states of the adjacent segments. This results in reduced update equations for the boundary segments, given by

$$\begin{aligned} v_{up}(k+1) &= \max \left[ v_i(k) + \frac{T}{\tau} (v_{des,up}(k) - v_{up}(k)) - \frac{\eta_{des,up}(k)T}{\tau L} \frac{\rho_1(k) - \rho_{up}(k)}{\rho_{up}(k) + \kappa}, v_{min} \right] \\ \rho_{up}(k+1) &= \frac{q_{up}(k+1)}{\lambda v_{up}(k+1)} = \frac{q_{up}(k)}{\lambda v_{up}(k+1)} = \rho_{up}(k) \frac{v_{up}(k)}{v_{up}(k+1)} \\ v_{down}(k+1) &= \max \left[ v_{down}(k) + \frac{T}{\tau} (v_{des,up}(k) - v_{up}(k)) + \frac{T}{L} v_{down}(k) (v_{18}(k) - v_{down}(k)), v_{min} \right] \\ \rho_{down}(k+1) &= \rho_{down}(k) + \frac{T}{L} (\rho_{down-1}(k)v_{down-1}(k) - \rho_{down}(k)v_{down}(k)) \end{aligned} \quad (6.6)$$

where  $v_{des,up}$  and  $v_{des,down}$  are calculated as in (6.3), where the first term is used, since there are no speed limits on the boundary segments. The parameter value  $\eta_{up}$  can be determined

as in (6.3.1). In the update for the upstream density  $\rho_{\text{up}}(k+1)$ , we have assumed a constant flow, so  $q_{\text{up}}(k+1)=q_{\text{up}}(k)$ . Also,  $v_{18}(k)$  is the mean speed on segment 18. The parameters of the METANET model ( $\tau$ ,  $\alpha$ ,  $\kappa$ ,  $k_{\text{crit}}$ ,  $v_{\text{free}}$ ,  $a$ ,  $\eta_{\text{high}}$  and  $\eta_{\text{low}}$ ) need to be determined by calibration as described in Section 6.6.

### 6.3.2 Performance criteria and constraints

Possible performance criteria  $J_{\text{perf}}(k)$  for MPC are the total time spent in a traffic network, the total throughput, the total fuel consumption, safety, or a combination of these, all evaluated over the time period  $[kT_{\text{ctrl}}, (k+N_p)T_{\text{ctrl}}]$ . Moreover, in order to prevent oscillations and frequent shifting in the control signals, one often adds a penalty on variations in the control signal  $u$  in both time and space, which results in the total performance function

$$\begin{aligned} TTS = J_{\text{perf}}(k) &+ \lambda_1 \sum_{j=0}^{N_c-1} \sum_{i=1}^{N_s} \|u_i(k+j) - u_i(k+j-1)\|_2 \\ &+ \lambda_2 \sum_{j=0}^{N_c-1} \sum_{i=1}^{N_s-1} \|u_{i+1}(k+j) - u_i(k+j)\|_2, \end{aligned} \quad (6.7)$$

at control step  $k$ , where  $\lambda_1, \lambda_2 > 0$  are weighting factors, and  $u_i(k)$  is the control signal in segment  $i$  at time step  $k$ .

The MPC controller also explicitly takes into account operational constraints such as minimum and maximum speeds, etc.

## 6.4 Simulation model - Paramics

Paramics is a suite of software tools used to model the movement and behaviour of individual vehicles on urban roads and highway networks. It uses microscopic characteristics such as speed, position, etc, to simulate the interaction between vehicles and the road network. We are interested in this software program, mainly due to the possibility to add extra functionalities by creating plugins.

During a simulation, the acceleration for every vehicle in the network is calculated, based on the speed limit, headway, speeds of vehicles nearby, and the desired free-flow speed of the vehicle. For this research, we will change the speed limits dynamically, and implement ISA-equipped vehicle behaviour in Paramics, by changing the desired free-flow speed of vehicles, which will be discussed in Section 6.5.

The longitudinal aspects of the driver tasks can be classified as free-flow, car-following, and stop-and-go behaviour. In free-flow behaviour, the vehicles can travel at their desired speed (corresponding to the speed limit, e.g, 120 km/h). As the traffic demand increases, the vehicles start to follow their predecessors on the same lane at closer distances and at reduced speeds (50–80 km/h). Once the capacity of the highway is being utilised at its maximum, the vehicles move with stop-and-go movements (0–40 km/h). There are two important microscopic submodels used in Paramics, namely for the car-following behaviour and the way the desired headway is determined per vehicle.

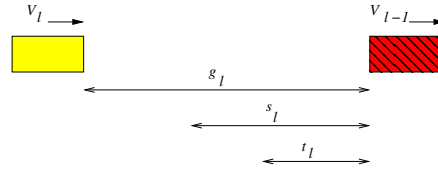


Figure 6.3: Representation of target points for car-following in Paramics

### Car-following behaviour

The car-following behaviour is split into three modes; braking, cruising, and accelerating. For all modes, the concept of a target point is used. For vehicle  $l$ , this target point  $s_l$  is calculated as

$$s_l = h_{\text{des}} v_{l-1} \quad (6.8)$$

where  $s_l$  is the spacing,  $h_{\text{des}}$  is the desired headway and  $v_{l-1}$  is the speed of the preceding vehicle. The situation is shown in Figure 6.3. In order to account for the reaction time of human drivers the speed  $v_{l-1}$  used is actually the speed of the preceding vehicle some time in the past. In order to make the vehicles reach the desired headway faster, an adjusted target point  $t_l$  is used, calculated by

$$t_l = \frac{s_l^2}{g_l}$$

where  $g_l$  is the current distance between the vehicles. Note that this indeed will speed up reaching the target point dependent on the desired headway and vehicle speed, since

$$\begin{aligned} t_l < s_l & \text{ if } g_l > s_l \\ t_l = s_l & \text{ if } g_l = s_l \\ t_l > s_l & \text{ if } g_l < s_l \end{aligned}$$

and using  $t_l$  as the desired spacing the difference  $|(t_l - g_l)|$  between the current spacing and desired spacing will always be bigger or equal than when we use  $s_l$  as desired spacing.

The cruising mode is split up into three separate regions, called A, B, and C. The braking mode will be called region D, and the acceleration mode is region E. The characteristics of the regions are

- Region A: the following vehicle has overshoot the target point (the spacing is less than the target value), and an attempt is made to achieve the target speed as quickly as possible.
- Region B: the leading vehicle is pulling away from the following vehicle.
- Region C: the vehicles are at a constant separation or coming closer together.
- Region D: the leading vehicle is decelerating with a value above a certain threshold so that, the following vehicle will notice the leading vehicle is braking. The following vehicle will start to brake as well.

- Region E: the leading vehicle is accelerating with a value above a certain threshold, the following vehicle will notice the leading vehicle is accelerating. The following vehicle will start to accelerate as well.

The accelerations for the following vehicle for the different regions are given by

$$\begin{aligned}
 a_A &= k_1 \Delta v + k_2 \frac{g_l - t_l}{g_l} \\
 a_B &= k_1 \Delta v + k_3 \frac{g_l - t_l}{t_l} \\
 a_C &= c - \frac{(\Delta v)^2}{g_l - t_l} \\
 a_D &= a_{\min} \\
 a_E &= a_{\max}
 \end{aligned}$$

where  $k_1 = 1.0 [s^{-2}]$ ,  $k_2 = 1.0 [s^{-2}]$ ,  $k_3 = 0.005 [s^{-2}]$ ,  $\Delta v = v_{l-1} - v_l$ ,  $c$  the bunching acceleration to bring the vehicles together rapidly, and  $a_{\min}$  and  $a_{\max}$ , are the minimum and maximum acceleration of the vehicle respectively.

### Headway of vehicles

The desired headway is dependent on the type of vehicle, the situation in the network, and the driver behaviour. It is based on an average headway, which can be set for the simulation. Each vehicle has several multiplication factors, which together determine the desired headway of the individual vehicle. Each vehicle type has a type-dependent multiplication factor. Using this factor, we can, e.g., include the larger headway a heavy goods truck needs compared to a car, since it can decelerate less strongly. Two important driver characteristics used in Paramics for manipulating the driver behaviour are the aggressiveness and awareness. Aggressiveness is used to influence the headway everywhere in the network. The multiplication factor for aggressiveness is calculated using

$$F_{ag} = \begin{cases} \frac{36 - 5\theta_{ag}}{12 - \theta_{ag}}, & \theta_{ag} \leq 4 \\ \frac{16}{8}, & \theta_{ag} > 4 \end{cases} \quad (6.9)$$

where  $\theta_{ag}$  is the assigned level of aggressiveness of the driver, indicated by a value between 1 and 8. A lower value for  $\theta_{ag}$  means that the driver is less aggressive, and will have a larger desired headway. The awareness factor is used to calculate the desired headway when a vehicle is near a lane drop. It is used to simulate to what extent a driver will keep distance to the vehicle in front in order to let other vehicles merge into the other lane. This multiplication factor is given by

$$F_{aw} = \begin{cases} \frac{\theta_{aw} + 4}{8}, & \theta_{aw} \leq 4 \\ \theta_{aw} - 3, & \theta_{aw} > 4 \end{cases} \quad (6.10)$$

where  $\theta_{aw}$  is the awareness factor of the driver, indicated by a value between 1 and 8. A higher value means that the driver is more aware of other vehicles wanting to get onto the lane, and the driver will have a larger desired headway near lane drops.

## 6.5 Coupling Paramics and Matlab

Since there is no existing interface between the Paramics and Matlab, we will create a plug-in to simulate the behaviour of IV-based vehicles. We will develop plug-ins for the partially automated vehicles to capture their reaction to the control measures. The first plugin is created to be able to measure mean speeds, densities, flows, and the number of passing vehicles on a road segment, and to send this data to a Matlab engine. A traffic controller created in Matlab can then be used to calculate control actions for the traffic network in Paramics. The second plugin is created to change the behaviour of the vehicles in Paramics. Using this plugin, ISA-equipped vehicles can be used in the simulation. Also the vehicle behaviour when using unenforced and enforced speed limits can be influenced by this plugin.

### 6.5.1 Interface plugin

The Interface plugin can be used as a stand-alone plugin and is created in Paramics. It is created to measure data in Paramics, and also to use it to control the simulated traffic in Paramics, via a controller implemented in Matlab. The traffic measurements needed to calculate the control actions are obtained from Paramics. These control actions will then be sent back to Paramics to simulate the response of traffic on these control actions.

#### Measurement data

The plugin makes use of data collected by detectors. When a vehicle passes a detector, it will be counted, and its speed will be used to calculate the mean speed on the segment the detector is on. Therefore, in order to be able to use the Interface plugin, some detectors should be placed in the Paramics project and the details on how to do this can be found in the Paramics Modeller manual. Every simulated minute the plugin will determine the values of the measurements in Paramics, and send this data to the Matlab engine. When the simulation is finished, the data will be stored in three different mat-files namely `detector.mat`, `control.mat` and `odmatrix.mat`. These mat-files can be found in the same directory from where we started the Paramics program.

The file `detector.mat` contains a three-dimensional matrix representing simulated minute, detector number, and index number of the detector data. The data obtained from the detectors is stored in an array with index numbers indicating simulation time, estimated space-mean density, time-mean density, approximated space-mean density, average flow, estimated space-mean speed, time-mean speed, approximated space-mean speed, vehicle count (minute) and vehicle count (total). The approximated values are calculated by determining the values (e.g., space mean density) of a segment at every second, sum these values for one minute, and determine the average value for that minute. The estimated values are determined using the basic macroscopic traffic characteristics equations. For example, `detector(20, 7, 5)` will give the flow (index 5) at detector 7 for the 20<sup>th</sup> simulated minute.

The file `control.mat` contains a matrix representing simulated minute, controlled segment number, and control action number. The control action data mat-

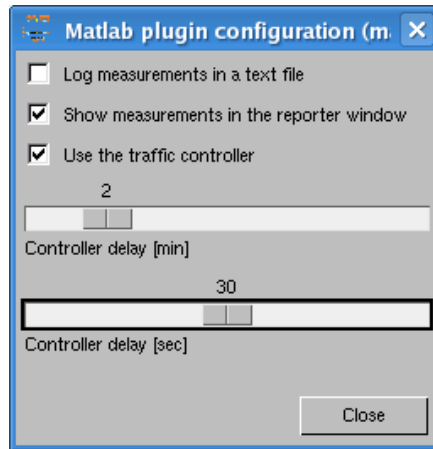


Figure 6.4: The Interface plugin dialog box

rix contains the dynamic speed limits ( $ctrnr=1$ ), and the dynamic time headways ( $ctrnr=2$ )<sup>2</sup>, for all timesteps at all controlled segments. The plugins are made with more types of intelligent vehicles in mind. Another interesting type of intelligent vehicles are ACC equipped vehicles. Besides dynamic speed limits, also the time headway of these vehicles might be changed in order to improve the network performance. In this way both means speeds and densities can be influenced, giving the traffic controller an extra degree of freedom. This is why the traffic controller is supposed to return both speed limits and time headways.

The file `odmatrix.mat` contains four-dimensional matrix representing simulated minute, detector number, origin number, and destination number. The origin-destination data array contains the number of vehicles that came from a certain origin, and want to go to a certain destination in the network, counted in the passed simulated minute. The index numbers of the origins and destinations correspond to the zone numbers in Paramics.

### Interface plugin parameters

Different parameters of the Interface plugin shown in Figure 6.4 are described:

- Log and show measurements: The checkboxes indicate to the plugin whether it should log the measurements in a textfile as `data.txt` or show the measurements in the reporter window.
- Use the traffic controller: This checkbox tells whether the traffic controller should be called at every simulated minute. When the traffic controller is set to 'true', the Interface plugin will call an m-file called `traffic_controller` at every simulated minute. This function can be used as `[speedlimitdata headwaydata]=traffic_controller(minute, state)`, where
  - `speedlimitdata` is a vector containing the dynamic speed limits for the controlled segments, in kilometres per hour. These speed limits are the control

<sup>2</sup>Desired headway control can be used for controlling ACC-equipped vehicles.

actions of the controller, which will be applied to the network during the next simulation minute.

- `headwaydata` is a vector containing the dynamic time headways for the controlled segments, in seconds. These time headways are the control actions of the controller, which will be applied to the network during the next simulation minute.
  - `minute` is the current simulation minute, for which the measurements are taken.
  - `state` is a matrix containing the mean speeds and densities from the measured segments. These measurements can be used by the controller to determine its next control actions.
- **Controller delay [min]/[sec]:** This is the amount of minutes plus seconds the plugin will wait after the start of the calculation of new control values, before the control actions are applied. The Paramics simulation will wait until the controller is finished. In this way, the time a controller may use to calculate the next control values can be taken into account in the simulation. The value for the minutes is an integer between 0 and 10, the value for the seconds is an integer between 0 and 59.

### 6.5.2 ISA plugin

The ISA plugin is dependent on the Interface plugin, so the Interface plugin should also be loaded if one intends to use the ISA plugin. It is created to simulate vehicles equipped with an ISA system. This is done by giving all vehicles of a certain type (“ISA vehicles”) a compliance factor. First, we motivate the reason for creating this plugin, followed by an explanation of the parameters of the plugin.

#### Intended use

We want to investigate the differences in controlling three scenarios, namely traffic driving with unenforced speed limits, traffic driving with enforced speed limits, and traffic consisting of vehicles equipped with ISA systems. Vehicles equipped with ISA can be modelled using their free-flow speed distribution. The effects on mean speeds and the speed distribution for vehicles can be modelled by a normal distribution. They are based on research using ISA with an active acceleration pedal [70]. This research was carried out in the city of Lund, Sweden. It covered measurements on main streets and arterial roads. On the main streets, there was no significant difference in speed distribution between vehicles with and without active acceleration pedal. On the arterial roads (with speed limits of 50 and 70 km/h), the standard deviation of the active acceleration pedal equipped vehicles was about 5.7% of the speed limit, and without active acceleration pedal it was about 9.9% of the speed limit. The decrease in the standard deviation on arterial road is due to the decrease in speed of the fastest vehicles, but also due to the increase of speed of the slowest vehicles. The mean speed on the arterial roads is about 110% of the speed limit for vehicles without active acceleration pedal, and about 100% for vehicles with active acceleration pedal. A similar conclusion was drawn from measurements on a highway in the Netherlands [67]. Here it was found that vehicles drive about 10% faster when the speed limits are unenforced, and about 10% too slow when they are enforced.

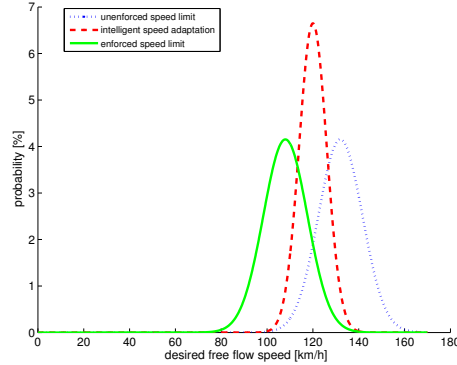


Figure 6.5: Gaussian distributions of the desired free-flow speeds

We assign a compliance factor to every vehicle, based on the distribution belonging to its type. This factor is used to give a vehicle a desired speed, based on the current speed limit on the road. The calculation for the desired speed thus becomes

$$v_{des,n} = \alpha_{cf,n} \cdot v_{speedlimit}$$

where,  $\alpha_{cf,n}$  represents the compliance factor of vehicle  $n$  and is a Gaussian random number. The used mean value and standard deviation used depend on the scenario that is considered.

In the case of simulating ISA equipped vehicles, a distribution is used with the mean value near 1.00, and a small standard deviation. In literature, values are found for the standard deviation of ISA vehicles [70]. Measurements on arterial roads with speed limits of 50 and 70 km/h give standard deviations of about 5.7%. Since this article also shows that the standard deviation decreases with increasing speeds, and we simulate a highway with a speed limit of 120 km/h, a standard deviation of 5.0% is used.

Also for the non-ISA vehicles, a compliance factor can be used. For the situation of unenforced speed limits a Gaussian random number is determined with a mean value near 110% of the speed limit, and a larger standard deviation. This value can be found in both [67] and [70]. In the latter article, the standard deviation on the arterial roads is 10.2% at 50 km/h and 9.6% at 70 km/h. Here we assume that the standard deviation is 8.0% at 120 km/h.

When the speed limits are enforced (e.g. by trajectory control), human drivers tend to drive slower than the speed limit [67], at about 90% of the speed limit. This behaviour can also be described by using a random compliance factor. The standard deviation is taken equal to the one in the unenforced situation. The probability distribution functions for the three different scenarios are shown in Figure 6.5. The used speed limit is 120 km/h. Due to the smaller standard deviation for the ISA equipped vehicles compared to the other two types, the chance of having a desired free-flow speed near the mean value becomes bigger, as can be seen by the higher and narrower peak for this type.



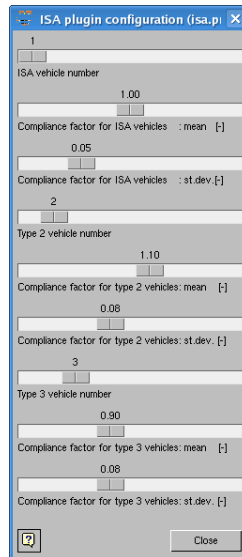


Figure 6.6: The ISA plugin dialog box

### ISA plugin parameters

The different parameters of the ISA plugin shown in Figure 6.6 are defined as follows,

- Vehicle numbers: This parameter indicates which vehicle type in Paramics should behave as ISA-equipped vehicles and which vehicles should behave as second-type and third-type vehicles. The value is an integer between 1 and 10.
- Compliance factor for vehicles: mean and st.dev.: These parameters give the mean value and the standard deviation for the compliance factor for ISA-equipped, type 2 and type 3 vehicles. The value can be chosen between 0.50 and 1.50.

## 6.6 Case study

Now we present a simple case study in which the MPC control strategy for the roadside controller layer that has been described in Section 6.3 is applied. First we will describe the set-up of the network, which will be followed by a description of how this network can be modelled by the METANET model as well as other implementation details. Next, we will discuss and analyse the results obtained from the simulations. For our simulations, we will consider dynamic speed limits as the control measure.

### 6.6.1 Set-up

The network used for the simulation is similar to the stretch discussed in Figure 6.2 and is a two-lane highway stretch, with one on-ramp at the end. The stretch consist of 20 segments

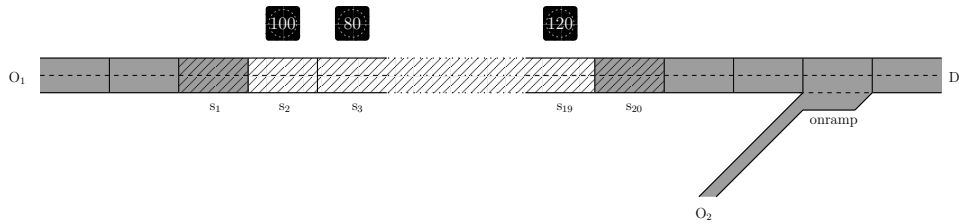


Figure 6.7: Schematic layout of the network

with a length of 500 m each. The first and last segments are used to emulate boundary values, and the other 18 segments are used to control the traffic. A schematic drawing is shown in Figure 6.7. For this case study, we compare three different scenarios:

- Unenforced speed limits,
- Enforced speed limits,
- ISA-equipped vehicles.

### 6.6.2 Scenario

A shock wave is a sudden change in flow or density moving upstream or downstream on the road. In an unstable region, where the density is higher than the critical density, shock waves may occur only when there are disturbances in traffic, like braking or merging. In a stable region with the density less than the critical density, any disturbance will vanish without intervention. Since the difference between the stable and unstable region will not be sharp in practice, the notion of metastability is introduced [66]. Hegyi *et al.* [66] argue that shock waves can only be dissolved in the metastable region.

In our case study, the on-ramp is used to create shock waves in the network at desired moments, by temporarily adding extra vehicles to the stretch. When the capacity of the road is reached, a traffic jam will start to build up. This traffic jam will move upstream with a speed of about 15 km/h. The aim of the traffic controller is to reduce or dissolve this traffic jam.

Also, on the upstream part of the measurement area, some segments are added to keep the created shock wave from reaching the origin, where vehicles enter the network, within the hour. Between the 20 hatched measurement area and the on-ramp, there are two segments without detectors. This is done to avoid measuring mean speeds based on two lanes with a large difference in speed.

For every situation, the following cases are simulated:

- No shock wave case: no traffic is entering the main road at the on-ramp,
- Uncontrolled case: a shock wave enters the measured area, but no control actions are taken,
- Controlled case: a shock wave enters the measured area, and dynamic speed limit control is used to dissolve the shock wave.

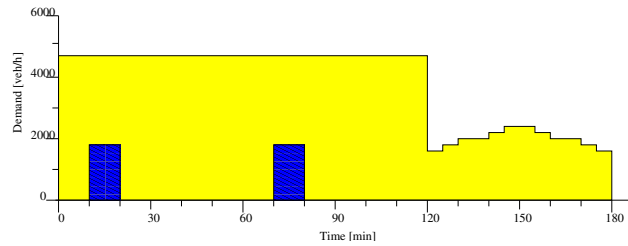


Figure 6.8: Demand patterns for the calibration of the model

### 6.6.3 Models

We will use the microscopic traffic simulator Paramics for traffic simulation purposes. The MPC traffic controller that is implemented in the Matlab needs a traffic model to predict the states when the speed limits are applied. We implement a METANET model for prediction purpose. Since we use different models for simulation and prediction purposes, we need to calibrate the prediction model.

#### Calibration of METANET model

For this calibration, a scenario is created with a duration of 3 hours. In Figure 6.8, the demand patterns for the calibration are shown. The light pattern shows the demand on the main road ( $O_1$  to  $D_1$  in Figure 6.7), and the dark pattern shows the demand for the on-ramp ( $O_2$  to  $D_1$  in Figure 6.7). In the first hour, a traffic jam is created, which is (almost) dissolved by a manually created speed limit pattern. This part is used to fit the model for capturing the traffic response to dynamic speed limits at high density. In the second hour, the traffic jam is created using the same method as in the first hour, but here the speed limits are kept constant at 120 kilometres per hour. This part is used to fit the model to capture the shock wave phenomenon. In the third hour, the demand is low and changing, and a manually created dynamic speed limit pattern is applied. Here the model is fit to capture the traffic behaviour at low densities. At these low densities the desired speed  $v_{des}$  is more likely to be limited by the dynamic speed limits than at high densities. The combination of these three scenarios should give a good basis for calibrating the prediction model for the traffic.

Using this calibration scenario, two simulations are performed for each of the three traffic scenarios. One obtained data set is used for calibrating the METANET model, the other is used for validating the METANET model. The set for validation is needed, because both the generation of vehicles and the assignment of the speed limit compliance factor are stochastic processes, and therefore there is a difference in traffic behaviour for different simulation runs. A set of parameters that represents the traffic behaviour for one simulation run very well, might therefore not be very suited for another run.

**Calibration objective** The METANET model is calibrated by optimisation of the model parameters. This optimisation is done in a similar way as discussed in Section 6.2.1, using a rolling horizon. Instead of using the speed limits as the variables to optimise, the parameters

Table 6.1: Values accounted for the three scenarios

Scenario	calibration data			validation data		
	VAF <sub><math>\rho</math></sub>	VAF <sub><math>v</math></sub>	average	VAF <sub><math>\rho</math></sub>	VAF <sub><math>v</math></sub>	average
unenforced	97.36	99.41	98.39	96.72	99.26	97.99
enforced	96.49	98.94	97.71	94.05	98.54	96.29
ISA	96.42	98.96	97.69	95.69	98.60	97.14

of the METANET model are used. The vector containing the parameters is

$$P = [\tau \quad \alpha \quad \kappa \quad \rho_{\text{crit}} \quad v_{\text{free}} \quad a \quad \eta_{\text{high}} \quad \eta_{\text{low}}]^T$$

This vector is changed by the Matlab function `fmincon`, until it finds a minimum value for the objective function

$$J_{\text{cal}}(P) = \frac{1}{N_p} \sum_{i=1}^{N_s} \sum_{m=1}^{M-N_p} \sum_{k=m}^{m+N_p} \left\{ \left( \frac{\rho_{\text{dat},i}(k) - \rho_{\text{sim},i}(k)}{\bar{\rho}_{\text{dat},i,k}} \right)^2 + \beta_1 \left( \frac{v_{\text{dat},i}(k) - v_{\text{sim},i}(k)}{\bar{v}_{\text{dat},i,k}} \right)^2 \right\} \quad (6.11)$$

where  $N_p$  is the prediction horizon,  $N_s$  is the number of segments,  $M$  is the length of the simulation in minutes,  $\rho_{\text{dat},i}$  and  $v_{\text{dat},i}$  are the measured densities and mean speeds at segment  $i$ ,  $\rho_{\text{sim},i}$  and  $v_{\text{sim},i}$  are the simulated densities and mean speeds at segment  $i$  as predicted by the METANET model using the current values for the parameter vector  $P$ , and  $\bar{\rho}_{\text{dat},i,k}$  and  $\bar{v}_{\text{dat},i,k}$  are the mean density and speed over the measured values from  $k$  to  $k+N_p$  for segment  $i$ . The multiplication factor  $\beta_1$  can be used to set the relative importance for getting accurate predictions for either the densities or the mean speeds. If, e.g.,  $\beta_1 > 1$ , the predictions for the densities will become more accurate, thereby reducing the accuracy of the predicted mean speeds. In this research  $\beta_1$  is 1.

**Validating METANET model** For each of the three scenarios, more than 50 calibration runs have been done on a data set containing measurement data for the particular scenario. The value obtained by (6.11) gives a measure on how well the densities and mean speeds can be predicted using the METANET model with the accompanying parameter values. The 20 parameter sets with the lowest values for  $J_{\text{cal}}$  are tested on the validation data. The first test is calculating the Variance Accounted For (VAF). The VAF is calculated as

$$\text{VAF}_x = 100\% \cdot \left\{ 1 - \frac{\sum_{k=1}^M \sum_{i=1}^{N_s} (x_{\text{dat},i}(k) - x_{\text{sim},i}(k))^2}{\sum_{k=1}^M \sum_{i=1}^{N_s} x_{\text{dat},i}^2(k)} \right\} \quad (6.12)$$

where  $M$  is the length of the simulation,  $N_s$  is the number of segments,  $x$  is the variable of interest (either  $\rho$  or  $v$ ),  $x_{\text{dat},i}$  is the measured data from Paramics at segment  $i$ , and  $x_{\text{sim},i}$  is the simulated data for segment  $i$ , using the METANET model. The VAF gives a percentage of how well the simulated data represents the measured data.

The mean values for the VAFs of the 20 lowest values of  $J_{\text{cal}}$  for the three scenarios are given in Table 6.1. The variances for these mean percentages are all smaller than  $2.0 \cdot 10^{-3}$ ,

Table 6.2: Parameter values for the METANET model

Scenario	$\tau$	$\alpha$	$\kappa$	$\rho_{\text{crit}}$	$v_{\text{free}}$	$a$	$\eta_{\text{high}}$	$\eta_{\text{low}}$
unenforced	0.0042	-0.0521	126.91	26.69	176.78	1.62	204.60	153.59
enforced	0.0025	-0.2737	176.69	29.28	169.10	1.37	167.35	91.96
ISA	0.0061	-0.1257	137.05	29.39	169.21	2.65	233.81	128.31

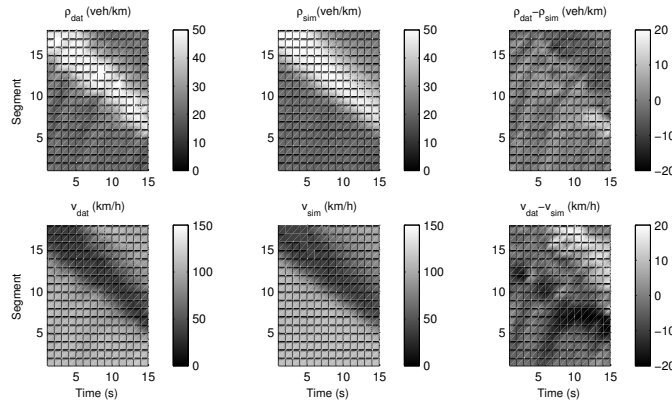


Figure 6.9: Illustration of validation

so the 20 best parameter sets give quite similar results. From the table it can be seen that the values for the validation data are lower compared to the values for the calibration data. This is as expected, since the parameters are optimised to represent the calibration data. Another observation from the table is that the densities are predicted less accurate than the mean speeds. The prediction of the densities can be improved by taking a smaller value for the weighting factor  $\beta_1$  in (6.11). This is not done due to limited time, but it will probably improve the overall performance of the controller, since it will give more accurate predictions. Despite the lower accuracy of the densities, the average VAFs for both the calibration data and the validation data are good. Per scenario, the parameter set resulting in the highest average VAF is used for the controller prediction model. The parameter values used are shown in Table 6.2.

Besides validating the obtained parameters by calculation, we also provide an illustration of this validation. In order to do so, some 15 minute predictions are done, and plotted in a figure, together with the measured data, and the difference of both. An example of such a plot is given in Figure 6.9. The figure shows the prediction from the 80th to the 95th minute, for the unenforced speed limit scenario.  $\rho_{\text{dat}}$  and  $v_{\text{dat}}$  are the measured densities and mean speeds from Paramics,  $\rho_{\text{sim}}$  and  $v_{\text{sim}}$  are the simulated densities and mean speeds as predicted using the METANET model. At this time, a shock wave is starting to enter the network, and no control is used. The shock wave is clearly visible, starting at the upper left corner. This visualisation shows that the speed at which the shock wave moves upstream is quite accurate, and also the width is predicted well. The prediction is much smoother than the measured situation, but it needs not give more detail for use in the controller. For

both the calibration and validation, the measured data is used to get the future states at the boundary segments.

#### 6.6.4 Control problem

When no control is applied to the network, and the traffic demand stays high, the traffic jam will continue to move upstream, thereby slowing down all vehicles passing it. The aim of the traffic controller is to reduce or dissolve this traffic jam, and thereby reducing the time vehicles spend in the monitored part of the traffic network. The corresponding performance function  $J_{\text{perf}}(k)$  used in the calculation of the total time spent as in (6.3.2), is then given by

$$J_{\text{perf}}(k) = T_{\text{ctrl}}N(0)K + T_{\text{ctrl}}^2 \sum_{k=0}^K (K-k)(q_{\text{dem}} - q_{\text{out}}(k)) \quad [\text{veh} \cdot \text{h}] \quad (6.13)$$

where  $T_{\text{ctrl}}$  is the controller time step ( $\frac{1}{60}$  hour),  $N(0)$  is the number of vehicles on the segments 1 to 20 at  $k = 0$ <sup>3</sup>,  $K$  is the number of time steps taken into account,  $q_{\text{dem}}$  is the constant demand used for the simulation (desired inflow), and  $q_{\text{out}}$  is the outflow measured at the 20<sup>th</sup> segment. In the total MPC objective function given in (6.3.2), we have also included a penalty term with  $\lambda_1 = 10$  and  $\lambda_2 = 10$ .

In order to be able to compare the different situations for all scenarios at different traffic demands, the mean travel time per vehicle is calculated. This is done using

$$\text{TPV}(K) = 60 \cdot \frac{\text{TTS}(K)}{T_{\text{ctrl}}Kq_{\text{dem}}} \quad [\text{min}] \quad (6.14)$$

where  $\text{TTS}(K)$  is the TTS calculated using (6.3.2). The factor 60 is used to convert hours into minutes. The time per vehicle gives the average time it took a vehicle to pass the 10 kilometres long highway. In order to compare the different situations and demands, the times per vehicle are normalised based on an ideal situation, where the mean speed is 120 km/h in every segment at every time instant. In this case the TPV for the 10 kilometre long highway part is

$$\text{TPV}_{\text{norm}} = 60 \cdot \frac{10}{120} = 5 \quad [\text{min}] \quad (6.15)$$

As a constraint, the dynamic speed limits are given a maximum and minimum allowed value. The upper bound for the speed limits is 120 km/h, and the lower bound value is 40 km/h. For the calculation of the optimal control values, all speed limits are constrained to this range. When the optimal values are found, they are rounded to a multiplicity of 10 km/h, since this is more clear for human drivers, and also technically feasible without large investments.

#### 6.6.5 Results and analysis

When the density is high, it is more difficult to control the traffic, since the mean speed might already be below the control speed (see the definition of  $v_{\text{des}}$  in (6.1)). Therefore, simulations are done using densities at which the shock wave can dissolve without using

<sup>3</sup>For the calculation of the TTS, the first 10 minutes of the measured data are rejected. Since the simulation starts with no vehicles in the network, we take this time to initialise.

Table 6.3: Measured results for the unenforced speed limit scenario

$q_{\text{dem}}$	Case	#1	#2	#3	#4	#5	TTS: mean (std)	TPV
4700	no shock wave	494.73	452.15	435.98	414.88	428.30	445.21 (6.9%)	5:41
4700	uncontrolled	520.42	517.48	536.13	475.98	539.58	517.92 (4.9%)	6:36
4700	controlled	513.45	488.43	521.35	479.75		500.75 (4.0%)	6:24
4900	no shock wave	493.90	472.60	492.78	521.10	489.43	493.96 (3.5%)	6:03
4900	uncontrolled	635.10	584.92	643.72	571.85	588.63	604.84 (5.3%)	7:24
4900	controlled	575.30	654.12	589.77	572.15		597.84 (6.4%)	7:19

control, and at densities where the shock wave remains. For each scenario, five simulations for three different cases are done, each with a duration of one hour. The results of the simulations are reported in Tables 6.3, 6.4, and 6.5. Each table contains the traffic demand  $q_{\text{dem}}$ , the case as discussed in Section 6.6, the total time spent calculated using (6.13) for the five simulations, the mean value and standard deviation (as a percentage of the mean value) of the total time spent, and the resulting travel time per vehicle calculated using (6.14). The standard deviation gives a measure of the accuracy of the mean value. For lower percentages, the obtained mean value for the TTS is more accurate.

### Unenforced speed limits

When the speed limits are unenforced, the desired free-flow speed is assumed to be about 10% higher than the speed limit. The desired free-flow speed is modelled as a Gaussian distribution, with a mean value of 110% of the speed limit, and a standard deviation of 8% of the speed limit. Therefore the influence of dynamic speed limits is expected to be high, because the density will be low compared to the other scenarios. For this scenario, shock waves occur and dissolve at a traffic demand of 4700 veh/h when the on-ramp is used. At a traffic demand of 4900 veh/h, the shock wave remains in the network.

From Table 6.3, it can be concluded that controlling the speed limits in the traffic network has a positive effect on the TTS and TPV for the unenforced speed limit scenario. Using the situation without shock waves as a reference, at a demand of 4700 veh/h the TPV is 16.3% higher in the uncontrolled situation, and 12.5% in the controlled situation. Considering the demand of 4900 veh/h, the uncontrolled situation gives an increase of 22.4% in travel time, while the controlled situation gives an increase of 21.0%. Although the improvements are not very large in this single hour, the long-term improvement is significant. Since the traffic jam is dissolved in the controlled situation, the increase in travel time will be the average between the controlled hour and the shock-wave-free travel time, resulting in an increased travel time of 10.5% for the high demand situation.

Typical traffic conditions at the traffic demand of 4700 veh/h without control are shown in Figure 6.10. For all four plots in this figure, the vertical axis shows the segment numbers. The traffic flow direction is from the bottom to the top, and since all segments have the same length, this axis can also be seen as the distance traveled. The upper plot shows that all speed limits are set to 120 km/h at all times. The second plot from the top shows the mean speeds at the segments. The shock wave is shown as the dark area starting at the 20<sup>th</sup> segment after 10 min, moving upstream in time. This shock wave is also shown in the

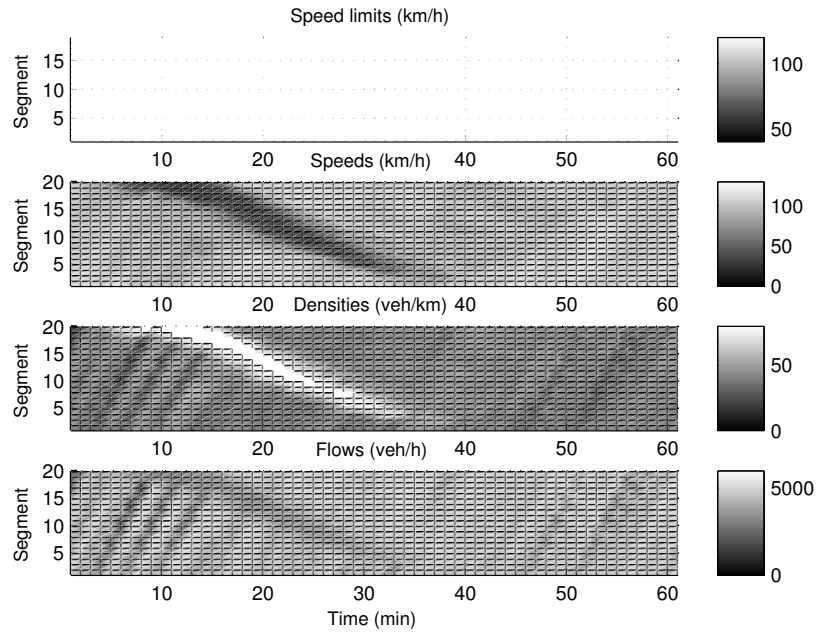


Figure 6.10: Uncontrolled traffic at  $q_{\text{dem}}=4700$  veh/h

third plot from the top, which gives the densities. Here the light colour represents the high density of the traffic jam. Notice that there are also light, small areas which go downstream in time. They are the areas of high density, caused by the difference in desired speed for the vehicles. Fast vehicles will meet slower vehicles as they drive on the highway, causing high density areas, while creating areas of low density in the upstream direction. These stripes of low and high density are also visible in the fourth plot, which shows the flow. The traffic jam is visible as the dark area of low flow.

When we apply model predictive control to the traffic network, using the same situation as before, the resulting traffic conditions in the network are shown in Figure 6.11. At the 10<sup>th</sup> minute, the shock wave enters the part of the highway containing detectors. The controller responds by lowering the speed limits at the segments close to the congestion. After 10 min, the traffic jam is almost completely dissolved, as can be seen by the plots.

### Enforced speed limits

When the speed limits are enforced, we assume that the desired free-flow speed becomes about 10% lower than the displayed speed limit. The desired free-flow speed is modelled as a Gaussian distribution, with a mean value of 90% of the speed limit, and a standard deviation of 8% of the speed limit. Therefore the influence of the dynamic speed limits is expected to be low, because the density will be high compared to the other scenarios.

For this scenario, shock waves occur and dissolve at a traffic demand of 4500 veh/h. At a traffic demand of 4700 veh/h, the shock wave remains in the network. Compared to the



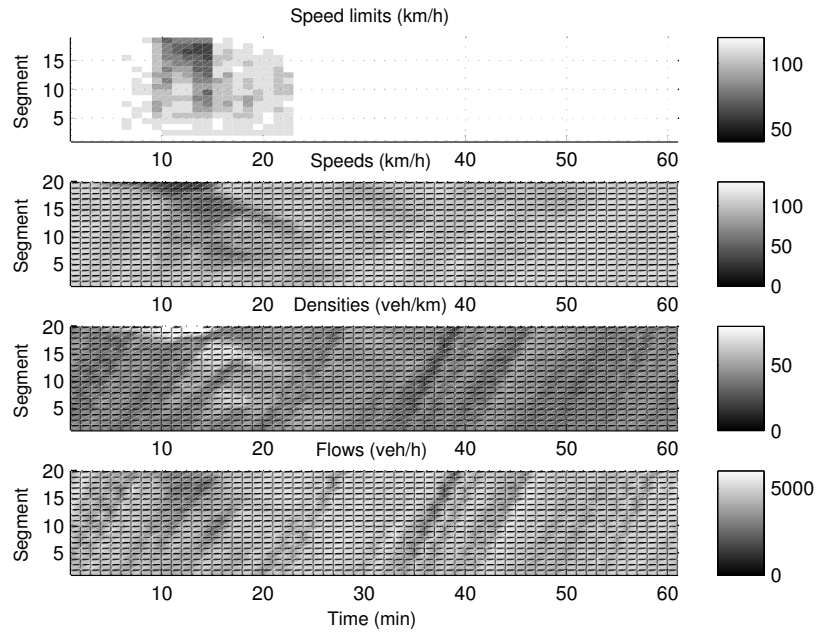


Figure 6.11: Controlled traffic at  $q_{\text{dem}}=4700$  veh/h

Table 6.4: Measured results for the enforced speed limit scenario

$q_{\text{dem}}$	Case	#1	#2	#3	#4	#5	TTS: mean (std)	TPV
4500	no shock wave	589.22	532.23	538.82	624.30	560.45	569.00 (6.7%)	7:35
4500	uncontrolled	580.85	571.20	603.40	627.30	623.12	601.17 (4.1%)	7:55
4500	controlled	615.60	600.32	636.12	575.60		606.91 (4.2%)	8:06
4700	no shock wave	635.52	614.97	583.73	693.12	588.07	623.08 (7.1%)	7:57
4700	uncontrolled	700.33	680.42	659.47	736.15	746.82	704.64 (5.2%)	9:00
4700	controlled	713.20	673.12	686.83	750.40		705.89 (4.8%)	9:01

unenforced speed limit scenario, these demands are lower. This is due to the lower desired free-flow speeds, which result in higher densities at the same demand. Therefore the critical density is reached at a lower demand. From Table 6.4, it can be seen that the TTS is larger in the uncontrolled case, due to the traffic jam. The controller is not able to lower the TTS for both demands, the travel times even slightly increase. At a demand of 4500 veh/h, the traffic jam increases the travel time by 5.7%. Using traffic control, the travel time increases by 6.7%, so there is a negative improvement in travel times by using dynamic speed limit control. Due to the small amount of simulations, no solid conclusion can be drawn from the mean values for the TTS for both cases, but they are expected to stay almost equal. We conclude that the controller will not improve the TTS, but it will improve traffic safety by creating a more homogenous traffic flow. The same conclusion can be drawn when the demand is 4700 veh/h, where the increase in travel time for the uncontrolled situation is

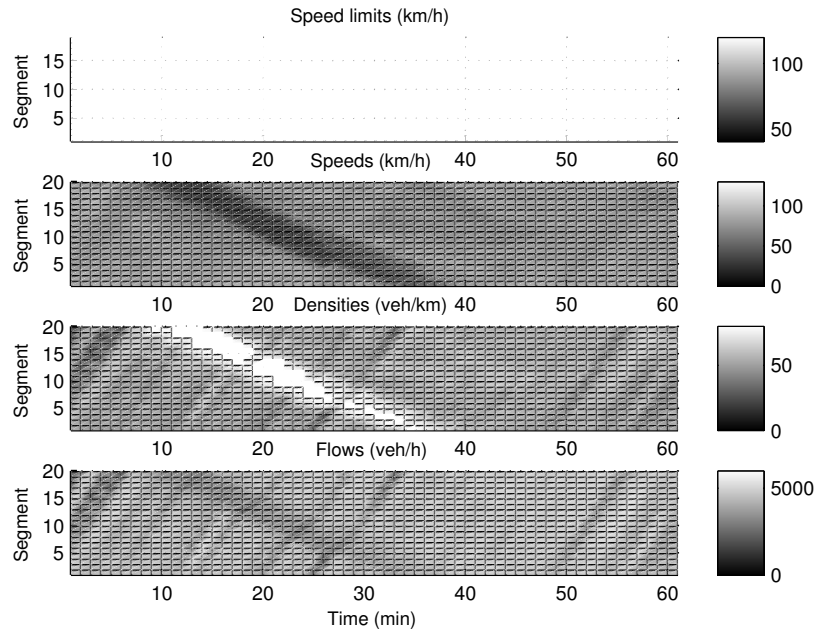


Figure 6.12: Uncontrolled traffic at  $q_{\text{dem}}=4700$  veh/h

13.1%, and 13.3% for the controlled situation.

The homogeneous traffic flow is shown in Figure 6.13. In Figure 6.12 the traffic states are shown, when the speed limits are uncontrolled, and with a high demand of 4700 veh/h that results in a remaining traffic jam. When the same demand is used, but dynamic speed limit control is applied to the network, the traffic jam is dissolved by spreading the vehicles more evenly over the highway, as shown in Figure 6.13.

The effect is a more constant speed over the measured area, as shown in the second plot from above. This increases the safety on the highway. So although the travel times will not decrease by applying control, the controller positively affects the traffic situation by improving the safety.

### Intelligent speed adaptation

For the ISA scenario, the desired free-flow speed is about 100% of the speed limit. The desired free-flow speed is modelled as a Gaussian distribution, with a mean value of 100% of the speed limit, and a standard deviation of 5% of the speed limit. Based on this percentage, the influence of the dynamic speed limits is expected to be good.

When the vehicles are equipped with ISA, shock waves occur and dissolve at a traffic demand of 4700 veh/h. At a traffic demand of 4900 veh/h, the shock wave remains in the network. The measured TTSs and TPVs for both demands are given in Table 6.5.

From Table 6.5, it is concluded that controlling the speed limits on the traffic network has a positive effect on the TTS and TPV for the intelligent speed adaptation scenario. Using the

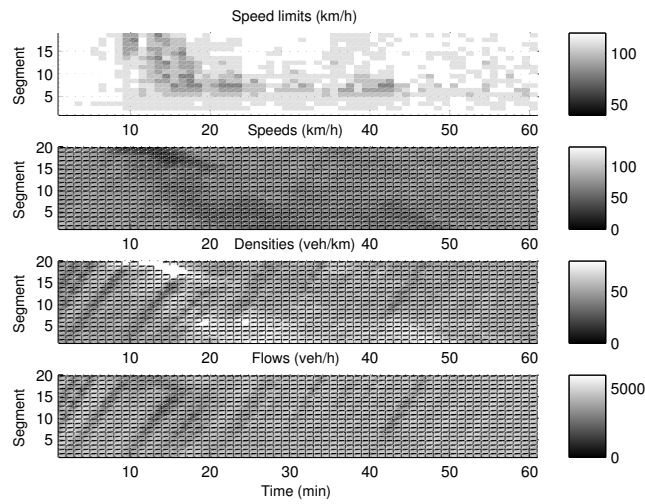


Figure 6.13: Controlled traffic at  $q_{\text{dem}}=4700$  veh/h

Table 6.5: Measured results for the intelligent speed adaptation scenario

$q_{\text{dem}}$	Case	#1	#2	#3	#4	#5	TTS: mean (std)	TPV
4700	no shock wave	432.38	407.60	461.85	481.47	446.35	445.93 (6.3%)	5:42
4700	uncontrolled	510.70	562.90	506.72	496.40	528.25	520.99 (5.0%)	6:39
4700	controlled	485.90	511.38	491.72	479.35		492.09 (2.8%)	6:17
4900	no shock wave	415.42	473.52	423.43	511.95	513.55	467.57 (9.9%)	5:44
4900	uncontrolled	596.58	619.15	540.70	629.22	658.70	608.87 (7.3%)	7:27
4900	controlled	468.77	475.83	535.53	492.45		493.15 (6.1%)	6:02

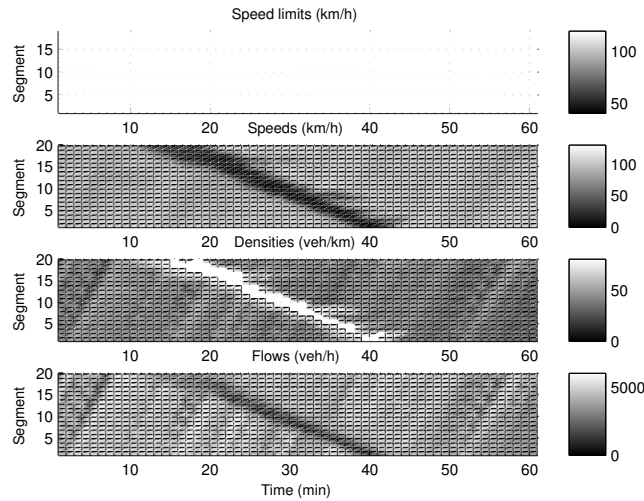


Figure 6.14: Uncontrolled traffic at  $q_{\text{dem}}=4900$  veh/h

case without shock waves as a reference, at a demand of 4700 veh/h the travel time is 16.8% longer in the uncontrolled situation, and 10.4% in the controlled situation. Considering the demand of 4900 veh/h, the uncontrolled situation gives an increase of 30.2% in travel time, while the controlled situation gives an increase of 5.5%.

Typical traffic conditions at the traffic demand of 4900 [veh/h] without traffic control are shown in Figure 6.14. The figure shows the remaining shock wave, which enters the network at the 10<sup>th</sup> minute, and remains in the network with a constant length. Without control, the traffic jam will continue to move upstream in the traffic network.

When control is applied to the traffic network using the same traffic demand, the traffic jam is dissolved, as shown in Figure 6.15. The traffic situation is improved in the part of the network where the measurements are done, and also the upstream part of the network will not suffer from higher travel times due to a traffic jam. Controlling a small part of the network thus has a positive effect for a larger part of the network.

## 6.7 Summary

In this chapter, we have investigated the possibility of using semi-autonomous IVs (especially IVs equipped with ISA systems) to improve the performance of the traffic system. We have discussed how model predictive control (MPC) approach can be used to determine optimal, dynamic speeds limits for the semi-autonomous traffic systems. We have also presented the details of the plugins that allow the simulation software Paramics to simulate the behaviour of semi-autonomous vehicles. This intermediate approach has been illustrated using a case study based on simulations. The improvement on the network performance obtained by deploying ISA systems has been compared with other scenarios, like unenforced and enforced speed limits.

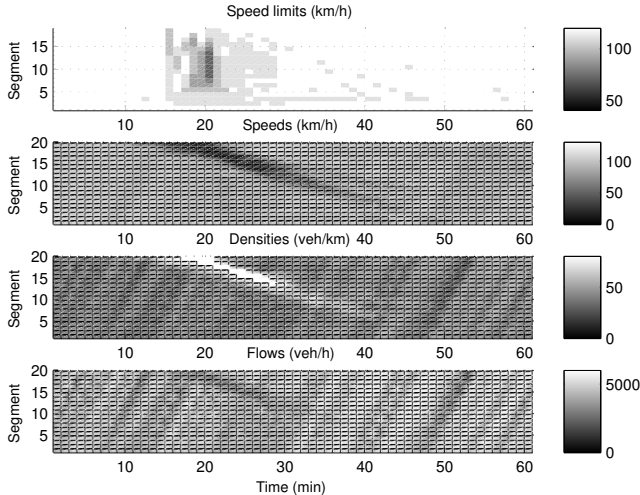


Figure 6.15: Controlled traffic at  $q_{dem}=4900$  veh/h



## Chapter 7

# Conclusions and Recommendations

This chapter mainly summarises our work in this thesis. Section 7.1 highlights the main contributions of this thesis to the state-of-the-art. In Section 7.2, the main conclusions drawn at the end of each chapter are summarised. Finally, in Section 7.3, open problems and recommendations for future research are discussed.

### 7.1 Main contributions

The approach we have considered in this thesis for advanced traffic system is based on the IVHS traffic system. An IVHS consist of interacting roadside infrastructures and vehicles equipped with advanced in-vehicle technologies (IVs). The fully automated IVs are organised in platoons with short intraplatoon distances, and larger distances between platoons. The proposed approach aims at enabling and sustaining an efficient operation of the IVHS system through the used of advanced control methods and for this purpose, it utilises technological advancements from both the automotive and the traffic management fields. Our main contributions to the state-of-the-art in IVHS-based traffic control and management approaches are:

- The proposal of a new hierarchical, distributed framework to control large-scale IVHS-based traffic networks,
- The development of control algorithms that can be deployed at different (roadside and area) levels of IVHS-based traffic management systems.

The first contribution of this thesis was the proposal of a new framework to control and manage large-scale IVHS-based traffic systems in a well-structured and distributed manner. In the proposed framework, the traffic control and coordination tasks are distributed among several levels in the hierarchy and also among various controllers at the same level.

Secondly, the focus was on the development of control methodologies that can be implemented within the context of our framework. Most of the existing IVHS frameworks primarily focus on vehicle level controlling problems. In particular, they mainly deal with

the control issues that result due to the actual implementation of platooning approaches and IV technologies in vehicles. However, the control problems at tactical and navigational levels did not yet receive much attention and thus motivated us to work on these lines of research. More specifically, in this thesis, we have designed control methodologies for the roadside and area controllers to improve the performance of large-scale IVHS-based traffic systems.

In particular, we have developed an MPC-based control algorithm that the roadside controller can use to combine ramp metering (or other conventional roadside-based traffic control measures) with intelligent speed adaptation (ISA), lane allocation measures, and other (future) IV-based control measures, and to determine system-optimum control signals for these control measures. In our framework, the navigational control tasks (route guidance) are handled by the area controller. Thus we have also developed and implemented an optimal control algorithm for the area controller to optimally route the platoons on the set of interconnected highways. To solve the routing problem, our control algorithm either used mixed-integer linear or real-valued nonlinear programming techniques.

For the other higher levels in the control hierarchy, similar control approaches, also based on MPC but with even more aggregate models can be used.

## 7.2 Conclusions

### 7.2.1 Main conclusions

The main conclusions of this thesis are:

- A multi-level framework is necessary for the traffic management and control purposes. This multi-level structure should range from a vehicle controller at the bottom level to the traffic management controllers at roadside, area, and regional at higher levels.
- MPC approach is a good method for traffic management and control purposes in both roadside and area controllers.
- The potential improvement in the traffic flow obtained from the multi-level framework has been illustrated using case studies. By the combination of IV-based platoons and multi-level framework, performance improvement of 10–30 % is possible. The accuracy of the performance improvement depends on the considered traffic demand, scenario, and set-up, and on more experiments.

### 7.2.2 Conclusions per chapter

In this section, we summarise the conclusions drawn from each chapter in this thesis.

In Chapter 2, we have presented a detailed review of some of the most frequently used control methodologies. These methods are used for controlling a human driven highway traffic system. We have also illustrated the working procedure of these control techniques using a common application such as ramp metering. Having discussed the state-of-the-art of currently applied control methods, we then focused on control architectures for IVHS. This included the discussion of various existing architectures such as PATH, Dolphin, Auto21



CDS, CVIS, SafeSpot, and PReVENT. Based on our literature survey, we have made a comparative analysis of the control methods and the control architectures to assess their similarities and their differences. The conclusions from this analysis clearly indicated the advantages offered by using a hierarchical structuring and also showed the importance of and need for a flexible and traffic-adaptive control design methodology. The other topic discussed in this chapter is IVs and their functionalities, which on the long-term include platooning. The platooning approach, which allows vehicles to maintain small intra-platoon distances and to travel with high speeds, along with the necessary driver assistance systems such as cooperative ACC (for maintaining safe distance), ISA, and dynamic route guidance systems will offer considerable benefits to traffic flow improvement. However, the roadside infrastructure is currently unaware of these potential benefits. This leads us to question why roadside traffic control measures are not (yet) adapted to include this intelligence.

Based on the knowledge gained from our survey and analysis, we have introduced a new framework for IVHS that integrates the intelligence in roadside infrastructure and in platoons of automated IVs. A detailed description of our new framework is given in Chapter 3. Our framework also allows vehicle-vehicle and vehicle-roadside communications similar to the other existing IVHS architectures. However, our framework differs mainly in terms of combining the existing conventional traffic control measures and the IV technologies, and viewing them as IV-based control measures.

Chapter 4 mainly dealt with the problem formulation and the control-related issues of roadside controllers. From our preliminary study on control methods, we have decided to implement an MPC approach to perform the IV-based roadside traffic control operations. More specifically, the MPC algorithm discussed in this chapter determines optimal ISA speed limits and lane allocations as well as optimal release times for the platoons at the on-ramps. Since MPC is an on-line, model-based approach with a receding horizon mechanism, we have presented some traffic models that could be used for simulation and prediction at the roadside level, and have discussed the control issues related to optimisation methods, and to the choice of controller settings such as the prediction and control horizons and the number of control signals. The simulations based on the MPC approach reported in Chapter 4, compare the performance of the IVHS traffic system to the performance of a human-driven traffic system. For the defined case study, the platoon-based approach results in a better performance compared to the situation with controlled human drivers. However, our approach poses some challenges on computational complexity. Determining optimal control signals that consist of both real and integer valued variables such as ISA speeds and lane allocation for each platoon, consumes more time and requires more system memory to perform the computations.

In Chapter 5, we have focused on area controllers, which control a set of interconnected highways and coordinate the activities of various roadside controllers in their area. In particular, this controller assigned optimal routes to the platoons in the IVHS. As mentioned earlier, determining control measures for each platoon proves to be computationally demanding, and hence motivated us to propose two simplified but fast simulation models to describe the flows of platoons in the network. For the first model, which only considers flows and queues lengths, we have shown that the optimal route choice control problem can be approximated by a linear or a mixed-integer linear problem. With a simple case study we have illustrated that this results in a balanced trade-off between optimality and computational efficiency. The second model is a macroscopic model based on an existing

traffic flow model for human drivers (METANET), which we have adapted to fit the platoon framework. For this model, the optimal route choice control problem results in an optimisation problem with only real-valued optimisation variables. This approach was also illustrated with a simple case study involving a comparison between human drivers and an IVHS with platoons of IVs. As expected, the deployment of IVs in the system resulted in a better performance than the human driven traffic system.

Chapter 6 mainly showed the possibility of equipping vehicles with advanced in-car systems in an existing simulation software and also investigated the improvement that could be attained by implementing ISA at the vehicle level. We have used the traffic simulation software PARAMICS to represent the current traffic system and we have developed plugins for the partially automated vehicles to capture their reaction to the control measures. We have implemented the roadside controller in Matlab and since there was no existing interface between the two software programs, we have also created a plug-in to simulate the behaviour of IV-based vehicles. We have investigated the effect of control on the traffic system driven by human drivers and on the system driven with partially automated vehicles using a simple case study. The system with partially automated vehicles offered better performance improvement than the current conventional traffic system.

### 7.3 Open problems and recommendations for future research

Every research work has a room for improvement. In this section, we discuss some proposals for future research in the field of traffic management and control with intelligent vehicles in various directions:

- Model considerations,
- Controller considerations,
- Complexity and implementation issues,
- Alternative approaches,
- Transition aspects.

#### Model considerations

- Since MPC is a model-based approach, we are in need of fast and simple models for prediction purposes. In this thesis except in Chapter 7, to filter out the effects of model mismatch errors, we have deployed the same simulation model for prediction purposes. The effect of such errors should be investigated by considering set-ups in which the simulation and the prediction model are different.
- Especially for the roadside controller described in Chapter 4, the currently used prediction model is still too time-consuming. Hence, we should also develop faster and simpler models for the traffic flows on individual freeway stretches, while still guaranteeing a sufficiently high degree of accuracy. However, e.g., for complex junctions,

we might still require a detailed representation of the traffic system. Therefore, we should develop so called multi-resolution prediction models with different degrees of accuracy in certain regions of interest.

- Each layer in our IVHS framework uses a different level of details of the traffic system, and also deals with different sets of inputs and control outputs. In particular, the roadside controller can use Paramics, or the big-car model, and the area controller can work with flow-based models. Therefore, we should develop interfaces to bring consistency among the inputs and outputs obtained from different layers of the framework.
- We should design and develop new, fast models with required accuracy for all levels.

### **Controller considerations**

- In this thesis, the roadside controller has performed the coordination and integration of different control measures. In Chapter 4, we have in particular combined ISA and optimal lane allocations (IV-based control measure) along with the ramp metering (roadside control measure). However, we can extend this combination to include other traffic IV-based control measures that might be developed by both the transportation and vehicle industry in the near future. Also, the computational complexity of the control problem can be analysed using analytical methods or empirical methods.
- Currently, the roadside controller had no control over the platoon sizes and the platoon size was assumed to be a constant value in all our case studies. If this control measure is also taken into account, then the roadside controller can have more control over the traffic scenario. Platoons can then vary from a single car to a very large number.
- Depending on the nature of the traffic problem, the controller can use continuous optimisation (for ISA and ramp metering), integer optimisation (for optimal lane allocations) or a combination of both techniques to find an optimal solution. Finding an optimal solution for an integer-valued problem poses more computational problems than real-valued problems. Since the determination of optimal platoon sizes also falls under the integer optimisation problems, the roadside controller should be able to consider these issues and to deal with the computational complexities (see next subsection).
- If we extend the scope of our approach to urban or mixed urban-freeway networks, then the controller should combine control measures from both traffic systems and should be able to handle the computational issues discussed above.
- In our framework, we allowed interactions only among different levels and not at the same level. For instance, the roadside controllers do not interact among themselves to coordinate some of their activities. For our future research, we should compare and investigate the potential benefits that could be achieved using decentralised and distributed coordination at the same level.
- Higher-level controllers such as regional and supraregional controllers will provide route guidance instructions to a collection of area controllers. This would then require

us to design control methods based on aggregated, simple flow models that would represent the traffic flows on different areas under consideration.

- Different objective functions or a combination of objective functions should be considered to analyse the performance of the traffic system. Some examples of possible cost functions are energy saving, fuel emissions, throughput, safety, etc. To solve the resulting multi-objective problems, we could use multi-objective optimisation methods such as evolutionary algorithms, weighted-sum single aggregate objective functions, goal attainment etc.

### **Complexity and implementation issues**

- The computational complexity of the proposed approach is still an issue and needs to be analysed, in particular for the roadside controller. Hence, we should investigate ways to increase the efficiency of the current approach or develop alternative control methods. Some suggestions for the current MPC approach can then be to use blocking techniques (same solution applied for a certain number of consecutive time steps), to start the optimisation with a good initial solution obtained through approximations, to use a multi-level approach to deal with the mixed integer optimisation problems, to reduce the control horizons, to increase the number of vehicles in a platoon, etc. The complexity of the control approach (i.e., execution time and memory requirements) mainly depends on the number of control variables and on the type of control variables (integer, real-valued). We should analyse this complexity either analytical or empirical ways.
- Extensive evaluation and assessment of the framework through a wide range of case studies and scenarios at different levels and with the entire framework should be performed. These set-ups could also use a detailed IV-based vehicle models for traffic simulations.
- Another topic will be the real-time application issues. Due to the optimisation over both real and integer variables, the controller implemented in Chapter 4 is unable to function in real time. Since multi-start optimisation has been used, an improvement in this direction could be attained by implementing the controller in such a way that it can take the advantage of the upcoming trends in multi-processor systems and distributed optimisation.

### **Alternative approaches**

- In this research, our framework was analysed mainly using the MPC approach. Later, we should also analyse the performance of the framework using other control approaches. For instance, we could use alternative control approaches such as fuzzy logic, artificial neural networks, gain scheduling, or LPV methods.
- On the other hand, one could consider a swarm-like autonomous driving system – a completely different approach to our hierarchical framework. More specifically, this approach does not allow/consider roadside-based controllers. In this approach, the vehicles try to aggregate themselves in groups, make their decisions based only

on local information. For this approach, one will have to develop control strategies for the movement of individual vehicles inside a swarm by means of cooperation, coordination or negotiation with their neighbours.

### **Transition aspects**

- In this research, vehicles in the traffic system were assumed to be completely automated and homogeneous (only IVs). An interesting research topic is to allow a traffic system with heterogeneous intelligent vehicles (cars, trucks, etc.) and also with different penetration rates of IVs, i.e., situations of mixed traffic with both human drivers and platoons of automated IVs, if tractable at all.
- In an urban-freeway traffic system, the roadside controller should be able to handle the transition situation from human-driven vehicles in urban networks to autonomously driven platoons on freeways, and vice versa. As a suggestion to support this transition, we could allow the roadside controller to use traffic signals as a control measure so that the vehicles do not have to wait too long at the on-ramp, by which one can prevent the on-ramp queue from spilling back to the urban network.
- Legal, social, economic and other issues involved in the transition of the conventional traffic system to the proposed IVHS system also need to be addressed. These issues can be stated in terms of social acceptance of the system, investments required for implementations, etc.



# Bibliography

- [1] A. Aamodt and E. Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1):39–59, March 1994.
- [2] AHSRA web site. <http://www.ahsra.or.jp/>. Last visited on November 4, 2008.
- [3] K. J. Åström and B. Wittenmark. *Computer-Controlled Systems — Theory and Applications*. Prentice-Hall, Upper Saddle River, New Jersey, 3rd edition, 1997.
- [4] K. J. Åstrom, P. Albertos, M. Blanke, A. Isidori, W. Schaufelberger, and R. Sanz, editors. *Control of Complex Systems*. Springer-Verlag, New York, 2000.
- [5] A. Atamtürk and M. W. P. Savelsbergh. Integer-programming software systems. *Annals of Operations Research*, 140(1):67–124, November 2005.
- [6] C. Audet and J. E. Dennis Jr. Analysis of generalized pattern searches. *SIAM Journal on Optimization*, 13(3):889–903, 2007.
- [7] Auto21 CDS web site. <http://www.damas.ift.ulaval.ca/projets/auto21/en/index.html>. Last visited on December 2, 2008.
- [8] S. Bana. Coordinating automated vehicles via communication. Technical Report UCB-ITS-PRR-2001-20, PATH Research Report, Berkely, California, 2001.
- [9] L. D. Baskar, B. De Schutter, and H. Hellendoorn. Decentralized traffic control and management with intelligent vehicles. In *Proceedings of the 9th TRAIL Congress 2006 — TRAIL in Motion — CD-ROM*, Rotterdam, The Netherlands, November 2006.
- [10] L. D. Baskar, B. De Schutter, and H. Hellendoorn. Hierarchical traffic control and management with intelligent vehicles. In *Proceedings of the 2007 IEEE Intelligent Vehicles Symposium (IV'07)*, pages 834–839, Istanbul, Turkey, June 2007.
- [11] L. D. Baskar, B. De Schutter, and H. Hellendoorn. Model predictive control for intelligent speed adaptation in intelligent vehicle highway systems. In *Proceedings of the 17th IEEE International Conference on Control Applications*, pages 468–473, San Antonio, Texas, September 2008.

- [12] L. D. Baskar, B. De Schutter, and H. Hellendoorn. Intelligent speed adaptation in intelligent vehicle highway systems — A model predictive control approach. In *Proceedings of the 10th TRAIL Congress 2008 — TRAIL in Perspective — CD-ROM*, Rotterdam, The Netherlands, October 2008.
- [13] L. D. Baskar, B. De Schutter, and J. Hellendoorn. Model-based predictive traffic control for intelligent vehicles: Dynamic speed limits and dynamic lane allocation. In *Proceedings of the 2008 IEEE Intelligent Vehicles Symposium (IV'08)*, pages 174–179, Eindhoven, The Netherlands, June 2008.
- [14] L. D. Baskar, B. De Schutter, and J. Hellendoorn. Dynamic speed limits and on-ramp metering for IVHS using model predictive control. In *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems (ITSC 2008)*, pages 821–826, Beijing, China, October 2008.
- [15] L. D. Baskar, B. De Schutter, J. Hellendoorn, and A. Tarău. Traffic management for intelligent vehicle highway systems using model-based predictive control. In *Proceedings of the 88th Annual Meeting of the Transportation Research Board*, Washington, DC, January 2009. Paper 09-2107.
- [16] L.D. Baskar, B. De Schutter, Z. Papp, and J. Hellendoorn. Traffic control and intelligent vehicle highway systems: A survey. Technical Report 09-043, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, August 2009. Accepted for publication in *IET Intelligent Transport Systems*.
- [17] T. Bellemans. *Traffic Control on Motorways*. PhD thesis, Faculty of Applied Sciences, K.U.Leuven, Leuven, Belgium, May 2003.
- [18] T. Bellemans, B. De Schutter, and B. De Moor. Model predictive control for ramp metering of motorway traffic: A case study. *Control Engineering Practice*, 14(7): 757–767, July 2006.
- [19] R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1957.
- [20] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.
- [21] R. Bishop. A survey of intelligent vehicle applications worldwide. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 25–30, Dearborn, Michigan, 2000.
- [22] R. Bishop. *Intelligent Vehicles Technology and Trends*. Artech House, 2005.
- [23] R. Bishop. <http://www.ivsource.net/>, 2005. Last visited on June 2, 2009.
- [24] J. Blum and A. Eskandarian. Managing effectiveness and acceptability in intelligent speed adaptation systems. In *International Conference on Intelligent Transportation Systems*, pages 319–324, Toronto, Canada, September 2006.
- [25] K. Bogenberger, H. Keller, and A. D. May. A neuro-fuzzy approach for ramp metering. In *Proceedings of the 10th International Conference on Road Transport Information and Control*, pages 101–105, London, England, 2000.



- [26] A. Bose and P. Ioannou. Mixed manual/semi-automated traffic: a macroscopic analysis. *Transportation Research Part C*, 11(6):439–462, 2003.
- [27] M. Brackstone and M. McDonald. Car-following: A historical review. *Transportation Research Part F*, 2(4):181–196, 1999.
- [28] M. Broucke and P. Varaiya. The automated highway system: A transportation technology for the 21st century. *Control Engineering Practice*, 5(11):1583–1590, November 1997.
- [29] M. Burger. *Advanced Traffic Management with Integrated Model Predictive Control*. Msc thesis, Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, September 2007.
- [30] E. F. Camacho and C. Bordons. *Model Predictive Control in the Process Industry*. Springer-Verlag, Berlin, Germany, 1995.
- [31] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems Magazine*, 22(1):44–52, February 2002.
- [32] O. Carsten and F. Tate. Intelligent speed adaptation: The best collision avoidance system? In *Proceedings of the 17th International Technological Conference on the Enhanced Safety of Vehicles*, pages 1–10, Amsterdam, The Netherlands, 2001.
- [33] C. G. Cassandras, D. L. Pepyne, and Y. Wardi. Optimal control of systems with time-driven and event-driven dynamics. In *Proceedings of the 37th IEEE Conference on Decision and Control*, pages 7–12, Tampa, Florida, December 1998.
- [34] R. Chandler, R. Herman, and E. Montroll. Traffic dynamics studies in car following. *Operations Research*, 6(2):165–184, June 1958.
- [35] H. Chen and F. Allgöwer. Quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability. *Automatica*, 34(10):1205–1217, January 1998.
- [36] S. H. Chen, A. J. Jakeman, and J. P. Norton. Artificial intelligence techniques: An introduction to their use for modelling environmental systems. *Mathematics and Computers in Simulation*, 78(2):379–400, July 2008.
- [37] S. Cheon. An overview of automated highway systems and the social institutional challenges they face. Technical Report UCTC No. 624, University of California, Berkeley, California, 2002.
- [38] D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized predictive control – Part I. The basic algorithm. *Automatica*, 23(2):137–148, March 1987.
- [39] L. S. Comte. New systems: new behaviour? *Transportation Research Part F*, 3(2): 95–111, May 2000.
- [40] C. Cordier, H. Marchand, R. Laundry, and L. A. Wolsey. *bc-opt*: A branch-and-cut code for mixed integer programs. *Mathematical Programming, Series A*, 86(2):335–353, 1999.

- [41] J. Cuenca, J. Hernández, and M. Molina. Knowledge-based models for adaptive traffic management systems. *Transportation Research Part C*, 3(5):311–337, October 1995.
- [42] CVIS web site. <http://www.cvisproject.org/>. Last visited on November 4, 2008.
- [43] C. F. Daganzo. *Fundamentals of Transportation and Traffic Operations*. Pergamon Press, 1997.
- [44] C. F. Daganzo. A continuum theory of traffic dynamics for freeways with special lanes. *Transportation Research Part B*, 31(2):83–102, 1997.
- [45] S. Darbha and K. R. Rajagopal. Intelligent cruise control systems and traffic flow stability. *Transportation Research Part C*, 7(6):329–352, December 1999.
- [46] L. Davis. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.
- [47] L. C. Davis. Effect of adaptive cruise control systems on traffic flow. *Physical Review E*, 69:1–8, 2004.
- [48] B. De Schutter, S. P. Hoogendoorn, H. Schuurman, and S. Stramigioli. A multi-agent case-based traffic control scenario evaluation system. In *Proceedings of the IEEE 6th International Conference on Intelligent Transportation Systems (ITSC'03)*, pages 678–683, Shanghai, China, October 2003.
- [49] R. W. Eglese. Simulated annealing: A tool for operations research. *European Journal of Operational Research*, 46(3):271–281, June 1990.
- [50] R. E. Fenton. IVHS/AHS: Driving into the future. *IEEE Control Systems Magazine*, 14(6):13–20, December 1994.
- [51] R. Fletcher and S. Leyffer. Numerical experience with lower bounds for MIQP branch-and-bound. *SIAM Journal on Optimization*, 8(2):604–616, May 1998.
- [52] A. V. Fursikov. *Optimal Control of Distributed Systems: Theory and Applications*. American Mathematical Society, Boston, Massachusetts, 2000.
- [53] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, San Francisco, California, 1979.
- [54] V. Gazi and K. M. Passino. Stability of a one-dimensional discrete-time asynchronous swarm. *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, 35(4):834–841, August 2005.
- [55] D. Gazis, R. Herman, and R. Rothery. Nonlinear follow the leader models of traffic flow. *Operations Research*, 9(4):545–567, June 1961.
- [56] M. Gelfand and S. V. Fomin. *Calculus of Variations*. Dover Publications, New York, 1991.

- [57] D. N. Godbole, J. Lygeros, E. Singh, A. Deshpande, and A. E. Lindsey. Design and verification of communication protocols for degraded modes of operation of AHS. In *Proceedings of the 34th IEEE Conference on Decision and Control*, pages 427–432, New Orleans, Louisiana, 1995.
- [58] F. L. Hall and K. Agyemang-Duah. Freeway capacity drop and the definition of capacity. *Transportation Research Record*, (1320):91–98, 1991.
- [59] R. Hall and C. Chin. Vehicle sorting for platoon formation: Impacts on highway entry and throughput. *Transportation Research Part C*, 13(5):405–420, 2005.
- [60] S. Hallé, B. Chaib-draa, and J. Laumonier. Car platoons simulated as a multiagent system. In *Proceedings of the 4th Workshop on Agent-Based Simulation*, pages 57–63, Montpellier, France, 2003.
- [61] D. Hammerstrom. Working with neural networks. *IEEE Spectrum*, 30(7):46–53, July 1993.
- [62] F. Hayes-Roth. Rule-based systems. *Communications of the ACM*, 28(9):921–932, 1985.
- [63] J. K. Hedrick, M. Tomizuka, and P. Varaiya. Control issues in automated highway systems. *IEEE Control Systems Magazine*, 14(6):21–32, December 1994.
- [64] A. Hegyi. *Model Predictive Control for Integrating Traffic Control Measures*. PhD thesis, Delft University of Technology, Delft, The Netherlands, February 2004. TRAIL Thesis Series T2004/2.
- [65] A. Hegyi, B. De Schutter, S. P. Hoogendoorn, R. Babuška, and H. van Zuylen. Fuzzy decision support system for traffic control centers. In *Proceedings of the European Symposium on Intelligent Techniques (ESIT 2000)*, pages 389–395, Aachen, Germany, September 2000. Paper BC-01-2.
- [66] A. Hegyi, B. De Schutter, and J. Hellendoorn. Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):102–112, March 2005.
- [67] A. Hegyi, B. De Schutter, and J. Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C*, 13(3):185–209, June 2005.
- [68] D. Helbing. Gas-kinetic derivation of navier-stokes-like traffic equations. *Physical Review E*, 53(3):2266–2381, 1996.
- [69] M. R. Hestenes. *Calculus of Variations and Optimal Control Theory*. John Wiley and Sons, New York, 1966.
- [70] M. Hjalmdahl, S. Almqvist, and A. Várhelyi. Speed regulation by in-car active accelerator pedal - effects on speed and speed distribution. *IATSS Research*, 26(2):60–66, May 2002.

- [71] R. Holve, P. Protzel, and K. Naab. Generating fuzzy rules for the acceleration control of an adaptive cruise control system. In *Proceedings of the 1996 Biennial Conference of the North American Fuzzy Information Processing Society*, pages 451–455, Berkeley, California, 1996.
- [72] S. P. Hoogendoorn and P. H. L. Bovy. Multiclass macroscopic traffic flow modelling: A multilane generalisation using gas kinetic theory. In *Proceedings of the 14th International Symposium of Transportation and Traffic Theory*, pages 27–50, Adelaide, Australia, 1999. Elsevier Science Ltd.
- [73] S. P. Hoogendoorn, B. De Schutter, and H. Schuurman. Decision support in dynamic traffic management. Real-time scenario evaluation. *European Journal of Transport and Infrastructure Research*, 3(1):21–38, 2003.
- [74] R. Horowitz and P. Varaiya. Control design of an automated highway system. *Proceedings of the IEEE: Special Issue on Hybrid Systems*, 88(7):913–925, July 2000.
- [75] A. Hsu, F. Eskafi, S. Sachs, and P. Varaiya. Design of platoon maneuver protocols for IVHS. Technical Report 96-21, University of California, Berkeley, California, 1991.
- [76] A. Hsu, F. Eskafi, S. Sachs, and P. Variaya. Protocol design for an automated highway system. *Discrete Event Dynamic Systems: Theory and Applications*, 2(1):183–206, 1993.
- [77] P. Ioannou and C. C. Chien. Autonomous intelligent cruise control. *IEEE Transactions on Vehicular Technology*, 42(4):657–672, November 1993.
- [78] N. R. Jennings. A roadmap of agent research and development. *Autonomous Agents and Multi-Agent Systems*, 1(1):7–38, July 1998.
- [79] D. R. Jones. DIRECT global optimization algorithm. In *Encyclopedia of Optimization*, volume 1, no. 1, pages 431–440. Kluwer Academic Publishers, Boston, Massachusetts, 2001.
- [80] R. K. Jurgen. Smart cars and highways go global. *IEEE Spectrum*, 28(5):26–36, May 1991.
- [81] P. Kachroo and K. Özbay. *Feedback Control Theory for Dynamic Traffic Assignment*. Advances in Industrial Control. Springer-Verlag, Berlin, 1999.
- [82] A. Karimi and A. Gupta. Incident management system for Santa Monica Smart Corridor. In *Proceedings of the 63rd Annual Meeting of the Institute of Transportation Engineers*, pages 180–185, The Hague, The Netherlands, 1993.
- [83] A. Kesting, M. Treiber, M. Schönhof, F. Kranke, and D. Helbing. Jam-avoiding adaptive cruise control ACC and its impact on traffic dynamics. In *Traffic and Granular Flow*, pages 633–643. Berlin, Germany: Springer-Verlag, 2005.
- [84] S. Kim, M. E. Lewis, and C. C. White III. Optimal vehicle routing with real-time traffic information. *IEEE Transactions on Intelligent Transportation Systems*, 6(2): 178–188, June 2005.

- [85] D. E. Kirk. *Optimal Control Theory: An Introduction*. Prentice-Hall, Englewood Cliffs, New Jersey, 1970.
- [86] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice-Hall, Upper Saddle River, New Jersey, 1995.
- [87] E. Kometani and T. Sasaki. Dynamic behaviour of traffic with a nonlinear spacing speed relationship. In *Proceedings of the Symposium for Theory Traffic Flow*, pages 105–109, Research Laboratories, General Motors, New York, 1959.
- [88] A. Kotsialos and M. Papageorgiou. Freeway ramp metering: An overview. In *Proceedings of the 3rd Annual IEEE Conference on Intelligent Transportation Systems (ITSC 2000)*, pages 228–239, Dearborn, Michigan, October 2000.
- [89] A. Kotsialos and M. Papageorgiou. Efficiency and equity properties of freeway network-wide ramp metering with AMOC. *Transportation Research Part C*, 12(6): 401–420, 2004.
- [90] A. Kotsialos, M. Papageorgiou, and A. Messmer. Optimal coordinated and integrated motorway network traffic control. In *Proceedings of the 14th International Symposium of Transportation and Traffic Theory (ISTTT)*, pages 621–644, Jerusalem, Israel, July 1999.
- [91] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET. *IEEE Transactions on Intelligent Transportation Systems*, 3(4): 282–292, December 2002.
- [92] A. Kotsialos, M. Papageorgiou, M. Mangeas, and H. Haj-Salem. Coordinated and integrated control of motorway networks via non-linear optimal control. *Transportation Research Part C*, 10(1):65–84, 2002.
- [93] A. Kotsialos, M. Papageorgiou, M. Mangeas, and H. Haj-Salem. Coordinated and integrated control of motorway networks via non-linear optimal control. *Transportation Research Part C*, 10(1):65–84, February 2002.
- [94] B. Krause and C. von Altrock. A complete fuzzy logic control approach for existing traffic control systems. In *Mobility for Everyone, Proceedings of the 4th World Congress on Intelligent Transportation Systems*, Berlin, Germany, October 1997. Paper no. 2045.
- [95] J. A. Laval and C. F. Daganzo. Lane-changing in traffic streams. *Transportation Research Part B*, 40(3):251–264, 2006.
- [96] N. E. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *Proceedings of 40th IEEE Conference on Decision and Control*, pages 2968–2973, Orlando, Florida, 2001.
- [97] K. Li and P. Ioannou. Modeling of traffic flow of automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 5(2):99–113, June 2004.

- [98] M. J. Lighthill and G. B. Whitham. On kinematic waves: a theory of traffic flow on long crowded roads. In *Proceedings Royal Society*, pages 317–345, London, England, August 1955.
- [99] J. Linderoth and T. Ralphs. Noncommercial software for mixed-integer linear programming. *Optimization Online*, January 2005.
- [100] Y. Liu, K. M. Passino, and M. M. Polycarpou. Stability analysis of one-dimensional asynchronous swarms. *IEEE Transactions on Automatic Control*, 48(10):1848–1854, October 2003.
- [101] J. M. Maciejowski. *Predictive Control with Constraints*. Prentice-Hall, Harlow, England, 2002.
- [102] A. D. May. *Traffic Flow Fundamentals*. Prentice-Hall, Englewood Cliffs, New Jersey, 1990.
- [103] M. McDonald, N. B. Hounsell, and S. R. Njoze. Strategies for route guidance systems taking account of driver response. In *Proceedings of the 6th Vehicle Navigation and Information Systems*, pages 328 – 333, Seattle, Washington, July 1995.
- [104] A. Messmer and M. Papageorgiou. METANET: A macroscopic simulation program for motorway networks. *Traffic Engineering and Control*, 31(9):466–470, 1990.
- [105] R. M. Michaels. Perceptual factors in car following. In *Proceedings of the 2nd International Symposium for Theory Road Traffic Flow*, pages 44–59, Paris, France, 1963.
- [106] F. Middelham. Fuzzy logic demo. Technical report, Rijkswaterstaat, AVV, Rotterdam, The Netherlands, 1993. In Dutch.
- [107] K. M. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, Massachusetts, 1999.
- [108] M. Molina, J. Hernández, and J. Cuenca. A structure of problem-solving methods for real-time decision support in traffic control. *International Journal of Human-Computer Studies*, 49(4):577–600, October 1998.
- [109] K. Nagel. Particle hopping models and traffic flow theory. *Physical Review E*, 53(5):4655–4672, 1996.
- [110] H. T. Nguyen and E. A. Walker. *A First Course in Fuzzy Logic*. Chapman & Hall Press, Boca Raton, Florida, 2nd edition, 1999.
- [111] M. Nookala and B. Estochen. Minnesota, USA intelligent vehicle initiative. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 533–536, Versailles, France, 2002.
- [112] S. A. Nsour, S. L. Cohen, J. E. Clark, and A. J. Santiago. Investigation of the impacts of ramp metering on traffic flow with and without diversion. *Transportation Research Record*, (1365):116–124, 1992.

- [113] U. Palmquist. Intelligent cruise control and roadside information. *IEEE Micro*, 13(1):20–28, 1993.
- [114] M. Papageorgiou. *Applications of Automatic Control Concepts to Traffic Flow Modeling and Control*. Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin, Germany, 1983.
- [115] M. Papageorgiou, J. M. Blosseville, and H. Hadj-Salem. Modelling and real-time control of traffic flow on the southern part of Boulevard Périphérique in Paris: Part I: Modelling. *Transportation Research Part A*, 24(5):345–359, September 1990.
- [116] M. Papageorgiou, J. M. Blosseville, and H. Haj-Salem. Modelling and real-time control of traffic flow on the southern part of Boulevard Périphérique in Paris: Part II: Coordinated on-ramp metering. *Transportation Research Part A*, 24(5):361–370, September 1990.
- [117] M. Papageorgiou, H. Hadj-Salem, and F. Middelham. ALINEA local ramp metering: Summary of field results. In *Proceedings of the 76th Annual Meeting of the Transportation Research Board*, Washington, DC, 1997. Paper no. 970032.
- [118] P. M. Pardalos and M. G. C. Resende. *Handbook of Applied Optimization*. Oxford University Press, Oxford, England, 2002.
- [119] L. E. Parker. Designing control laws for cooperative agent teams. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 582–587, Atlanta, Georgia, 1993.
- [120] M. Persson, F. Botling, E. Hesslow, and R. Johansson. Stop and go controller for adaptive cruise control. In *Proceedings of the IEEE International Conference on Control Applications*, pages 1692–1697, Kohala Coast, Hawaii, 1999.
- [121] PReVENT web site. <http://www.prevent-ip.org/>. Last visited on November 4, 2008.
- [122] I. Prigogine and R. Herman. *Kinetic Theory of Vehicular Traffic*. American Elsevier, New York, 1971.
- [123] S. J. Qin and T. A. Badgwell. An overview of industrial model predictive control technology. *American Institute of Chemical Engineers*, 93(316):232–256, June 1997.
- [124] B. S. Y. Rao and P. Varaiya. Roadside intelligence for flow control in an IVHS. *Transportation Research Part C*, 2(1):49–72, 1994.
- [125] M. J. Richardson and D. Smith. Design of the driver interface for autonomous intelligent cruise control. In *Proceedings of the IEE Colloquium on Design of the Driver Interface*, pages 7/1–7/4, London, England, 1995.
- [126] S. G. Ritchie. A knowledge-based decision support architecture for advanced traffic management. *Transportation Research Part A*, 24(1):27–37, January 1990.
- [127] S. G. Ritchie and N. A. Prosser. Real-time expert system approach to freeway incident management. *Transportation Research Record*, (1320):7–16, 1991.

- [128] D. Roseman and S. S. Tvedten. Santa Monica Freeway Smart Corridor project operational multi-agency traffic management and expert system. In *Mobility for Everyone, Proceedings of the 4th World Congress on Intelligent Transport Systems*, Berlin, Germany, October 1997. Paper no. 1032.
- [129] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, New Jersey, 2003.
- [130] SafeSpot web site. <http://www.safespot-eu.org/>. Last visited on November 4, 2008.
- [131] H. Shimizu, M. Kobayashi, and Y. Yonezawa. A route guidance system of a traffic network. In *Proceedings of the 34th SICE Annual Conference*, pages 1549–1552, Hokkaido, Japan, July 1995.
- [132] S. Shladover, C. A. Desoer, J. K. Hedrick, M. Tomizuka, J. Walrand, W. B. Zhang, D. H. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown. Automatic vehicle control developments in the PATH program. *IEEE Transactions on Vehicle Technology*, 40(1):114–130, February 1991.
- [133] S. E. Shladover. PATH at 20 – History and major milestones. *IEEE Transactions on Intelligent Transportation Systems*, 8(4):584–592, December 2007.
- [134] D. D. Šiljak. *Decentralized Control of Complex Systems*. Academic Press, San Diego, California, 1991.
- [135] Y. J. Stephanedes, E. Kwon, and K. Chang. Control emulation method for evaluating and improving traffic-responsive ramp metering strategies. *Transportation Research Record*, (1360):42–45, 1992.
- [136] J. M. Sussman. Intelligent vehicle highway systems: Challenge for the future. *IEEE Micro*, 1(14-18):101–104, June 1993.
- [137] H. J. Sussmann and J. C. Willems. 300 years of optimal control: From the brachistochrone to the maximum principle. *IEEE Control Systems Magazine*, 17(3):32–44, June 1997.
- [138] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, Massachusetts, 1998.
- [139] D. Swaroop and K. R. Rajagopal. Intelligent cruise control systems and traffic flow stability. *Transportation Research Part C*, 7(6):329–352, 1999.
- [140] H. Taale and F. Middelham. Ten years of ramp-metering in The Netherlands. In *Proceedings of the 10th International Conference on Road Transport Information and Control*, pages 106–110, London, England, April 2000.
- [141] R. Thomas. Less is more. *IEE Review*, 49(5):40–43, May 2003.
- [142] Tomlab web site. <http://www.tomlab.biz/>. Last visited on January 8, 2008.



- [143] G. Toulminet, J. Boussuge, and C. Laugeau. Comparative synthesis of the 3 main european projects dealing with cooperative systems (CVIS, SAFESPOT and COOPERS) and description of COOPERS demonstration site 4. In *Proceedings of the 11th IEEE Conference on Intelligent Transportation Systems*, pages 809–814, Beijing, China, 2008.
- [144] S. Tsugawa, S. Kato, K. Tokuda, T. Matsui, and H. Fujii. An architecture for cooperative driving of automated vehicles. In *Proceedings of the IEEE Intelligent Transportation Symposium*, pages 422–427, Dearborn, Michigan, 2000.
- [145] S. Tsugawa, S. Kato, K. Tokuda, T. Matsui, and H. Fujii. A cooperative driving system with automated vehicles and inter-vehicle communications in demo 2000. In *Proceedings of the 2001 IEEE Conference on Intelligent Transportation Systems*, pages 918–923, Oakland, California, August 2001.
- [146] B. van Arem. Cooperative vehicle-infrastructure systems: An intelligent way forward. Technical Report 2007-D-R0158/B, Verkeer en Vervoer, TNO, Delft, The Netherlands, 2007. In Dutch.
- [147] B. van Arem, C. J. G. van Driel, and R. Visser. The impact of cooperative adaptive cruise control on traffic-flow characteristics. *IEEE Transactions on Intelligent Transportation Systems*, 7(4):429–436, 2006.
- [148] M. van den Berg, B. De Schutter, H. Hellendoorn, and A. Hegyi. Control of day-to-day route choice in traffic networks with overlapping routes. In *Proceedings of the 12th IFAC Symposium on Control in Transportation Systems*, pages 556–561, Redondo Beach, California, September 2009.
- [149] P. Varaiya. Smart cars on smart roads: Problems of control. *IEEE Transactions on Automatic Control*, 38(2):195–207, February 1993.
- [150] P. Varaiya and S. E. Shladover. Sketch of an IVHS systems architecture. In *Vehicle Navigation and Information Systems*, pages 909–922, Dearborn, Michigan, October 1991.
- [151] D.J. Verburg, A.C.M. van der Knaap, and J. Ploeg. VEHIL: Developing and testing intelligent vehicles. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 537–544, Versailles, France, 2002.
- [152] S. Vukanovic and O. Ernhofner. Evaluation and field implementation of the fuzzy logic based ramp metering algorithm accezz. In *Proceedings of the 9th IEEE Conference on Intelligent Transportation Systems*, pages 437–441, Toronto, Canada, 2006.
- [153] D. Watling and T. van Vuren. The modelling of dynamic route guidance systems. *Transportation Research Part C*, 1(2):159–182, June 1993.
- [154] G. Weiss, editor. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. MIT Press, Cambridge, Massachusetts, 1999.

- 
- [155] J. Weymann, J. L. Fargas, and J. J. Henry. Optimization of traffic dynamic route guidance with drivers' reactions in a queue-based model. *IEEE Transactions on Systems, Man and Cybernetics*, 25(7):1161–1165, July 1995.
  - [156] Q. Yang and H. N. Koutsopoulos. A microscopic traffic simulator for evaluation of dynamic traffic management systems. *Transportation Research Part C*, 4(3):113–129, 1996.
  - [157] X. Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.
  - [158] J. Yi and R. Horowitz. Macroscopic traffic flow propagation stability for adaptive cruise controlled vehicles. *Transportation Research Part C*, 14(2):81–95, April 2006.
  - [159] H. Zhang and S. G. Ritchie. Real-time decision-support system for freeway management and control. *Journal of Computing in Civil Engineering*, 8(1):35–51, 1994.

# Glossary

## Conventions

The following conventions are used in this thesis for notation and symbols:

- A lower case character typeset in boldface, e.g.,  $\mathbf{x}$ , represents a column vector.
- The number of elements in a vector  $\mathbf{x}$  is indicated by  $n_{\mathbf{x}}$ .

## List of symbols and notations

Below follows a list of the most frequently used symbols and notations in this thesis. Symbols particular to area controller and roadside controller are explained only in the relevant chapters.

### Simulation model

$u$	input variable
$x$	state variable
$\mathbf{f}$	function
$T_{\text{sim}}$	simulation time step
$N_s$	total number of simulation steps

### Vehicle model

$i$	vehicle $i$
$i + 1$	predecessor vehicle $i + 1$
$x_i$	position of vehicle $i$
$v_i$	velocity of vehicle $i$
$a_i$	acceleration of vehicle $i$
$a_{\text{acc,max}}$	maximum acceleration
$a_{\text{dec,max}}$	maximum deceleration
$S_0$	minimum safe distance headway
$h_{\text{ref},i}$	reference distance headway for vehicle $i$
$T_{\text{head},i}$	desired time headway for vehicle $i$

$L_i$  length of vehicle  $i$

### MPC controller

$N_p$  length of prediction horizon  
 $N_c$  length of control horizon  
 $T_{ctrl}$  controller time step  
 $t$  continuous time instant  
 $k$  discrete time instant (corresponding to the time instant  $t = kT_{ctrl}$ )  
 $\mathbf{x}(k)$  vector with current state of the system  
 $\mathbf{u}(k)$  vector with control sequence for the system  
 $M$  an integer, equals  $\frac{T_{ctrl}}{T_{sim}}$   
 $J_{perf}$  performance index  
 $n_{veh}(k)$  number of vehicles that are present within network at time  $t = kT_{sim}$   
 $q_{main}(k)$  number of vehicles in queue at the mainstream origin at time  $t = kT_{sim}$   
 $q_{on}(k)$  number of vehicles present in on-ramp queues at time  $t = kT_{sim}$

### Flow model

$O$  set of origin nodes  
 $I$  set of internal nodes  
 $D$  set of destination nodes  
 $\mathcal{V}$  set of all nodes  
 $\mathcal{L}$  set of all links  
 $O \times D$  origin-destination pair  
 $D_{o,d}$  demand of vehicles at origin  $o$  with destination  $d$   
 $v$  internal nodes  
 $L_v^{in}$  incoming links at  $v$   
 $L_v^{out}$  set of outgoing links from  $v$   
 $x_{l,o,d}$  flow of vehicles from origin  $o$  to destination  $d$  that enter link  $l$

### METANET model

$v$  mean speed  
 $\rho$  traffic density  
 $q$  traffic flow  
 $\rho_{crit}$  critical density  
 $v_{free}$  free-flow speed  
 $q_{max}$  maximum flow  
 $m$  link  
 $i$  segment  
 $o$  origin

---

$j$	destination
$w_{o,j}$	queue length
$d_o(k)$	total demand
$Q_{\text{cap},o}$	capacity (veh/h) of origin $o$ under free-flow conditions
$\beta_{n,m,j}(k)$	splitting rate in node $n$

## List of abbreviations

The following abbreviations are used in this thesis:

IVHS	Intelligent Vehicle Highway Systems
ITS	Intelligent Transportation Systems
AHS	Automated Highway Systems
IVs	Intelligent Vehicles
ISA	Intelligent Speed Adaptation
ACC	Adaptive Cruise Control
MPC	Model Predictive Control
MILP	Mixed-Integer Linear Programming
TTS	Total Time Spent



# Samenvatting

Files leveren problemen op die de meesten van ons dagelijks ervaren. Naast diverse systemen voor verkeersmanagement biedt een efficiënt gebruik van de bestaande wegkantinfrastructuur gecombineerd met technologie in de auto een nieuw perspectief om de fileproblematiek en daarmee verbonden problemen op te lossen. Deze aanpak heeft geleid tot het ontstaan van “Intelligent Vehicle Highway Systems” (IVHSs). Een IVHS bestaat in wezen uit een wegkantinfrastructuur die communiceert met automatische intelligente voertuigen (IVs), die georganiseerd zijn in groepen van dicht op elkaar rijdende voertuigen die “platoons” genoemd worden. Door “platooning” kunnen meer voertuigen gebruik maken van de snelweg, waardoor de verkeersdoorstroming verbetert.

In de huidige situatie maken de meeste verkeersmanagement- en regelcentra gebruik van bestaande wegkantgebaseerde maatregelen om de verkeerssituatie te verbeteren. Het doel van dit proefschrift is het ontwikkelen van een raamwerk voor en een systematische aanpak van geïntegreerde verkeersregelingsmethoden voor IVHS.

De focus van dit proefschrift ligt op het combineren van de regeltechnische mogelijkheden die automatische “platoons” en wegkantssystemen bieden. We streven ernaar om op verschillende niveaus verkeersmanagement- en regeltechnische methoden te ontwikkelen, waarbij we de intelligentie van de voertuigen combineren. Langs deze weg gebruiken de wegkantregelaars zowel IV-gebaseerde als bestaande verkeersmaatregelen om “platoons” te formeren en aan te sturen. Hierdoor wordt de verkeersintensiteit verhoogd.

Ten eerste geven we een overzicht van IV-gebaseerde verkeersmaatregelen die “platooning” ondersteunen en die door de wegkantssystemen kunnen worden gebruikt als verkeersmaatregelen. Daarna geven we een overzicht van de stand van zaken van diverse verkeersmanagement- en regeltechnische raamwerken voor de verdeling van intelligentie tussen de wegkantinfrastructuur en automatische voertuigen. Op basis van dit overzicht stellen we een nieuwe, hiërarchische verkeersmanagement- en regeltechnische architectuur voor IVHS voor, die de sterke punten van elk van de bestaande IV-gebaseerde verkeersmanagementsraamwerken bevat en die bestaat uit verschillende niveaus, van voertuig- en platoonregelaars, via wegkant- en gebiedsregelaars tot regionale en bovenregionale regelaars. In dit proefschrift richten we ons vooral op de wegkant- en gebiedsregelaars.

De regelstrategie die wij voorstellen is gebaseerd op “model-based predictive control” (MPC) en maakt gebruik van voorspellingsmodellen gecombineerd met optimalisatiemethodes om voor een gegeven prestatie criterium de beste verkeerssignalering over een gegeven tijdshorizon te bepalen. De daaruit volgende reeks van regelingen wordt toegepast, gebruikmakend van schuivende tijdshorizon.

De wegkantregelaars zijn in staat om aparte wegvakken op de snelweg te controleren.

Wij laten zien hoe wegkantregelaars MPC kunnen gebruiken om op een geïntegreerde wijze optimale snelheden voor de leiders van “platoons” te bepalen, een optimale toewijzing van rijbanen voor “platoons” te vinden en optimale toegangstijdstippen voor “platoons” bij opritten te bepalen. We voeren ook het zogenaamde “big car” model in als een methode om de numerieke complexiteit van het MPC-optimalisatieprobleem aan te pakken. De mogelijkheden van de voorgestelde wegkantverkeersregelmethode wordt met enkele eenvoudige case studies geïllustreerd.

Voor het regelen en organiseren van een netwerk van snelwegen maakt het voorgestelde raamwerk gebruik van gebiedsregelaars. De gebiedsregelaar is hoofdzakelijk verantwoordelijk voor het bieden van route-instructies voor “platoons”; daarnaast coördineert een gebiedsregelaar de activiteiten van de diverse wegkantregelaars in zijn gebied. Het bieden van route-instructies in een netwerk blijkt echter een ingewikkelde zaak te zijn, die bovendien rekentechnisch complex is. Om het probleem handelbaar te maken stellen we twee aanpakken voor, beide gebaseerd op MPC. In de eerste aanpak regelen en dirigeren we de verkeersstromen in het netwerk in plaats van elk “platoon” apart te behandelen. We tonen aan dat deze aanpak uitmondt in een lineair programmeringsprobleem met reële en integer variabelen, waarvoor efficiënte oplossingsmethoden bestaan. De tweede aanpak die wij voorstellen is gebaseerd op een macroscopisch METANET-model dat we aanpassen om het geschikt te maken voor “platoons”. Hierbij maken we gebruik van MPC om de optimale splitsingsverhoudingen in de knooppunten te bepalen. Beide aanpakken illustreren we met een eenvoudig voorbeeld.

Ten slotte bespreken we hoe bestaande softwarepakketten voor verkeerssimulatie zoals PARAMICS kunnen worden uitgebreid met IV-gebaseerde regelmethode. Dit zou een tussenstap kunnen vormen tussen bestaande verkeersmethoden en volledig geautomatiseerde IVHS. We bespreken hoe deze IV-technologieën met enkele uitbreidingen betere verkeersprestaties zouden kunnen leveren dan bestaande regelmethode.

Het proefschrift sluit af met een samenvatting van de belangrijkste resultaten en een overzicht van toekomstige onderzoeksrichtingen.

L.D. Baskar



# Summary

Traffic congestion is a problem experienced daily by most of us. Among various traffic management schemes, efficient utilisation of the existing roadside infrastructure combined with in-vehicle technologies offers a promising solution to address traffic congestion and related problems. This approach has resulted in the development of Intelligent Vehicle Highway Systems (IVHSs). An IVHS basically consists of roadside infrastructures interacting with automated intelligent vehicles (IVs) that are organised in a closely spaced groups called platoons. With platooning, more vehicles can be accommodated on the highway, thus increasing the traffic flow.

In the present situation, most of the existing traffic management and control centers use conventional roadside-based control measures to improve the traffic performance. The objective of the thesis is to provide a framework and a systematic approach for integrating traffic control and management methods into the IVHS.

The focus of the thesis is on combining the control capabilities offered by automated platoons with those of the roadside infrastructure. We aim at developing traffic management and control methods to be implemented at various control levels, by incorporating intelligence from and within vehicles. Thus the roadside controllers use both IV-based and conventional traffic control measures for controlling and managing platoons, such that the performance of the traffic is improved.

First, we present a survey on IV-based traffic control measures that could support platooning and that also could also be used as a control measure by the roadside controllers. Then, we give a review of current state of the art of various traffic management and control frameworks that distribute intelligence between roadside infrastructure and automated vehicles. On the basis of this survey, we propose a new hierarchical traffic management and control architecture for IVHS that incorporates the strong points of each of the existing IV-based traffic management frameworks, and that consists of several levels ranging from vehicle and platoons controllers, over roadside and area controllers, to regional and supra-regional controller. In this thesis we focus in particular on the roadside and area controllers.

The control approach we propose is based on model-based predictive control (MPC) and makes use of prediction models in combination with optimisation to determine the control signals that optimise a given performance criterion over a given time horizon. The resulting control sequence is then applied using a moving horizon approach.

The roadside controllers are capable of managing single stretches of highway. We show how MPC can be used by the roadside controllers to determine in an integrated way optimal speeds for the platoon leaders, optimal lane allocations for the platoons, and optimal on-ramp release times for the platoons. We also introduce the so-called “big” car model as

a way to address the computational complexity of the MPC optimisation problem. The potential of the proposed roadside traffic control method is illustrated with some simple case studies.

To control and manage a network of highways, the proposed framework uses area controllers. The area controller is mainly responsible to provide for providing route guidance instructions for platoons and also to coordinate the activities of various roadside controllers in their area. However, providing route guidance for each platoon in the network is a cumbersome task and also computationally intensive. To make the problem tractable, we propose two approaches, both based on MPC. In the first approach we control and route the traffic flows in the network rather than controlling each platoon individually. We show that this results in a mixed integer linear programming problem, for which efficient solvers exist. The second approach we propose is based on the macroscopic METANET model that we adapt to suit the platoon framework. Here we use MPC to determine the optimal splitting rates at the nodes. Both approaches are illustrated using a simple set-up.

Finally we also discuss how existing traffic simulation software such as PARAMICS can be extended with IV-based traffic control measures. This serves as an intermediate step between the current traffic control practice and fully automated IVHS. We discuss how these IV technologies with some extensions could improve traffic performance when compared to the standard control methods.

The thesis concludes with a summary of the main results and an outlook for future research directions.

L.D. Baskar

# Curriculum vitae

L.D. Baskar was born on October 15, 1981 in Erode, TamilNadu, India. In 2003, she graduated from Bharathiyar University in India with Computer Science Engineering as a specialisation. She received her Master of Science in Computer Science in 2005, with a specialisation in Simulation and Modelling from Otto-von-Guericke Universität, Magdeburg, Germany.

Since 2005, L.D. Baskar has been working on her PhD at the Delft Center for Systems and Control of the Delft University of Technology, The Netherlands. The PhD project has been on traffic management and control in Intelligent Vehicle Highway Systems and has been supervised by Prof.dr.ir. J. Hellendoorn and Prof.dr.ir. B. De Schutter. Her research interests are on applied sciences and thus include traffic and transportation systems, simulation and modelling of systems, and optimal control of systems. During her PhD research, she has been a member of The Netherlands Research School for Transport, Infrastructure, and Logistics (TRAIL).



# TRAIL Thesis Series

A series of The Netherlands TRAIL Research School for theses on transport, infrastructure and logistics.

Baskar, L.D., *Traffic Control and Management in Intelligent Vehicle Highway Systems*, T2009/12, November 2009, TRAIL Thesis Series, the Netherlands

Konings, J.W., *Intermodal Barge Transport: Network Design, Nodes and Competitiveness*, T2009/11, November 2009, TRAIL Thesis Series, the Netherlands

Kusumaningtyas, I., *Mind Your Step: Exploring aspects in the application of long accelerating moving walkways*, T2009/10, October 2009, TRAIL Thesis Series, the Netherlands

Gong, Y., *Stochastic Modelling and Analysis of Warehouse Operations*, T2009/9, September 2009, TRAIL Thesis Series, the Netherlands

Eddia, S., *Transport Policy Implementation and Outcomes: the Case of Yaounde in the 1990s*, T2009/8, September 2009, TRAIL Thesis Series, the Netherlands

Platz, T.E., *The Efficient Integration of Inland Shipping into Continental Intermodal Transport Chains. Measures and decisive factors*, T2009/7, August 2009, TRAIL Thesis Series, the Netherlands

Tahmasseby, S., *Reliability in Urban Public Transport Network Assessment and Design*, T2009/6, June 2009, TRAIL Thesis Series, the Netherlands

Bogers, E.A.I., *Traffic Information and Learning in Day-to-day Route Choice*, T2009/5, June 2009, TRAIL Thesis Series, the Netherlands

Amelsfort, D.H. van, *Behavioural Responses and Network Effects of Time-varying Road Pricing*, T2009/4, May 2009, TRAIL Thesis Series, the Netherlands

Li, H., *Reliability-based Dynamic Network Design with Stochastic Networks*, T2009/3, May 2009, TRAIL Thesis Series, the Netherlands

Stankova, K., *On Stackelberg and Inverse Stackelberg Games and their Applications in the Optimal Toll Design Problem, the Energy Markets Liberalization Problem, and in the Theory of Incentives*, T2009/2, February 2009, TRAIL Thesis Series, the Netherlands

Li, T., *Informedness and Customer-Centric Revenue*, T2009/1, January 2009, TRAIL Thesis Series, the Netherlands

Agusdinata, D.B., *Exploratory Modeling and Analysis. A promising method to deal with*

- deep uncertainty*, T2008/17, December 2008, TRAIL Thesis Series, the Netherlands
- Kreutzberger, E., *The Innovation of Intermodal Rail Freight Bundling Networks in Europe. Concepts, Developments, Performances*, T2008/16, December 2008, TRAIL Thesis Series, the Netherlands
- Taale, H., *Integrated Anticipatory Control of Road Networks. A game theoretical approach*, T2008/15, December 2008, TRAIL Thesis Series, the Netherlands
- Li, M., *Robustness Analysis for Road Networks. A framework with combined DTA models*, T2008/14, December 2008, TRAIL Thesis Series, the Netherlands
- Yu, M., *Enhancing Warehouse Performance by Efficient Order Picking*, T2008/13, October 2008, TRAIL Thesis Series, the Netherlands
- Liu, H., *Travel Time Prediction for Urban Networks*, T2008/12, October 2008, TRAIL Thesis Series, the Netherlands
- Kaa, E.J. van de, *Extended Prospect Theory. Findings on Choice Behaviour from Economics and the Behavioural Sciences and their Relevance for Travel Behaviour*, T2008/11, October 2008, TRAIL Thesis Series, the Netherlands
- Nijland, H., *Theory and Practice of the Assessment and Valuation of Noise from Roads and Railroads in Europe*, T2008/10, September 2008, TRAIL Thesis Series, the Netherlands
- Annema, J.A., *The Practice of Forward-Looking Transport Policy Assessment Studies*, T2008/9, September 2008, TRAIL Thesis Series, the Netherlands
- Ossen, S.J.L., *Theory and Empirics of Longitudinal Driving Behavior*, T2008/8, September 2008, TRAIL Thesis Series, the Netherlands
- Tu, H., *Monitoring Travel Time Reliability on Freeways*, T2008/7, April 2008, TRAIL Thesis Series, the Netherlands
- ÁAriano, A., *Improving Real-Time Train Dispatching: Models, Algorithms and Applications*, T2008/6, April 2008, TRAIL Thesis Series, the Netherlands
- Quak, H.J., *Sustainability of Urban Freight Transport. Retail Distribution and Local Regulations in Cities*, T2008/5, March 2008, TRAIL Thesis Series, the Netherlands
- Hegeman, G., *Assisted Overtaking. An assessment of overtaking on two-lane rural roads*, T2008/4, February 2008, TRAIL Thesis Series, the Netherlands
- Katwijk, R.T. van, *Multi-Agent Look-ahead Traffic Adaptive Control*, T2008/3, January 2008, TRAIL Thesis Series, the Netherlands
- Argiolu, R., *Office Location Choice Behaviour and Intelligent Transport Systems*, T2008/2, January 2008, TRAIL Thesis Series, the Netherlands
- Houtenbos, M., *Expecting the Unexpected. A study of interactive driving behaviour at intersections*, T2008/1, January 2008, TRAIL Thesis Series, the Netherlands