Efficient Predictive Model-Based and Fuzzy Control for Green Urban Mobility

Anahita Jamshidnejad

Efficient Predictive Model-Based and Fuzzy Control for Green Urban Mobility

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof.ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op 22 juni 2017 om 10.00 uur
door **Anahita JAMSHIDNEJAD**,
Master of Science in Mechanical Engineering, University of Tehran,
geboren te Teheran, Iran.

This dissertation has been approved by the promotors:

Prof.dr.ir. H. Hellendoorn
Prof.dr. M. Papageorgiou

Composition of the doctoral committee:

Rector Magnificus	chairperson
Prof.dr.ir. H. Hellendoorn	Technische Universiteit Delft, promotor
Prof.dr. M. Papageorgiou	Technical University of Crete, promotor

Independent members:

Prof.dr. Y. Xi	Shanghai Jiao Tong University
Prof.dr. S. Sacone	University of Genova
Prof.dr. M. Cao	Rijksuniversiteit Groningen
Prof.dr.ir. H. van Lint	Technische Universiteit Delft
Prof.dr.ir. C. Witteveen	Technische Universiteit Delft

This dissertation has been completed in partial fulfillment of the requirements of the Dutch Institute of Systems and Control (DISC) for graduate studies. Research described in this thesis was supported by the NWO-NSFC project “Multi-level predictive traffic control for large-scale urban networks” (629.001.011), which is partly financed by the Netherlands Organization for Scientific Research (NWO), and by the European COST Action TU1102.

TRAIL Thesis Series T2017/6, 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: Anahita Jamshidnejad
E-mail: ana.jamshidnejad@gmail.com

ISBN 978-90-5584-224-7

Printed in the Netherlands

To my precious parents,

to Nina and Ashkan,

&

to my dear Bart

for their endless support and love

Contents

Part II. Efficient MPC: Sustainable urban traffic control 93

Chapter 1

Introduction

Efficient modeling, estimation, and control approaches for various dynamical systems are the main concerns of control engineers. It is important to investigate and develop models that can provide reliable estimates of the system's dynamics and that require a low computation time, and also control approaches that can efficiently steer the dynamical systems towards a desired or an optimal performance. In this regards, for obtaining the main aim of this thesis which is to efficiently control urban traffic networks for green mobility, we have investigated the problem from different points-of-view, i.e., the main focus of this thesis is on developing efficient estimation, modeling, and control approaches for traffic networks. The thesis includes three parts, where The main aim of Part I is to develop algorithms and formulas that increase the accuracy of the traffic computations and estimations. Part II focuses on efficient control and optimization methods for achieving green mobility in urban areas. The aim of Part III is to introduce a control framework that is efficient, adaptive, and that has the potential to be applied to large-scale systems such as traffic networks.

This chapter motivates the need for designing efficient estimation and control approaches for traffic networks, in particular for achieving green urban mobility. Moreover, it motivates the choice of using model-predictive control (MPC) and fuzzy logic control approaches to achieve green urban mobility. Next, the main contributions of the thesis are given, and the chapter closes by an overview and a road map of the thesis.

1.1 Motivation

Due to the various negative effects of highly congested traffic (especially in urban areas) on different aspects of life (e.g., economics, health and well-being of the urban population, environment, etc.), this issue should be tackled efficiently. Spending time in very long queues on streets is a waste of time and energy. Moreover, it increases the level of the fuel consumption and the amount of harmful substances emitted into the air. Hence, the solutions that are considered for traffic problems should take into account both the time that is spent by the urban population in traffic, and the environmental effects (i.e., the level of emissions, fuel consumption, and noise) of traffic.

In order to tackle the issues that are caused by congested traffic in urban areas, several solutions have been proposed. These solutions include increasing the capacity of the available roads by constructing additional roads, decreasing the traffic demand (e.g., via road pricing or by restricting regulations for entering certain traffic areas during the day or at particular times of the day), and managing the existing capacity of the roads via efficient control approaches. The first two approaches may often be of less interest, since they require high budgets and also create restrictions for the urban population in their daily trips. This highlights the role of efficient control approaches in managing traffic in urban areas. In this thesis, we therefore focus on the third strategy, i.e., on developing efficient control approaches for urban traffic networks.

MPC is an optimization-based control approach that has proven to be efficient for handling problems with input and state constraints, with nonlinear dynamics and constraints, and with multi-objective cost functions. For the urban traffic problem discussed above, we face a multi-objective problem that aims to reduce both the total travel time and the total emissions caused by urban traffic, where the dynamics of urban traffic networks can be highly nonlinear. In this thesis, the control variable is the green time of the traffic lights (assuming that we have fixed cycle times for the traffic lights). The green time is constrained by a lower bound (due to safety reasons) and an upper bound (due to the fixed cycle time). Moreover, the summation of the green, yellow, and red light for each direction of movement should not exceed the cycle time of the traffic light, which itself introduces an additional constraint on the control variables. In summary, we will have a multi-objective nonlinear minimization problem that is subject to state and input constraints. Hence, considering the mentioned characteristics for MPC, this control method seems to be a promising approach for solving our problem.

Fuzzy logic control uses fuzzy logic to design a control system that translates human-defined linguistic rules into automatic strategies. Moreover, it is simple and flexible, and requires a low computation time. Fuzzy logic control can handle problems with incomplete or imprecise information, and the resulting control system can easily be updated and maintained over time. All these characteristics, and in particular the fact that traffic regulations are by nature rule-based (i.e., they have originally been controlled via human-defined rules), motivate the use of a rule-based control approach, such as fuzzy logic control for traffic networks.

1.2 Main focus and contributions

The main focus of this thesis is on developing efficient estimation and control approaches for traffic networks. Two control approaches are considered for that aim: MPC and fuzzy logic control. We discuss different methods (including smooth and nonsmooth) to efficiently solve the nonsmooth optimization problem of the MPC controller. Moreover, in the last part of the thesis, we combine the idea of fuzzy logic control with a predictive and model-based control approach to develop a coordinative multi-agent control architecture that can be used to control large-scale urban traffic networks.

The main contributions of the three parts of the thesis are listed below:

- In Part I of the thesis, we address an issue that is missing in the existing literature, i.e., we propose efficient formulas and algorithms to estimate the temporal-spatial fundamental traffic variables (including the time-space mean speed (TSMS)). The focus is on developing accurate formulas that are based on a microscopic point-of-view, and on using data from point measurements. We show via extensive simulations based on real-life datasets and on MATLAB that the proposed approaches for estimating the TSMS are much more accurate in approximating the average traffic speed compared with the methods available in literature, which consider the space mean speed as an approximation for the average traffic speed. Part I of the thesis is based on [63–65].

- In Part II of the thesis, we develop efficient optimization-based control approaches for achieving green urban mobility. To that aim, we first develop a general framework for integrating traffic flow and emission models. The resulting model is in nature mesoscopic, which makes it fast for real-time computations. We show via simulations that this model produces very accurate estimations for the emission levels. Next, we develop general smoothening methods that can be used to make the nonsmooth formulations of a cost function or the constraints of an optimization problem smooth. Using the smoothening approaches and the mesoscopic flow and emission model, we develop an MPC controller that uses efficient gradient-based approaches to solve the MPC optimization problem online. Finally, we develop formulas for endpoint penalties that can be used in the formulation of the cost function to approximate a finite-horizon MPC problem with an infinite-horizon one with a low computational burden. Part II of the thesis is based on [67–70, 72, 73].

- In Part III of the thesis, we propose an approach for integrating fast intelligent control methods (e.g., fuzzy logic control or artificial neural networks) and optimization-based and model-based control approaches to obtain a fast real-time control system with a highly satisfactory performance. We propose a general treatment of type-2 fuzzy sets and fuzzy membership functions by introducing two forms of type-2 fuzzy membership functions called probabilistic-fuzzy and fuzzy-fuzzy. We then introduce a two-layer adaptive type-2 fuzzy control scheme that is capable of tuning its parameters online. We use a predictive optimization-based controller next to the fuzzy controller to provide estimates of the future states for the controller and also to make it capable of coordinating with neighboring fuzzy controllers in a multi-agent control architecture. Part III of the thesis is based on [66, 71].

1.3 Overview of the thesis

An overview of the thesis is shown in Figure 1.1. This thesis has been written in three main parts, where each part starts with a background chapter. Part I of the thesis includes Chapters 2-4. The main focus of this part is on the temporal-spatial definitions for the fundamental

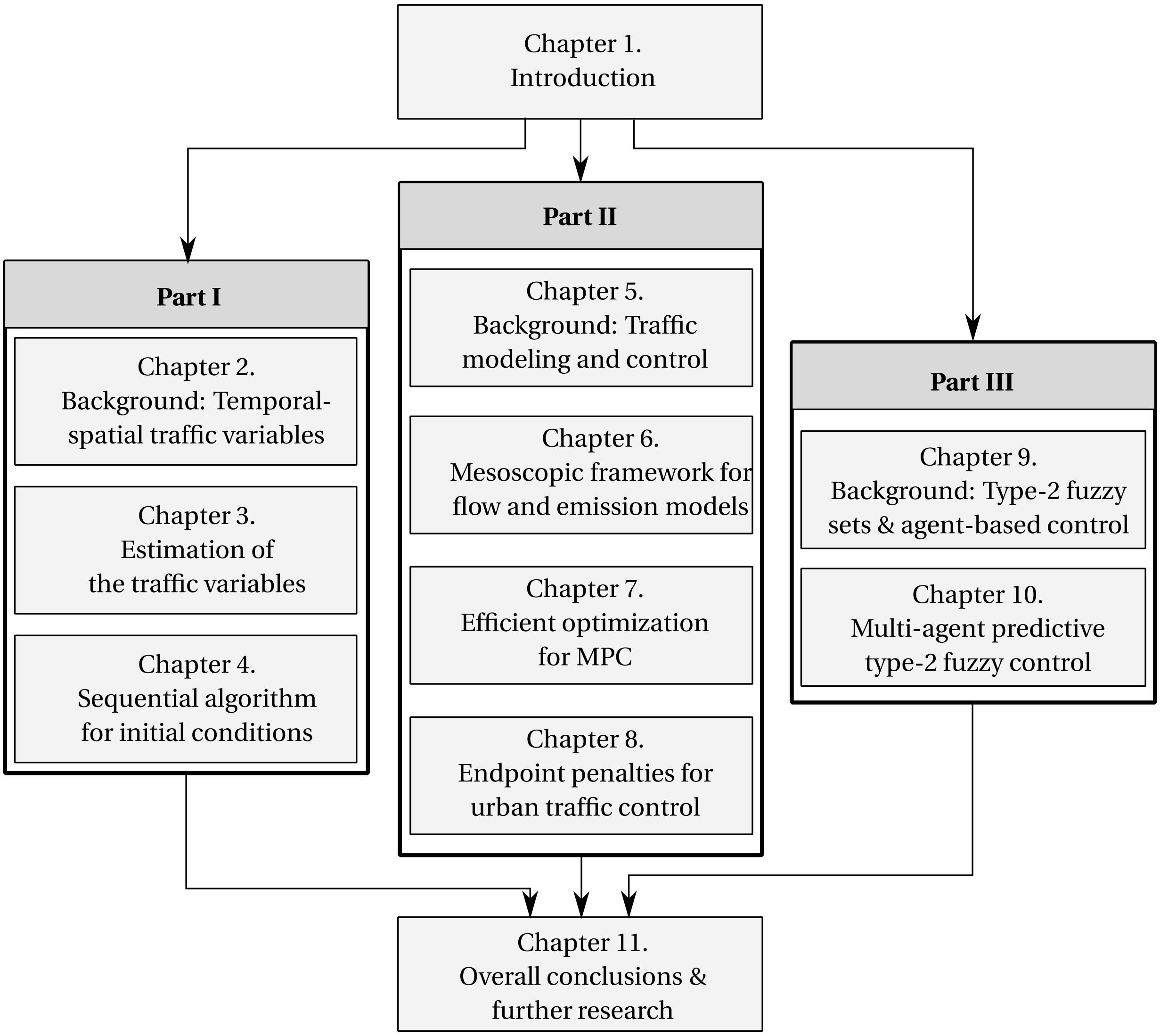

Chapter 1.
Introduction
Part I
Chapter 2.
Background: Temporal-spatial traffic variables
Chapter 3.
Estimation of the traffic variables
Chapter 4.
Sequential algorithm for initial conditions
Part II
Chapter 5.
Background: Traffic modeling and control
Chapter 6.
Mesoscopic framework for flow and emission models
Chapter 7.
Efficient optimization for MPC
Chapter 8.
Endpoint penalties for urban traffic control
Part III
Chapter 9.
Background: Type-2 fuzzy sets & agent-based control
Chapter 10.
Multi-agent predictive type-2 fuzzy control
Chapter 11.
Overall conclusions & further research

Figure 1.1: Road map of the thesis.

traffic variables and on developing efficient and accurate estimating formulas and algorithms for computing the temporal-spatial traffic variables, in particular the time-space mean speed. Part II of the thesis, which includes Chapters 5-8, focuses on fast and accurate modeling of traffic for real-time model-based analysis and control of traffic. Moreover, Part II focuses on efficient control approaches for achieving green mobility in urban traffic networks. In this part, we develop fast and accurate methods for jointly modeling the traffic flow and the emissions in urban networks. Moreover, we propose efficient methods for solving the optimization problem of an MPC controller that finds a balanced trade-off between reduction of the total time spent and reduction of the total emissions of the vehicles in the urban traffic network. Finally, Part III of the thesis is allocated to the theory and application of fuzzy control approaches in adaptive and agent-based control schemes. We extend the definitions and concepts available for type-2 fuzzy sets and we combine the idea of fuzzy logic control and predictive model-based control to provide an efficient control architecture for multi-agent control of large-scale networks. An overall conclusion of the thesis together with the main contributions and some suggestions for further research are given in Chapter 11.

Part I

Microscopic estimation approaches for road traffic

Chapter 2

Background: Temporal-spatial point-of-view of the fundamental traffic variables

2.1 Introduction

Macroscopic fundamental traffic variables (density, flow, and average speed) are important in theory, analysis, control, and performance measurement of traffic (see, e.g., [34, 88, 91, 115, 139, 152]). In particular, the average speed of the vehicles traveling on a specific segment of a road plays an important role in many model-based traffic applications and in performance and control-related studies of traffic. Most of the traffic simulation models use the average speed in accident analyses, in economic studies, as an indication of the service level on the road, and as an input in order to estimate other performance indicators such as the fuel consumption, the vehicle emissions, the travel time, and the traffic noise [111]. Moreover, the average speed is known to be a fundamental measurement in traffic studies. Indeed, the average speed together with the flow and the density are commonly called the *fundamental variables* of traffic, since a basic relationship is established between the flow and the density by means of the average speed, which is known as the *fundamental relationship* or the *fundamental equation* of traffic [35, 153]. We will later use this equation, specifically in Chapter 4, to develop an algorithm that produces the fundamental traffic variables from point measurements.

In addition to the discussed importance of the average speed, the availability of a reliable value of the travel time is important for traffic engineers in applications such as traffic signal coordination, in 'before' and 'after' studies of traffic, and also in estimation of other traffic states [18, 111]. Furthermore, the travel time together with the average speed is used to identify and assess operational problems within highways. Two main methodologies are applied in order to measure or estimate the travel time on a road: the direct measurement method and the indirect estimation method [117].

In the direct measurement method, the total travel time between two predefined reference points is measured after all the vehicles have finished their itineraries. For the measurement, it is necessary to identify the time instant at which a particular vehicle appears at each of the two reference points. Therefore, the identifying technologies (e.g., the automatic vehicle identification system) are implemented by means of license plates or toll tag IDs [142]. However, to apply the identifying techniques, one needs to assume that the vehicles do not make intermediate stops in between the reference points. Besides this limitation, if additional information including the time stopped, the fuel consumption, the number of brake applications, or the travel time between other reference points were are needed, the license plate technique might not be sufficient. The other shortcoming of direct measurements is the information delay in real-time applications, since the measurements are obtained after the vehicle has finished its trip.

The indirect travel time estimation is recommended as an alternative, where the fundamental traffic variables are measured at a specific point of the road link and an algorithm is applied for estimation of the travel time. Mostly, the travel time is considered to be the ratio of the length of the road's section and the average speed [142]. To calculate the travel time, therefore, the average speed on the particular section of the road is needed [56, 142, 153]. Further on, in Section 2.2, we introduce different averaging methods for computing the traffic average speed, and we argue that the temporal-spatial definition (see Section 2.2.3), which is an equivalent for the generalized speed given by Edie [43], is the best averaging method for the speeds obtained from point measurements or data collecting methods that represent the data as trajectories within the time-space plane.

The rest of the chapter is organized as follows. In Section 2.2 the average traffic speed is defined from three different points-of-view, i.e., temporal, spatial, and temporal-spatial averages. Section 2.3 discusses the prevalence of point measurements and consequently, the importance of developing efficient approaches that can estimate the fundamental traffic variables from point measurements. In Section 2.4 we give an overview of the work done on both estimation of the average traffic speed and estimation of the trajectories of the vehicles in between two consecutive point measurements. Finally, an overview of Part I of the thesis is given in Section 2.5. The road map of Chapter 2 is represented in Figure 2.1, and the mathematical notations that are used frequently in this chapter are given in Table 2.1.

2.2 Temporal, spatial, and temporal-spatial definitions of the average traffic speed

The macroscopic fundamental traffic variables can be defined and formulated in two different ways: classical and generalized. In the classical definition within the time-space plane, the density is a spatial average (i.e., the average is made across the space axis), the flow is a temporal average (i.e., the average is made across the time axis), and the average speed is the ratio of the flow and the density [35, 153]. Edie [43] has proposed a more general definition for the fundamental traffic variables, where the averages are made within an area in the time-space plane. Next, we present more details regarding these definitions for the average speed.

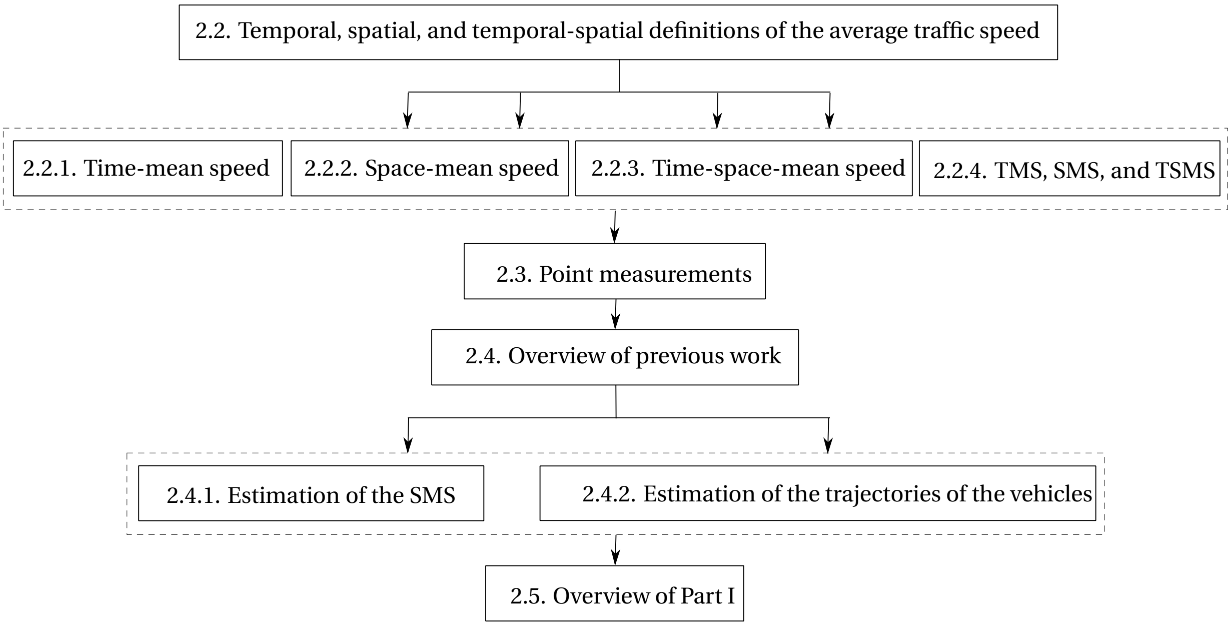

2.2. Temporal, spatial, and temporal-spatial definitions of the average traffic speed
2.2.1. Time-mean speed
2.2.2. Space-mean speed
2.2.3. Time-space-mean speed
2.2.4. TMS, SMS, and TSMS
2.3. Point measurements
2.4. Overview of previous work
2.4.1. Estimation of the SMS
2.4.2. Estimation of the trajectories of the vehicles
2.5. Overview of Part I

Figure 2.1: Road map of Chapter 2.

Table 2.1: Frequently used mathematical notations for Chapter 2.

$p\,(t, x_1, x_2, \ldots, x_N)$	joint probability function
$p_i^{\mathrm{m}}(t, x)$	marginal probability of vehicle i at time instant t and at position x
$p_i^{\mathrm{d}}(t, x)$	probability density of vehicle i at time instant t and at position x
ρ_A	generalized traffic density within the time-space area A
q_A	generalized traffic flow within the time-space area A
$\bar{v}_A$	generalized average traffic speed within the time-space area A
N_A	total number of the trajectories of vehicles that are observed within the time-space area A
$\|A\|$	surface of the time-space area A
$t_{A,i}$	time spent by the i^{th} vehicle within the time-space area A
$d_{A,i}$	distance traveled by the i^{th} vehicle within the time-space area A
TMS	time-mean speed
SMS	space-mean speed
TSMS	time-space mean speed
σ^2_{TMS}	temporal standard deviation of the observed speeds of the vehicles
σ^2_{SMS}	spatial standard deviation of the observed speeds of the vehicles

2.2.1 Time-mean speed (TMS)

The *time-mean speed (TMS)* involves averaging the individual speeds of the vehicles at a fixed location across a time interval (the mathematical formulation of the TMS will be given later

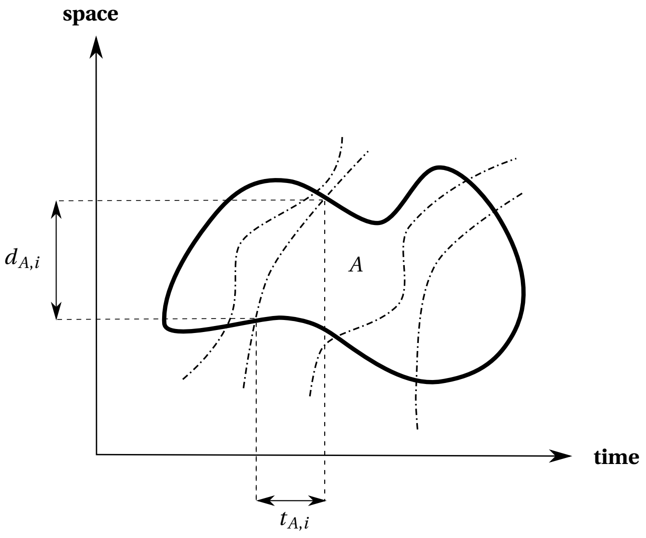


Figure 2.2: An arbitrary region in the time-space plane for defining the generalized fundamental traffic variables.

in Section 2.2.4). A stationary observer such as a loop detector observes the vehicles during a sample period and at a fixed position. Hence, from the definition given above, the arithmetic mean of the individual speeds observed by a loop detector gives the TMS.

2.2.2 Space-mean speed (SMS)

The *space-mean speed (SMS)* involves averaging the individual speeds of the vehicles observed at a specific instant of time across a stretch of the road (the mathematical formulation of the SMS will be given later in Section 2.2.4). Therefore, considering a traffic camera that captures at a specific time instant a photograph that covers a stretch of road of a fixed length, the arithmetic mean of the individual speeds captured in the photograph is the SMS.

2.2.3 Time-space-mean speed (TSMS)

The generalized point-of-view for defining the traffic average speed results in an average called the *time-space-mean speed (TSMS)*. The fundamental traffic equation relates two fundamental traffic variables, the flow and the density (note that the flow is defined across a time interval and the density is defined across a length interval). Considering the ratio of these two variables, we need an intermediary variable (with the unit of length per time, i.e., the speed unit) that is neither a local nor an instantaneous variable, but a variable defined simultaneously through both the time and the space axes. We call this variable the TSMS, which is equivalent to the generalized speed introduced by Edie [43].

Since the definition of the intermediary variable that relates the flow and the density, depends on how we define these two concepts, we will discuss these definitions. According to the definitions given by Newell [122] for the flow and the density, which originate from a mathematical and physical perspective, the formulation given by Edie [43], which considers a temporal-spatial average for the traffic speed, is the most appropriate formulation. More specifically, Newell [122] starts with a purely physical interpretation, which because of the specific characteristics of traffic leads to a mathematical analysis. Newell [122] considers the analogy between the concepts of density in traffic (where we need to know the number of vehicles per unit length), and density in fluid dynamics (where we need to know the mass per unit volume).

To define density, both *homogeneity* and also a *sufficient* number of vehicles/atoms on the corresponding segment of the road/volume are required. For a small volume of a fluid, there is still a considerable number of particles, while based on physical laws, local concentrations of particles diffuse quite rapidly. Therefore, the possible observed fluctuations in the properties of the selected volume are negligible with respect to the huge number of particles. Thus existence of both homogeneity and sufficient number of atoms lets the density be defined simply. However, a small segment of the road with zero length limit contains either 0 or 1 vehicle. Consequently, traffic density on this small road segment will be either zero or infinite. Therefore, fulfillment of either homogeneity or sufficiency of numbers causes the other one to be violated.

Correspondingly, Newell [122] shifts to a mathematical interpretation for the traffic density. First a joint probability function[1] $p\,(t, x_1, x_2, \ldots, x_N)$ is defined, which is the probability that at time instant t vehicle i has a position strictly less than x_i, for $i \in \{1,\ 2, \ldots,\ N\}$, where N is the total number of vehicles. Hence, the marginal probability p_i^{m} for the i^{th} vehicle at time instant t and at position x (i.e., the probability that at time instant t, the i^{th} vehicle has a position less than x) is given by

$$p_i^{\mathrm{m}}(t,x) = p(t, L+\Delta, \ldots, L+\Delta, x, L+\Delta, \ldots, L+\Delta), \tag{2.1}$$

where L is the length of the road segment and $\Delta \gg 0$. Therefore, the probability for the vehicle to be located between position x and position $x + \mathrm{d}x$ (with $\mathrm{d}x$ an infinitesimally small positive length increment) at time instant t, is $p_i^{\mathrm{m}}(t, x+\mathrm{d}x) - p_i^{\mathrm{m}}(t,x)$. Then we define a probability density function p_i^{d} for the i^{th} vehicle at time instant t and at position x, which is given by

$$p_i^{\mathrm{d}}(t,x) = \frac{\partial}{\partial x} p_i^{\mathrm{m}}(t,x). \tag{2.2}$$

Finally, the total density for all the n observed vehicles at time instant t and at position x (which can indeed be considered as the *instantaneous* density apart from the *generalized* density, which will be defined afterwards) on the road segment is obtained by

$$\rho(t,x) = \sum_{i=1}^{N} p_i^{\mathrm{d}}(t,x). \tag{2.3}$$

The flow can similarly be defined through the time axis.

The practical way of implementing the mathematical interpretation of the density, the

[1] This function is assumed to be differentiable.

flow, and correspondingly the average speed for a traffic network is given by Edie [43] where the *generalized* definitions of the fundamental variables are represented. For a given time-space detection area A (see Figure 2.2), the generalized definitions of the fundamental variables describe the average behavior of a traffic stream in the time-space plane where the generalized density ρ_A and the generalized flow q_A are given by

$$\rho_A = \frac{1}{|A|}\sum_{i=1}^{N_A} t_{A,i}, \tag{2.4}$$

$$q_A = \frac{1}{|A|}\sum_{i=1}^{N_A} d_{A,i}, \tag{2.5}$$

with $t_{A,i}$ and $d_{A,i}$, respectively, the time spent and the distance traveled by the i^{th} vehicle in the time-space area A, $|A|$ the surface of the detection area, and N_A the number of the trajectories of vehicles that are observed in area A.

Newell [122] shows that Edie's definitions of the generalized fundamental variables are equivalent to the mathematical definitions of density given by (2.3) and flow (which is obtained following the same reasoning as for the density). Finally, the generalized average speed $\bar{v}_A$ within the detection area A is defined as the ratio of the generalized flow and the generalized density, i.e.,

$$\bar{v}_A = \frac{q_A}{\rho_A} = \frac{\sum\limits_{i=1}^{N_A} d_{A,i}}{\sum\limits_{i=1}^{N_A} t_{A,i}}. \tag{2.6}$$

2.2.4 Relation between TMS, SMS, and TSMS

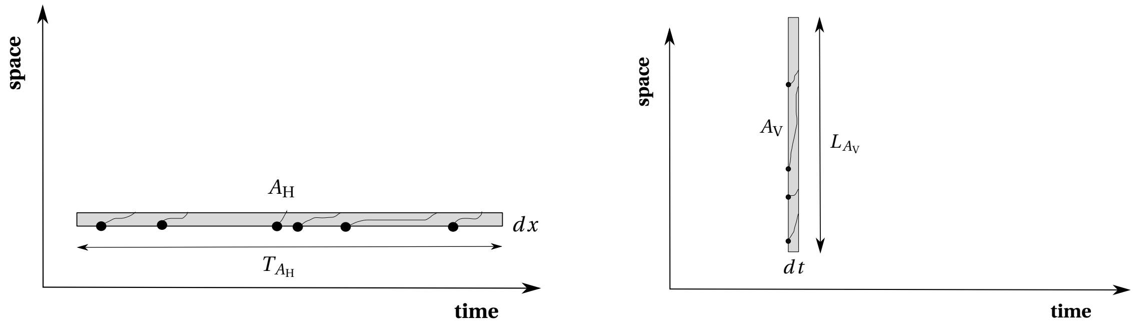
In the following discussion, we consider the relationships between the TMS and the TSMS, and also between the SMS and the TSMS. We apply the definition of the generalized average speed to the following two cases:

1. A thin horizontal sampling window with length dx and width $T_{A_{\text{H}}}$ (see Figure 2.3(a))

2. A thin vertical sampling window with length $L_{A_{\text{V}}}$ and width dt (see Figure 2.3(b))

The first case can represent the detection zone A_{H} of a loop detector. Due to the fact that dx is very small, the possibility that a trajectory enters or leaves the window through its left or right edge is negligible. Hence, from (2.6) and from the definition of TMS given in Section 2.2.1, we can write

$$\text{TSMS}_{A_{\text{H}}} = \frac{\sum\limits_{i=1}^{N_{A_{\text{H}}}} d_{A_{\text{H}},i}}{\sum\limits_{i=1}^{N_{A_{\text{H}}}} t_{A_{\text{H}},i}} = \frac{\mathrm{d}x \cdot N_{A_{\text{H}}}}{\sum\limits_{i=1}^{N_{A_{\text{H}}}} \frac{\mathrm{d}x}{v_{A_{\text{H}},i}}} = \frac{1}{\frac{1}{N_{A_{\text{H}}}}\sum\limits_{i=1}^{N_{A_{\text{H}}}} \frac{1}{v_{A_{\text{H}},i}}}, \tag{2.7}$$



(a) A thin horizontal sampling window.

(b) A thin vertical sampling window.

Figure 2.3: The TMS and SMS and their relationship with the TSMS.

and

$$\text{TMS}_{A_\text{H}} = \frac{1}{N_{A_\text{H}}} \sum_{i=1}^{N_{A_\text{H}}} v_{A_\text{H},i}, \tag{2.8}$$

where N_{A_H} is the number of vehicles observed within the thin horizontal sampling window during the sampling time T_{A_H}, and $v_{A_\text{H},i}$ is the observed speed for the i^th vehicle within the area. Thus, for a thin horizontal sampling window the TSMS is the harmonic mean of the detected speeds, while the TMS is the arithmetic mean of the speeds. This is also discussed extensively by Treiber and Kesting [149].

The observation area of a camera can be illustrated by the sampling window of case two. Due to the fact that dt is very small, the possibility that a trajectory enters or leaves the window through its bottom or top edge is negligible. Therefore, from (2.6) and from the definition of SMS given in Section 2.2.2, we have

$$\text{TSMS}_{A_\text{V}} = \frac{\sum_{i=1}^{N_{A_\text{V}}} d_{A,i}}{\sum_{i=1}^{N_{A_\text{V}}} t_{A_\text{V},i}} = \frac{\sum_{i=1}^{N_{A_\text{V}}} dt \cdot v_{A_\text{V},i}}{dt \cdot N_{A_\text{V}}} = \frac{1}{N_{A_\text{V}}} \sum_{i=1}^{N_{A_\text{V}}} v_{A_\text{V},i}, \tag{2.9}$$

and

$$\text{SMS}_{A_\text{V}} = \frac{1}{N_{A_\text{V}}} \sum_{i=1}^{N_{A_\text{V}}} v_{A_\text{V},i}, \tag{2.10}$$

where N_{A_V} is the number of vehicles observed in the thin vertical sampling window, and $v_{A_\text{V},i}$ is the observed speed for the i^th vehicle within the area. Then the TSMS at a specific instant of time is the arithmetic mean of the speeds being observed, where this is by definition the SMS. Therefore, for a thin vertical rectangle the TSMS and the SMS are equivalent.

2.3 Point measurements

Since the introduction of inductive loop detectors in the early 1960's, they have been widely used for the purpose of vehicle detection on roads and for point measurement of the speeds of vehicles. In fact, according to Klein et al. [83] inductive loop detectors are the most popular and the most widely used forms of traffic detection systems. Moreover, Bickel et al. [13] indicate that: "Point sensors implemented by inductive loops provide 95% of the data used by DoTs (Departments of Transportation) and traffic analysts worldwide". Hence, due to their extensive use and high costs of substituting the available inductive loop detectors with new detection technologies, there is a high chance that they are still in use for at least the next 10–15 years.

Both microscopic and macroscopic characteristics of the traffic flow can be determined using inductive loop detectors. Single-loop detectors have been deployed in order to provide information on the traffic flow (i.e., the number of vehicles passing a specific point on the road per unit time) and the lane occupancy (i.e., the fraction of the observation time interval that the loop's detection zone is occupied by vehicles). Microscopic traffic flow characteristics including the time headway[2], the vehicle occupancy time, and the space headway can be estimated using a single-loop detector [104]. A double-loop detector is constructed by installing two single-loop detectors consecutively a few meters apart from one another. The main advantage of a double-loop detector over a single-loop detector is that it can provide individual speed data [151].

The extensive use of inductive loop detectors in traffic systems all around the world and the relatively high costs of substituting them with modern detecting instruments has incited the development of efficient ways for estimating the fundamental traffic variables based on the information provided by these loop detectors.

2.4 Overview of previous work on estimation of the traffic average speed

2.4.1 Estimation of the SMS

As motivated in this chapter, the temporal-spatial definition of the fundamental traffic variables plays an important role in traffic theory and applications. However, the available literature mostly focuses on estimation of the traffic variables in their classical definition. To the best of our knowledge, there is no work on estimation of the generalized average speed from point measurements. In this section, we briefly review some of the most significant work done on estimation of the classical traffic average speed.

Dailey [36] presents an algorithm to estimate the mean speed using data from a single-loop detector. Dailey [36] considers the statistical nature of the measurements made by single-

[2] The time between consecutive vehicle observations at a fixed location is usually called the time headway, and the distance separation between consecutive vehicles at a given time instant is called the distance headway [111] or alternatively the space headway if we follow the terminology used by Daganzo [35].

loop detectors and presents an algorithm that estimates the speed. However, Dailey [36] does not distinguish between the TMS, the SMS, and the TSMS and only uses the notation $\bar{v}$ in general (note that the SMS best applies to the formula given by Dailey [36]). The measurements are by their nature realizations taken from the probability distributions of the underlying variables. Two typical measurements, i.e., the flow q and the occupancy o, are taken into account. Hence,

$$o = \frac{1}{T}\sum_{i=1}^{q\cdot T}\frac{l_i}{v_i}, \tag{2.11}$$

where l_i is the effective length (i.e., the length of the vehicle plus the length of the detection zone) of the i^{th} vehicle, and T is the duration of the measurement.

Furthermore, the speed and the length of the vehicles are random variables that can be rewritten as the summation of their expected value (respectively, $\bar{v}$ and $\bar{l}$) and the deviation that occurs for the observations (respectively, Δv_i and Δl_i). Using these expressions, Dailey [36] obtains the following equation for the mean speed, $\bar{v}$:

$$o\frac{T}{\bar{l}}\bar{v}^3 - q\bar{v}^2 - q\sigma_v^2 = 0. \tag{2.12}$$

Two new estimation methods are also introduced by Dailey [36]; the first one is the '*root finding*' method, which is based on the deterministic speed values and yields an unbiased estimator for $\bar{v}$ when there are idealized noiseless measurements (which is almost never the case in practice), and the second one is the '*filtering*' method, which addresses the reliability of the measurements.

Wardrop [153] develops a relationship between the SMS and the TMS using a macroscopic point-of-view, which estimates the TMS from a known SMS. The equation given by Wardrop states that

$$\text{TMS} = \text{SMS} + \frac{\sigma_{\text{SMS}}^2}{\text{SMS}}, \tag{2.13}$$

with σ_{SMS}^2 the spatial standard deviation of the observed speeds. However, since in the case of a loop detector the only available information is the TMS, it is not straightforward to estimate σ_{SMS}^2.

Han et al. [56] suggest a different representation of (2.13) that does not involve σ_{SMS}, but adds the mean value of the individual speed values. Han et al. [56] discuss a theoretical approach in combination with an empirical method to solve (2.13) for the SMS using the definition of σ_{SMS}^2. The applied procedure is as follows:

$$\begin{aligned}\sigma_{\text{SMS}}^2 &= \text{E}\left[(v_i - \text{SMS})^2\right] \\ &= \text{E}\left[v_i^2\right] + \text{SMS}^2 - 2\text{SMS}\cdot\text{TMS},\end{aligned} \tag{2.14}$$

where v_i is the speed of the i^{th} entering vehicle passing through the loop detector, and E[.] denotes the expected value operator. The above relation is applied to (2.13) and the resulting quadratic equation is solved, which finally results in

$$\mathrm{SMS} = \frac{3\mathrm{TMS} \pm \sqrt{9\mathrm{TMS}^2 - 8\mathrm{E}\left[v_i^2\right]}}{4}. \tag{2.15}$$

In (2.15) the value of $\mathrm{E}\left[v_i^2\right]$ is unknown. Han et al. [56] propose a quadratic relationship between $\mathrm{E}\left[v_i^2\right]$ and $\mathrm{E}\left[v_i\right]$, which is given by

$$\mathrm{E}\left[v_i^2\right] = a \cdot \mathrm{TMS}^2 + b \cdot \mathrm{TMS} + c, \tag{2.16}$$

where a set of data from the loop detectors should be used to estimate the constant coefficients a, b, and c empirically.

Rakha and Zhang [134] propose an approach for extracting the formula that has been introduced by Khisty and Lall [82] for estimating the SMS. This formula is given by

$$\mathrm{SMS} \simeq \mathrm{TMS} - \frac{\sigma_{\mathrm{TMS}}^2}{\mathrm{TMS}}, \tag{2.17}$$

where σ_{TMS} is the temporal standard deviation of the individual speeds of the observed vehicles. In (2.17) for estimation of the SMS, we need to know the values of the TMS and also of σ_{TMS}^2. However, loop detectors only report the value of the TMS. As a consequence, Soriguera and Robusté [141] propose to use a normal distribution for the vehicle speeds on a particular lane of the road to find an estimate for σ_{TMS} and to apply it to (2.17). A confidence interval is also formulated by Soriguera and Robusté [141] for the estimated value of the SMS, which delimits the error for a desired confidence level. The formula proposed by Soriguera and Robusté [141] is given by

$$\sigma_{\mathrm{TMS}} = \frac{v^* - \mathrm{TMS}}{\phi^{-1}\left(\dfrac{N_{v^*}}{N}\right)}, \tag{2.18}$$

with v^* a particular speed threshold, $\phi(\cdot)$ the cumulative distribution function of the standard normal distribution, N_{v^*} the number of observed vehicles moving with speed v^*, and N the total number of observed vehicles.

2.4.2 Estimation of the trajectories of the vehicles

In Chapter 4, we will propose a new method for approximating the trajectories of the vehicles in the time-space plane, which is inspired by Coifman [31]. Hence, we also explain this approach briefly here. Coifman [31] proposes a method to estimate trajectories of vehicles between two consecutive double-loop detectors based on available point measurements. According to Lighthill and Whitham [94], if a change in speed occurs at a point of a certain traffic stream, this change will back-propagate through the traffic stream with a fixed speed. The speed of the back-propagation of the change depends on the governing traffic situation (free-flow or congested). Coifman [31] applies this result to find the approximate trajectories. Sup-

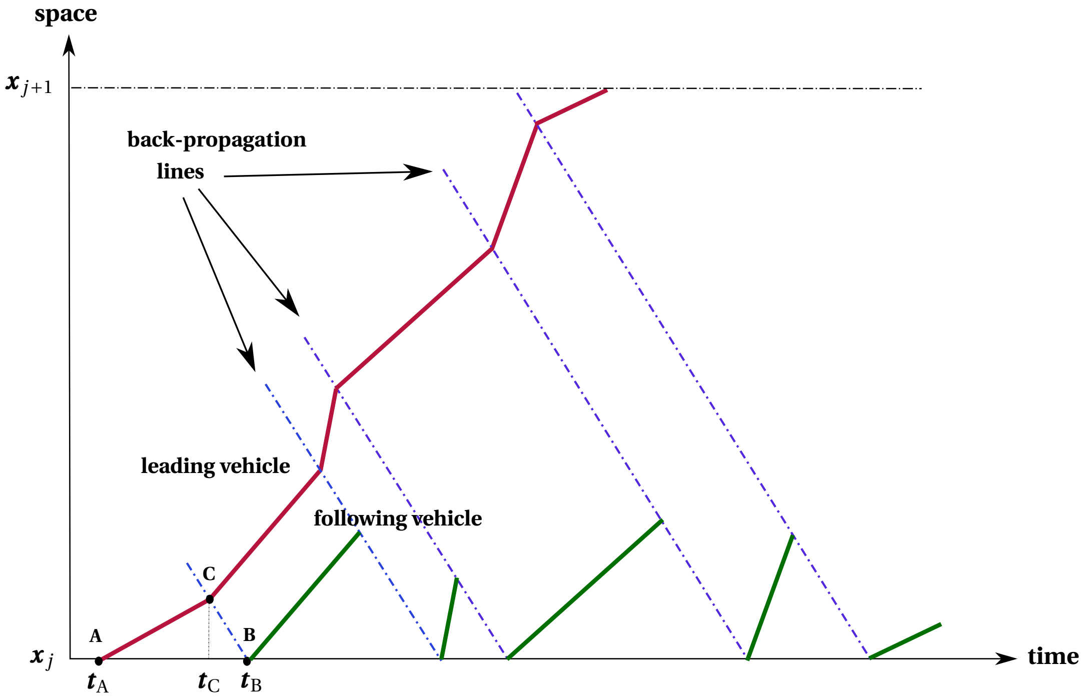


Figure 2.4: Estimating the trajectories of vehicles based on the approach of Coifman [31].

pose that speed measurements of two successive vehicles are available at points “A” and “B” within the time-space plane (see Figure 2.4), with these points located at the loop detector line and corresponding to time instants t_A and t_B.

Coifman [31] plots the back-propagation lines (see the blue dashed-dotted lines in Figure 2.4) with a fixed slope equal to the back-propagation speed. These back-propagation lines originate at points in the time-space plane, where speed measurements are available (e.g., at point “B” in Figure 2.4). As soon as a vehicle’s trajectory intersects with one of these back-propagation lines, it changes its slope to the one of the trajectory of the leading vehicle (e.g., at point “C” in Figure 2.4, the trajectory changes its slope to the slope of the trajectory of the vehicle observed at point “B”).

This actually means that the vehicle observed at point “B” has been following the vehicle observed at point “A”, and at point “B”, this vehicle adapts its speed to the speed of the leading vehicle at point “C”. In practice, since the speed measurements are available only at the loop detector line (i.e., at location x_j in Figure 2.4), the slope of the trajectory of the leading vehicle at point “C” can be extracted from the slope of the trajectory of the following vehicle at point “B”. This is not a problem for off-line processing of the dataset of course, but this approach cannot be implemented online.

Note that the approach proposed by Coifman [31] is based on the Newell’s car-following model [123] and on the assumption that the adapting point of the following vehicle, i.e., the point in the time-space plane at which the following vehicle changes its speed to that of the leading vehicle, is located at the loop detector line.

2.5 Overview of Part I

The aim of this part of the thesis is to cover the current gap in the existing literature on the topic of estimating the TSMS. For this aim, in Chapter 3, we propose a novel approach for approximating the TSMS based on a microscopic point-of-view and point measurements. An upper and a lower bound for the TSMS are developed first, and later the approach is extended with the aim of making the bounds tighter. Finally, a convex combination of the upper and lower bound is proposed to estimate the TSMS, while coping with the cases where the trajectories of the vehicles in the time-space plane might not be straight lines. Moreover, in Chapter 4 a sequential algorithm is introduced that gives an estimate for the generalized traffic variables, including the TSMS, using point measurements. The algorithm considers those vehicles that stay on the segment of the road between two consecutive measurement points for more than one sampling cycle and hence, are not detected by either of the upstream or downstream loop detectors. The proposed algorithm can be used to upgrade or to reprogram inductive double-loop detectors so as to provide a more accurate approximation of the TSMS.

Chapter 3

Estimation of the temporal-spatial traffic variables

3.1 Introduction

The average speed of vehicles plays an important role in traffic engineering. Almost in any model-based traffic monitoring, analysis, and control application the average speed is required as a measure of performance or as an input for traffic models where fuel consumption, vehicle emissions, or traffic noise is to be estimated. The average speed is also used in algorithms that estimate the travel time. It also appears in the fundamental equation of traffic where density is calculated based on measurements of average speed and flow. This chapter presents a new methodology for estimating the time-space-mean speed (TSMS), which is an equivalent for the definition initially given by Edie [43] as the *generalized speed*. To this aim, first tight upper and lower bounds are developed for the TSMS using individual speeds of the vehicles that are obtained via point measurements. To estimate the TSMS from the bounds, and to deal with the cases where the trajectories of the vehicles might not be straight lines, a convex combination of the upper and lower bounds is introduced.

In order to assess the convex combination and to compare its performance with other formulas in literature, real-life datasets including the NGSIM dataset and the Rotterdam-Delft dataset are used. NGSIM data provides detailed information of the trajectories for the I-80 freeway in the San Francisco Bay Area, while the Rotterdam-Delft dataset provides microscopic data on freeway A13 in the Netherlands. At the end, we include simulations in MATLAB to cover the possible scenarios that are not included in real-life datasets. The results produced by the new formula, both for the real-life datasets and for the MATLAB simulations, are found to be more exact compared with other available formulas in literature.

Contributions and organization of the chapter

The main contributions of this chapter are as follows:

1. We present a new approach based on a microscopic point-of-view that produces a tight

upper and a tight lower bound for the time-space-mean speed.

2. We construct a formula that uses a convex combination of these bounds to find a reliable estimate of the time-space-mean speed.

3. We assess and compare the new formula with available formulas in literature using real-life datasets, i.e., the NGSIM dataset and the Rotterdam-Delft dataset.

The rest of the chapter is organized as follows; Section 3.2 describes the problem that is going to be discussed in this chapter. Section 3.3 proposes a new method for finding a tight upper and a tight lower bound for the time-space-mean speed (TSMS), a convex combination for estimating the TSMS from these tight bounds, and the proofs for developing the tight upper and lower bounds. Moreover, at the end of this section, we show that the developed formulas can be used for an arbitrary time headway distribution and the assumptions made before can easily be relaxed. Finally, the results for real-life datasets and also for MATLAB simulations are presented and discussed in Section 3.4. Section 3.5 is allocated to the conclusions and topics for further research. The schematic view of the chapter's road map is represented in Figure 3.1, and Table 3.1 presents the frequently used mathematical notations of this chapter.

This chapter of the thesis is based on [63].

3.2 Problem definition

In this chapter, we solve the problem of finding the appropriate average speed (which we have argued to be the TSMS in Section 2.2.3) giving a formula that covers microscopic data of point measurement type. Note that from (2.7), the TSMS is equal to the harmonic mean of the individual speeds of the observed vehicles, in case the time-space area is a very thin horizontal rectangle (e.g., see the detection zone of loop detector 1 of the road section illustrated in Figure 3.4(a), where this detection zone is represented by a dashed thin horizontal rectangle in Figure 3.4(b)). However, in case the area of interest in the time-space plane is a rectangle with a considerable length (see rectangle $v_1v_2v_3v_4$ in Figure 3.4(b), which illustrates the time-space area corresponding to a piece of a single-direction road section between two consecutive loop detectors), the harmonic mean might not be a reliable estimate of the TSMS anymore.

Defining the sampling windows for estimation of the TSMS

Since for the TSMS, the traffic information is averaged within an area (i.e., over two dimensions, the space and the time, simultaneously), and since we use the trajectories of the vehicles to consider features of the traffic stream, the area of interest is the time-space plane. We consider a rectangular window in the time-space plane, which forms the main idea of the new methodology given in this chapter. This window represents the considered segment of the road during one sampling cycle. From now, we simply call this rectangular time-space window the *sampling window*.

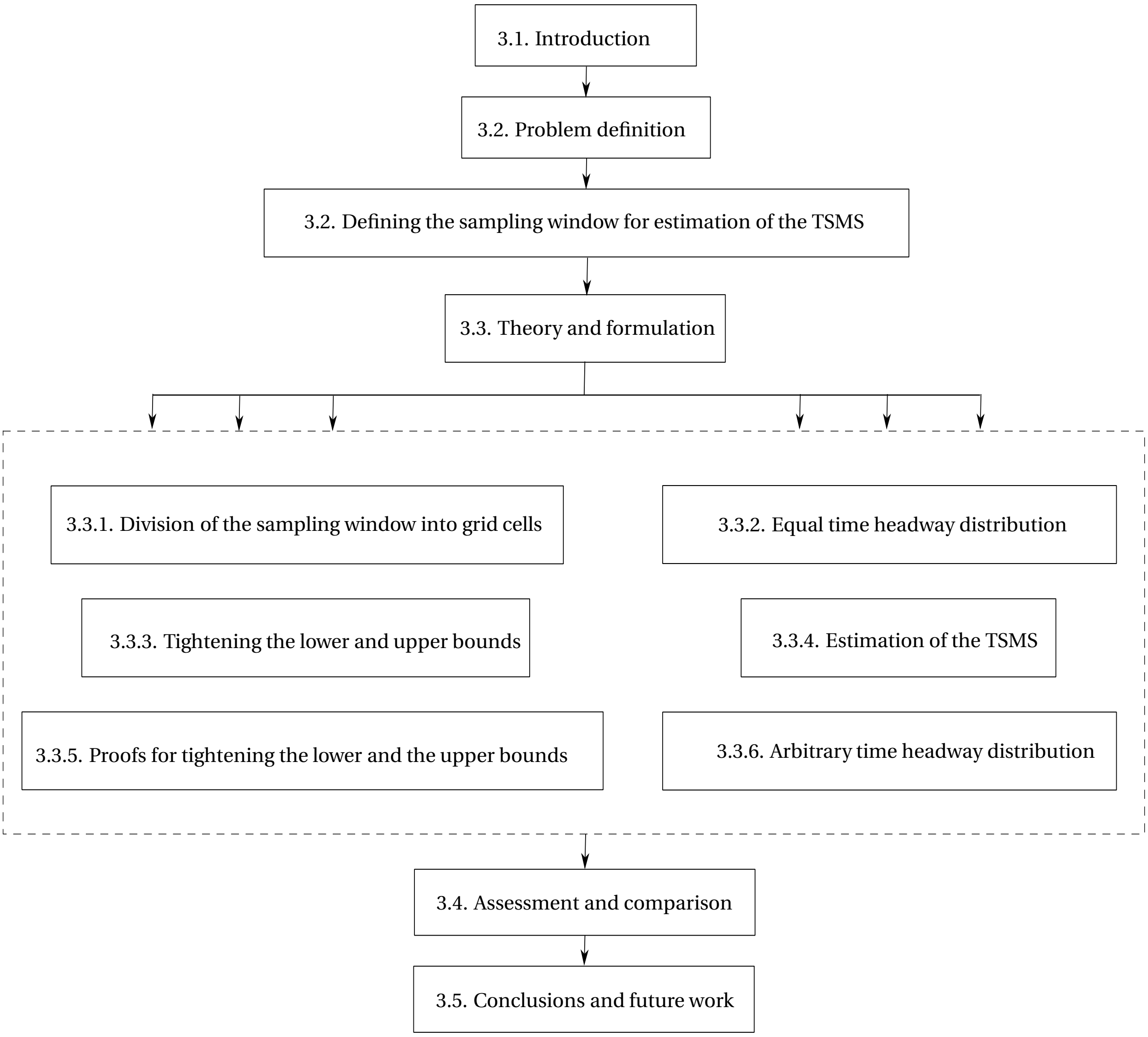

3.1. Introduction
3.2. Problem definition
3.2. Defining the sampling window for estimation of the TSMS
3.3. Theory and formulation
3.3.1. Division of the sampling window into grid cells
3.3.2. Equal time headway distribution
3.3.3. Tightening the lower and upper bounds
3.3.4. Estimation of the TSMS
3.3.5. Proofs for tightening the lower and the upper bounds
3.3.6. Arbitrary time headway distribution
3.4. Assessment and comparison
3.5. Conclusions and future work

Figure 3.1: Road map of Chapter 3.

First we explain how the equivalent sampling window of a road section within the time-space plane can be constructed. Consider a road of length L^{road}, with n^{loop} inductive loop detectors indicated by D_j installed at positions x_j, $j \in \{1,2,\dots,n^{\text{loop}}\}$, where the distance L_j between any two consecutive loop detectors D_j and D_{j+1} is represented by

$$L_j = x_{j+1} - x_j. \tag{3.1}$$

Moreover, the distance $L_{n^{\text{loop}}}^{\text{endpoint}}$ between the last loop detector and the endpoint of the road is obtained via

$$L_{n^{\text{loop}}}^{\text{endpoint}} = x^{\text{endpoint}} - x_{n^{\text{loop}}}.$$

Table 3.1: Frequently used mathematical notations for Chapter 3.

N_A	total number of vehicles observed within the sampling road section A
T_A	sampling cycle of the loop detector installed at the beginning of the sampling road section A
L_A	spatial length of the sampling road section A
$v_{A,i}$	speed of the i^{th} vehicle observed on the sampling road section A
$H_{A,1\rightarrow n_{\text{cell},A}-m_A+1}$	harmonic mean of the speeds of vehicles that correspond to the first $n_{\text{cell},A} - m_A + 1$ grid cells assuming equal time headway distribution for the vehicles
$v_{\text{cell},A}$	the cell speed corresponding to the sampling window A
$v_{\min,A}$	minimum observed speed on the sampling road section A
$v_{\max,A}$	maximum observed speed on the sampling road section A
$h_{A,i}$	time headway between the i^{th} and the $(i+1)^{\text{th}}$ vehicle on the sampling road section A
h_A	mean time headway of the observed vehicles on the sampling road section A
$n_{\text{cell},A}$	number of grid cells in the sampling road section A for an equal time headway distribution
$t_{A,i}$	travel time of vehicle i within the sampling road section A
$d_{A,i}$	traveled distance of vehicle i within the sampling road section A
$x_{A,i}$	position of vehicle i within the sampling road section A
x_{loop}	position of the loop detector "loop"
$\bar{v}_A$	generalized average traffic speed (TSMS) corresponding to the sampling road section A
$\bar{v}_A^{\text{lower}}$	a lower bound for $\bar{v}_A$
$\bar{v}_A^{\text{upper}}$	an upper bound for $\bar{v}_A$
$\text{TSMS}_A^{\text{lower}}$	the tight lower bound for $\bar{v}_A$
$\text{TSMS}_A^{\text{upper}}$	the tight upper bound for $\bar{v}_A$
$\text{TSMS}_A^{\text{est}}$	the estimated $\bar{v}_A$ from $\text{TSMS}_A^{\text{lower}}$ and $\text{TSMS}_A^{\text{upper}}$

with x^{endpoint} the position of the endpoint of the road. Note that to define the positions and distances on a road, we first consider a virtual curve that coincides with the centerline of the road (see the dashed curve in Figure 3.2). Then we consider an equivalent straight road for which the length is equal to the length of the centerline curve (see Figure 3.3). Note that all positions are measured along the centerline of the road.

Now Consider the road section illustrated in Figure 3.4(a), where two consecutive loop detectors are shown. Suppose that the sampling cycle of the loop detectors is T_A, and the distance between the two loop detectors, i.e., the length of the black dashed line, representing the middle of the road, is L_A. Then the length and the width of the sampling window are, respectively, L_A and T_A (see Figure 3.4(b)). Loop detector 1 collects data in its detection zone, which is illustrated by the thin dashed rectangle at $x = x_{\text{loop1}}$ in Figure 3.4(b). Under such a configuration, where point measurements are available for discrete points on the road at the locations at which the loop detectors are installed, the challenge is to find an approach to estimate the average speed during each sampling cycle.

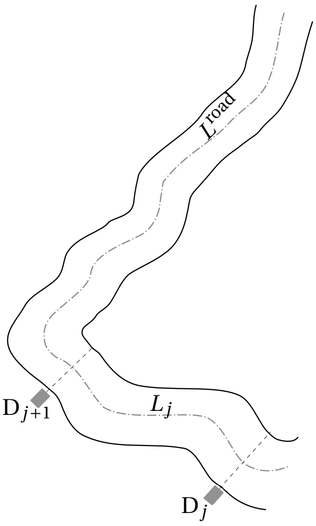


Figure 3.2: Real shape of the single-lane road (D_j and D_{j+1} are two consecutive loop detectors on the road and L_j is the distance between these loop detectors along the centerline of the road). The dashed dotted line indicates the virtual centerline of the road.

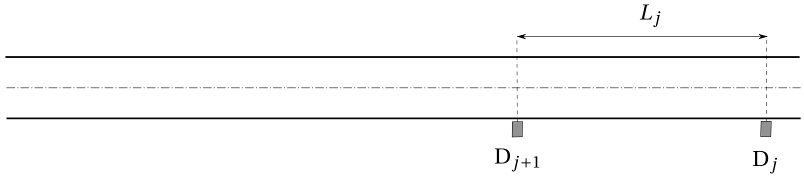
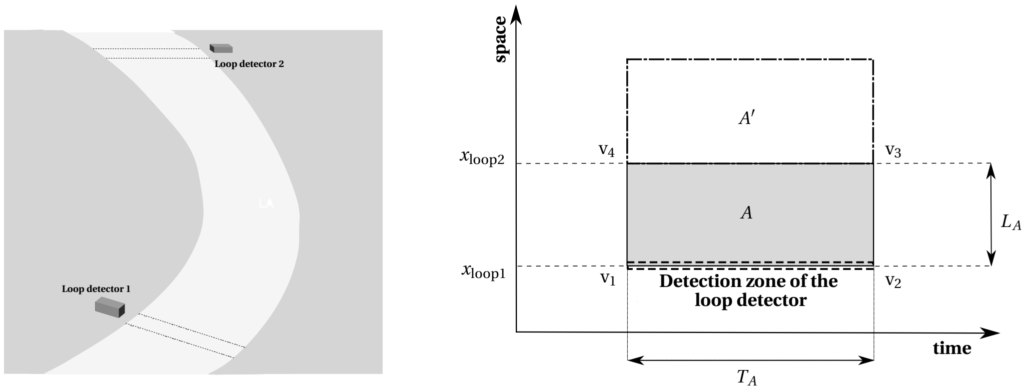


Figure 3.3: Mapping the real road into Cartesian coordinates.



(a) A section of the road stretched between two consecutive loop detectors.

(b) The sampling window A with length L_A and width T_A corresponding to the sampling road section.

Figure 3.4: Specifying a sampling window for a sampling road section in the time-space plane to estimate the TSMS.

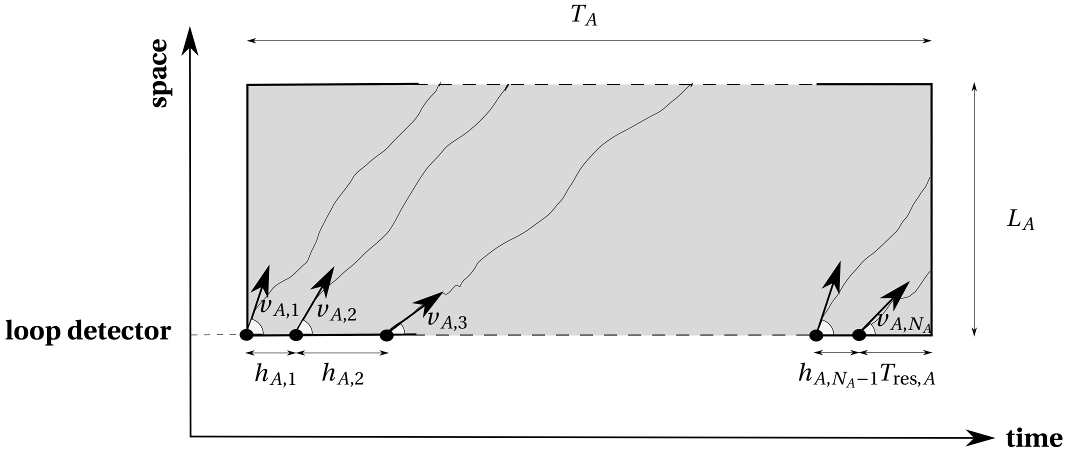


Figure 3.5: Trajectories of the observed vehicles within the sampling window A.

3.3 New formulas for a tight upper and a tight lower bound for the TSMS

3.3.1 Division of the sampling window into grid cells

Consider the sampling window (T_A, L_A) shown in Figure 3.5, which contains the trajectories of the vehicles that have been observed by the loop detector installed at the beginning of the road section. The number of observed vehicles by the loop detector within the sampling window A is N_A, and the corresponding microscopic data that can be extracted from the loop detector during one sampling cycle are given by

$$\begin{aligned} \mathbf{V}_A &= \{v_{A,i} \mid i = 1, 2, \ldots, N_A\}, \\ \mathbf{H}_A &= \{h_{A,i-1} \mid i = 1, 2, \ldots, N_A\}, \end{aligned} \tag{3.2}$$

where $v_{A,i}$ and $h_{A,i-1}$ are, respectively, the speed of the i^{th} observed vehicle, and the time headway between the $(i-1)^{\text{th}}$ and the i^{th} vehicle if $i \neq 1$, where for $i = 1$, $h_{A,0}$ indicates the time duration from the beginning of the observation cycle until the first vehicle is observed. We assume that for the sampling window A, $h_{A,0} = 0$. From the given measurements, we obtain the following information

$$v_{\min,A} = \min_{i=1,\ldots,N_A} (v_{A,i}), \tag{3.3}$$

$$v_{\max,A} = \max_{i=1,\ldots,N_A} (v_{A,i}), \tag{3.4}$$

where $v_{\min,A}$ and $v_{\max,A}$ are the minimum and maximum speed of the observed vehicle.

The area is first divided into $n_{\text{cell},A}$ grid cells of length L_A and width h_A (see Figure 3.6). A grid cell, which is defined corresponding to a sampling window within the time-space plane, is indeed a time-space rectangular area of the same length as its corresponding sampling window and of a width that is less than or equal to the width of the sampling window. These cells

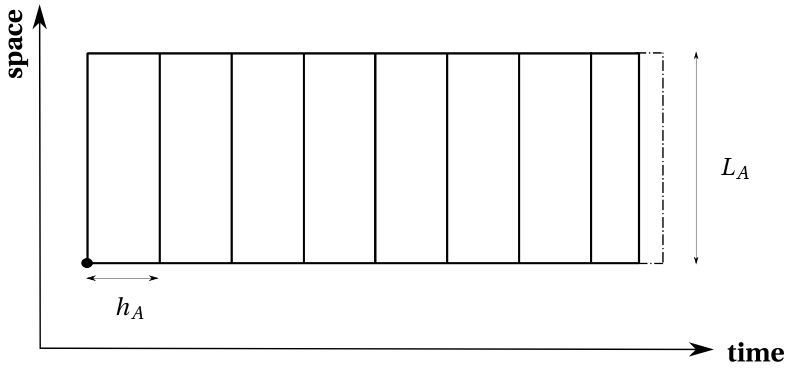


Figure 3.6: Dividing the time-space sampling window into $n_{\text{cell},A}$ grid cells (the dashed-dotted lines show the added part to the original sampling window, which is done to make the dimensions of the last grid cell equal to those of the other grid cells).

are obtained from splitting the original sampling window into $n_{\text{cell},A}$ identical rectangular areas. Note that we assume that at the left bottom corner of each grid cell, one and only one vehicle is located. We have

$$n_{\text{cell},A} := \left\lceil \frac{T_A}{h_A} \right\rceil, \tag{3.5}$$

where $\lceil\cdot\rceil$ denotes the ceiling function. Note that the two parameters $n_{\text{cell},A}$ and N_A are not necessarily equal. However, in the following case they are the same:

$$T_{\text{res},A} < h_A \Rightarrow n_{\text{cell},A} = N_A, \tag{3.6}$$

where $T_{\text{res},A}$ is the time duration from the last observation until the end of the sampling cycle. The above fact comes from

$$N_A = \frac{1}{h_A}\sum_{i=1}^{N_A} h_{A,i-1} = \frac{1}{h_A}(T_A - T_{\text{res},A}) \xRightarrow[\text{since } N_A\in\mathbb{N}]{\text{if } T_{\text{res},A}<h_A,} N_A = \left\lceil \frac{T_A}{h_A} \right\rceil := n_{\text{cell},A}. \tag{3.7}$$

From now, as shown in Figure 3.6, we assume $n_{\text{cell},A} = N_A$ to ease the formulations. Extension of the proposed approach to the case where $n_{\text{cell},A} \neq N_A$ is straightforward.

Next, we introduce a new parameter called the *cell speed*, which depends on the dimensions of the sampling window and is defined by

$$v_{\text{cell},A} = \frac{L_A}{h_A}. \tag{3.8}$$

The cell speed defined for the sampling window A, is indeed the least required speed for traveling the complete length L_A of the road within one grid cell. We also introduce two integers denoted by m_A and M_A with the following physical definitions for the sampling window A. The value of $m_A - 1$ indicates the total number of those vehicles that arrive in the detection zone of the loop detector corresponding to the sampling window A, when it is impossible for them to leave the sampling road section during the current sampling cycle assuming that they are moving by the speed $v_{A,\min}$. Moreover, $M_A - 2$ indicates the total number of vehicles that have arrived in the detection zone of the loop detector during the current sampling cycle so late that even if they keep on moving forward by the speed $v_{A,\max}$, they cannot leave the sampling road section within the current sampling cycle. Hence, we can write

$$m_A = \Big(\text{number of vehicles arriving in the last } \tfrac{L_A}{v_{A,\min}} \text{ time units of current sampling cycle}\Big) + 1, \tag{3.9}$$

$$M_A = \Big(\text{number of vehicles arriving in the last } \tfrac{L_A}{v_{A,\max}} \text{ time units of current sampling cycle}\Big) + 1. \tag{3.10}$$

For the case of equal time headways, these integers are given by

$$m_A - 1 \leq \frac{v_{\text{cell},A}}{v_{\min,A}} < m_A \Leftrightarrow m_A = \left\lfloor \frac{L_A}{h_A v_{\min,A}} \right\rfloor + 1. \tag{3.11}$$

$$M_A - 1 \leq \frac{v_{\text{cell},A}}{v_{\max,A}} < M_A \Leftrightarrow M_A = \left\lfloor \frac{L_A}{h_A v_{\max,A}} \right\rfloor + 1. \tag{3.12}$$

Remark 3.1 The expressions given for m_A and M_A by (3.11) and (3.12) are based on the assumption of $v_{\max,A} < v_{\text{cell},A}$, which in practice is the most possible case to happen. Consequently, the cases $m_A = 1$ and $M_A = 1$ are discarded. In case we have $v_{\max,A} \geq v_{\text{cell},A}$, we propose to use the following equation to compute M_A:

$$M_A - 1 \leq \frac{v_{\max,A}}{v_{\text{cell},A}} < M_A \Rightarrow M_A = \left\lfloor \frac{h_A v_{\max,A}}{L_A} \right\rfloor + 1. \tag{3.13}$$

Moreover, if we also have $v_{\min,A} \geq v_{\text{cell},A}$, then we can use the following equation to determine m_A:

$$m_A - 1 \leq \frac{v_{\min,A}}{v_{\text{cell},A}} < m_A \Rightarrow m_A = \left\lfloor \frac{h_A v_{\min,A}}{L_A} \right\rfloor + 1. \tag{3.14}$$

□

3.3.2 Equal time headway distribution

In Figure 3.7, a sampling window is shown that is divided into $n_{\text{cell},A}$ grid cells. We first consider the case for which the time headways of the vehicles are all the same. Later on, we will expand the approach for arbitrary time headway distribution (see Section 3.3.6. Therefore, at the left bottom corner of each grid cell, one and only one vehicle is located (taking into account the assumption of $h_{A,0} = 0$. In the following discussions, we divide the vehicles into two sets based on the grid cell that they are at. We call these two sets the *first set* and the *second set*. By the first set, we refer to the vehicles in the first $n_{\text{cell},A} - m_A + 1$ grid cells (we indicate the corresponding part of the sampling window A by A_1), and by the second set, we mean the vehicles within the last $m_A - 1$ grid cells (the corresponding part of the sampling window A is denoted by A_2). Furthermore, the second set is also divided into two subsets called the *first*

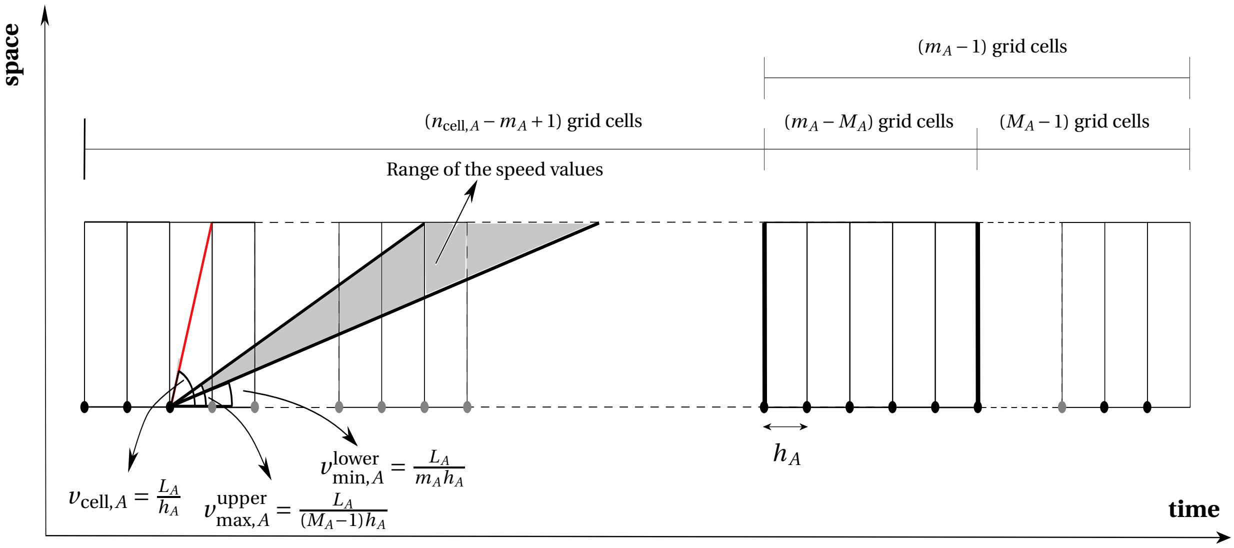

space
$(m_A - 1)$ grid cells
$(n_{\text{cell},A} - m_A + 1)$ grid cells
$(m_A - M_A)$ grid cells
$(M_A - 1)$ grid cells
Range of the speed values
h_A
$v^{\text{lower}}_{\min,A} = \frac{L_A}{m_A h_A}$
$v_{\text{cell},A} = \frac{L_A}{h_A}$
$v^{\text{upper}}_{\max,A} = \frac{L_A}{(M_A - 1)h_A}$
time

Figure 3.7: Dividing the grid cells into three parts in order to find a lower and an upper bound of the TSMS.

subset and the *second subset*, where the first subset includes the first $m_A - M_A + 1$ grid cells of the second set and the second subset includes the remaining $M_A - 2$ grid cells (see Figure 3.7).

From (3.11), we have

$$v_{\min,A} > \frac{v_{\text{cell},A}}{m_A}, \tag{3.15}$$

which, considering the definition of the cell speed, indicates that the vehicles that face m_A or more grid cells in front of them, travel a length greater than or at least equal to L_A during the observation time interval. From Figure 3.7, the vehicle with index $n_{\text{cell},A} - m_A + 1$ and all its predecessors in A (i.e., the first set) fulfills this condition. Consequently, those vehicles that enter the detection zone within the first $(n_{\text{cell},A} - m_A + 1)h_A$ seconds of the observation time interval always leave the sampling window A through its upper edge, while for the vehicles that enter the detection zone in the last $(m_A - 1)h_A$ seconds of the observation time interval, the trajectories *might* intersect the right edge of the sampling window. Therefore, we need to consider the vehicles in the second set more carefully. We expand (2.6) as

$$\bar{v}_A = \frac{\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} L_A + \sum\limits_{i=1}^{m_A-1} t_{A,n_{\text{cell},A}-i+1} \cdot v_{A,n_{\text{cell},A}-i+1}}{\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} \dfrac{L_A}{v_{A,i}} + \sum\limits_{i=1}^{m_A-1} t_{A,n_{\text{cell},A}-i+1}}, \tag{3.16}$$

reminding that $t_{A,i}$ indicates the time spent by the i^{th} vehicle in area A. Next, we determine a lower bound and an upper bound for (3.16). The detailed proofs and explanations are given in Section 3.3.5.

Lower bound

To find a lower bound for $\bar{v}_A$ in (3.16), we consider the case in which all vehicles within the second group move with $v_{\text{min},A}$. The TSMS for such a case is definitely a lower bound for any other possible scenario. For this situation, all vehicles in the second set stay within A until the end of the observation period. Therefore,

$$t_{A,n_{\text{cell},A}-j+1} = jh_A, \qquad j = 1,\, 2,\, 3,\, \ldots,\, m_A - 1. \tag{3.17}$$

For the generalized mean speed given by (3.16), the following holds:

$$\bar{v}_A \geq \frac{\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} L_A + h_A \sum\limits_{i=1}^{m_A-1} i \cdot v_{A,n_{\text{cell},A}-i+1}}{\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} \dfrac{L_A}{v_{A,i}} + h_A \sum\limits_{i=1}^{m_A-1} i}. \tag{3.18}$$

To continue, we find an upper bound for the denominator using the definition of M_A in (3.12). For the first $n_{\text{cell},A} - m_A + 1$ vehicles entering the area, we can write

$$h_A \leq \frac{L_A}{(M_A-1)\,v_{A,i}} \Rightarrow (n_{\text{cell},A} - m_A + 1)h_A \leq \frac{1}{(M_A-1)} \sum_{i=1}^{n_{\text{cell},A}-m_A+1} \frac{L_A}{v_{A,i}}. \tag{3.19}$$

Substituting h_A for $i = 1, 2, \ldots, n_{\text{cell},A} - m_A + 1$ in the denominator of (3.18) by its upper bound from (3.19), and $v_{A,i}$, $i = n_{\text{cell},A} - m_A + 2, \ldots, n_{\text{cell},A}$ in the numerator by the lower bound of $v_{\text{min},A}$ from (3.11) a lower bound for $\bar{v}_A$ is obtained by

$$\bar{v}_A > \frac{\left(\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} L_A\right) + h_A \left(\sum\limits_{i=1}^{m_A-1} i\right) \dfrac{L_A}{m_A h_A}}{\left(\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} \dfrac{L_A}{v_{A,i}}\right) + \dfrac{1}{(M_A-1)(n_{\text{cell},A}-m_A+1)} \left(\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} \dfrac{L_A}{v_{A,i}}\right) \sum\limits_{i=1}^{m_A-1} i}. \tag{3.20}$$

Finally, we obtain a lower bound for $\bar{v}_A$, which is given by

$$\bar{v}_A^{\text{lower}} = \frac{n_{\text{cell},A} - \dfrac{m_A-1}{2}}{(n_{\text{cell},A} - m_A + 1) + \dfrac{m_A-1}{2}\left(\dfrac{m_A}{M_A-1}\right)} H_{A,1\to n_{\text{cell},A}-m_A+1}, \tag{3.21}$$

where $H_{A,1\to n_{\text{cell},A}-m_A+1}$ stands for the harmonic mean of the speeds of those vehicles, which appear in the first $n_{\text{cell},A} - m_A + 1$ grid cells within the time-space plane.

Upper bound

To find an upper bound for $\bar{v}_A$ in (3.16), the case is considered in which all vehicles within the second set move with $v_{\text{max},A}$. The TSMS for any other scenario does not exceed the TSMS for the considered scenario here. From (3.12), we have

$$v_{\max,A} < \frac{v_{\text{cell},A}}{M_A - 1}. \tag{3.22}$$

From (3.22) and Figure 3.7 for those vehicles that are located in the first subset (i.e., in the first $m_A - M_A$ grid cells of the second set) a distance larger than or at least equal to L_A is traveled during the sampling cycle T_A, whereas for vehicles in the second subset (i.e., in the last $M_A - 1$ grid cells of the second set) the traveled distance is less than L_A.

Accordingly, to calculate an upper bound for $\bar{v}_A$, the second term of the distance traveled in the numerator of (3.16) is split into two terms, i.e.,

$$\bar{v}_A < \frac{\left(\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} L_A\right) + (m_A - M_A + 1)L_A + \frac{1}{M_A - 1}\left(\sum\limits_{i=1}^{M_A-1} i\right) L_A}{\left(\sum\limits_{i=1}^{n_{\text{cell},A}-m_A+1} \frac{L_A}{v_{A,i}}\right) + (m_A - M_A + 1)(M_A - 1)h_A + \left(\sum\limits_{i=1}^{M_A-1} i\right) h_A}, \tag{3.23}$$

where the following expressions, which are obtained from Figure 3.7, for the travel time and the traveled distance of vehicles in the last $M_A - 2$ grid cells have been substituted in (3.16). Hence,

$$\begin{aligned} &t_{A,n_{\text{cell},A}-j+1} = j h_A, \\ &d_{A,n_{\text{cell},A}-j+1} = v_{A,n_{\text{cell},A}-j+1} \cdot t_{A,n_{\text{cell},A}-j+1} = \frac{j}{M_A - 1} L_A, \end{aligned} \qquad \text{for} \qquad j = 1, 2, \ldots, M_A - 2.$$

Applying (3.11) to find a lower bound for h_A in the denominator, we finally obtain an upper bound for $\bar{v}_A$, which is given by

$$\bar{v}_A^{\text{upper}} = \frac{n_{\text{cell},A} - \frac{M_A}{2} + 1}{(n_{\text{cell},A} - m_A + 1) + \frac{(M_A - 1)}{2m_A}(2m_A - M_A)} H_{A,1 \to n_{\text{cell},A}-m_A+1}. \tag{3.24}$$

3.3.3 Tightening the lower and upper bounds

In this section, we use the results of (3.21) and (3.24) in Section 3.3.2 with the aim of finding bounds that are very tight. Suppose that we have the datasets given by (3.2) from a loop detector, and therefore, we have the maximum and minimum speeds, $v_{\max,A}$ and $v_{\min,A}$, and the mean time headway, h_A, given by (3.3) and (3.4). From $v_{\max,A}$ and $v_{\min,A}$ we derive the parameters m_A and M_A by (3.11) and (3.12). Equations (3.21) and (3.24) yield a lower and an upper bound for the TSMS. Therefore, we can write

$$\bar{v}_A^{\text{lower}} \leq \bar{v}_A \leq \bar{v}_A^{\text{upper}}. \tag{3.25}$$

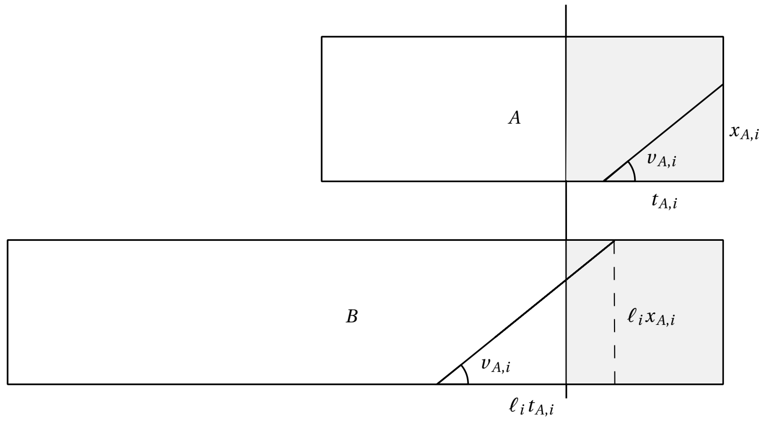


Figure 3.8: Sampling windows A and B with the same datasets (i.e., speed values, number of vehicles, and length of the window), but different time headways (we have $\ell_i > 1$ and $\ell_i x_{A,i} = L_A$).

Now we construct a new sampling window B (see Figure 3.8) with all speed data the same as those of A, but with a different constant time headway, i.e.,

$$h_B = h_A + \Delta h, \tag{3.26}$$

and

$$L_B = L_A, \qquad T_B = n_{\text{cell},A} h_B, \tag{3.27}$$

and we want to find Δh, such that

$$\bar{v}_B^{\text{lower}} \leq \bar{v}_A \leq \bar{v}_B^{\text{upper}}. \tag{3.28}$$

Note that $n_{\text{cell},A} = n_B$. From Figure 3.8 we see that for $\Delta h > 0$ some of the vehicles that are located in the second set for A, might be located in the first set for B. Let W be the number of vehicles from the second set for A that are in the first set for B. Therefore, we have

$$m_B = m_A - W. \tag{3.29}$$

Now we consider the upper and lower bounds separately, so we denote the extended sampling window for the upper bound by B_{u} and for the lower bound by B_{l}. In addition, let W_{u} and W_{l} be the number of vehicles from the second set for A that are, respectively, in the first set for B_{u} and B_{l}.

First we consider the conditions under which the right-hand inequality in (3.28) is satisfied for B_{u}. Now it can be verified (see Section 3.3.5 for the detailed proof) that the upper bound calculated for the sampling window B_{u}, is also an upper bound for $\bar{v}_A$ if we have

$$W_{\text{u}} \leq \min\left\{ m_A \frac{M_A - 2}{M_A - 1}, n_{\text{cell},A} - \frac{M_A}{2} + 1 \right\}. \tag{3.30}$$

Likewise, based on (3.28) we can show that if

$$W_\mathrm{l} \leq m_A - M_A + 1, \tag{3.31}$$

then the lower bound calculated for the sampling window B is also a lower bound for $\bar{v}_A$.

Finally, if we seek for a very tight upper bound for the TSMS, from (3.29) and (3.30), we can write

$$m_{B_\mathrm{u}} = m_A - \min\left\{ m_A \frac{M_A - 2}{M_A - 1}, n_{\mathrm{cell},A} - \frac{M_A}{2} + 1 \right\}.$$

Then we can find h_{B_u} from (3.11), and correspondingly we can calculate M_{B_u} from (3.12). Substituting m_{B_u} and M_{B_u} in (3.24), we have the desired upper bound. Similarly, a desired tight lower bound can be found.

3.3.4 Estimation of the TSMS from the upper and lower bounds

In the previous section, we have determined tight upper and lower bounds for the TSMS (note that the tightened $\bar{v}_A^\mathrm{lower}$ is denoted by $\mathrm{TSMS}_A^\mathrm{lower}$ and the tightened $\bar{v}_A^\mathrm{upper}$ is denoted by $\mathrm{TSMS}_A^\mathrm{upper}$). In this section, we introduce a formula for estimating the TSMS from the introduced upper and lower bounds. For that we propose a convex combination of the lower and upper bounds. Using a convex combination, we can also deal with the cases where the trajectories of vehicles are not necessarily straight lines. The expression for the estimated TSMS, $\mathrm{TSMS}_A^\mathrm{est}$, is proposed as

$$\mathrm{TSMS}_A^\mathrm{est} = \frac{\mathrm{TSMS}_A^\mathrm{lower} + \gamma_A \mathrm{TSMS}_A^\mathrm{upper}}{1 + \gamma_A} \quad \text{with} \quad \gamma_A \geq 0. \tag{3.32}$$

In the above expression γ_A can be obtained from a dataset including individual speeds of the vehicles if such a dataset is initially available for different traffic scenarios.

However, such an extensive dataset might not always be available or there might be a need for immediate application of the formula for a new area (without a priori knowledge about the possible traffic scenarios or any available data). For such cases, a parametric expression for γ_A is more applicable. The following formula, which is a convex combination with the weight being a function of the speed range is proposed for the TSMS:

$$\mathrm{TSMS}_A^\mathrm{est} = \frac{\mathrm{TSMS}_A^\mathrm{lower} + \dfrac{v_{\mathrm{max},A}}{v_{\mathrm{min},A}} \cdot \mathrm{TSMS}_A^\mathrm{upper}}{1 + \dfrac{v_{\mathrm{max},A}}{v_{\mathrm{min},A}}}. \tag{3.33}$$

In (3.33) when the ratio of $v_{\mathrm{max},A}$ and $v_{\mathrm{min},A}$ increases, the share of the upper bound in the estimated TSMS will be higher compared with the share of the lower bound.

Now we explain how to estimate the TSMS from the convex combination given by (3.33) using the tight bounds for the TSMS given by (3.21) and (3.24). We use the bounds W_u and W_l (see (3.30) and (3.31)) for the same sampling window to apply (3.33). Therefore, for the estimation of TSMS we substitute W with the following expression

$$W = \min\{W_{\mathrm{u}}, W_{\mathrm{l}}\}, \tag{3.34}$$

and correspondingly, we determine m_B and M_B and substitute them in (3.21) and (3.24) to obtain $\bar{v}_B^{\mathrm{lower}} = \mathrm{TSMS}_A^{\mathrm{lower}}$ and $\bar{v}_B^{\mathrm{upper}} = \mathrm{TSMS}_A^{\mathrm{upper}}$. Finally, we substitute $\mathrm{TSMS}_A^{\mathrm{lower}}$ and $\mathrm{TSMS}_A^{\mathrm{upper}}$ in (3.33) to find an estimate for $\bar{v}_A$.

3.3.5 Proofs for tightening the lower and the upper bounds

This section contains the extended proof on the method for tightening the lower bound given by (3.21) and the tight upper bound (3.24), which has been presented in Section 3.3.3.

First we consider the upper bound of $\bar{v}_B$ computed within the sampling window B in order to find the conditions under which this upper bound is also an upper bound for $\bar{v}_A$. From (3.24) for the factor that is multiplied by $H_{A,1\rightarrow n_{\mathrm{cell},A}-m_A+1}$, i.e., for

$$\begin{aligned} f(m_A) &= \frac{n_{\mathrm{cell},A} - \dfrac{M_A}{2} + 1}{(n_{\mathrm{cell},A} - m_A + 1) + \dfrac{(M_A - 1)}{2m_A}(2m_A - M_A)}, \\ \text{with:} \quad & M_A = \alpha_A m_A + 1, \qquad 0 < \alpha_A < 1, \end{aligned} \tag{3.35}$$

we can easily show that

$$\frac{\partial f(m_A)}{\partial m_A} = \frac{(2\alpha_A^2 - 6\alpha_A + 4)n_{\mathrm{cell},A} + 2\alpha_A^2 - 4\alpha_A + 2}{4n_{\mathrm{cell},A}^2 + (f_1(\alpha_A)m_A - 4\alpha_A + 8)n_{\mathrm{cell},A} + f_2(\alpha_A)m_A^2 + f_3(\alpha_A)m_A + \alpha_A^2 - 4\alpha_A + 4} \geq 0, \tag{3.36}$$

with

$$f_1(\alpha_A) = -4\alpha_A^2 + 8\alpha_A - 8, \tag{3.37}$$

$$f_2(\alpha_A) = \alpha_A^4 - 4\alpha_A^3 + 8\alpha_A^2 - 8\alpha_A + 4, \tag{3.38}$$

$$f_3(\alpha_A) = 2\alpha_A - 8\alpha_A^2 + 12\alpha_A - 8. \tag{3.39}$$

Therefore, by reducing m_A to m_B (i.e., by increasing h_A to h_B), the factor (3.35) becomes smaller and can produce a tighter upper bound (note that the equality occurs for $\alpha_A = 1$, i.e., for uniform speeds).

Now we should find the extreme/worst case, where $\bar{v}_A$ might violate $\bar{v}_B^{\mathrm{upper}}$. Thus, we aim to increase $\bar{v}_A$ and to decrease $\bar{v}_B^{\mathrm{upper}}$ at the same time in order to produce the worst possible case. For the given A and B, the case where

$$v_{A,j} = v_{\max,A}, \qquad \text{for} \qquad j = n_{\mathrm{cell},A} - m_A + W, \ldots, n_{\mathrm{cell},A}, \tag{3.40}$$

makes $\bar{v}_A$ increased with respect to $\bar{v}_B^{\text{upper}}$, because the speed values of the vehicles in (3.40) do not appear in (3.24) for $\bar{v}_B^{\text{upper}}$, but increases $\bar{v}_A$.

Now we consider the vehicles $v_{A,i}, i = n_{\text{cell},A} - m_A, \ldots, n_{\text{cell},A} - m_A + W - 1$, where these vehicles appear in both the harmonic mean of the first set for B and in $\bar{v}_A$. We first introduce the following lemmas:

Lemma 3.1 *If $v_{A,i} < v_{A,j}$, for $j = 1,2,\ldots,n_{\text{cell},A} - m_A + 1$, then*

$$\mathrm{H}\left(\{v_{A,j} | j = 1,2,\ldots,n_{\text{cell},A} - m_A + 1\} \cup \left\{v_{A,i}\right\}\right) < \mathrm{TSMS}\left(\{v_{A,j} | j = 1,2,\ldots,n_{\text{cell},A} - m_A + 1\} \cup \left\{v_{A,i}\right\}\right),$$

where $\mathrm{H}(\cdot)$ *provides the harmonic mean of its arguments and* $\mathrm{TSMS}(\cdot)$ *gives the TSMS of the speeds given as its arguments.*

Proof: We denote the traveled distance and the travel time of vehicle $v_{A,k}$ for $k = 1,\ldots,n_{\text{cell},A} - m_A + 1, i$ by, respectively, $x_{A,k}$ and $t_{A,k}$ and also $j = 1,2,\ldots,n_{\text{cell},A} - m_A + 1$ by $j \in A_1$. Then, we have

$$\mathrm{TSMS}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right) = \frac{x_{A,1} + \ldots + x_{A,n_{\text{cell},A}-m_A+1} + x_{A,i}}{t_{A,1} + \ldots + t_{A,n_{\text{cell},A}-m_A+1} + t_{A,i}}, \tag{3.41}$$

$$\mathrm{H}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right) = \frac{n_{\text{cell},A} - m_A}{\frac{t_{A,1}}{x_{A,1}} + \ldots + \frac{t_{A,n_{\text{cell},A}-m_A+1}}{x_{A,n_{\text{cell},A}-m_A+1}} + \frac{t_{A,i}}{x_{A,i}}}, \tag{3.42}$$

then

$$\begin{aligned}
&\mathrm{H}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right) < \mathrm{TSMS}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right)\\
&\Leftrightarrow (x_{A,1} - x_{A,i}) \frac{(x_{A,1} t_{A,i} - x_{A,i} t_{A,1})}{x_{A,i}} x_{A,2} \ldots x_{A,n_{\text{cell},A}-m_A+1} +\\
&\vdots\\
&+(x_{A,n_{\text{cell},A}-m_A+1} - x_{A,i}) \frac{(x_{A,n_{\text{cell},A}-m_A+1} t_{A,i} - x_{A,i} t_{A,n_{\text{cell},A}-m_A+1})}{x_{A,i}} x_{A,1} \ldots x_{A,n_{\text{cell},A}-m_A}\\
&+ (x_{A,2} - x_{A,1})(x_{A,2} t_{A,1} - x_{A,1} t_{A,2}) x_{A,3} \ldots x_{A,n_{\text{cell},A}-m_A+1} +\\
&\vdots\\
&+ (x_{A,n_{\text{cell},A}-m_A+1} - x_{A,n_{\text{cell},A}-m_A}) \cdot\\
&\quad (x_{A,n_{\text{cell},A}-m_A+1} t_{A,n_{\text{cell},A}-m_A} - x_{A,n_{\text{cell},A}-m_A} t_{A,n_{\text{cell},A}-m_A+1}) x_{A,1} \ldots x_{A,n_{\text{cell},A}-m_A-1} > 0.
\end{aligned} \tag{3.43}$$

We want to find $v_{A,i}$ such that the left-hand side term in the above inequality is positive. Since

$$x_{A,1} = \ldots = x_{A,n_{\text{cell},A}-m_A+1} = L_A \quad , \quad x_{A,i} \leq L_A.$$

We have

$$x_{A,j} - x_{A,i} \geq 0 \qquad \text{for} \qquad j \in A_1. \tag{3.44}$$

Then if we have

$$\frac{x_{A,j}\,t_{A,i}}{x_{A,i}} - t_{A,j} > 0 \Leftrightarrow v_{A,j} > v_{A,i}, \tag{3.45}$$

then the red term in the left-hand side of (3.43) is definitely positive (note that the red and the black terms vary independently, and $v_{A,i}$ only has effect on the red term).

If we consider

$$v_{A,j} = v_{\min,A}, \qquad \text{for} \qquad j = n_{\text{cell},A} - m_A, \ldots, n_{\text{cell},A} - m_A + W - 1, \tag{3.46}$$

then (3.43) is satisfied while the red term adopts its maximum value. Moreover, the expressions given by (3.44) adopt their maximum possible value since $x_{A,i}$ adopts its minimum. Therefore, the largest difference between the realized value of $\mathrm{H}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right)$ and the value of $\mathrm{TSMS}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right)$ takes place. □

Lemma 3.2 *For any two real-valued and non-negative parameters $N, a \geq 0$ and any two real-valued and positive parameters $D, b > 0$, we have $\frac{a}{b} \geq \frac{N}{D} \Leftrightarrow \frac{N+a}{D+b} \geq \frac{N}{D}$.*

Lemma 3.3 *If for the real-valued and positive parameters $w_1, w_1', w_2, w_2' \geq 0$, we have either of*

C1.

$$w_1 > w_1' \quad \textit{and} \quad w_2 < w_2' \quad \textit{and} \quad \frac{w_1}{w_2} \geq \frac{w_1'}{w_2'}, \tag{3.47}$$

C2.

$$w_1 > w_1' \quad \textit{and} \quad w_2 > w_2' \quad \textit{and} \quad \frac{w_1 + 1}{w_2} \geq \frac{w_1' + 1}{w_2'}, \tag{3.48}$$

C3.

$$w_1 < w_1' \quad \textit{and} \quad w_2 < w_2' \quad \textit{and} \quad \frac{w_1}{w_2 + 1} \geq \frac{w_1'}{w_2' + 1}, \tag{3.49}$$

then, considering the real-valued parameter X within the range $[X_{\min}, X_{\max}]$, we have

$$\frac{X + w_1 X_{\max} + w_2 X_{\min}}{1 + w_1 + w_2} \geq \frac{X + w_1' X_{\max} + w_2' X_{\min}}{1 + w_1' + w_2'}. \tag{3.50}$$

Proof: First, we consider the following definition:

$$D = (X_{\max} - X_{\min}) \underbrace{(w_1 w_2' - w_1' w_2)}_{F_1} + (X_{\max} - X) \underbrace{(w_1 - w_1')}_{F_2} + (X - X_{\min}) \underbrace{(w_2' - w_2)}_{F_3}. \tag{3.51}$$

It is easy to verify that $D \geq 0$. The equivalence of (3.50) and (3.51) being positive is as follows

$$\begin{aligned}
& D \geq 0 \Leftrightarrow \\
& (X_{\max}-X_{\min})(w_1 w_2' - w_1' w_2) + (X_{\max}-X)(w_1 - w_1') + (X - X_{\min})(w_2' - w_2) \geq 0 \Leftrightarrow \\
& w_1 w_2' X_{\max} - w_1 w_2' X_{\min} - w_1' w_2 X_{\max} + w_1' w_2 X_{\min} + \\
& \qquad w_1 X_{\max} - w_1 X - w_1' X_{\max} + w_1' X + \\
& \qquad w_2' X - w_2' X_{\min} - w_2 X + w_2 X_{\min} \geq 0 \Leftrightarrow \\
& w_1 w_2' X_{\max} + w_1' w_2 X_{\min} + w_1 X_{\max} + w_1' X + w_2' X + w_2 X_{\min} + X + w_1' w_1 X_{\max} + w_2' w_2 X_{\min} \\
& \quad \geq w_1' w_2 X_{\max} + w_1 X + w_1' X_{\max} + w_2' X_{\min} + w_2 X + X + w_1' w_1 X_{\max} + w_2' w_2 X_{\min} \Leftrightarrow \\
& (X + w_1 X_{\max} + w_2 X_{\min})(1 + w_1' + w_2') \geq (X + w_1' X_{\max} + w_2' X_{\min})(1 + w_1' + w_2') \\
& \xLeftrightarrow{\text{since } w_1, w_1', w_2, w_2' \geq 0} \frac{X + w_1 X_{\max} + w_2 X_{\min}}{1 + w_1 + w_2} \geq \frac{X + w_1' X_{\max} + w_2' X_{\min}}{1 + w_1' + w_2'}.
\end{aligned} \tag{3.52}$$

All the terms $X_{\max} - X_{\min}$, $X_{\max} - X$, and $X - X_{\min}$ in (3.51) are non-negative. Now we study three cases:

1. Suppose that we have case C1 defined by (3.47). Hence, we already know that F_1, F_2, and F_3 in (3.51) are positive. Then, we know definitely that D is non-negative.

2. Suppose that we have case C2, i.e., (3.48) holds. Considering (3.51) the minimum value of D is obtained when $X = X_{\max}$ and hence the value that is multiplied by the negative factor F_3 adopts its maximum value. Therefore (3.51) reduces to

$$D_{\mathrm{C2}} = (X_{\max} - X_{\min})(w_1 w_2' - w_1' w_2 + w_2' - w_2). \tag{3.53}$$

   Then from (3.48) we can write

$$\begin{aligned}
& \frac{w_1 + 1}{w_2} \geq \frac{w_1' + 1}{w_2'} \Leftrightarrow \\
& (w_1 + 1) w_2' \geq (w_1' + 1) w_2 \Leftrightarrow w_1 w_2' - w_1' w_2 + w_2' - w_2 \geq 0.
\end{aligned} \tag{3.54}$$

   Hence, D_{C2} is definitely non-negative and so is D.

3. Suppose that we have case C3 defined by (3.49). Then the minimum value of D is obtained when $X = X_{\min}$ and the value multiplied by the negative factor F_2 adopts its maximum value. Therefore, (3.51) reduces to

$$D_{\mathrm{C3}} = (X_{\max} - X_{\min})(w_1 w_2' - w_1' w_2 + w_1 - w_1'). \tag{3.55}$$

   Then from (3.49) we can write

$$\begin{aligned}
& \frac{w_1}{w_2 + 1} \geq \frac{w_1'}{w_2' + 1} \Leftrightarrow \\
& w_1 (w_2' + 1) \geq w_1' (w_2 + 1) \Leftrightarrow w_1 w_2' - w_1' w_2 + w_1 - w_1' \geq 0.
\end{aligned} \tag{3.56}$$

Then D_{C3} is definitely non-negative. This concludes the proof. □

Lemma 3.4 *For the real-valued and non-negative parameters* $X, a \geq 0$ *and the real-valued and positive parameters* $Y, b > 0$ *we have* $\frac{X+a}{Y+b} \leq \frac{X}{Y} \Leftrightarrow \frac{a}{b} \leq \frac{X}{Y}$.

Based on Lemma 3.1, Lemma 3.2, and (3.40), we can conclude that

$$\bar{v}_A \geq \mathrm{TSMS}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right), \tag{3.57}$$

and hence,

$$\mathrm{H}\left(\{v_{A,j} | j \in A_1\} \cup \left\{v_{A,i}\right\}\right) \geq \bar{v}_A. \tag{3.58}$$

Thus the worst case occurs if we have both (3.40) and (3.46).

We can write $\bar{v}_A$ and $\bar{v}_B^{\mathrm{upper}}$ for the worst case, as follows. First for $\bar{v}_A$, we have

$$\bar{v}_A = \frac{\sum_{A,x} + v_{\mathrm{min},A}\left[W\frac{L_A}{v_{\mathrm{min},A}} - \frac{W(W+1)}{2}h_A\right] + v_{\mathrm{max},A}\left[(m_A - W - 1)\frac{L_A}{v_{\mathrm{max},A}} - \frac{(M_A-1)(M_A-2)}{2}h_A\right]}{\sum_{A,t} + W\frac{L_A}{v_{\mathrm{min},A}} - \frac{W(W+1)}{2}h_A + (m_A - W - 1)\frac{L_A}{v_{\mathrm{max},A}} - \frac{(M_A-1)(M_A-2)}{2}h_A}, \tag{3.59}$$

where

$$\sum_{A,x} = \sum_{j=1}^{n_{\mathrm{cell},A}-m_A+1} v_{A,j} t_{A,j} \quad , \quad \sum_{A,t} = \sum_{j=1}^{n_{\mathrm{cell},A}-m_A+1} t_{A,j},$$

are the traveled distance and the travel time of the first set of vehicles for the sampling window A. The second term of the denominator corresponds to the travel time of the first W vehicles located in the second set of vehicles for the sampling window A, for which we have

$$\begin{aligned}\sum_{j=n_{\mathrm{cell},A}-m_A+1}^{n_{\mathrm{cell},A}-m_A+W} t_{A,j} &= \left(\frac{L_A}{v_{\mathrm{min},A}} - h_A\right) + \left(\frac{L_A}{v_{\mathrm{min},A}} - 2h_A\right) + \ldots + \left(\frac{L_A}{v_{\mathrm{min},A}} - Wh_A\right) \\ &= W\frac{L_A}{v_{\mathrm{min},A}} - \frac{W(W+1)}{2}h_A,\end{aligned}$$

and the second term of the numerator corresponds to the traveled distance of the first W vehicles located in the second set of vehicles for the sampling window A.

The third term of the denominator corresponds to the travel time of the last $m_A - W - 1$ vehicles located in the second set of vehicles for the sampling window A, for which we have

$$\begin{aligned}\sum_{j=n_{\mathrm{cell},A}-m_A+W+1}^{n_{\mathrm{cell},A}} t_{A,j} &= (m_A - W - 1 - (M_A - 2))\frac{L_A}{v_{\mathrm{max},A}} + \\ &(M_A-2)\left[\left(\frac{L_A}{v_{\mathrm{max},A}} - h_A\right) + \left(\frac{L_A}{v_{\mathrm{max},A}} - 2h_A\right) + \ldots + h_A\right] \\ &= (m_A - W - 1)\frac{L_A}{v_{\mathrm{max},A}} - \frac{(M_A-1)(M_A-2)}{2}h_A,\end{aligned} \tag{3.60}$$

and the third term of the numerator corresponds to the traveled distance of the last $m_A - W - 1$ vehicles located in the second set of vehicles for the sampling window A.

Now we consider the upper bound of $\bar{v}_B$ for the sampling window B. Here we know that the first W vehicles in the second set of vehicles for the sampling window A are located in the first set of vehicles for the sampling window B. Extension of the time headway (or the width of the sampling window) might also have an effect on the number of vehicles in the second subset of B_2 (i.e., the last $M_B - 2$ vehicles in the second set of vehicles for the sampling window B). Therefore, in general we consider W' vehicles that were in the second subset of A_2 for A, are now out of this subset for B. Consequently, we obtain

$$M_B = M_A - W'.$$

For the upper bound for B from (3.23), we can write

$$\bar{v}_B^{\text{upper}} = \frac{\sum_x + v_{\min,A} \dfrac{WL_A}{v_{\min,A}} + v_{\max,A} \left[\dfrac{(m_A - W - 1)L_A}{v_{\max,A}} - \dfrac{(M_A - W' - 1)(M_A - W' - 2)}{2} (h_A - \Delta h_A) \right]}{\sum_{A,t} + \dfrac{WL_A}{v_{\min,A}} + \dfrac{(m_A - W - 1)L_A}{v_{\max,A}} - \dfrac{(M_A - W' - 1)(M_A - W' - 2)}{2} (h_A - \Delta h)}, \tag{3.61}$$

where the following summations in the numerator and denominator of $\bar{v}_B^{\text{upper}}$ represent the traveled distance and the travel time of the first set of vehicles for the sampling window B, respectively.

Traveled distance of the vehicles in B_1: $\sum_{A,x} + v_{\min,A} \frac{WL_A}{v_{\min,A}}$.

Travel time of the vehicles in B_1: $\sum_{A,t} + \frac{WL_A}{v_{\min,A}}$.

Also the following two terms are the traveled distance and the travel time of the second set of vehicles for the sampling window B.

Traveled distance of the vehicles in B_2:

$$v_{\max,A} \left[\frac{(m_A - W - 1)L_A}{v_{\max,A}} - \frac{(M_A - W' - 1)(M_A - W' - 2)}{2} (h_A - \Delta h) \right].$$

Travel time of the vehicles in B_2:

$$\frac{(m_A - W - 1)L_A}{v_{\max,A}} - \frac{(M_A - W' - 1)(M_A - W' - 2)}{2} (h_A - \Delta h).$$

We apply Lemma 3.3 to see what conditions are needed for (3.50) in order to be able to apply it to (3.59) and (3.61). Equations (3.59) and (3.61) can be rewritten as

$$\bar{v}_A = \frac{\text{TSMS}\left(\{v_{A,j} | j \in A_1\}\right) \sum_{A,t} + c'_1 \cdot v_{\max,A} + c'_2 \cdot v_{\min,A}}{\sum_{A,t} + c'_1 + c'_2}, \tag{3.62}$$

where $c'_1 = \frac{(m_A - W - 1)L_A}{v_{\max,A}} - \frac{(M_A - 1)(M_A - 2)}{2} h_A$ and $c'_2 = \frac{WL_A}{v_{\min,A}} - \frac{W(W+1)}{2} h_A$.

Next we normalize the coefficients with respect to $\sum_{A,t}$,

$$\bar{v}_A = \frac{\text{TSMS}\left(\{v_{A,j} | j \in A_1\}\right) + \frac{c_1'}{\sum_{A,t}} \cdot v_{\max,A} + \frac{c_2'}{\sum_{A,t}} \cdot v_{\min,A}}{1 + \underbrace{\frac{c_1'}{\sum_{A,t}}}_{w_1'} + \underbrace{\frac{c_2'}{\sum_{A,t}}}_{w_2'}}. \tag{3.63}$$

Similarly,

$$\bar{v}_B^{\text{upper}} = \frac{\text{TSMS}\left(\{v_{A,j} | j \in A_1\}\right) + \frac{c_1}{\sum_{A,t}} \cdot v_{\max,A} + \frac{c_2}{\sum_{A,t}} \cdot v_{\min,A}}{1 + \underbrace{\frac{c_1}{\sum_{A,t}}}_{w_1} + \underbrace{\frac{c_2}{\sum_{A,t}}}_{w_2}}, \tag{3.64}$$

with $c_1 = \frac{(m_A - W - 1)L_A}{v_{\max,A}} - \frac{(M_A - W' - 1)(M_A - W' - 2)}{2}(h_A - \Delta h_A)$ and $c_2 = \frac{W L_A}{v_{\min,A}}$.

Therefore, case C2 of the lemma 3.3 holds for (3.63) and (3.64) if we have

$$\frac{\overbrace{(m_A - W - 1)\frac{L_A}{v_{\max,A}} - \frac{(M_A - W' - 1)(M_A - W' - 2)}{2} h_A + (n_{\text{cell},A} - m_A + 1)(M_A - 1) h_A}^{N'} + \frac{(M_A - W' - 1)(M_A - W' - 2)}{2} \Delta h}{\underbrace{W \frac{L_A}{v_{\min,A}} - \frac{W(W+1)}{2} h_A}_{D} + \frac{W(W+1)}{2} h_A}$$

$$\geq \frac{\overbrace{(m_A - W - 1)\frac{L_A}{v_{\max,A}} - \frac{(M_A - 1)(M_A - 2)}{2} h_A + (n_{\text{cell},A} - m_A + 1)(M_A - 1) h_A}^{N}}{\overbrace{W \frac{L_A}{v_{\min,A}} - \frac{W(W+1)}{2} h_A}^{D}}. \tag{3.65}$$

Note that from Lemma 3.3 the extreme case where the inequality given by (3.50) might be violated takes place for $X = X_{\max}$ or equivalently

$$\text{TSMS}\left(\{v_{A,j} | j \in A_1\}\right) = v_{\max,A} \Rightarrow \sum_{A,t} = (n_{\text{cell},A} - m_A + 1)(M_A - 1) h_A.$$

Next we use Lemma 3.2, where for (3.65),

$$a = \frac{(M_A - W' - 1)(M - W' - 2)}{2} \Delta h + \frac{W'(2M_A - W' - 3)}{2} h_A,$$
$$b = \frac{W(W+1)}{2} h.$$

Since $W' < M_A - 2$, then $W' = M_A - 3$ makes a minimal. Now, if the following relationship holds

$$\frac{\frac{(M_A-(M_A-3)-1)(M_A-(M_A-3)-2)}{2}\Delta h + \frac{(M_A-3)(2M_A-(M_A-3)-3)}{2}h_A}{\frac{\cancel{W}(W+1)}{2}\cancel{h_A}} \geq \frac{(m_A-W-1)(M_A-1)h_A - \frac{(M_A-1)(M_A-2)}{2}h_A + (n_{\text{cell},A}-m_A+1)(M_A-1)h_A}{\cancel{W}m_A\cancel{h_A} - \frac{\cancel{W}(W+1)}{2}\cancel{h_A}}, \tag{3.66}$$

which can be simplified to the following expression:

$$\Delta h \geq \frac{n_{\text{cell},A}M_AW - M_AW^2 - n_{\text{cell},A}W^2 - n_{\text{cell},A}W - M_A^2m_A - \frac{1}{2}M_AW}{2m_A - W - 1}h_A. \tag{3.67}$$

In addition, we have

$$(m_A - W)(h_A + \Delta h) = m_Ah_A \Rightarrow \Delta h_A = \frac{W}{m_A - W}h_A. \tag{3.68}$$

Then by substituting (3.68) in (3.67), we finally get

$$\boxed{W \geq \frac{n_{\text{cell},A}(M_A-1)}{n_{\text{cell},A}+M_A},} \tag{3.69}$$

and if we also have

$$\begin{gathered}(m_A-W-1)(M_A-1)h_A - \frac{(M_A-1)(M_A-2)}{2}h_A + (n_{\text{cell},A}-m_A+1)(M_A-1)h_A \geq 0 \Leftrightarrow \\ W \leq n_{\text{cell},A} - \frac{M_A}{2} + 1,\end{gathered} \tag{3.70}$$

then based on Lemma 3.2, (3.61) is an upper bound for (3.59).

In addition, we know that in (3.61), the vehicles in the second subset of B compensate for the reduction of $\bar{v}_B^{\text{upper}}$ due to the W last vehicles in the first set of vehicles for the sampling window B, which move with $v_{\min,A}$. Therefore, we need to make sure that

$$h_B \leq (M_A - 1)h_A. \tag{3.71}$$

Then for the extreme case, we have

$$m_A - W \geq \frac{m_Ah_A}{(M_A-1)h_A} \Rightarrow W \leq m_A\frac{(M_A-2)}{M_A-1}.$$

Hence, the following holds:

$$\boxed{W \leq \min\left\{m_A \frac{M_A - 2}{M_A - 1}, n_{\text{cell},A} - \frac{M_A}{2} + 1\right\}.} \tag{3.72}$$

Finally, we need to select W such that both (3.69) and (3.72) are satisfied at the same time. Afterwards, from W we can obtain m_B using (3.29), and then from (3.11) we have h_B. Hence, we can obtain M_B from (3.12). Then using m_B and M_B, $\text{TSMS}_B^{\text{upper}}$ is calculated by (3.24). The obtained value is an upper bound for $\bar{v}_A$.

Now, we consider the lower bound of $\bar{v}_B$ corresponding to the sampling window B, and we seek for conditions under which this bound is also a lower bound for $\bar{v}_A$. With a similar reasoning as we had for the upper bound, here the extreme case where $\bar{v}_A$ might become equal to or less than $\bar{v}_B^{\text{lower}}$ is when we have

$$v_{A,j} = v_{\min,A}, \quad \text{for} \quad j = n_{\text{cell},A} - m_A + W, \ldots, n_{\text{cell},A}. \tag{3.73}$$

Additionally, the worst case corresponds to the situation where the W vehicles (that are located in the second set of A, but in the first set of B) all move with $v_{\max,A}$, i.e.,

$$v_{A,j} = v_{\max,A}, \quad \text{for} \quad j = n_{\text{cell},A} - m_A, \ldots, n_{\text{cell},A} - m_A + W - 1. \tag{3.74}$$

Finally, for $\bar{v}_A$ and $\bar{v}_B^{\text{lower}}$ (from (3.18)), we have

$$\bar{v}_A = \frac{\sum_{A,x} + v_{\max,A}\left[\frac{WL_A}{v_{\max,A}} - \frac{(W-m_A+M_A-1)(W-m_A+M_A)h_A}{2}\right] + v_{\min,A}\frac{(m_A-W-1)(m_A-W)h_A}{2}}{\sum_{A,t} + \frac{WL_A}{v_{\max,A}} - \frac{(W-m_A+M_A-1)(W-m_A+M_A)h_A}{2} + \frac{(m_A-W-1)(m_A-W)h_A}{2}}, \tag{3.75}$$

$$\bar{v}_B^{\text{lower}} = \frac{\sum_{A,x} + v_{\max,A}\frac{WL_A}{v_{\max,A}} + v_{\min,A}\frac{(m_A-W-1)(m_A-W)}{2}(h_A+\Delta h)}{\sum_{A,t} + \frac{WL_A}{v_{\max,A}} + \frac{(m_A-W-1)(m_A-W)}{2}(h_A+\Delta h)}. \tag{3.76}$$

The second term of the numerator of (3.75) corresponds to the first W vehicles in the second set of A. From (3.74) these move with $v_{\max,A}$. Hence, the travel time of the first W vehicles in the 2^{nd} set of A is

$$\underbrace{\frac{L_A}{v_{\max,A}}(m_A - M_A + 1)}_{\text{vehicles in set 2, but outside of subset 2 of } A_2} + \underbrace{\left[\left(\frac{L_A}{v_{\max,A}} - h_A\right) + \ldots + \left(\frac{L_A}{v_{\max,A}} - (W - m_A + M_A - 1)h_A\right)\right]}_{\text{vehicles in subset 2 of } A}. \tag{3.77}$$

For (3.75) the third term of the numerator is corresponding to the last $m_A - W - 1$ vehicles in the second set of A, i.e., the vehicles given by (3.73). We know that the travel time of $v_{A,n_{\text{cell},A}}$ is h_A, for $v_{A,n_{\text{cell},A}-1}$ it is $2h_A$, and so on. Therefore, we have the following statement.
Travel time of the last $m_A - W - 1$ vehicles in the 2^{nd} set for the sampling window A:

$$h_A + 2h_A + \ldots + (m_A - W - 1)h_A = \frac{(m_A - W - 1)(m_A - W)h_A}{2}.$$

For (3.76) the second term corresponds to the W vehicles that are in the second set of A, but in the first set of B. The third term corresponds to the vehicles in the second set for the sampling window B. Moreover, the following statement holds.
Travel time of the vehicles in the 2^{nd} set for the sampling window B:

$$(h_A + \Delta h) + 2(h_A + \Delta h) + \ldots + (m_A - W - 1)(h_A + \Delta h).$$

At the end, comparing (3.75) and (3.76) with the cases given in Lemma 3.3, case C3 is applicable here. Therefore, if we have

$$\frac{\dfrac{WL_A}{v_{\max,A}} - \dfrac{(W - m_A + M_A - 1)(W - m_A + M_A)}{2}h_A}{\dfrac{(m_A - W - 1)(m_A - W)}{2}h_A + {\color{red}(n_{\text{cell},A} - m_A + 1)m_A h_A}} \geq \frac{\dfrac{WL_A}{v_{\max,A}} - {\color{blue}\dfrac{(W - m_A + M_A - 1)(W - m_A + M_A)}{2}h_A} + {\color{blue}\dfrac{(W - m_A + M_A - 1)(W - m_A + M_A)}{2}h_A}}{\dfrac{(m_A - W - 1)(m_A - W)}{2}h_A + {\color{red}(n_{\text{cell},A} - m_A + 1)m_A h_A} + \dfrac{(m_A - W - 1)(m_A - W)}{2}\Delta h}. \tag{3.78}$$

Next we apply the Lemma 3.4. We have

$$\frac{\dfrac{WL_A}{v_{\max,A}} - \dfrac{(W - m_A + M_A - 1)(W - m_A + M_A)}{2}h_A}{\dfrac{(m_A - W - 1)(m_A - W)}{2}h_A + {\color{red}(n_{\text{cell},A} - m_A + 1)m_A h_A}} \geq \frac{\dfrac{(W - m_A + M_A - 1)(W - m_A + M_A)}{2}h_A}{\dfrac{(m_A - W - 1)(m_A - W)}{2}\Delta h}, \tag{3.79}$$

which reduces to

$$\Delta h \geq \frac{(W - m_A + M_A - 1)(W - m_A + M_A)\left[(W + m_A)(W - m_A + 1) + 2m_A(n_{\text{cell},A} - W)\right]}{(m_A - W - 1)(m_A - W)[-W^2 + (2m - 1)W + (M^2 - 2mM + M^2 - M + m)]}h_A, \tag{3.80}$$

and if in addition to (3.80), we have

$$2W(M_A - 1) - (W - m_A + M_A - 1)(W - m_A + M_A) \geq 0, \tag{3.81}$$

which reduces to

$$W \leq m_A + \sqrt{(2m_A - M_A)(M_A - 1)} - \frac{1}{2}, \tag{3.82}$$

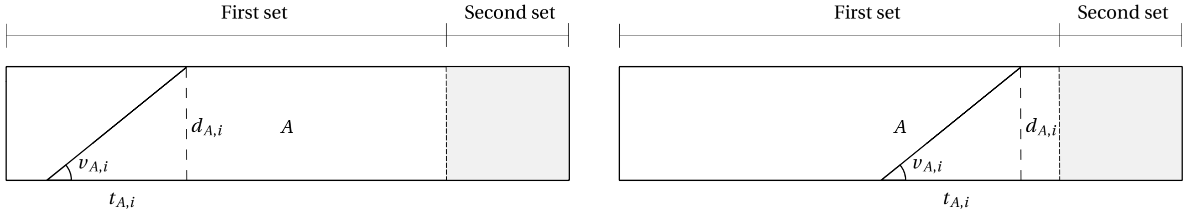


Figure 3.9: Considering a vehicle that is originally in the first set for the sampling window A, and shifting its trajectory to the left (shown in the left-hand side figure) or to the right (shown in the right-hand side figure) does not change the traveled distance and the travel time of the vehicle as long as it stays in the first set.

then we can make sure that based on case C3 of Lemma 3.3 that

$$\text{TSMS}_B^{\text{lower}} \le \bar{v}_A.$$

Then similar to what we explained for the upper bound before, we have m_B and M_B and correspondingly $\text{TSMS}_B^{\text{lower}}$ from (3.21), which can be used as a lower bound for $\bar{v}_A$.
Note that a simple proposed range for W from (3.80) and (3.82) is the following, which indeed satisfies both conditions:

$$\boxed{W \le m_A - M_A + 1.} \tag{3.83}$$

3.3.6 Arbitrary time headway distribution

In Section 3.3.2, we have developed equations for finding a lower and an upper bound for the TSMS assuming that all the vehicles have an equal time headway. In this section, we generalize our proposed approach such that the formulas given in Section 3.3.2, and correspondingly the tight upper and lower bounds and the estimated TSMS determined in Sections 3.3.3 and 3.3.4 can be used for a general case, i.e., even if the time headways of different vehicles are not necessarily the same. We will also show that the assumption of $h_{A,0} = 0$, which has been used in Section 3.3.2 to explain the proposed approach in a simpler way, can easily be relaxed, while the proposed formulas remain valid.

First, we show that (3.21) and (3.24) are valid for an arbitrary distribution of the time headways. We first consider those vehicles that are in the first set of the sampling window A (i.e., they are able to leave the sampling road section by the end of the current sampling cycle, even if they move forward by the speed $v_{A,\text{min}}$). For these vehicles, if the time headway is changed from h_A (which is the case considered in Figure 3.7 and also in extracting the formulas given in Sections 3.3.2, 3.3.3, and 3.3.4), then it actually means that the corresponding trajectories of these vehicles are moved to the left or to the right in the sampling window A within the time-space plane. However, as long as these vehicles still stay in the first set, their effect on the TSMS does not change as their traveled distance and their travel time are unchanged (see Figure 3.9).

Next, we start by bringing the new vehicles one by one to either the first or the second set

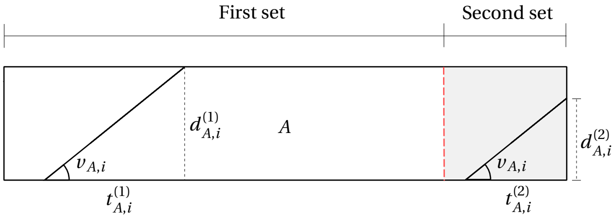


Figure 3.10: Illustration of the traveled distance and the travel time of a vehicle in two cases, i.e., if the vehicle belongs to the first set of the sampling window A, and if the vehicle belongs to the second set of the sampling window A.

of the sampling window A. For the first vehicle, if we add it to the first set, i.e., if we bring its trajectory to the left-hand side of the red dashed splitting line shown in Figure 3.10, the traveled distance and the travel time of the vehicle are indicated by $d_{A,i}^{(1)}$ and $t_{A,i}^{(1)}$, respectively. If we add this vehicle to the second set of vehicles, i.e., if we bring the trajectory of this vehicle to the right-hand side of the red dashed splitting line shown in Figure 3.10, and indicate the traveled distance and the travel time of the vehicle by, respectively, $d_{A,i}^{(2)}$ and $t_{A,i}^{(2)}$, then $d_{A,i}^{(1)} > d_{A,i}^{(2)}$ and $t_{A,i}^{(1)} > t_{A,i}^{(2)}$ hold (see Figure 3.10). We also use $T^{(1)}$ and $D^{(1)}$ for the total traveled distance and the total travel time of the vehicles in the first set.

Suppose that the added vehicle moves with $v_{A,\min}$. Since we have $v_{A,\min} \leq \frac{D^{(1)}}{T^{(1)}}$, hence $v_{A,\min} T^{(1)} \leq D^{(1)}$ (note that $D^{(1)}$, $T^{(1)}$, and $v_{A,\min}$ are positive quantities). Since $t_{A,i}^{(1)} - t_{A,i}^{(2)} > 0$, we can multiply both sides of the resulting inequality by $t_{A,i}^{(1)} - t_{A,i}^{(2)}$, i.e.,

$$v_{A,\min} T^{(1)} \left(t_{A,i}^{(1)} - t_{A,i}^{(2)}\right) \leq D^{(1)} \left(t_{A,i}^{(1)} - t_{A,i}^{(2)}\right).$$

Next we add $v_{A,\min} t_{A,i}^{(1)} t_{A,i}^{(2)} + D^{(1)} T^{(1)}$ to both sides of the resulting inequality. We have

$$v_{A,\min} T^{(1)} \left(t_{A,i}^{(1)} - t_{A,i}^{(2)}\right) + v_{A,\min} t_{A,i}^{(1)} t_{A,i}^{(2)} + D^{(1)} T^{(1)} \leq D^{(1)} \left(t_{A,i}^{(1)} - t_{A,i}^{(2)}\right) + v_{A,\min} t_{A,i}^{(1)} t_{A,i}^{(2)} + D^{(1)} T^{(1)}.$$

The speed of the added vehicle is assumed to be $v_{A,\min}$. Hence, we can substitute $v_{A,\min} t_{A,i}^{(1)}$ by $d_{A,i}^{(1)}$ and $v_{A,\min} t_{A,i}^{(2)}$ by $d_{A,i}^{(2)}$. We obtain

$$T^{(1)} \left(d_{A,i}^{(1)} - d_{A,i}^{(2)}\right) + d_{A,i}^{(1)} t_{A,i}^{(2)} + D^{(1)} T^{(1)} \leq D^{(1)} \left(t_{A,i}^{(1)} - t_{A,i}^{(2)}\right) + d_{A,i}^{(2)} t_{A,i}^{(1)} + D^{(1)} T^{(1)},$$

which reduces to

$$\left(D^{(1)} + d_{A,i}^{(1)}\right) \left(T^{(1)} + t_{A,i}^{(2)}\right) \leq \left(D^{(1)} + d_{A,i}^{(2)}\right) \left(T^{(1)} + t_{A,i}^{(1)}\right).$$

Then we have

$$\frac{D^{(1)} + d_{A,i}^{(1)}}{T^{(1)} + t_{A,i}^{(1)}} \leq \frac{D^{(1)} + d_{A,i}^{(2)}}{T^{(1)} + t_{A,i}^{(2)}},$$

which indicates that in order to find the lower bound $\bar{v}_A^{\text{lower}}$ for the TSMS, we should add the vehicle to the first set. The same reasoning holds if we add the second, third, and ... vehicles one by one. Hence, $\bar{v}_A^{\text{lower}}$ is realized when all the trajectories of the vehicles are added to the first set. Taking into account the physical definition of the integer m_A given in Section 3.3.1, for the case explained above, we have $m_A - 1 = 0$. From (3.21), if we substitute $m_A = 1$, we obtain $\bar{v}_A^{\text{lower}} = H_{A,1\to n_{\text{cell},A}-m_A+1}$, which is true since all the vehicles are now added to the first set.

Additionally, we can show in a similar way that if the added vehicle moves forward by the speed $v_{A,\max}$, then we have

$$\frac{D^{(1)} + d_{A,i}^{(1)}}{T^{(1)} + t_{A,i}^{(1)}} > \frac{D^{(1)} + d_{A,i}^{(2)}}{T^{(1)} + t_{A,i}^{(2)}},$$

which indicates that for $\bar{v}_A^{\text{upper}}$ to be realized, all the vehicles should be in the first set. From the physical definition of the integers m_A and M_A given in Section 3.3.1, the case described above results in $M_A - 2 = 0$ and also $m_A - 1 = 0$, which together with (3.23) gives $\bar{v}_A^{\text{upper}} = H_{A,1\to n_{\text{cell},A}-m_A+1}$. This is again true, as all the vehicles have been added to the first set for the sampling window A.

Finally, considering the definitions given for m_A and M_A by (3.9) and (3.10), for any arbitrary case with a randomly distributed time headway for the vehicles, we can still use the formulas developed earlier on in this chapter. Additionally, now that we have shown that by changing the individual time headways $h_{A,i}$ for $i \in \{0, \ldots, N_A - 1\}$, the formulas developed in this chapter can still be used, we can conclude that this also holds for $h_{A,0}$, i.e., the assumption $h_{A,0} = 0$ made before can easily be relaxed and the formulas will still hold.

3.4 Case study

In this section we present the results for using the formulas given by Wardrop [153], i.e., (2.13), and by Rakha and Zhang [134], i.e., (2.17), and the new formula given in this chapter, i.e., (3.33), for estimation of the TSMS. We first use real-life data from the NGSIM dataset for the I-80 freeway (http://gateway.path.berkeley.edu/ngsimdocs/I-80/) and the Rotterdam-Delft dataset for the A13 freeway (http://data.3tu.nl/repository/collection:traffic_flow_obs) to determine the relative errors of these formulas with respect to the real TSMS. As a comparison, we also present the relative error of the harmonic mean of the speeds with respect to the TSMS.

The NGSIM dataset provides detailed information on trajectories of vehicles and has been generated as part of the Next Generation SIMulation (NGSIM) project by the Federal Highway Administration from a segment of the interstate freeway I-80 in San Francisco, California, US, on April 13, 2005. The data has been collected via seven video cameras from 2.00 PM till 7.00 PM, and is available in 3 sets for the time periods between 4.00 PM and 4.15 PM, between 5.00 PM and 5.15 PM, and between 5.15 PM and 5.30 PM. In addition to that, real-life data for the traffic flow on the A13 freeway, Rotterdam-Delft, extracted from movies that were captured by a helicopter is used. Figure 3.11 shows the map of the A13 freeway where the

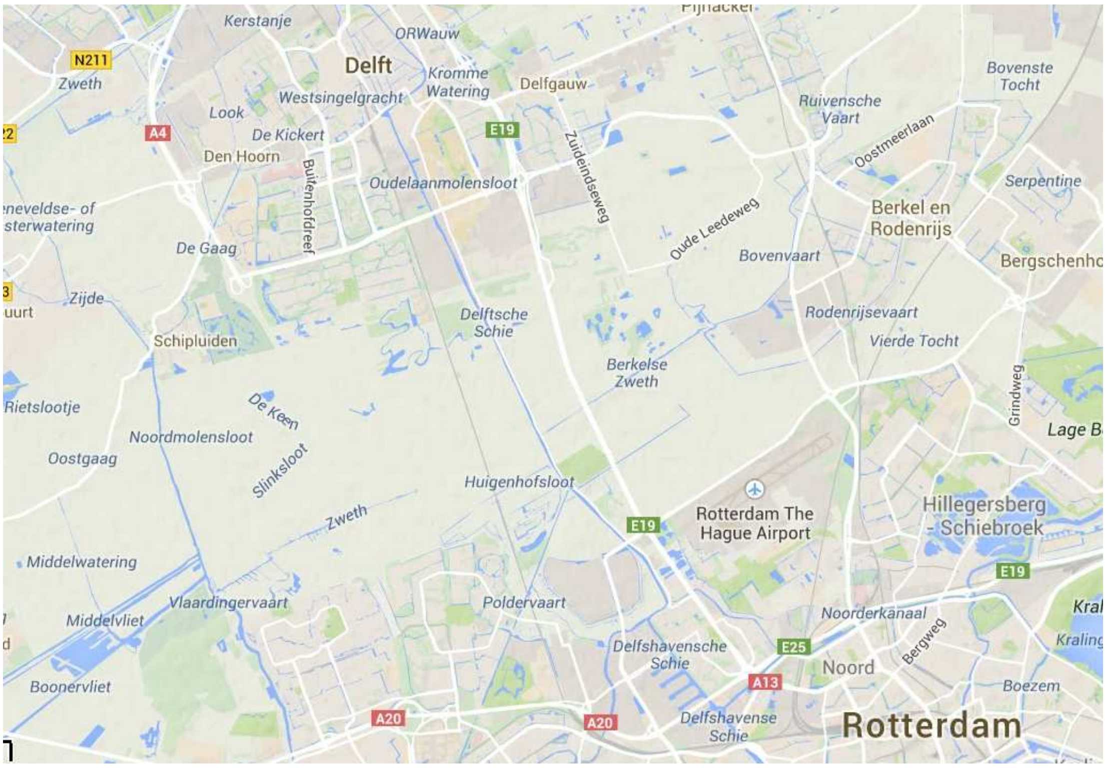


Figure 3.11: Map displaying the A13 freeway, Rotterdam-Delft for which the real-life flow data is available.

data has been collected. The movies and the corresponding point speed data are provided by 4TU.ResearchData, the joint center for research data of Delft University of Technology, Eindhoven University of Technology, University of Twente, and University of Wageningen.

Furthermore, to cover the possible scenarios that are not included in the real-life datasets, we also include simulations in MATLAB at the end of this section.

3.4.1 Real-life data (NGSIM, I-80 and Rotterdam-Delft, A13)

On a road section, we might have a single scenario or a combination of different traffic scenarios. For cases involving a multi-lane road where lane-changing is also permitted, there are two possible ways of estimating the average speed: the first considers the average speed separately for each lane, and the other considers multiple lanes together and estimates their overall average speed. Due to the lane changes, a measurement on one lane might affect the speed on the other lane(s), which is a good reason for considering the collection of lanes. For the real-life data in this section we therefore consider this joined case for the lanes.

Figure 3.12 demonstrates the relative error of the convex combination, the formulas by Rakha and Zhang [134] and by Wardrop [153], and the harmonic mean of the speeds with respect to the real TSMS for the NGSIM I-80 dataset. We have extracted four sub-datasets from the NGSIM I-80 dataset, where the corresponding quantities for L_A, h_A, T_A, $v_{\min,A}$, and

Table 3.2: The values of the traffic parameters for the NGSIM (I-80 freeway) dataset.

Dataset	L_A(m)	h_A(s)	T_A (min)	$v_{\min,A}$ ($\frac{\text{m}}{\text{s}}$)	$v_{\max,A}$ ($\frac{\text{m}}{\text{s}}$)
DS 1	500	0.34	102.00	10.00	34.50
DS 2	500	0.34	102.56	18.25	33.09
DS 3	500	0.34	102.65	11.33	32.87
DS 4	500	0.34	102.36	17.96	30.85

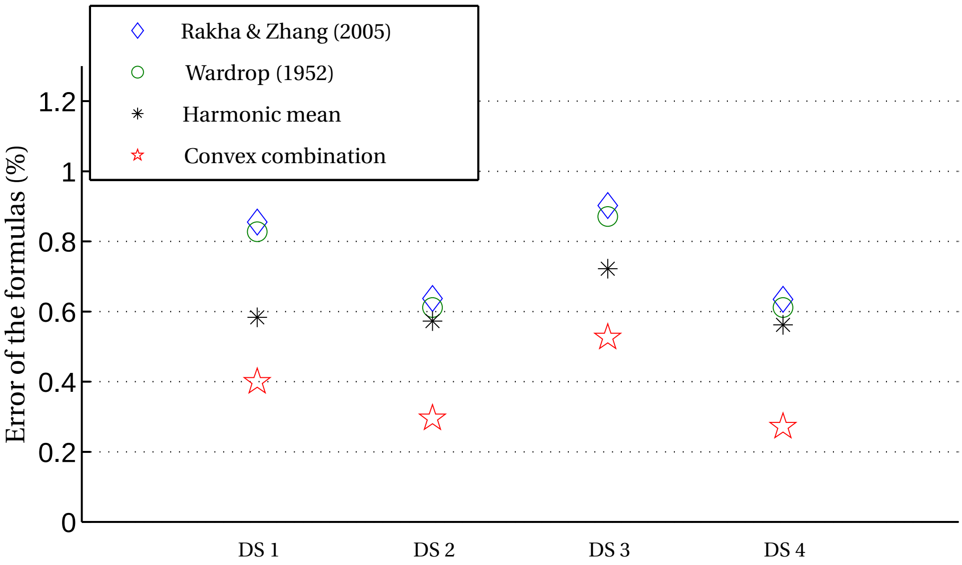


Figure 3.12: The percentage of errors for the formulas by Rakha and Zhang [134], by Wardrop [153], the harmonic mean of the individual speeds, and the convex combination developed in this chapter for the NGSIM (I-80 freeway) dataset.

$v_{\max,A}$ are given in Table 3.2. From the relative errors shown in Figure 3.12 we see that the convex combination (3.33) has the best performance.

Next we performed the calculations for estimation of the TSMS for the eight datasets that are extracted from the Rotterdam-Delft A13 real-life data. The values for L_A, h_A, T_A, $v_{\min,A}$, and $v_{\max,A}$ are given in Table 3.3.

Figure 3.13 demonstrates the relative error of the convex combination, the formulas by Rakha and Zhang [134] and by Wardrop [153], and the harmonic mean of the speeds with respect to the real TSMS for the Rotterdam Delft A13 real-life dataset. From Figure 3.13 for most datasets the convex combination shows the best performance. Among the datasets, only for DS 9 and DS 10 the harmonic mean and the formula by Rakha and Zhang [134] respectively, produce smaller errors.

Table 3.3: The values of the traffic parameters for the Rotterdam-Delft (A13 freeway) dataset.

Dataset	L_A(m)	h_A(s)	T_A (min)	$v_{\min,A}$ ($\frac{m}{s}$)	$v_{\max,A}$ ($\frac{m}{s}$)
DS 5	300	0.81	183.00	6.82	38.94
DS 6	300	1.07	204.00	12.24	35.30
DS 7	300	1.01	220.00	13.33	34.24
DS 8	300	0.87	192.00	10.30	35.15
DS 9	300	0.92	213.00	12.42	39.55
DS 10	300	0.99	192.00	12.88	35.15
DS 11	300	0.93	387.00	6.81	38.94
DS 12	300	1.94	400.00	13.63	40.60

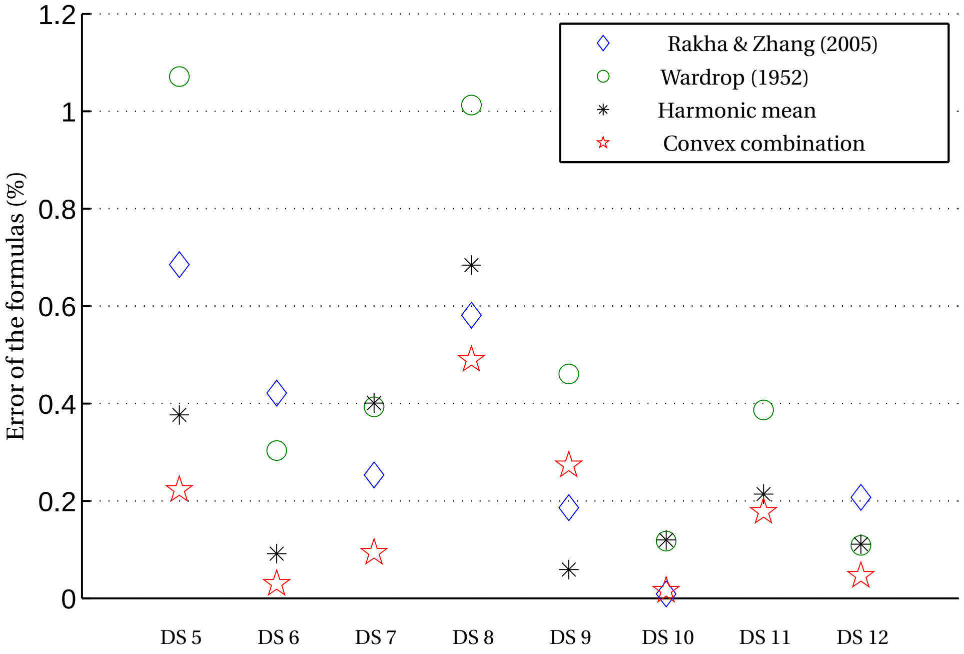


Figure 3.13: The percentage of errors for the formulas by Rakha and Zhang [134], by Wardrop [153], the harmonic mean of the individual speeds, and the convex combination developed in this chapter for the Rotterdam-Delft (A13 highway) dataset.

3.4.2 MATLAB simulations

In this section we present some simulations using MATLAB in order to consider the scenarios that are not covered by the real-life datasets given in Section 3.4.1. Using MATLAB simulations we can repeat the experiments for different ranges of h_A, $v_{\min,A}$, and $v_{\max,A}$ as many times as is desired, and construct the boxplots of the errors being produced by different averaging formulas. Such a boxplot shows the shape of the distribution of the errors, its central value, and variability. The produced picture consists of the most extreme values in the dataset, the lower and upper quartiles, and the median.

Table 3.4: Classification of speed ranges for flow scenarios.

Traffic scenario	**Range of speed** ($\frac{\text{m}}{\text{s}}$)
Over-saturated (Breakdown)	< 15.5
Queue discharge	$15.5 - 25.0$
Under-saturated (Free flow)	≥ 25.0

Table 3.5: The values of the traffic variables for different traffic scenarios.

Scenario	L_A(m)	h_A(s)	T_A (min)	$v_{\text{min},A}$ ($\frac{\text{m}}{\text{s}}$)	$v_{\text{max},A}$ ($\frac{\text{m}}{\text{s}}$)
Over-saturated	500	2.0	5	2.0	15.5
Queue discharge	500	1.0	5	15.5	25.0
Under-saturated	500	0.5	5	25.0	40.0

We produce the set of individual speeds using a normal[1] distribution. In order to produce the individual speeds, we applied the standard deviations given by Huey et al. [61]. To find the real TSMS using (2.6), the trajectories of the vehicles are assumed to be straight lines, an assumption which is confirmed by the trajectories obtained from the NGSIM and the Rotterdam-Delft data. Since we consider loop detectors, we use $L_A = 500$ m, which is the prevalent distance between two consecutive loop detectors in most parts of Europe, and also in many areas of the US.

We consider each of the scenarios given in Table 3.4, which is based on [57]. For the mean time headway of the vehicles, h_A, we can apply the results provided by Zou et al. [163] and Brackstone et al. [19], where the correlation between the speed and the time headway is studied. Based on the results of these papers, we have selected h_A for each scenario as is shown in Table 3.5.

The results for MATLAB simulations are given in Figure 3.14. From this figure we see that in general the errors by all formulas reduce when the flow scenario changes from the over-saturated to the under-saturated case. In all cases the convex combination shows the best performance. The difference in the performance is more highlighted for the over-saturated case where the best formula is clearly the convex combination. In Figure 3.14(c) the errors for the convex combination and the harmonic mean are negligible.

3.5 Conclusions and future work

We have developed tight upper and lower bounds for the TSMS, which is an equivalent for the generalized average speed defined by Edie [43], using microscopic traffic point measurements. Then we have introduced a convex combination of the upper and lower bounds such that an appropriate estimate of the time-space-mean speed is obtained.

[1] According to May [111] one of the most commonly used mathematical distributions for representing the measured speed values is the normal distribution.

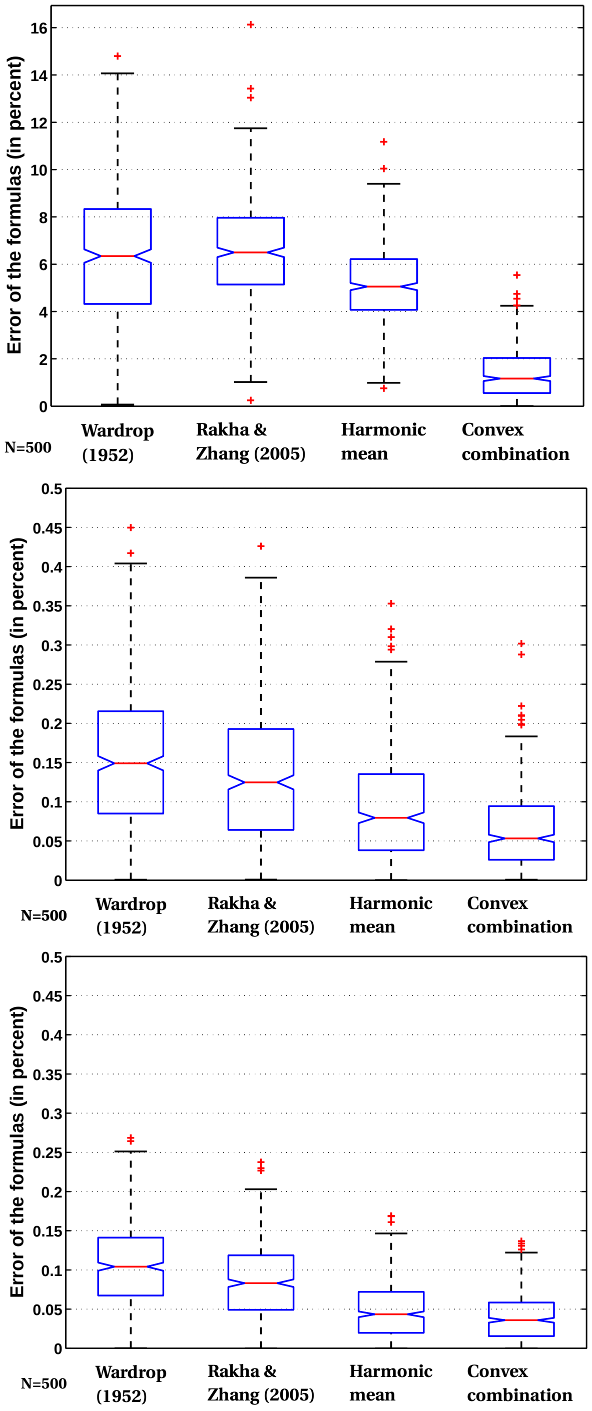


Figure 3.14: The box plots for the percentage of errors for the formulas by Wardrop [153], and by Rakha and Zhang [134], the harmonic mean of the individual speeds, and the convex combination developed in this chapter for MATLAB simulations, for $N = 500$ *tests for the over-saturated (top figure), queue discharge (middle figure), and under-saturated (bottom figure) scenarios.*

In order to assess and to compare performance of the different formulas, we have applied real-life traffic datasets, the NGSIM dataset for the I-80 freeway (US) and the Rotterdam-Delft dataset for the A13 freeway (The Netherlands). We have also included a limited number of MATLAB simulations to consider some of the traffic scenarios that are not covered by the real-life datasets. This way the three scenarios under-saturated, queue discharge, and over-saturated flow are covered. The results from both the real-life data and from the MATLAB simulations show that the values produced by the new formula give better estimates of the TSMS compared with the formulas by Wardrop [153] and by Rakha and Zhang [134], and the pure harmonic mean of the individual speeds.

For future work, the approach proposed in this chapter can be expanded such that the trajectories of the vehicles are approximated in a more accurate way, and not by straight lines only. A car-following model together with a lane-changing model may help us to approximate these trajectories in a more accurate way.

Chapter 4

Sequential estimation algorithm for estimating generalized fundamental traffic variables

4.1 Introduction

This chapter proposes a new efficient sequential algorithm for estimating the generalized traffic variables, including the generalized average speed or the TSMS, using point measurements. The algorithm takes into account those vehicles that stay in between two consecutive measurement points for more than one sampling cycle and that are not detected in the meantime. The relative position of these vehicles w.r.t. the beginning point of the sampling road section is computed and stored by the algorithm to be used as initial conditions for the next sampling cycle. Consequently, we distinguish two main groups of vehicles per sampling cycle: those vehicles that are detected at the beginning of the sampling road section within the current sampling cycle are put in group 1, and those vehicles that have already been on the sampling road section and their relative position is now used as an initial condition by the sequential algorithm, are stored in group 2. The algorithm is introduced for single-lane roads first, and then is extended to multi-lane roads. The sequential algorithm is developed based on the assumption that vehicles will keep their original speed (i.e., the speed that has been detected by the loop detector at the beginning of the sampling road section) while they have not reached the next loop detector. Then the speed would immediately change to that detected by the next loop detector.

In order to extend the sequential algorithm, and to make a more realistic and smooth trajectory for each vehicle within the time-space plane that does not show sudden changes, in Section 4.4 we use Newell's car-following model to propose an approach for approximating the trajectory of each vehicle in between two successive measurement points. In this section, we briefly explain the main idea behind our proposed approach. The formulas for estimation of the generalized fundamental traffic variables can then be developed using the same trend and logic as in the previous sections. The proposed approach is inspired by the one given by Coifman [31]. The main advantage of the proposed method is that the trajectory of the leading vehicle can be estimated without any need to capture measurements from the following vehicle at some time into the future. We propose a model for the movement of the leading vehicle of a traffic stream that is based on the assumption that the leading vehicle intends to reach the free-flow speed as soon as possible. Then, we just need to know the initial speed of the leading vehicle in a sampling cycle in order to approximate its trajectory, where this initial speed is either measured at a measurement point, or is computed by the proposed sequential algorithm in case the leading vehicle is located in between two consecutive measurement points at the beginning of the sampling time. The trajectories of the following vehicles are then found using Newell's car-following model. Hence, the trajectory of a leading vehicle is extracted independently from that of the following vehicle, and this makes our proposed methodology suitable for online applications.

For evaluation of the proposed sequential algorithm, the NGSIM dataset, which provides detailed information on trajectories of the vehicles on a segment of the interstate freeway I-80 in San Francisco, California has been used. The simulation results illustrate the excellent performance of the sequential procedure for estimating the generalized traffic variables compared with previous approaches. The sequential algorithm proposed in this chapter is suitable for upgrading or reprogramming the inductive loop detectors for a more accurate estimation of the generalized fundamental traffic variables.

Contributions and organization of the chapter

The main contributions of this chapter include the following topics:

1. We propose a new efficient method for estimation of the generalized fundamental traffic variables from point measurements. The proposed approach is suitable especially for cases where there is missing, disrupted, or limited information about the vehicles in between two consecutive measurement points.

2. We propose two methods to deal with the missing or disrupted information in between two measurement points. For the two methods, vehicles in between two consecutive measurement points follow two paradigms: constant speed and non-constant speed. The non-constant speed case is inspired by the approach proposed by Coifman [31], but we have changed the method so that it does not need the future information of a leading vehicle to produce the approximate trajectory of the following vehicle at the current time instant. Therefore, our approach can also be used for estimation of the trajectory of vehicles in between two consecutive measurement points in real time.

3. We develop a new sequential algorithm that takes into account those vehicles on the sampling road section that are not detected by the loop detector at the beginning of the road section during the current sampling cycle. These are vehicles that are already on the sampling road section at the beginning of the current sampling cycle. The proposed sequential algorithm keeps track of the detected vehicles from the time instant they are observed at one measurement point, until they reach the next measurement point.

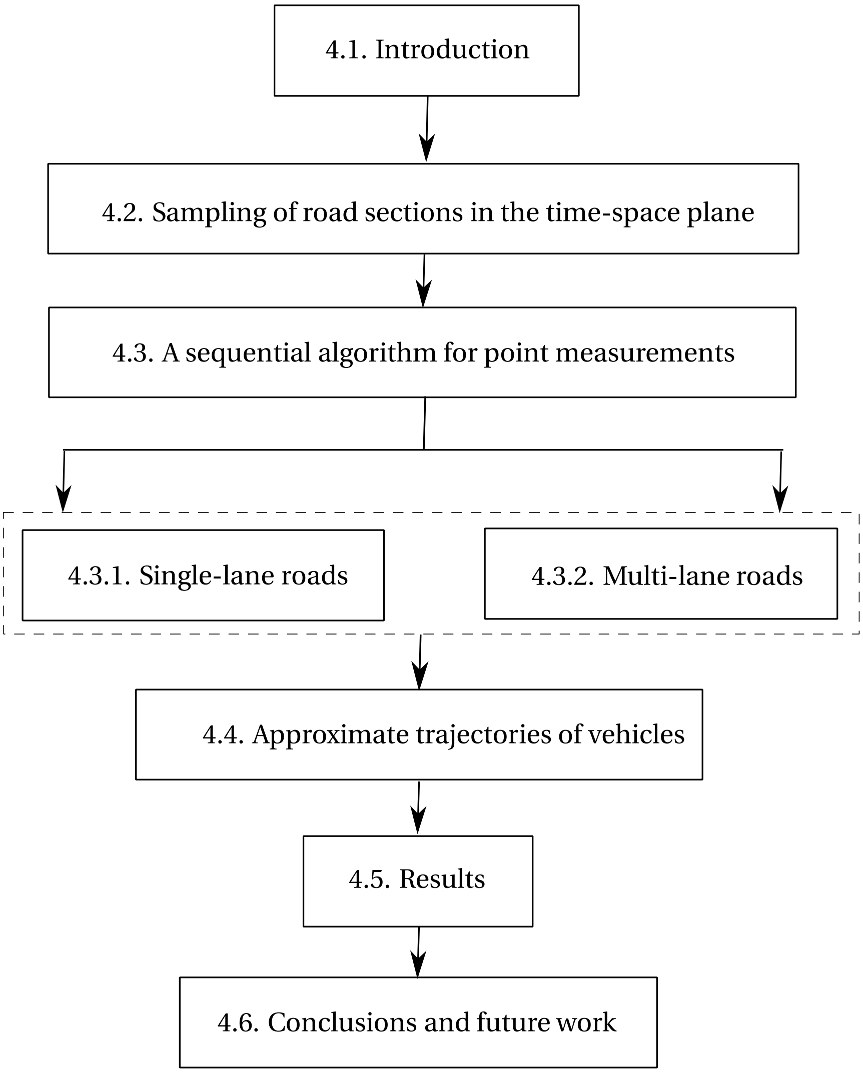

4.1. Introduction
4.2. Sampling of road sections in the time-space plane
4.3. A sequential algorithm for point measurements
4.3.1. Single-lane roads
4.3.2. Multi-lane roads
4.4. Approximate trajectories of vehicles
4.5. Results
4.6. Conclusions and future work

Figure 4.1: Road map of Chapter 4.

Hence, compared with previous work that does not take these vehicles into account, our new approach can produce more accurate results.

4. We show via real-life NGSIM data (captured on a segment of the interstate freeway I-80 in San Francisco, California, US) that the proposed approach produces more accurate results compared with the available approaches.

The rest of the chapter is organized as follows. In Section 4.2 we propose a representation of the sampling road section for which we want to find the fundamental traffic variables, in the time-space plane. We also introduce the concept of sampling window, which is a rectangular time-space area representing the road section within a specific time interval. Section 4.3 introduces the new sequential algorithm that keeps track of the vehicles that are detected by inductive loop detectors, and produces the generalized density, flow, and average speed of vehicles. In Section 4.3.1 first the algorithm is developed for single-lane roads assuming a constant speed and hence a linear trajectory for all the vehicles in between of two consecutive loop detectors. Next, in Section 4.3.2 the algorithm is extended to multi-lane roads. Section 4.4 explains how to find approximate trajectories of the vehicles that are traveling on a sampling road section in between of two successive loop detectors relaxing the assumption of having linear trajectories, i.e., in this section the possible accelerating and decelerating behaviors of the vehicles are also considered. In Section 4.5 we present the results of a case study using NGSIM data in order to assess the efficiency of the proposed sequential algorithm compared with formulas available in the literature. Finally, the chapter is concluded in Section 4.6 and suggestions for future work are proposed. A road map of the chapter is illustrated in Figure 4.1. In addition to that, Table 4.1 presents the frequently used mathematical notations used within the chapter.

This chapter of the thesis is based on [64, 65].

4.2 Sampling of road sections in the time-space plane

Consider a single-lane road of length L^{road} with n^{loop} loop detectors D_j, $j \in \{1, 2, \ldots, n^{\text{loop}}\}$ installed at positions x_j w.r.t. the beginning point of the road. We assume that the first loop detector D_1 is located at the beginning point of the road, and that the endpoint of the road has the position x^{endpoint}. This road will be processed within the time interval $\left[\bar{t}^{\text{initial}}, \bar{t}^{\text{final}}\right]$ of length T^{road}, i.e., the fundamental traffic variables in between each two consecutive loop detectors are determined for the sampling cycles in between $\bar{t}^{\text{initial}}$ and $\bar{t}^{\text{final}}$. Suppose that for $j \in \{1, 2, \ldots, n^{\text{loop}}\}$, loop detector D_j has a sampling cycle T_j, where the time interval $\left[\bar{t}^{\text{initial}}, \bar{t}^{\text{final}}\right]$ includes n_j^{cyc} sampling cycles of this loop detector. The section of the road of length L_j and beginning at x_j (i.e., D_j is installed at the beginning of the road section) is called a *sampling road section*, which is indexed by j. We first explain how to illustrate a sampling road section by a sampling window within the time-space plane (note that since our focus is on the computation of the generalized fundamental traffic variables and these variables are temporal-spatial ones, we should first map the problem into the time-space plane).

To present and process traffic data, trajectories of the vehicles can be plotted in the time-space plane [149], where the processing time is shown on the horizontal axis, and the processed length of the road is shown on the vertical axis (see Figure 4.2). The representation of the road of length L^{road} in the time-space plane (see the main frame in Figure 4.2) is a rectangular frame of length L^{road} (in between x_1 and x^{endpoint} along the space axis) and of width T^{road} (in between $\bar{t}^{\text{initial}}$ and $\bar{t}^{\text{final}}$ along the time axis). Note that for the single-lane road, the space axis is along the direction of movement of the vehicles. Then sampling road section j, $j \in \{1, \ldots, n^{\text{loop}}\}$ in the time-space plane is illustrated by a window of length L_j (in between x_j and x_{j+1} along the space axis) and of width T_j (in between $\bar{t}_{i-1,j}$ and $\bar{t}_{i,j}$ along the time axis) called a *sampling window* that is indexed by the time step counter i and by index j of the loop detector located at its beginning point. Figure 4.2 illustrates the time-space sampling windows with a length of L_j and a width of T_j for $j \in \{1, \ldots, n^{\text{loop}}\}$ corresponding to the road. For a sampling window $\text{w}_{i,j}$, with $j \in \{1, 2, \ldots, n^{\text{loop}}\}$ and $i \in \{1, 2, \ldots, n_j^{\text{cyc}}\}$, the right, left, top, and bottom edges of the sampling window are denoted by, respectively, $\mathrm{E}|_{i,j}$, $|\mathrm{E}_{i,j}$, $\overline{\mathrm{E}}_{i,j}$, and $\underline{\mathrm{E}}_{i,j}$ (see Figure 4.2). The trajectories of those vehicles that are observed by loop detector D_j during the current sampling cycle will intersect the lower edge $\underline{\mathrm{E}}_{i,j}$ of the sampling window.

Next it should be checked whether any of the trajectories of the vehicles observed within the area of $\mathrm{w}_{i,j}$ enter $\mathrm{w}_{i,j}$ via the left-hand edge $|\mathrm{E}_{i,j}$ of the window. The vehicles that correspond to these trajectories will spend some time on the sampling road section j and therefore, will affect the average speed and the average density of the sampling window $\mathrm{w}_{i,j}$. However, since they will not pass through the detection zone of loop detector D_j within the current sampling cycle, they will not be detected. Therefore, we should keep track of them from the previous cycle(s), and then use their speed and location at $\bar{t}_{i-1,j}$ as the initial conditions for the current sampling cycle.

The main question here is whether we need to check all previous windows $\mathrm{w}_{a,b}$, for $a = 1, 2, \ldots, i-1$ and $b = 1, 2, \ldots, j$ to process $\mathrm{w}_{i,j}$, or whether it is possible to reduce the effort. Considering the rectangles in the same row of the grid shown in Figure 4.2 (e.g., the grey windows for processing $\mathrm{w}_{i,j}$) will be sufficient, because the trajectories of all vehicles that enter $\mathrm{w}_{i,j}$ and have previously traveled in the sampling windows located in lower rows of the grid (i.e., with $x_b < x_j$), should definitely cross the line $x = x_j$ before they enter $\mathrm{w}_{i,j}$ and hence they will once be observed by loop detector D_j. However, the vehicles that are located at position x_c at $\bar{t}_{i,j}$ with $x_j < x_c < x_{j+1}$, i.e., vehicles that were (partly) traveling in the same road section during the previous cycle, will not be detected by the upstream loop detector. To keep track of these vehicles, we should consider the sampling window located in the left-hand side of $\mathrm{w}_{i,j}$ in the same row, i.e., $\mathrm{w}_{i-1,j}$ (see Figure 4.2); and similarly for $\mathrm{w}_{i-1,j}$ we should keep track of the information in the sampling window $\mathrm{w}_{i-2,j}$, and so on. Consequently, we consider separate time indicators $\bar{t}_{i,j}$, $i \in \{1, \ldots, n_j^{\mathrm{cyc}}\}$ for each row of the sampling windows in Figure 4.2.

Note that in this chapter, we define four groups of vehicles for every sampling window $\mathrm{w}_{i,j}$: $G_{1,i,j}$, $G_{2,i,j}$, $G_{3,i,j}$, and $G_{4,i,j}$ referring to, respectively, the group of vehicles that enter the sampling road section j during cycle i, the group of vehicles that are already in the sampling road section j at the beginning of cycle i, the group of vehicles that leave the sampling road section by the end of cycle i, and the group of vehicles that will stay on the sampling road section at the end of cycle i.

4.3 A sequential algorithm for point measurements to keep track of all vehicles

4.3.1 Single-lane roads

All discussions presented in this section are based on the following assumptions:

Assumption 1. The first loop detector on the processed road is located at the beginning point of the road.

Assumption 2. The road is considered to have only a single lane (this assumption will later on be relaxed in Section 4.3.2).

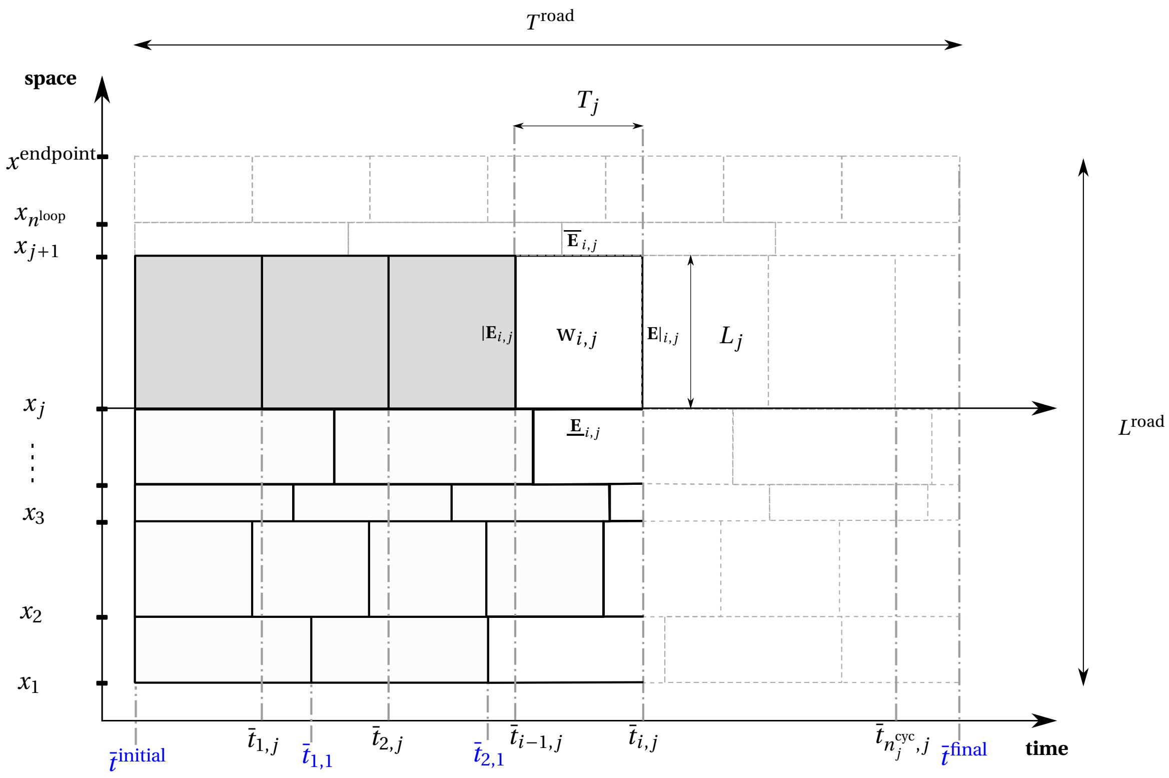


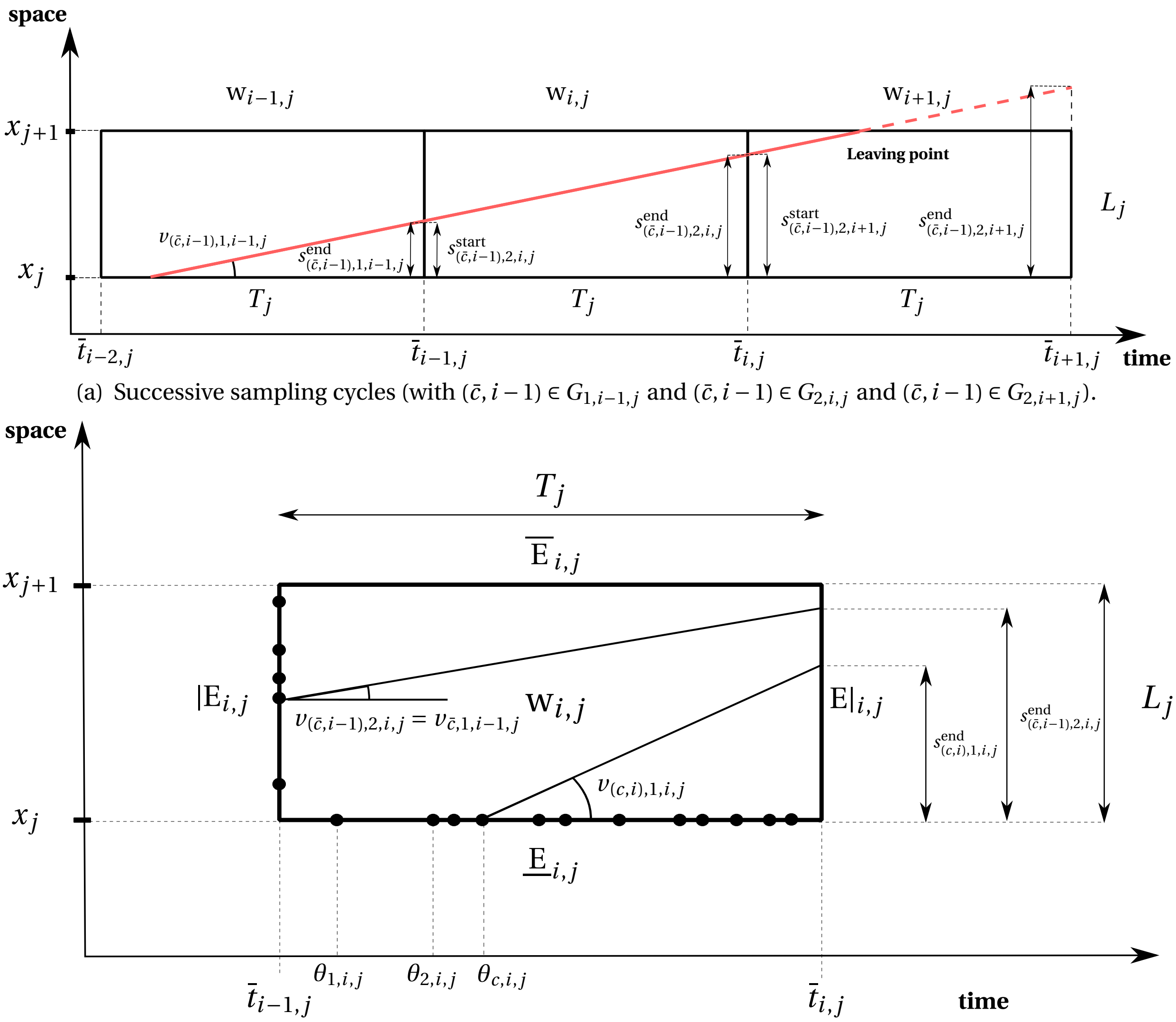
Figure 4.2: The time-space plot for the sampling road sections on a road of length L^{road} being processed during a total time of T^{road} using data from inductive loop detectors.

Assumption 3. All vehicles travel with a constant speed (reported by the upstream loop detector) on each sampling road section, until a new speed value is detected for them by the downstream loop detector (this assumption will be relaxed later on in Section 4.4).

For the work presented in this chapter, the assumption of stationary traffic conditions is not required.

Figure 4.3(a) illustrates three sampling windows $\mathrm{w}_{i-1,j}$, $\mathrm{w}_{i,j}$, and $\mathrm{w}_{i+1,j}$ in the time-space plane that represent sampling road section j during three successive sampling cycles of length T_j starting at $\bar{t}_{i-2,j}$, $\bar{t}_{i-1,j}$, and $\bar{t}_{i,j}$. Figure 4.3(b) illustrates $\mathrm{w}_{i,j}$ and different groups of vehicles that are observed in this window. Note that for each sampling road section j, each vehicle is indicated by a pair[1] (c, ι_c). The first index c denotes the number of the vehicle in the set of vehicles that are detected by the loop detector at the beginning of the sampling road section. The vehicle keeps this index until it leaves the sampling road section, and it reaches the next loop detector on the road. By using the index c only, a vehicle is not completely distinguished from the rest of the vehicles. Hence, the second index ι_c is also kept for the vehicle, which indicates the number of the sampling cycle at which the vehicle has entered the sampling road section. The pair of indices corresponding to those vehicles for which the trajectories pass

[1] Since the proposed sequential algorithm computes the fundamental traffic variables on a particular sampling road section j, for the sake of simplicity in the notations, we have used c instead of c_j. Note that c may vary on different sampling road sections.


space
x_{j+1}
x_j
$\mathrm{w}_{i-1,j}$
$\mathrm{w}_{i,j}$
$\mathrm{w}_{i+1,j}$
Leaving point
$v_{(\bar{c},i-1),1,i-1,j}$
$s^{\mathrm{end}}_{(\bar{c},i-1),1,i-1,j}$
$s^{\mathrm{start}}_{(\bar{c},i-1),2,i,j}$
$s^{\mathrm{end}}_{(\bar{c},i-1),2,i,j}$
$s^{\mathrm{start}}_{(\bar{c},i-1),2,i+1,j}$
$s^{\mathrm{end}}_{(\bar{c},i-1),2,i+1,j}$
L_j
T_j
$\bar{t}_{i-2,j}$
$\bar{t}_{i-1,j}$
$\bar{t}_{i,j}$
$\bar{t}_{i+1,j}$
time

(a) Successive sampling cycles (with $(\bar{c}, i-1) \in G_{1,i-1,j}$ and $(\bar{c}, i-1) \in G_{2,i,j}$ and $(\bar{c}, i-1) \in G_{2,i+1,j}$).

space
T_j
$\overline{\mathrm{E}}_{i,j}$
x_{j+1}
x_j
$|\mathrm{E}_{i,j}$
$v_{(\bar{c},i-1),2,i,j} = v_{\bar{c},1,i-1,j}$
$\mathrm{w}_{i,j}$
$\mathrm{E}|_{i,j}$
$v_{(c,i),1,i,j}$
$s^{\mathrm{end}}_{(c,i),1,i,j}$
$s^{\mathrm{end}}_{(\bar{c},i-1),2,i,j}$
L_j
$\underline{\mathrm{E}}_{i,j}$
$\theta_{1,i,j}$
$\theta_{2,i,j}$
$\theta_{c,i,j}$
$\bar{t}_{i-1,j}$
$\bar{t}_{i,j}$
time

(b) Sampling window $\mathrm{w}_{i,j}$ and different groups of vehicles observed (with $(c,i) \in G_{1,i,j}$ and $(\bar{c}, i-1) \in G_{2,i,j}$).

Figure 4.3: Sampling windows corresponding to the same road section during successive sampling cycles.

through $\underline{\mathrm{E}}_{i,j}$ are stored in $G_{1,i,j}$, and the pair of indices corresponding to those vehicles for which the trajectories intersect $|\mathrm{E}_{i,j}$ are stored in $G_{2,i,j}$. From Figure 4.3(a), for any vehicle $(c,i) \in G_{1,i,j}$ in the sampling window $\mathrm{w}_{i,j}$, we obtain

Case 1. If $s^{\mathrm{end}}_{(c,i),1,i,j} \geq L_j$, then the vehicle leaves $\mathrm{w}_{i,j}$ through its upper edge and will not enter $\mathrm{w}_{i+1,j}$. Indices of those vehicles in $G_{1,i,j}$ that satisfy this condition are stored in subgroup $G^{(1)}_{1,i,j}$ of $G_{1,i,j}$.

Case 2. If $s^{\mathrm{end}}_{(c,i),1,i,j} < L_j$, then the vehicle leaves $\mathrm{w}_{i,j}$ through its right-hand edge, and enters $\mathrm{w}_{i+1,j}$ through its left-hand edge. Indices of those vehicles that satisfy this condition are stored in subgroup $G^{(2)}_{1,i,j}$ of $G_{1,i,j}$.

Similarly, for any vehicle $(\bar{c},\bar{\imath}) \in G_{2,i,j}$ in the sampling window $\mathrm{w}_{i,j}$, where $\bar{\imath} \in \{1,\ldots,i-1\}$, we have

Case 3. If $s^{\text{end}}_{(\bar{c},\bar{\imath}),2,i,j} \geq L_j$, then the vehicle leaves $\mathrm{w}_{i,j}$ through its upper edge and will not enter $\mathrm{w}_{i+1,j}$. Indices of those vehicles in $G_{2,i,j}$ that satisfy this condition are stored in subgroup $G^{(1)}_{2,i,j}$ of $G_{2,i,j}$.

Case 4. If $s^{\text{end}}_{(\bar{c},\bar{\imath}),2,i,j} < L_j$, then the vehicle leaves $\mathrm{w}_{i,j}$ through its right-hand edge, and enters $\mathrm{w}_{i+1,j}$ through its left-hand edge. Indices of those vehicles that satisfy this condition are stored in subgroup $G^{(2)}_{2,i,j}$ of $G_{2,i,j}$.

Therefore, $G^{(1)}_{1,i,j}$ and $G^{(1)}_{2,i,j}$ will not play any role in the sampling window $\mathrm{w}_{i+1,j}$, while $G^{(2)}_{1,i,j}$ and $G^{(2)}_{2,i,j}$ will form a set indicated by $G_{0,i,j}$ (the index zero is chosen to show that the end position of those vehicles for which the indices are stored in set $G_{0,i,j}$ will be used as the initial conditions for the next sampling cycle). The set $G_{0,i,j}$ is indeed the same as the set $G_{2,i+1,j}$, which will be used in the sampling window $\mathrm{w}_{i+1,j}$.

From Figure 4.3(b), the total travel time of a vehicle $(c,i) \in G_{1,i,j}$ within one cycle time is obtained by

$$t_{(c,i),1,i,j} = T_j + \bar{t}_{i-1,j} - \theta_{c,i,j}. \tag{4.1}$$

The total traveled distance by vehicle (c,i) during one sampling cycle equals

$$d_{(c,i),1,i,j} = v_{(c,i),1,i,j} \cdot t_{(c,i),1,i,j}, \tag{4.2}$$

and for the relative position of (c,i) at $\bar{t}_{i-1,j}$ and at $\bar{t}_{i,j}$ w.r.t. $x = x_j$,

$$s^{\text{start}}_{(c,i),1,i,j} = 0, \tag{4.3}$$

$$s^{\text{end}}_{(c,i),1,i,j} = d_{(c,i),1,i,j}. \tag{4.4}$$

The total travel time of a vehicle $(\bar{c},\bar{\imath}) \in G_{2,i,j}$ during one sampling cycle is given by

$$t_{(\bar{c},\bar{\imath}),2,i,j} = T_j. \tag{4.5}$$

The total distance traveled by vehicle $(\bar{c},\bar{\imath})$ during one sampling cycle is computed by

$$d_{(\bar{c},\bar{\imath}),2,i,j} = v_{(\bar{c},\bar{\imath}),2,i,j} \cdot t_{(\bar{c},\bar{\imath}),2,i,j}, \tag{4.6}$$

in which

$$v_{(\bar{c},\bar{\imath}),2,i,j} = v_{(\bar{c},\bar{\imath}),g_{(\bar{c},\bar{\imath}),2,i-1,j},i-1,j}, \tag{4.7}$$

where $g_{(\bar{c},\bar{\imath}),2,i-1,j} \in \{1,2\}$ denotes the group that vehicle $(\bar{c},\bar{\imath}) \in G_{2,i,j}$ belongs to for that piece of its trajectory that is located within the sampling window $\mathrm{w}_{i-1,j}$. Note that $g_{(\bar{c},\bar{\imath}),2,i-1,j} = 1$, if $i - \bar{\imath} = 1$, and $g_{(\bar{c},\bar{\imath}),2,i-1,j} = 2$, if $i - \bar{\imath} > 1$. Hence the start and the end positions of vehicle $(\bar{c},\bar{\imath})$

are obtained by

$$s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j} = s^{\text{end}}_{(\bar{c},\bar{\imath}),g_{(\bar{c},\bar{\imath}),2,i-1,j},i-1,j}, \tag{4.8}$$

$$s^{\text{end}}_{(\bar{c},\bar{\imath}),2,i,j} = d_{(\bar{c},\bar{\imath}),2,i,j} + s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j}. \tag{4.9}$$

The new procedure sequentially uses (4.1)-(4.9), and computes the generalized average speed, the generalized flow, and the generalized density using the following equations:

$$\rho_{i,j} = \frac{1}{A_{i,j}} \left(\sum_{(c,i)\in G_{1,i,j}} \min\left\{ t_{(c,i)),1,i,j}, \frac{L_j}{v_{(c,i),1,i,j}} \right\} + \sum_{(\bar{c},\bar{\imath})\in G_{2,i,j}} \min\left\{ t_{(\bar{c},\bar{\imath}),2,i,j}, \frac{L_j - s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j}}{v_{(\bar{c},\bar{\imath}),2,i,j}} \right\} \right), \tag{4.10}$$

$$q_{i,j} = \frac{1}{A_{i,j}} \left(\sum_{(c,i)\in G_{1,i,j}} \min\left\{ d_{(c,i),1,i,j}, L_j \right\} + \sum_{(\bar{c},\bar{\imath})\in G_{2,i,j}} \min\left\{ d_{(\bar{c},\bar{\imath}),2,i,j}, L_j - s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j} \right\} \right), \tag{4.11}$$

$$\bar{v}_{i,j} = \frac{\rho_{i,j}}{q_{i,j}}. \tag{4.12}$$

Algorithm 4.1 on page 62 represents the sequential algorithm proposed for a single-lane road. The total number of operations that the sequential algorithm involves can be determined as follows. Suppose that for a specific loop detector, within one sampling cycle, n vehicles are observed in total. Then, for Algorithm 4.1, the total number of multiplications is $3(n+1)$ and the total number of additions is less than $3.5n$, where the total number of additions depends on the number of vehicles that are in either of groups 1 and 2 introduced in Section 4.2. Note that $3.5n$ is the maximum possible number of the additions and is reached for the case where there are the same number of vehicles in groups 1 and 2.

4.3.2 Multi-lane roads

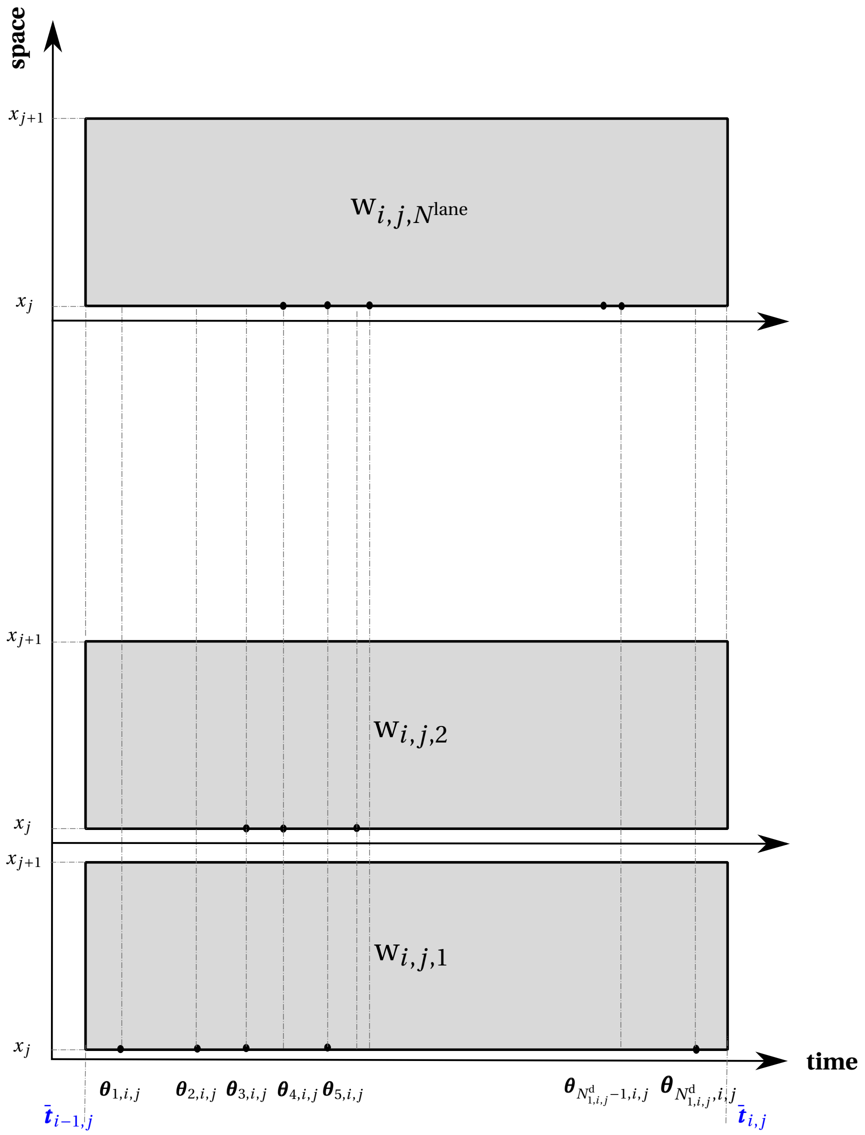
In Section 4.3.1, we considered a single-lane road (see **Assumption 2** of Section 4.3.1). Here we will relax this assumption by considering a multi-lane road and by extending the sequential procedure to a multi-lane road case (in order to avoid making the derived equations too complicated by involving a lane-changing model, here we assume that there are no lane changes).

Suppose that the road has N^{lane} lanes; we consider the sampling road section j that is extended between two consecutive loop detectors D_j and D_{j+1}. Suppose that during $[\bar{t}_{i-1,j}, \bar{t}_{i,j}]$, $N_{1,i,j}$ vehicles have entered the sampling road section (through all N^{lane} lanes) at time instants $\theta_{1,i,j}, \theta_{2,i,j}, \ldots, \theta_{N_{1,i,j},i,j}$, where the values of $\theta_{c,i,j}$ for $(c,i) \in G_{1,i,j}$ are not necessarily distinct, i.e., it is possible to have two vehicles entering the sampling road section j at the same time instant via different lanes. Define $v_{(c,i),1,i,j,\ell}$ as the speed of the c^{th} vehicle that enters the sampling road section j at time instant $\theta_{c,i,j}$ via lane ℓ, where $\ell \in \{1,\ldots,N^{\text{lane}}\}$.

Now we can correspond to each lane of the sampling road section in the time-space plane a sampling window $\mathrm{w}_{i,j,\ell}$, which indicates the sampling window corresponding to the ℓ^{th} lane of the sampling road section j during the time interval $[\bar{t}_{i-1,j}, \bar{t}_{i,j}]$. Figure 4.4 illustrates

Algorithm 4.1 Computation of the fundamental traffic variables on the single-lane road section $j,\ j \in \{1,2,\dots,n^{\text{loop}}\}$.

```
Input: $\begin{cases} L^{\text{road}},\ T^{\text{road}},\ L_j,\ T_j \\ v_{c_1,1,i,j},\ \theta_{c_1,i,j},\ \bar{t}_{i-1,j} & \forall c_1 \in \{1,\dots,N_{1,i,j}\},\ \forall i \in \{1\dots,n_j^{\text{cyc}}\},\ n_j^{\text{cyc}} = \left\lfloor \frac{T^{\text{road}}}{T_j} \right\rfloor \\ v_{c_2,2,0,j} & \forall c_2 \in \{1,\dots,N_{2,0,j}\} \end{cases}$
Output: $\rho_{i,j},\ q_{i,j},\ \bar{v}_{i,j}$
for $i = 1$ to $n_j^{\text{cyc}}$ do
    for $c_1 = 1$ to $N_{1,i,j}$ do
        compute $t_{c_1,1,i,j}$ from (4.1),
        compute $d_{c_1,1,i,j}$ from (4.2),
        compute $s^{\text{end}}_{c,1,i,j}$ from (4.4),
        use Case 1 and Case 2 to construct $G^{(1)}_{1,i,j}$ and $G^{(2)}_{1,i,j}$,
        $v_{c_2'',2,i+1,j} \leftarrow v_{c_1,1,i,j}$    {Where $c_2''$ is the index of vehicle $c_1$ in $G_{2,i+1,j}$}
    end for
    for $c_2 = 1$ to $N_{2,i,j}$ do
        compute $t_{c_2,2,i,j}$ from (4.5),
        compute $d_{c_2,2,i,j}$ from (4.6),
        compute $s^{\text{end}}_{c,2,i,j}$ from (4.9),
        use Case 3 and C4 to construct $G^{(1)}_{2,i,j}$ and $G^{(2)}_{2,i,j}$,
        $v_{c_2'',2,i+1,j} \leftarrow v_{c_2,2,i,j}$,    {Where $c_2''$ is the index of vehicle $c_2$ in $G_{2,i+1,j}$}
        $s^{\text{start}}_{c,2,i+1,j} \leftarrow s^{\text{end}}_{c,2,i,j}$
    end for
    $G_{2,i+1,j} \leftarrow G^{(2)}_{1,i,j} \bigcup G^{(2)}_{2,i,j}$
    compute $\rho_{i,j}$ from (4.10),
    compute $q_{i,j}$ from (4.11),
    compute $\bar{v}_{i,j}$ from (4.12)
end for
```


space
x_{j+1}
$\mathrm{w}_{i,j,N^{\text{lane}}}$
x_j
x_{j+1}
$\mathrm{w}_{i,j,2}$
x_j
x_{j+1}
$\mathrm{w}_{i,j,1}$
x_j
time
$\boldsymbol{\theta}_{1,i,j}$
$\boldsymbol{\theta}_{2,i,j}$
$\boldsymbol{\theta}_{3,i,j}$
$\boldsymbol{\theta}_{4,i,j}$
$\boldsymbol{\theta}_{5,i,j}$
$\boldsymbol{\theta}_{N^{\text{d}}_{1,i,j}-1,i,j}$
$\boldsymbol{\theta}_{N^{\text{d}}_{1,i,j},i,j}$
$\bar{\boldsymbol{t}}_{i-1,j}$
$\bar{\boldsymbol{t}}_{i,j}$

	$\boldsymbol{\theta}_{1,i,j}$	$\boldsymbol{\theta}_{2,i,j}$	$\boldsymbol{\theta}_{3,i,j}$	$\boldsymbol{\theta}_{4,i,j}$	$\boldsymbol{\theta}_{5,i,j}$	— — —	$\boldsymbol{\theta}_{N^{\text{d}}_{1,i,j}-1,i,j}$	$\boldsymbol{\theta}_{N^{\text{d}}_{1,i,j},i,j}$
lane 1	$\boldsymbol{v}_{(1,i),1,i,j,1}$	$\boldsymbol{v}_{(2,i),1,i,j,1}$	$\boldsymbol{v}_{(3,i),1,i,j,1}$		$\boldsymbol{v}_{(5,i),1,i,j,1}$			$\boldsymbol{v}_{(N^{\text{d}}_{1,i,j},i),1,i,j,1}$
lane 2			$\boldsymbol{v}_{(3,i),1,i,j,2}$	$\boldsymbol{v}_{(4,i),1,i,j,2}$				
⋮								⋮
lane N^{lane}				$\boldsymbol{v}_{(4,i),1,i,j,N^{\text{lane}}}$		— — —	$\boldsymbol{v}_{\left(N^{\text{d}}_{1,i,j}-1,i\right),1,i,j,N^{\text{lane}}}$	

Figure 4.4: Time-space plots for a multi-lane road.

such sampling windows. Suppose that the set of all distinct time instants in the time-space plane for all sampling windows $\mathrm{w}_{i,j,\ell}$, has $N^{\mathrm{d}}_{1,i,j}$ elements ($N^{\mathrm{d}}_{1,i,j} \leq N_{1,i,j}$). Then $v_{(c,i),1,i,j,\ell}$ for $(c,i) \in G_{1,i,j,\ell}$ will be substituted by either the observed speed on lane ℓ at time instant $\theta_{c,i,j}$, or by zero if no vehicles have been observed (i.e., we consider a virtual vehicle with zero speed in this case). This way we can summarize all the data corresponding to the sampling road section in a matrix of dimension $N^{\mathrm{lane}} \times N^{\mathrm{d}}_{1,i,j}$ as follows (see also Figure 4.4):

$$V_{1,i,j} =$$
$$\begin{bmatrix} v_{(1,i),1,i,j,1} & v_{(2,i),1,i,j,1} & v_{(3,i),1,i,j,1} & 0 & \cdots & 0 & v_{(N^{\mathrm{d}}_{1,i,j},i),1,i,j,1} \\ 0 & 0 & v_{(3,i),1,i,j,2} & v_{(4,i),1,i,j,2} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & v_{(4,i),1,i,j,N^{\mathrm{lane}}} & \cdots & v_{(N^{\mathrm{d}}_{1,i,j}-1,i),1,i,j,N^{\mathrm{lane}}} & 0 \end{bmatrix}. \tag{4.13}$$

The parameters $\theta_{(1,i),j}$, $\theta_{(2,i),j}$, ..., $\theta_{(N^{\mathrm{d}}_{1,i,j},i),j}$ are the time instants at which each of the vehicles (real or virtual) is positioned at x_j. We use an $N^{\mathrm{d}}_{1,i,j} \times N^{\mathrm{d}}_{1,i,j}$ matrix defined by

$$\Theta_{1,i,j} = \begin{bmatrix} \theta_{(1,i),j} & 0 & 0 & \cdots & 0 \\ 0 & \theta_{(2,i),j} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & \theta_{(N^{\mathrm{d}}_{1,i,j},i),j} \end{bmatrix}. \tag{4.14}$$

We have

$$t_{(c,i),1,i,j,\ell} = \left(T_j + \bar{t}_{i-1,j} - \theta_{(c,i),j}\right) \cdot \mathrm{sign}\left(v_{(c,i),1,i,j,\ell}\right), \tag{4.15}$$

$$d_{(c,i),1,i,j,\ell} = v_{(c,i),1,i,j,\ell} \cdot t_{(c,i),1,i,j,\ell}, \tag{4.16}$$

$$s^{\mathrm{start}}_{(c,i),1,i,j,\ell} = 0, \tag{4.17}$$

$$s^{\mathrm{end}}_{(c,i),1,i,j,\ell} = d_{(c,i),1,i,j,\ell}, \tag{4.18}$$

where $i \in \{1,\ldots,n^{\mathrm{cyc}}_j\}$, $j \in \{1,\ldots,n^{\mathrm{loop}}\}$, $(c,i) \in G_{1,i,j,\ell}$, $\ell \in \{1,\ldots,N^{\mathrm{lane}}\}$, and for all ℓ we have $G_{1,i,j,\ell} = \{1,\ldots,N^{\mathrm{d}}_{1,i,j}\}$. Finally, we obtain the following equation for all those vehicles that are in $G_{1,i,j} = \bigcup\limits_{\ell \in \{1,\ldots,N^{\mathrm{lane}}\}} G_{1,i,j,\ell}$:

$$T_{1,i,j} = \mathrm{sign}\left(V_{1,i,j}\right) \cdot \left(\left(T_j + \bar{t}_{i-1,j}\right) \cdot I_{N^{\mathrm{d}}_{1,i,j} \times N^{\mathrm{d}}_{1,i,j}} - \Theta_{1,i,j}\right), \tag{4.19}$$

$$D_{1,i,j} = V_{1,i,j} \cdot \left(\left(T_j + \bar{t}_{i-1,j}\right) \cdot I_{N^{\mathrm{d}}_{1,i,j} \times N^{\mathrm{d}}_{1,i,j}} - \Theta_{1,i,j}\right), \tag{4.20}$$

$$S^{\mathrm{start}}_{1,i,j} = 0, \tag{4.21}$$

$$S^{\text{end}}_{1,i,j} = D_{1,i,j}, \tag{4.22}$$

where sign(·) operates element-wise on a matrix, and produces a matrix with the same dimension as the input matrix, i.e., if $S = \text{sign}(A)$ with A an $m \times n$ matrix, then S is also an $m \times n$ matrix with $s_{ij} = \text{sign}\left(a_{ij}\right)$. Moreover, $T_{1,i,j}$, $D_{1,i,j}$, $S^{\text{start}}_{1,i,j}$, and $S^{\text{end}}_{1,i,j}$ are matrices of dimension $N^{\text{lane}} \times N^{\text{d}}_{1,i,j}$.

For the second group $G_{2,i,j} = \bigcup\limits_{\ell \in \{1,\ldots,N^{\text{lane}}\}} G_{2,i,j,\ell}$ of vehicles that have entered the sampling road section within previous cycles, and their trajectories intersect the left-hand side edge of the sampling windows, $V_{2,i,j}$ will again be a matrix of the following form (where $\bar{\imath} \in \{1,\ldots,i-1\}$):

$$V_{2,i,j} =$$
$$\begin{bmatrix} v_{(\bar{c}_{11},\bar{\imath}_{11}),2,i,j,1} & v_{(\bar{c}_{12},\bar{\imath}_{12}),2,i,j,1} & \cdots & v_{(\bar{c}_{1N^{\text{d}}_{2,i,j}},\bar{\imath}_{1N^{\text{d}}_{2,i,j}}),2,i,j,1} \\ v_{(\bar{c}_{21},\bar{\imath}_{21}),2,i,j,2} & v_{(\bar{c}_{22},\bar{\imath}_{22}),2,i,j,2} & \cdots & v_{(\bar{c}_{2N^{\text{d}}_{2,i,j}},\bar{\imath}_{2N^{\text{d}}_{2,i,j}}),2,i,j,2} \\ \vdots & \vdots & \cdots \; \vdots & \vdots \\ v_{(\bar{c}_{N^{\text{lane}}1},\bar{\imath}_{N^{\text{lane}}1}),2,i,j,N^{\text{lane}}} & v_{(\bar{c}_{N^{\text{lane}}2},\bar{\imath}_{N^{\text{lane}}2}),2,i,j,N^{\text{lane}}} & \cdots & v_{(\bar{c}_{N^{\text{lane}}N^{\text{d}}_{2,i,j}},\bar{\imath}_{N^{\text{lane}}N^{\text{d}}_{2,i,j}}),2,i,j,N^{\text{lane}}} \end{bmatrix}, \tag{4.23}$$

which is of dimension $N^{\text{lane}} \times N^{\text{d}}_{2,i,j}$, with $N^{\text{d}}_{2,i,j} = \max\limits_{\ell \in \{1,\ldots,N^{\text{lane}}\}} \left\{N_{2,i,j,\ell}\right\}$. Note that $N_{2,i,j,\ell}$ is the number of vehicles in $G_{2,i,j,\ell}$, i.e., the number of vehicles within group 2 for the sampling window $\text{w}_{i,j}$ that are traveling on lane ℓ. In order to construct the matrix $V_{2,i,j}$, we start by filling the matrix from the first element of each row ℓ and put the speed of the vehicles (the order does not matter) belonging to group 2 of the sampling window $\text{w}_{i,j}$ one by one at the columns. If at any moment, the number of columns is more than the number of vehicles, we put zero in that column. Then, we correspond to each vehicle $(\bar{c},\bar{\imath})$ moving on lane ℓ, a value $k_{\bar{c},\bar{\imath},\ell}$, which shows the number of column of $V_{2,i,j}$ at which the speed of this vehicle is stored. Moreover, for all ℓ we have $G_{2,i,j,\ell} = \{1,\ldots,N^{\text{d}}_{2,i,j}\}$. For lanes with $N_{2,i,j,\ell} < N^{\text{d}}_{2,i,j}$, the corresponding elements of $V_{2,i,j}$ on columns $N_{2,i,j,\ell}+1$ are set to zero. We obtain

$$t_{(\bar{c},\bar{\imath}),2,i,j,\ell} = T_j \cdot \text{sign}\left(v_{(\bar{c},\bar{\imath}),2,i,j,\ell}\right), \tag{4.24}$$

$$d_{(\bar{c},\bar{\imath}),2,i,j,\ell} = v_{(\bar{c},\bar{\imath}),2,i,j,\ell} \cdot t_{(\bar{c},\bar{\imath}),2,i,j,\ell}, \tag{4.25}$$

$$s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j,\ell} = s^{\text{end}}_{(\bar{c},\bar{\imath}),g_{(\bar{c},\bar{\imath}),2,i-1,j,\ell},i-1,j,\ell}, \tag{4.26}$$

$$s^{\text{end}}_{(\bar{c},\bar{\imath}),2,i,j,\ell} = d_{(\bar{c},\bar{\imath}),2,i,j,\ell} + s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j,\ell}, \tag{4.27}$$

where $g_{(\bar{c},\bar{\imath}),2,i-1,j,\ell} \in \{1,2\}$ is the group that vehicle $(\bar{c},\bar{\imath}) \in G_{2,i,j,\ell}$ belongs to for that piece of its trajectory that is located within the sampling window $\text{w}_{i-1,j,\ell}$. Hence,

$$T_{2,i,j} = \text{sign}\left(V_{2,i,j}\right) \cdot T_j, \tag{4.28}$$

$$D_{2,i,j} = T_j \cdot V_{2,i,j}, \tag{4.29}$$

$$S^{\text{start}}_{2,i,j} = S^{\text{end}}_{0,i-1,j}, \tag{4.30}$$

$$S^{\text{end}}_{2,i,j} = D_{2,i,j} + S^{\text{start}}_{2,i,j}, \tag{4.31}$$

with $T_{2,i,j}$, $D_{2,i,j}$, $S^{\text{start}}_{2,i,j}$, and $S^{\text{end}}_{2,i,j}$ matrices of dimension $N^{\text{lane}} \times N^{\text{d}}_{2,i,j}$, and $S^{\text{end}}_{0,i-1,j}$ a matrix that includes all the end positions of those vehicles that their indices belong to $G_{0,i-1,j}$.

In order to find the generalized density, flow, and average speed for each lane separately, we can write

$$\begin{aligned}\rho_{i,j,\ell} = \frac{1}{A_{i,j}} \Bigg(&\sum_{(c,i)\in G_{1,i,j,\ell}} \min\left\{ \left(T_{1,i,j}\right)_{\ell,c}, \frac{L_j}{v_{(c,i),1,i,j,\ell}} \right\} + \\ &\sum_{(\bar{c},\bar{\imath})\in G_{2,i,j,\ell}} \min\left\{ \left(T_{2,i,j}\right)_{\ell,k_{\bar{c},\bar{\imath},\ell}}, \frac{L_j - s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j,\ell}}{v_{(\bar{c},\bar{\imath}),2,i,j,\ell}} \right\} \Bigg),\end{aligned} \tag{4.32}$$

$$\begin{aligned}q_{i,j,\ell} = \frac{1}{A_{i,j}} \Bigg(&\sum_{(c,i)\in G_{1,i,j,\ell}} \min\left\{ \left(D_{1,i,j}\right)_{\ell,c}, L_j \right\} + \\ &\sum_{(\bar{c},\bar{\imath})\in G_{2,i,j,\ell}} \min\left\{ \left(D_{2,i,j}\right)_{\ell,k_{\bar{c},\bar{\imath},\ell}}, L_j - s^{\text{start}}_{(\bar{c},\bar{\imath}),2,i,j,\ell} \right\} \Bigg),\end{aligned} \tag{4.33}$$

$$\bar{v}_{i,j,\ell} = \frac{\rho_{i,j,\ell}}{q_{i,j,\ell}}, \tag{4.34}$$

where $X_{a,b}$ indicates the element at row a and column b of the matrix X. Finally, to find the generalized traffic variables for the lane altogether, we have

$$\rho_{i,j} = \frac{1}{A_{i,j}} \sum_{\ell=1}^{N^{\text{lane}}} \rho_{i,j,\ell}, \tag{4.35}$$

$$q_{i,j} = \frac{1}{A_{i,j}} \sum_{\ell=1}^{N^{\text{lane}}} q_{i,j,\ell}, \tag{4.36}$$

$$\bar{v}_{i,j} = \frac{\rho_{i,j}}{q_{i,j}}. \tag{4.37}$$

Algorithm 4.1 can easily be extended for the multi-lane road using the introduced matrices.

4.4 Approximate trajectories of vehicles based on Newell's car-following model

4.4.1 Leading and following vehicles

In the previous sections, we have considered that each vehicle will move with a constant speed in the sampling window $\mathrm{w}_{i,j}$ (see **Assumption 3** of Section 4.3.1), i.e., for $(c,i) \in G_{1,i,j}$ and for

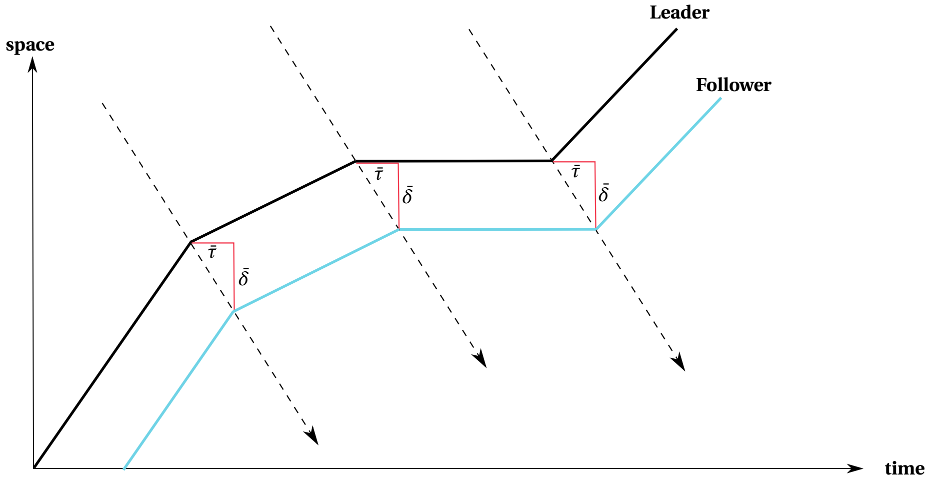


Figure 4.5: Trajectory of a following vehicle with respect to the leading vehicle according to Newell's car-following model.

$(\bar{c},\bar{\imath}) \in G_{2,i,j}$, we have

$$\forall \bar{t} \in \left[\bar{t}_{i-1,j}, \bar{t}_{i,j}\right], \qquad \begin{array}{l} v_{(c,i),1,i,j}(\bar{t}) = v_{(c,i),1,i,j}\left(\theta_{(c,i),j}\right), \\ v_{(\bar{c},\bar{\imath}),2,i,j}(\bar{t}) = v_{(\bar{c},\bar{\imath}),2,i,j}(\bar{t}_{i-1,j}). \end{array}$$

However, this assumption could result in some issues, e.g., intersecting trajectories, which is not realistic, especially for a single-lane road. Moreover, with the assumption of constant speeds in between loop detectors D_j and D_{j+1}, there might be a large difference between the estimated time-mean speed of the vehicles at position x_{j+1} and the reported value of the time-mean speed at position x_{j+1}. Therefore, using Newell's car-following model [123], in this section we will relax the assumption of having a constant speed for each vehicle that travels in between two consecutive loop detectors. The approach proposed in this section is inspired by the method given by Coifman [31] (see Chapter 2 for details). The main advantage of the approach proposed in this section, compared with the approach by Coifman [31], is that in our approach the trajectory of the leading vehicle is estimated without any need to capture measurements from the following vehicle at some time into the future. Therefore, our approach can also be used for real-time estimation of the trajectories for those vehicles that are in between two consecutive measurement points.

Note that when a group of vehicles move on the same lane of a sampling road section, each vehicle may act as a leading vehicle for the vehicle (if it exists) that moves behind it. Newell's car-following model gives a mathematical relationship between trajectories of a leading vehicle and its following vehicle. In particular, with Newell's car-following model, we can find a trajectory for the following vehicle with a piecewise affine behavior. It is assumed that the following vehicle adapts its speed to the speed of the leading vehicle in the time-space plane with a time delay (see Figure 4.5). Suppose that the set of all vehicles that act as leading vehicle within the sampling window $\mathrm{w}_{i,j}$ is denoted by $G_{\mathrm{lead},i,j}$, and the set of all vehicles that act as following vehicle within the sampling window $\mathrm{w}_{i,j}$ is denoted by $G_{\mathrm{follow},i,j}$ (note that a vehicle may appear in both sets for the sampling window $\mathrm{w}_{i,j}$, i.e., a vehicle that is a following vehicle for the vehicle in front of it, might act as a leading vehicle for the vehicle behind it). Newell proposes that the trajectory $s^{\mathrm{lead}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}$ of a leading vehicle $(c_{\mathrm{l}},\iota_{\mathrm{l}}) \in G_{\mathrm{lead},i,j}$ in the sampling window $\mathrm{w}_{i,j}$ and the trajectory $s^{\mathrm{follow}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ of its following vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}}) \in G_{\mathrm{follow},i,j}$ are always at a temporal distance of $\tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ and a spatial distance of $\delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ from each other in the time-space plane. Hence, assuming that vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}})$ acts as a following vehicle for vehicle $(c_{\mathrm{l}},\iota_{\mathrm{l}})$ within the sampling window $\mathrm{w}_{i,j}$, mathematically speaking we can write

$$s^{\mathrm{follow}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}(\bar{t}) = s^{\mathrm{lead}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}(\bar{t} - \tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}) - \delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}, \qquad \bar{t} \in \left[\bar{t}_{i-1,j}, \bar{t}_{i,j}\right]. \tag{4.38}$$

The parameters $\delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ and $\tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ may vary from driver to driver, as a result of different driving behaviors, and from one sampling window to another one, considering different traffic regimes. For example, the driver of vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}})$ chooses the spatial distance $\delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ based on her/his feeling of the safe distance from the front vehicle on the road. Additionally, $\tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ is the time delay of the driver of vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}})$ in responding to external stimuli. For the sake of simplicity, we consider the same values for these two parameters for all the following vehicles within all the sampling windows[2]. Hence, we use the average values of $\delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ and $\tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ of all the following vehicles $(c_{\mathrm{f}},\iota_{\mathrm{f}}) \in G_{\mathrm{follow},i,j}$ across all the sampling windows, and we denote them with $\bar{\delta}$ and $\bar{\tau}$, respectively. We have

$$\bar{\tau} = \frac{1}{\sum\limits_{j\in\{1,\dots,n^{\mathrm{loop}}\}} \sum\limits_{i\in\{1,\dots,n^{\mathrm{cyc}}_j\}} \left(N_{i,j}-1\right)} \sum_{j\in\{1,\dots,n^{\mathrm{loop}}\}} \sum_{i\in\{1,\dots,n^{\mathrm{cyc}}_j\}} \sum_{(c_{\mathrm{f}},\iota_{\mathrm{f}})\in G_{\mathrm{follow},i,j}} \tau_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}, \tag{4.39}$$

$$\bar{\delta} = \frac{1}{\sum\limits_{j\in\{1,\dots,n^{\mathrm{loop}}\}} \sum\limits_{i\in\{1,\dots,n^{\mathrm{cyc}}_j\}} \left(N_{i,j}-1\right)} \sum_{j\in\{1,\dots,n^{\mathrm{loop}}\}} \sum_{i\in\{1,\dots,n^{\mathrm{cyc}}_j\}} \sum_{(c_{\mathrm{f}},\iota_{\mathrm{f}})\in G_{\mathrm{follow},i,j}} \delta_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}, \tag{4.40}$$

where $N_{i,j}$ is the total number of vehicles that corresponds to the sampling window $\mathrm{w}_{i,j}$.

4.4.2 Determination of groups $G_{\mathrm{lead},i,j}$ and $G_{\mathrm{follow},i,j}$

Next we continue with explaining our proposed algorithm for estimation of the trajectories of the vehicles within one sampling window. In order to determine the two groups $G_{\mathrm{lead},i,j}$ and $G_{\mathrm{follow},i,j}$ for the sampling window $\mathrm{w}_{i,j}$, we first order the values of $s^{\mathrm{start}}_{(\bar{c},\bar{\iota}),2,i,j}$ for the vehicles $(\bar{c},\bar{\iota}) \in G_{2,i,j}$ in a descending order, and keep the corresponding pair of indices of these vehicles in an ordered set $O_{i,j}$. Next we put the pairs of indices corresponding to those vehicles that enter the sampling road section during the current sampling cycle in $O_{i,j}$, one by one and based on their order of entrance. Note that in construction of $O_{i,j}$, we just consider the relative position of the vehicles on the road (i.e., whether a vehicle is located behind the other vehicle or not). For a vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}})$ to act as a following vehicle for vehicle $(c_{\mathrm{l}},\iota_{\mathrm{l}})$, it is necessary, but not sufficient, that it moves behind vehicle $(c_{\mathrm{l}},\iota_{\mathrm{l}})$ on the road. Next, we give the additional

[2] Note however that the proposed approach given in this section, can easily be generalized to a case where the preferred temporal and spatial distances of drivers are different from each other.

conditions, which together with the previous condition provide the sufficient conditions for a vehicle to act as a following vehicle.

The temporal and spatial distance and the relative speed of the two vehicles might dictate some interaction between the two vehicles. Therefore, the group $G_{\text{follow},i,j}$ includes the e^{th} ($e > 1$) element of $O_{i,j}$, if this element satisfies at least one of the following conditions:

Spatial distance condition: If the spatial distance of two successive vehicles is lower than a threshold $\bar{\delta}$, then the front vehicle acts as a leading vehicle and the other vehicle acts as a following vehicle. Mathematically, if the following holds:

$$s^{\text{start}}_{O_{i,j}(e-1),g_{O_{i,j}(e-1),i,j},i,j} - s^{\text{start}}_{O_{i,j}(e),g_{O_{i,j}(e),i,j},i,j} < \bar{\delta},$$

then the vehicle that corresponds to $O_{i,j}(e)$ acts as a following vehicle for the vehicle that corresponds to $O_{i,j}(e-1)$. Note that $g_{O_{i,j}(e),i,j}$ is the group number (i.e., $g_{O_{i,j}(e),i,j} = 1$, if $O_{i,j}(e) \in G_{1,i,j}$ and $g_{O_{i,j}(e),i,j} = 2$, if $O_{i,j}(e) \in G_{2,i,j}$) in the sampling window $\text{w}_{i,j}$ of the vehicle that corresponds to the e^{th} element of $O_{i,j}$.

Temporal distance condition: If the temporal distance of two successive vehicles is less than a threshold $\bar{\tau}$, then the front vehicle acts as a leading vehicle and the other vehicle acts as a following vehicle. Mathematically speaking, if the following condition holds:

$$\theta^{\text{initial}}_{O_{i,j}(e),i,j} - \theta^{\text{initial}}_{O_{i,j}(e-1),i,j} < \bar{\tau},$$

where $\theta^{\text{initial}}_{O_{i,j}(e),i,j} = \max\left\{\theta_{O_{i,j}(e),i,j}, \bar{t}_{i-1,j}\right\}$, then the vehicle that corresponds to $O_{i,j}(e)$ acts as a following vehicle for the vehicle that corresponds to $O_{i,j}(e-1)$.

Relative speed condition: Assume that the vehicles corresponding to $O_{i,j}(e)$ and $O_{i,j}(e-1)$ keep on moving with their initial speeds within the sampling window $\text{w}_{i,j}$. If in that case their trajectories intersect at one point within the sampling window $\text{w}_{i,j}$, then the vehicle that moves in front should act as a leading vehicle and the other vehicle should act as a following vehicle. Mathematically, if we have

$$\frac{L_j - s^{\text{start}}_{O_{i,j}(e),g_{O_{i,j}(e),i,j},i,j}}{v^{\text{initial}}_{O_{i,j}(e),i,j}} + \theta^{\text{initial}}_{O_{i,j}(e),i,j} \leq \frac{L_j - s^{\text{start}}_{O_{i,j}(e-1),g_{O_{i,j}(e-1),i,j},i,j}}{v^{\text{initial}}_{O_{i,j}(e-1),i,j}} + \theta^{\text{initial}}_{O_{i,j}(e-1),i,j},$$

where $v^{\text{initial}}_{O_{i,j}(e),i,j}$ is the initial speed of the e^{th} element of $O_{i,j}$ within the sampling window $\text{w}_{i,j}$, then the vehicle that corresponds to $O_{i,j}(e)$ acts as a following vehicle for the vehicle that corresponds to $O_{i,j}(e-1)$.

Note that for each vehicle corresponding to $O_{i,j}(e)$ that goes into the group $G_{\text{follow},i,j}$, the vehicle corresponding to $O_{i,j}(e-1)$ goes into the group $G_{\text{lead},i,j}$ and acts as the leading vehicle for $O_{i,j}(e)$.

4.4.3 Trajectory approximation

First we consider those vehicles that appear in $G_{\text{lead},i,j}$, but do not appear in $G_{\text{follow},i,j}$. For these leading vehicles, we propose the following rules:

- If $v^{\text{initial}}_{(c_\text{l},\iota_\text{l}),i,j} = v^{\text{free}}_{i,j}$, where by $v^{\text{free}}_{i,j}$ we mean the free-flow speed of the vehicles[3] on the sampling road section j during $\left[\bar{t}_{i-1,j}, \bar{t}_{i,j}\right]$, then the leading vehicle keeps moving forward with $v^{\text{free}}_{i,j}$ till it leaves the sampling road section j.

- If $v^{\text{initial}}_{(c_\text{l},\iota_\text{l}),i,j} < v^{\text{free}}_{i,j}$, then the vehicle accelerates with $a^{\text{acc}}_{c_\text{l},\iota_\text{l}}$, i.e., the maximum possible acceleration rate for this vehicle, to obtain $v^{\text{free}}_{i,j}$. Then it keeps moving forward with $v^{\text{free}}_{i,j}$ till it leaves the sampling road section j.

A following vehicle $(c_\text{f}, \iota_\text{f})$ may face three different situations w.r.t. its leading vehicle $(c_\text{l}, \iota_\text{l})$, i.e.,

Situation 1: The leading and the following vehicles both belong to $G_{1,i,j}$.

Situation 2: The leading vehicle belongs to $G_{2,i,j}$, while the following vehicle belongs to $G_{1,i,j}$.

Situation 3: The leading and the following vehicles both belong to $G_{2,i,j}$.

We distinguish the following two cases: $v^{\text{initial}}_{(c_\text{l},\iota_\text{l}),i,j} = v^{\text{free}}_{i,j}$ and $v^{\text{initial}}_{(c_\text{l},\iota_\text{l}),i,j} < v^{\text{free}}_{i,j}$, and consequently obtain the following conclusions:

- Suppose that we have $v^{\text{initial}}_{(c_\text{l},\iota_\text{l}),i,j} = v^{\text{free}}_{i,j}$ (see Figure 4.6). In this case, for the following vehicle to keep the safe temporal and spatial distances from its leading vehicle, its trajectory in the time-space plane should not be located on the left-hand side of the dash-dotted curves (called the borderline curves) illustrated in Figures 4.6(a)-4.6(c), which are located at exactly a temporal distance $\bar{\tau}$ and a spatial distance $\bar{\delta}$ from the trajectory of the following vehicle.

  Then the borderline curves are defined as follows:

  1. For **situation 1**, the trajectory of the leading vehicle $(c_\text{l}, \iota_\text{l})$ is a straight line that intersects the lower edge of the sampling window $\text{w}_{i,j}$ and that has a slope of $v^{\text{free}}_{i,j}$ (see Figure 4.6(a)). Then the borderline curve corresponding to the leading vehicle $(c_\text{l}, \iota_\text{l})$ (indicated by $s^{\text{borderline}}_{(c_\text{l},\iota_\text{l}),i,j}$) is obtained by transferring the trajectory of the leading vehicle $\bar{\tau}$ units to the right and $\bar{\delta}$ units to the bottom in the time-space plane.
  2. For **situation 2** and **situation 3**, the trajectory of the leading vehicle is a straight line with the slope $v^{\text{free}}_{i,j}$, which intersects the left-hand edge of the sampling window $\text{w}_{i,j}$ at a distance of $s^{\text{start}}_{(c_\text{l},\iota_\text{l}),2,i,j}$ from the left bottom corner of the window (see

[3] We consider a constant value for $v^{\text{free}}_{i,j}$ throughout the sampling road section $\text{w}_{i,j}$.

Figures 4.6(b) and 4.6(c)). Then, the trajectory of the borderline curves corresponding to the leading vehicle $(c_\mathrm{l}, \iota_\mathrm{l})$ for these two situations are obtained via transferring the trajectory of the leading vehicle $\bar{\tau}$ units to the right and $\bar{\delta}$ units to the bottom in the time-space plane.

For **situation 1** and **situation 2** the trajectory of the following vehicle can then be characterized as follows::

- If $s^{\mathrm{borderline}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j}\left(\theta^{\mathrm{initial}}_{(c_\mathrm{f},\iota_\mathrm{f}),i,j}\right) \geq 0$, then in case the following vehicle has an initial speed of $v^{\mathrm{free}}_{i,j}$ it keeps on moving with a constant speed. If the initial speed of the following vehicle, however, is less than $v^{\mathrm{free}}_{i,j}$ it accelerates first until it reaches $v^{\mathrm{free}}_{i,j}$ and then keeps on moving with a constant speed.
- If $s^{\mathrm{borderline}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j}\left(\theta^{\mathrm{initial}}_{(c_\mathrm{f},\iota_\mathrm{f}),i,j}\right) < 0$, then the following vehicle is initially located on the left-hand side of the borderline curve and hence, it should first decelerate until it reaches the temporal and spatial distance of $\bar{\tau}$ and $\bar{\delta}$ and then accelerates to $v^{\mathrm{free}}_{i,j}$ (see Figure 4.6(a)).

For **situation 3** we have:

- If $s^{\mathrm{borderline}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j}\left(\bar{t}_{i-1,j}\right) \geq s^{\mathrm{start}}_{(c_\mathrm{f},\iota_\mathrm{f}),2,i,j}$, then in case the following vehicle has an initial speed of $v^{\mathrm{free}}_{i,j}$ it keeps on moving with a constant speed. If the initial speed of the following vehicle, however, is less than $v^{\mathrm{free}}_{i,j}$ it accelerates first until it reaches $v^{\mathrm{free}}_{i,j}$ and then keeps on moving with a constant speed.
- If $s^{\mathrm{borderline}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j}\left(\bar{t}_{i-1,j}\right) < s^{\mathrm{start}}_{(c_\mathrm{f},\iota_\mathrm{f}),2,i,j}$, then the following vehicle is initially located on the left-hand side of the borderline curve and hence, it should first decelerate until it reaches the temporal and spatial distance of $\bar{\tau}$ and $\bar{\delta}$ and then accelerates to $v^{\mathrm{free}}_{i,j}$ (see Figure 4.6(a)).

- Suppose that we have $v^{\mathrm{initial}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j} < v^{\mathrm{free}}_{i,j}$ (see Figure 4.7). Similarly to the previous case where $v^{\mathrm{initial}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j} = v^{\mathrm{free}}_{i,j}$, the trajectory of the following vehicle in the time-space plane should not be located on the left-hand side of the borderline curves illustrated by dash-dotted curves in Figures 4.7(a)-4.7(c).

  The borderline curves are obtained as follows:

  1. For **situation 1**, the trajectory of the leading vehicle is composed of a piece of a parabola that corresponds to the accelerating behavior of the leading vehicle from its initial speed $v^{\mathrm{initial}}_{(c_\mathrm{l},\iota_\mathrm{l}),i,j}$ to $v^{\mathrm{free}}_{i,j}$ (see Figure 4.7(a)) and a straight line that corresponds to the free-flow behavior of the leading vehicle after its speed reaches $v^{\mathrm{free}}_{i,j}$. The parabolic part of the trajectory intersects the lower edge of the sampling window $\mathrm{w}_{i,j}$. Then the borderline curve corresponding to the leading vehicle $(c_\mathrm{l}, \iota_\mathrm{l})$ is obtained by transferring the trajectory of the leading vehicle $\bar{\tau}$ units to the right and $\bar{\delta}$ units to the bottom in the time-space plane. The transition point between

the parabolic part and affine part of the borderline curve is indicated by $t^{\text{switch}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}$ in Figure 4.7(a).

2. For **situation 2** and **situation 3**, the trajectory of the leading vehicle again has a parabolic and an affine part. The parabolic part of the trajectory intersects the left-hand edge of the sampling window $\mathrm{w}_{i,j}$ at a distance of $s^{\text{start}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),2,i,j}$ from the left bottom corner of the window (see Figures 4.7(b) and 4.7(c)). Then the trajectory of the borderline curve corresponding to the leading vehicle $(c_{\mathrm{l}},\iota_{\mathrm{l}})$ is obtained by transferring the trajectory of the leading vehicle $\bar{\tau}$ units to the right and $\bar{\delta}$ units to the bottom in the time-space plane. Note that on the left-hand side of point 'A' in Figures 4.7(b) and 4.7(c), a linear extension of the transferred trajectory is considered.

For **situation 1** and **situation 2** the following facts hold:

- If $s^{\text{borderline}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}\left(\theta^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}\right) \geq 0$, then in case $v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} \leq v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j} + a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}} \cdot \theta^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ is satisfied, then the following vehicle accelerates with $a^{\text{acc}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}$ to reach $v^{\text{free}}_{i,j}$. Otherwise, if $v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} > v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j} + a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}} \cdot \theta^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}$ holds, the following vehicle tries to follow the borderline curve. Hence, it first decelerates for

$$\Delta\bar{t} = \frac{v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} - v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j} - a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}}\theta^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}}{a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}} - a^{\text{dec}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}}$$

  time units, with $a^{\text{dec}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}$ the deceleration rate of the following vehicle $(c_{\mathrm{f}},\iota_{\mathrm{f}})$. Then the vehicle accelerates until it reaches $v^{\text{free}}_{i,j}$.

- If $s^{\text{borderline}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}\left(\theta^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j}\right) < 0$, then the following vehicle starts to decelerate until its trajectory intersects the borderline curve in the time-space plane. Then the following vehicle starts to accelerate with $\min\left\{a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}}, a^{\text{acc}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}\right\}$ until it reaches the speed $v^{\text{free}}_{i,j}$.

For **situation 3** the following conclusions are made:

- If $s^{\text{borderline}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}\left(\bar{t}_{i-1,j}\right) \geq s^{\text{start}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),2,i,j}$, then in case $v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} \leq v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}$ is satisfied, then the following vehicle accelerates with $a^{\text{acc}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}$ to reach $v^{\text{free}}_{i,j}$. Otherwise, in case $v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} > v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}$ holds, the following vehicle tries to follow the borderline curve. Hence, it first decelerates for

$$\Delta\bar{t} = \frac{v^{\text{initial}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),i,j} - v^{\text{initial}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}}{a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}} - a^{\text{dec}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}}$$

  time units. Then the vehicle accelerates until it reaches $v^{\text{free}}_{i,j}$.

- If $s^{\text{borderline}}_{(c_{\mathrm{l}},\iota_{\mathrm{l}}),i,j}\left(\bar{t}_{i-1,j}\right) < s^{\text{start}}_{(c_{\mathrm{f}},\iota_{\mathrm{f}}),2,i,j}$, then the following vehicle starts to decelerate until its trajectory reaches the borderline curve in the time-space plane. Then the it accelerates with $\min\left\{a^{\text{acc}}_{c_{\mathrm{l}},\iota_{\mathrm{l}}}, a^{\text{acc}}_{c_{\mathrm{f}},\iota_{\mathrm{f}}}\right\}$ until it reaches the speed $v^{\text{free}}_{i,j}$.

4.5 Case study

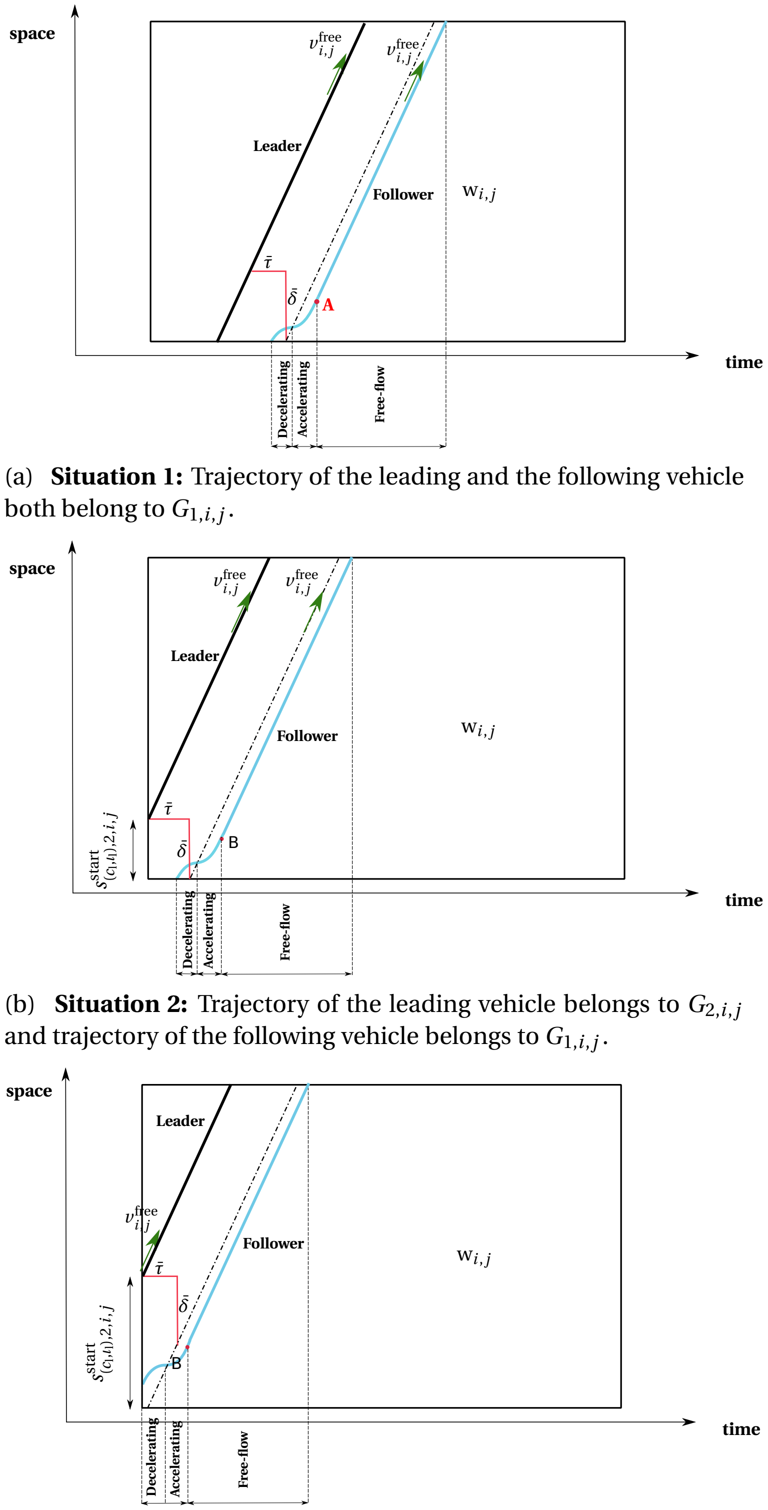
In this section, we present the results of a case study that uses real-life datasets of NGSIM. These datasets are available on either of the following three websites:

1. `http://www.ngsim-community.org/`,
2. `http://gateway.path.berkeley.edu/ngsimdocs/US-101/`,
3. `http://gateway.path.berkeley.edu/ngsimdocs/I-80/`,

and provide detailed information including the positions and the speeds of individual vehicles. For the experiment in this chapter, we have used the datasets that are available on the third website. These datasets have been generated as part of the Next Generation SIMulation (NGSIM) project by the Federal Highway Administration from a segment of the interstate freeway I-80 in San Francisco, California, US, on April 13, 2005. The data has been collected via seven video cameras from 2.00 PM till 7.00 PM, and is available in 3 sets for the time periods between 4.00 PM and 4.15 PM, between 5.00 PM and 5.15 PM, and between 5.15 PM and 5.30 PM. We indicate these three datasets by “dataset 1”, “dataset 2”, and “dataset 3”, respectively.

From these datasets, we can extract the trajectories of the vehicles and compute the real value of the generalized average speed. To assess the efficiency of the proposed sequential algorithm, we use it to determine the generalized average speed of the vehicles. Moreover, we implement the formulas given by Wardrop [153] and by Han et al. [56], by Rakha and Zhang [134], by Soriguera and Robusté [141], and by Jamshidnejad and De Schutter [63] as a comparison (see also Chapter 3), since these papers represent the state-of-the-art for estimation of the classical and the generalized average speed of the vehicles. The formulas proposed by Wardrop [153] and by Rakha and Zhang [134] are the most well-known formulas in the literature for estimating the space-mean speed of the vehicles on a road. Han et al. [56] and Soriguera and Robusté [141] give formulas for estimating the, respectively, spatial and temporal standard deviation of the speeds of the observed vehicles. The spatial standard deviation can be used in the formula by Wardrop [153] to estimate the space-mean speed of the vehicles, while the temporal standard deviation in combination with the formula by Rakha and Zhang [134] gives the estimated space-mean speed of the vehicles (see Chapter 2 for more details regarding these formulas). Since all these formulas estimate the space-mean speed, we can observe how in some cases the estimated value of the space-mean speed is close to the real value of the generalized average speed, while in other cases the estimated space-mean speed can deviate significantly from the real generalized average speed. Additionally, we use the formula given by Jamshidnejad and De Schutter [63] as the only available formula that takes into account estimation of the generalized average speed from point measurements. This formula has proven to be very accurate compared with other formulas (see [63]), and hence, it can provide a good comparison case for the proposed approach in this chapter.

For this case study, we extracted those sections of the trajectory plots from the NGSIM datasets for which enough information was available, as the dataset gives the trajectories for

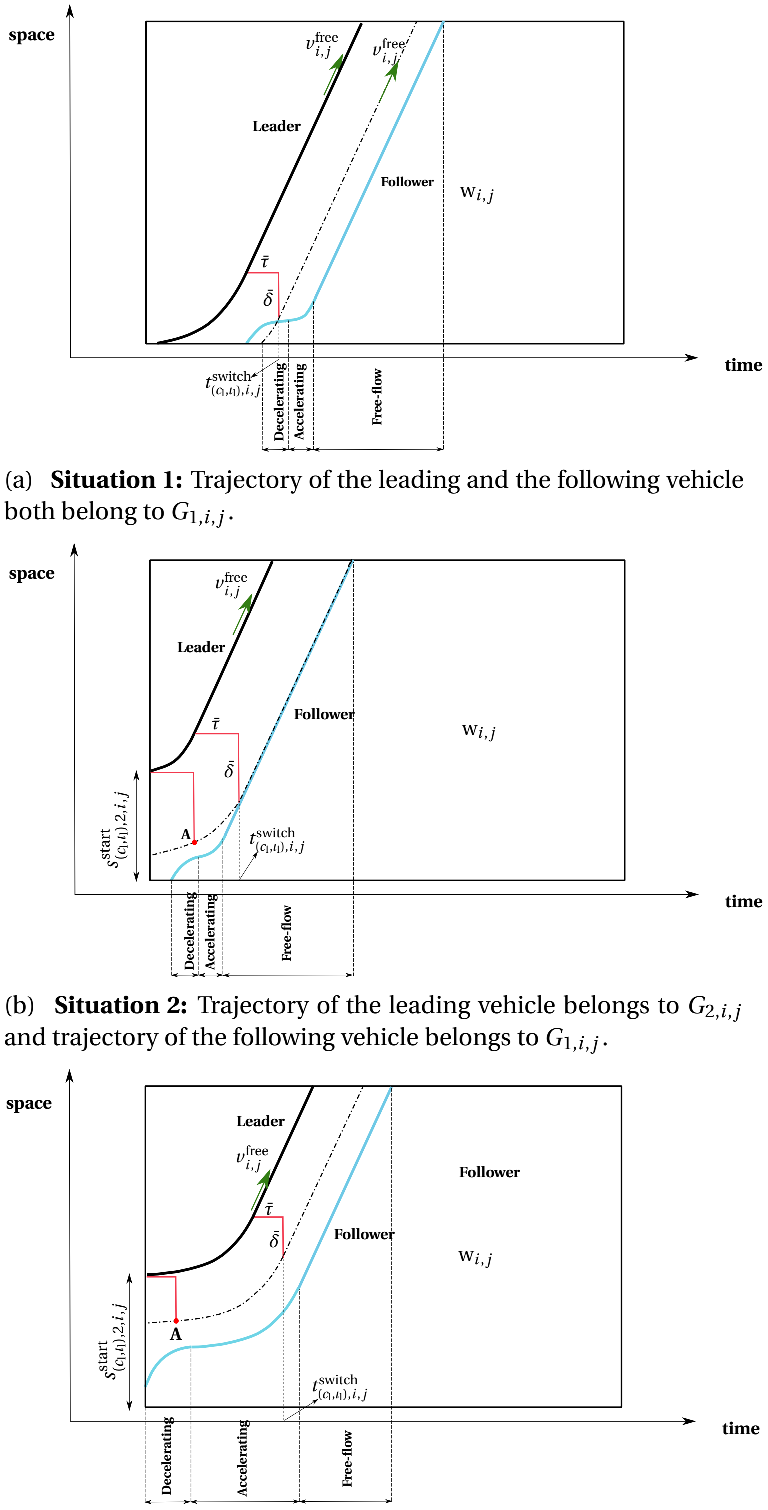


(a) **Situation 1:** Trajectory of the leading and the following vehicle both belong to $G_{1,i,j}$.

(b) **Situation 2:** Trajectory of the leading vehicle belongs to $G_{2,i,j}$ and trajectory of the following vehicle belongs to $G_{1,i,j}$.

(c) **Situation 3:** Trajectory of the leading and the following vehicle both belong to $G_{2,i,j}$.

Figure 4.6: Behavior of the leading and the following vehicle within the sampling window $\mathrm{w}_{i,j}$ for $v^{\text{initial}}_{(c_l,\iota_l),i,j} = v^{\text{free}}_{i,j}$.



(a) **Situation 1:** Trajectory of the leading and the following vehicle both belong to $G_{1,i,j}$.

(b) **Situation 2:** Trajectory of the leading vehicle belongs to $G_{2,i,j}$ and trajectory of the following vehicle belongs to $G_{1,i,j}$.

(c) **Situation 3:** Trajectory of the leading and the following vehicle both belong to $G_{2,i,j}$.

Figure 4.7: Behavior of the leading and the following vehicle within the sampling window $\mathrm{w}_{i,j}$ *for* $v^{\text{initial}}_{(c_1,\iota_1),i,j} < v^{\text{free}}_{i,j}$.

some time intervals, and there are gaps between these intervals where no trajectories are available. For each of the three datasets, we could extract four cases, i.e., 4 different sampling road sections. Since the length of the time intervals for which trajectories were available are short, we decided to choose sampling windows of dimensions $L_j = 150$ m and 200 m by $T_j = 5$ s. In this way we could obtain five successive sampling windows for each of the four selected cases that are illustrated in Figures 4.8(a)-4.11(a) for dataset 1, Figures 4.12(a)-4.15(a) for dataset 2, and Figures 4.16(a)-4.19(a) for dataset 3.

We have considered 4 different sampling road sections for each dataset. For dataset 1, the starting and end points of the first sampling road section are located at $x^{\mathrm{u}}_{1,1} = 50$ m and $x^{\mathrm{d}}_{1,1} = 250$ m (see Figure 4.8(a)), with the upstream and downstream loop detectors located at $x^{\mathrm{u}}_{1,1}$ and $x^{\mathrm{d}}_{1,1}$, respectively. The second sampling road section (see Figure 4.9(a)) corresponds to the starting and end points $x^{\mathrm{u}}_{1,2} = 400$ m and $x^{\mathrm{d}}_{1,2} = 600$ m, with the upstream and downstream loop detectors at $x^{\mathrm{u}}_{1,2}$ and $x^{\mathrm{d}}_{1,2}$. The third sampling road section (see Figure 4.10(a)) starts at $x^{\mathrm{u}}_{1,3} = 800$ m and ends at $x^{\mathrm{d}}_{1,3} = 1000$ m, where the upstream loop detector is located at $x^{\mathrm{u}}_{1,3}$ and the downstream loop detector is located at $x^{\mathrm{d}}_{1,3}$. Finally, the fourth sampling road section (see Figure 4.11(a)) has the starting and end points at $x^{\mathrm{u}}_{1,4} = 1000$ m and $x^{\mathrm{d}}_{1,4} = 1200$ m, with the upstream and downstream loop detectors located at $x^{\mathrm{u}}_{1,4}$ and $x^{\mathrm{d}}_{1,4}$. The time intervals considered for the first sampling road section is between $t = 920$ s and $t = 945$ s (see Figure 4.8(a)), for the second sampling road section it is between $t = 955$ s and $t = 980$ s (see Figure 4.9(a)), for the third sampling road it section is between $t = 925$ s and $t = 950$ s (see Figure 4.10(a)), and for the fourth sampling road section it is between $t = 955$ s and $t = 980$ s (see Figure 4.11(a)).

For dataset 3, the first sampling road section (see Figure 4.16(a)) corresponds to $x^{\mathrm{u}}_{3,1} =$ 950 m and $x^{\mathrm{d}}_{3,1} = 1150$ m, and the time interval between $t = 1875$ s and $t = 1900$ s. The second sampling road section (see Figure 4.17(a)) corresponds to $x^{\mathrm{u}}_{3,2} = 1300$ m and $x^{\mathrm{d}}_{3,2} = 1450$ m, and the time interval between $t = 2400$ s and $t = 2425$ s. The third sampling road section (see Figure 4.18(a)) corresponds to $x^{\mathrm{u}}_{3,3} = 150$ m and $x^{\mathrm{d}}_{3,3} = 300$ m, and the time interval between $t = 1905$ s and $t = 1930$ s. The fourth sampling road section (see Figure 4.19(a)) corresponds to $x^{\mathrm{u}}_{3,4} = 1000$ m and $x^{\mathrm{d}}_{3,4} = 1150$ m, and the time interval between $t = 2060$ s and $t = 2085$ s.

Figures 4.8(b)-4.11(b), 4.12(b)-4.15(b), and 4.16(b)-4.19(b) illustrate the relative errors w.r.t. the real value of the generalized average speed, which is computed by (2.6), i.e., the ratio of the absolute difference between the generalized average speed and the computed average speed via each of the formulas (by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm), and the generalized average speed for datasets 1, 2, and 3, respectively. For the 12 cases shown in Figures 4.8-4.19, the errors corresponding to the first sampling windows are not shown because the main aim of the assessment is to investigate the efficiency of the new sequential algorithm, including its capability for computing the initial conditions for the next sampling window and in using the computed initial conditions for the current sampling window. However, for the first sampling windows in Figures 4.8(a)-4.19(a), the initial conditions are just given to the algorithm as an input of the problem, and therefore are not estimated by the algorithm itself. Hence, the first sampling windows in the four cases should not be considered for assessment of the algorithm.

From Figure 4.8(b), we see that the proposed sequential algorithm shows the best performance for the 2$^{\text{nd}}$, 3$^{\text{rd}}$, and 4$^{\text{th}}$ windows, while for the 5$^{\text{th}}$ window, the formula proposed by Jamshidnejad and De Schutter [63] performs better. This can be explained by taking into account the main focus of each of these two approaches. Jamshidnejad and De Schutter [63] mainly focus on each sampling window from a microscopic point-of-view, and make accurate computations for each sampling window via partitioning it into smaller windows. The new sequential algorithm proposed in this chapter, however, focuses on the common edges of the sampling windows within the time-space plane, and those trajectories that intersect these edges. Therefore, it does consider many details for each sampling window, but instead it considers the details regarding transition of vehicle trajectories from one sampling window to the neighboring window. Consequently, if there are more trajectories that intersect the common edge of two neighboring sampling windows, we expect the proposed sequential algorithm to produce more accurate results compared with other approaches. In Figure 4.8(b), by considering the number of trajectories that intersect the left-hand edges of the sampling windows w.r.t. the total number of trajectories observed in each of these windows, we see that for the 2$^{\text{nd}}$, 3$^{\text{rd}}$, and 4$^{\text{th}}$ windows, this ratio is much larger (between 27% – 50%) than for the 5$^{\text{th}}$ window (6%). Therefore, we expect the effect of ignoring these trajectories in the computations to be more significant for the 2$^{\text{nd}}$, 3$^{\text{rd}}$, and 4$^{\text{th}}$ windows, and that the sequential algorithm shows a better performance. This expectation is well supported by the results illustrated in Figure 4.8(b). Moreover, from Figure 4.8, we see that when the formula proposed by Soriguera and Robusté [141] is combined with the formula by Rakha and Zhang [134], it produces more accurate results for 3 out of 4 experiments (see the results corresponding to the 3$^{\text{rd}}$, 4$^{\text{th}}$, and 5$^{\text{th}}$ windows).

In Figure 4.9(b), the performance of the new sequential algorithm and the formula by Jamshidnejad and De Schutter [63] are very close, i.e., for some windows the new sequential algorithm shows the best performance, and for some other windows the formula by Jamshidnejad and De Schutter [63] works better. For this case we see that, on the one hand, there is a relatively large number of vehicles for which the trajectory intersects the right-hand edge of the sampling windows. On the other hand, there are some vehicles that change their speed while traveling within the sampling windows (i.e., their trajectories do not have a straight linear shape). Since the sequential algorithm covers the first issue (intersecting trajectories) and the formula by Jamshidnejad and De Schutter [63] uses a convex combination of the lower and upper bounds of the generalized average speed (see Chapter 3), we can expect to see such a close performance for these two approaches.

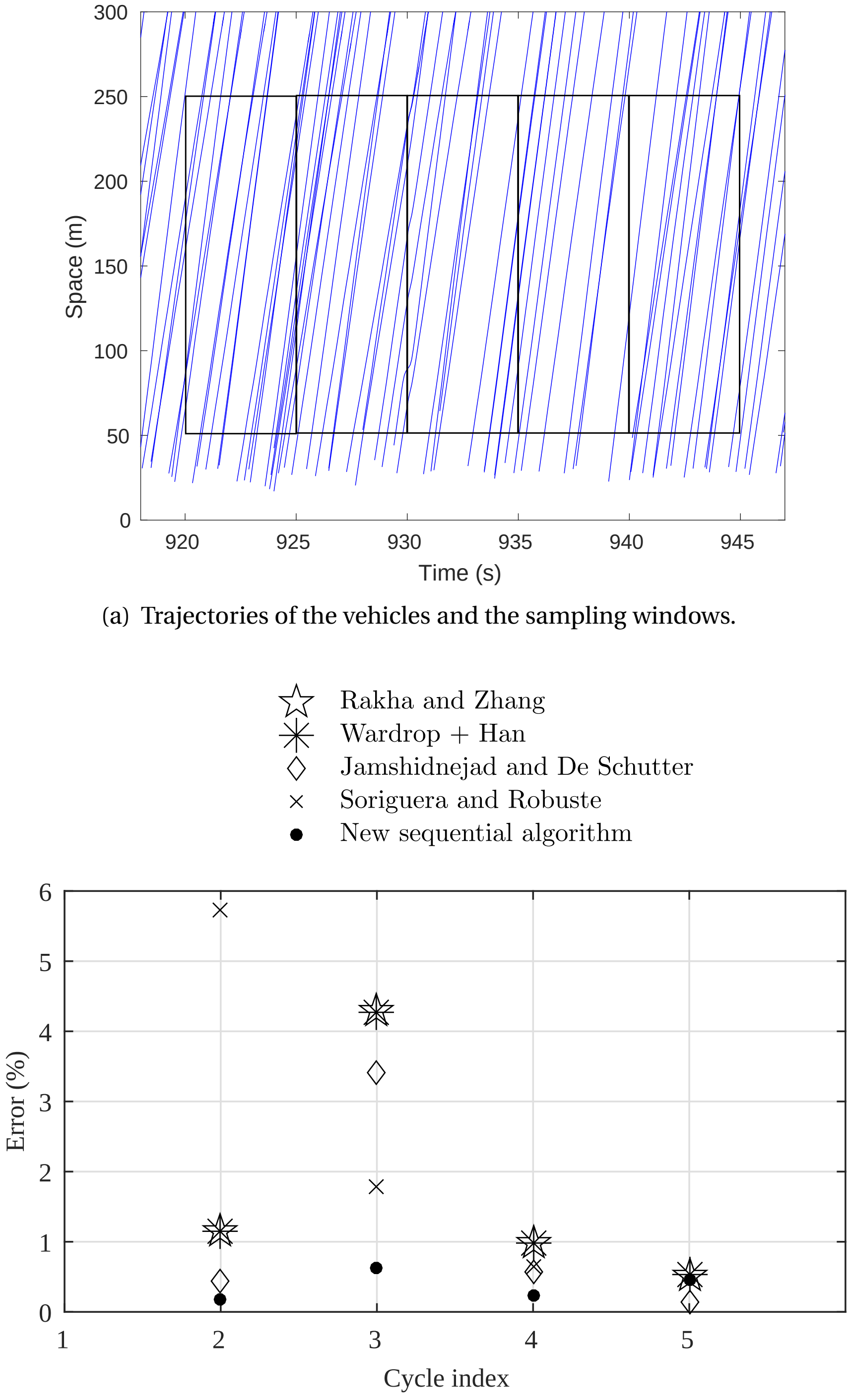
In Figure 4.10(b), for 3 out of the 4 sampling windows the best results correspond to the new sequential algorithm. As we see from the curves of trajectories, the ratio of the trajectories intersecting the left-hand edge of each window and the total number of trajectories in that window is relatively large (between 23% and 40%). Therefore, we could expect the new sequential algorithm, which keeps track of the trajectories that intersect the left-hand edge of the sampling windows, to produce the best results. For Figure 4.11(b), in 3 out of 4 windows, the sequential algorithm shows a significantly better performance, while for one window the formula by Jamshidnejad and De Schutter [63] is more accurate (note that the performance of the sequential algorithm proposed in this chapter is still very close to the performance of the formula by Jamshidnejad and De Schutter [63]).

Figures 4.12-4.15, which correspond to dataset 2, show that in 12 cases out of 16 cases (i.e., in 75% of the cases), the proposed sequential algorithm exhibits the best performance, and for 3 out of 16 cases the formula by Jamshidnejad and De Schutter [63] performs better. From Figures 4.16-4.19, which correspond to dataset 3, in 13 cases out of 16 cases (i.e., in 81.25% of the cases), the proposed sequential algorithm performs the best. Note that compared with the results obtained for dataset 1, where the relative error of the different formulas is always less than 6%, for datasets 2 and 3 this error may become close to or even exceed 50% for some formulas (but it never happens for the proposed sequential algorithm). In particular, in a few cases, for example for dataset 2, the 3rd window of case 1 (see Figure 4.12(b)) and the 5th window of case 4 (see Figure 4.15(b)), and for dataset 3, the 5th window of case 3 (see Figure 4.18(b) and the 3rd window of case 4 (see Figure 4.19(b)), for the formula proposed by Soriguera and Robusté [141] combined with the formula by Rakha and Zhang [134] this error exceeds 50%. However, for illustration purposes we have shown it at the highest percentage used for the illustrations (i.e., 50%). In all these cases, however, the relative error of the proposed sequential algorithm almost never exceeds 10% (except for the 5th window of case 2 of dataset 3 (see Figure 4.17(b))), where the relative error of the proposed sequential algorithm reaches almost 18%; note, however, that this is still the lowest percentage among all the formulas.

In general, considering Figures 4.8-4.19, we see that the new sequential algorithm shows excellent performance in most cases compared to the other formulas. In most experiments, either the new sequential procedure or the formula by Jamshidnejad and De Schutter [63] produces the most accurate results, while the other formulas are less accurate. For situations in which the formula by Jamshidnejad and De Schutter [63] is more accurate, the difference between its result and the result of the sequential algorithm is rather small (less than 5%). However, for the cases where the sequential algorithm is the most accurate approach, the difference between the results produced by the algorithm and by the formula by Jamshidnejad and De Schutter [63] is larger, i.e., the error difference could be more than 10% based on the results of the case study (e.g., see the 5th window of Figure 4.18(b)). The combination of the formulas given by Rakha and Zhang [134] and by Soriguera and Robusté [141] shows the best performance in 2 out of the 48 experiments (the proposed sequential algorithm in these cases ranks second, with the difference of the errors of the combined formula by Rakha and Zhang [134] and by Soriguera and Robusté [141], and the sequential algorithm being less than 0.3%), and in 1 case out of 48 cases, both the proposed sequential algorithm and the combined formula by Rakha and Zhang [134] and Soriguera and Robusté [141] perform best.

4.6 Conclusions and future work

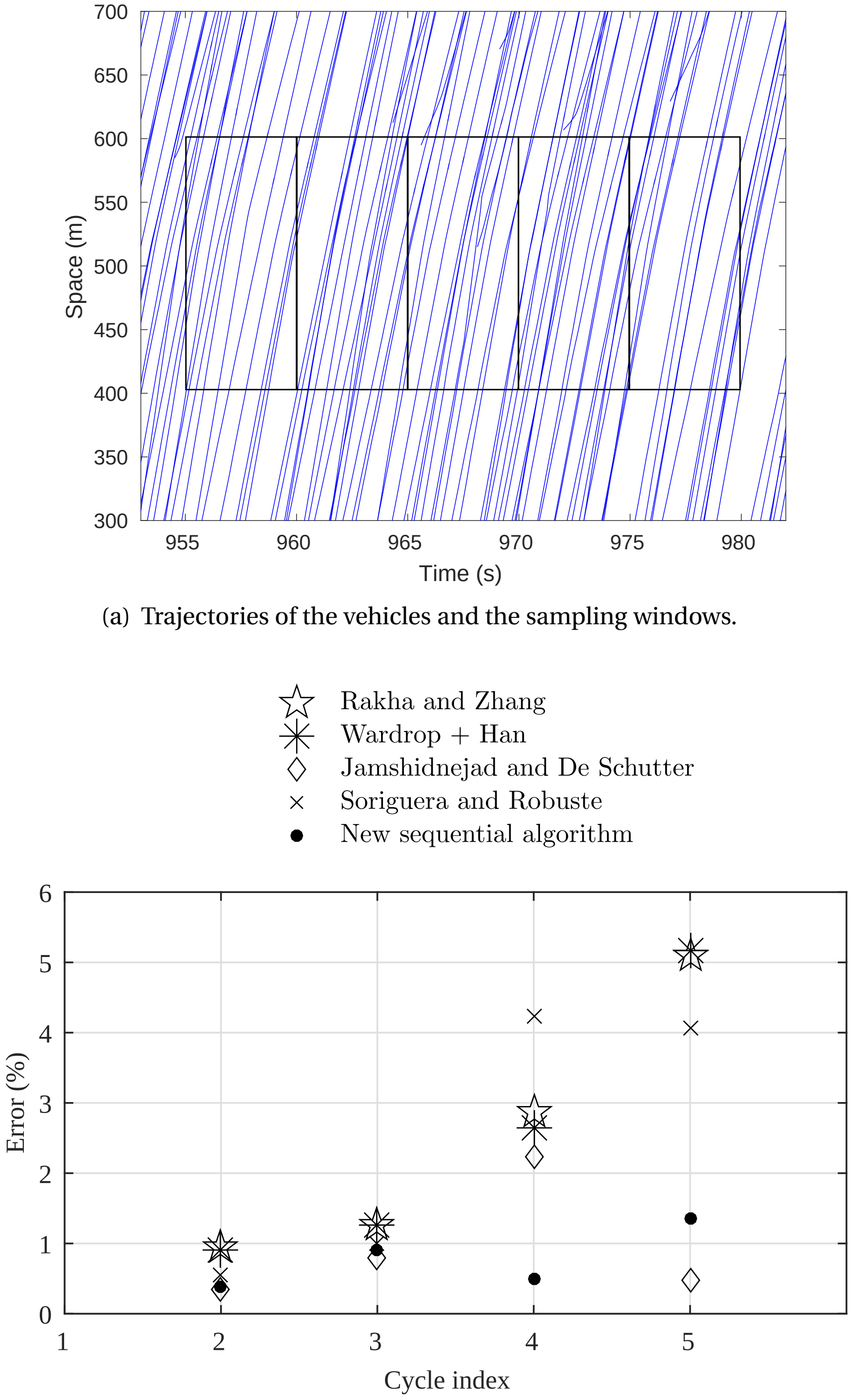
In this chapter, a new sequential algorithm has been proposed for finding an accurate estimate of the generalized traffic fundamental variables (i.e., the generalized density, flow, and average speed), taking into account the effect of those vehicles that remain on the same sam-



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

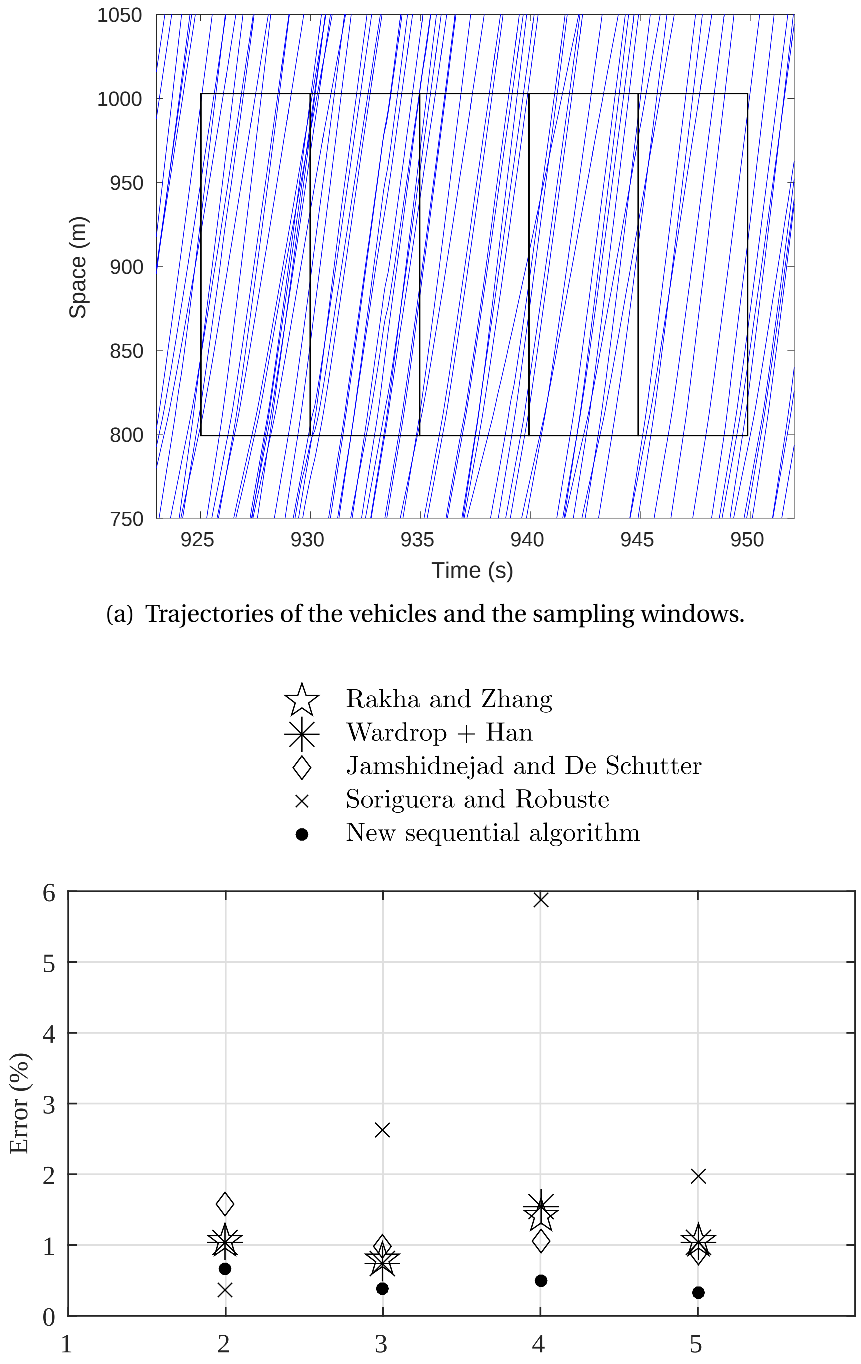
Figure 4.8: ***Dataset 1, first sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

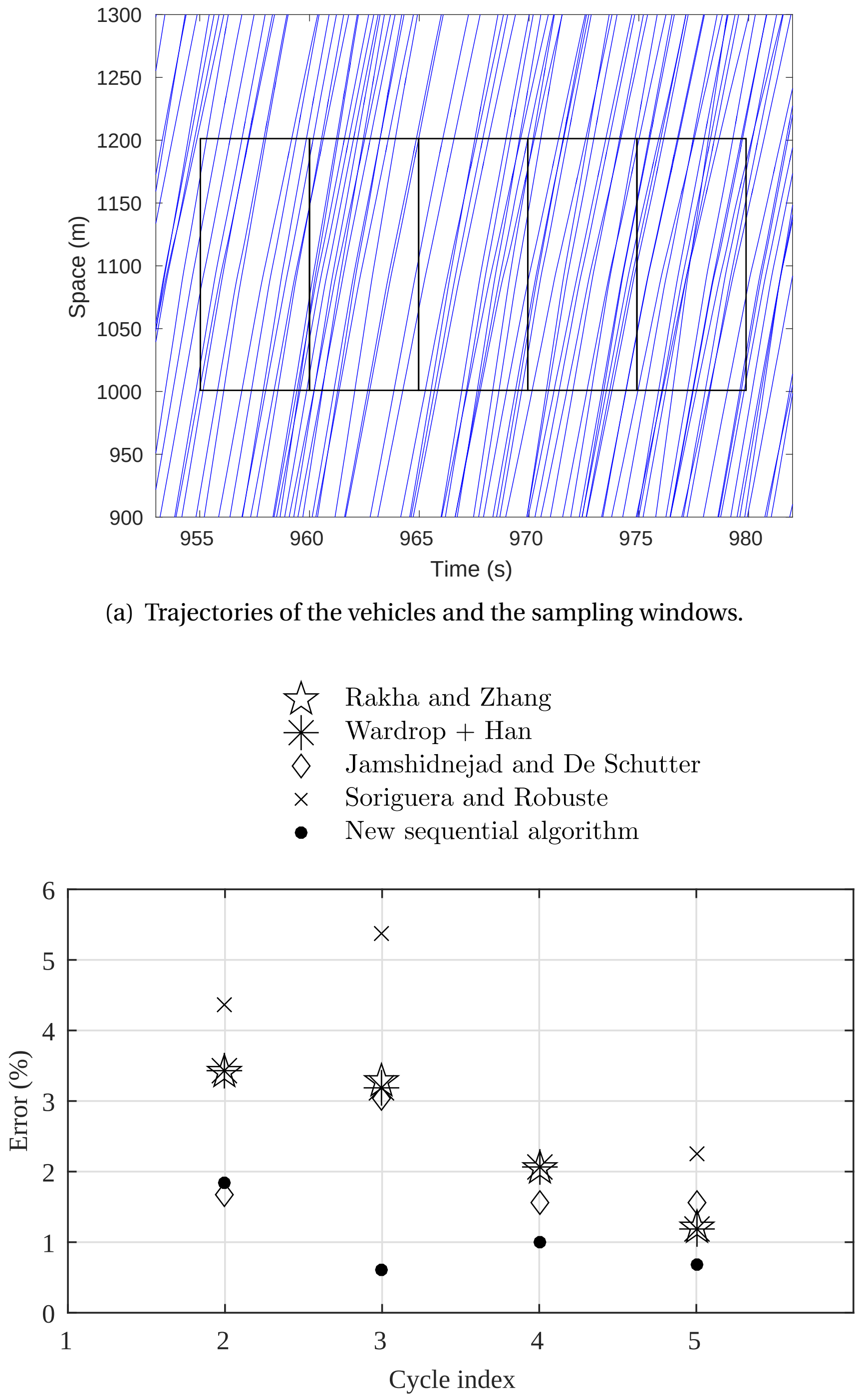
Figure 4.9: ***Dataset 1, second sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

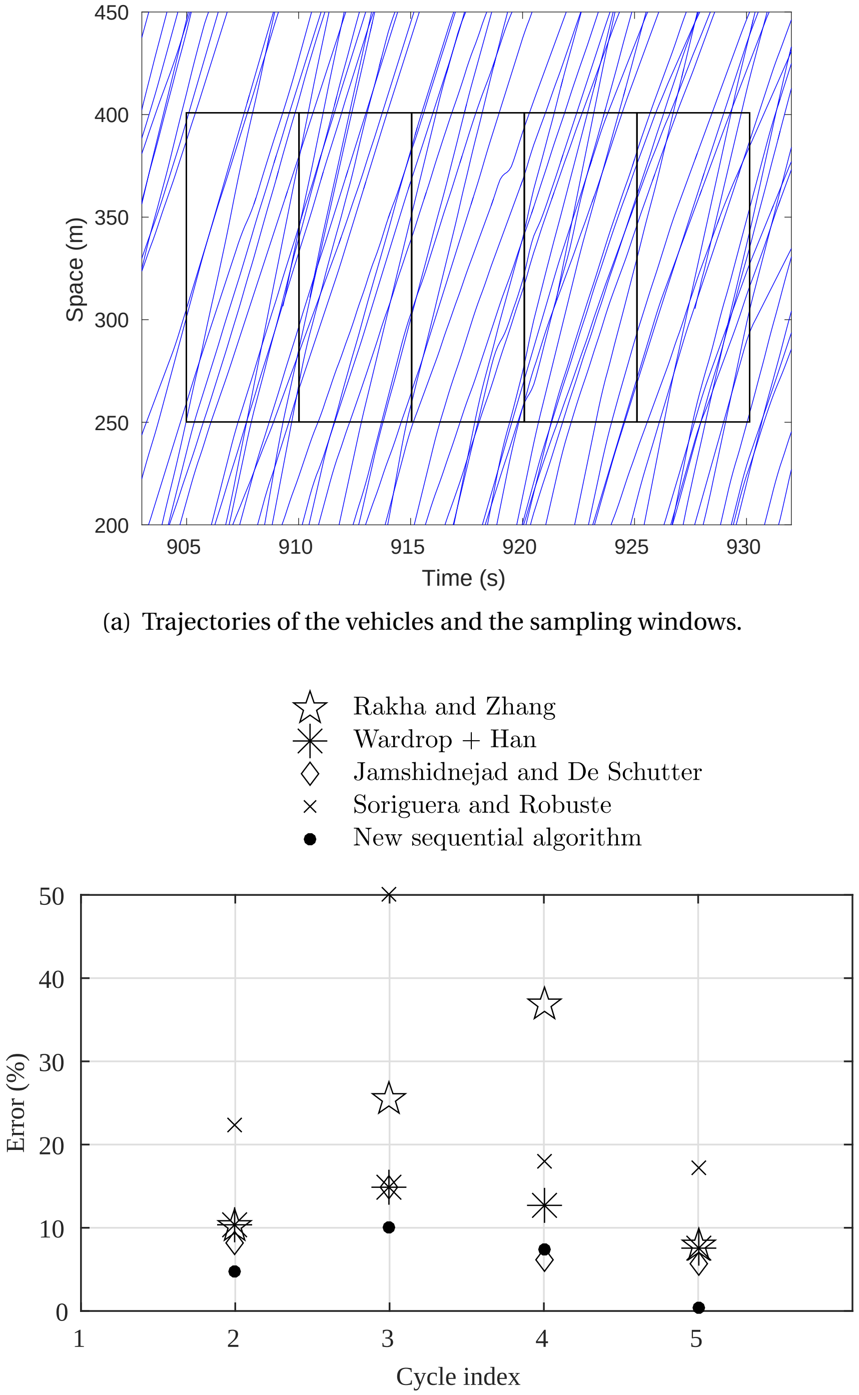
Figure 4.10: ***Dataset 1, third sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

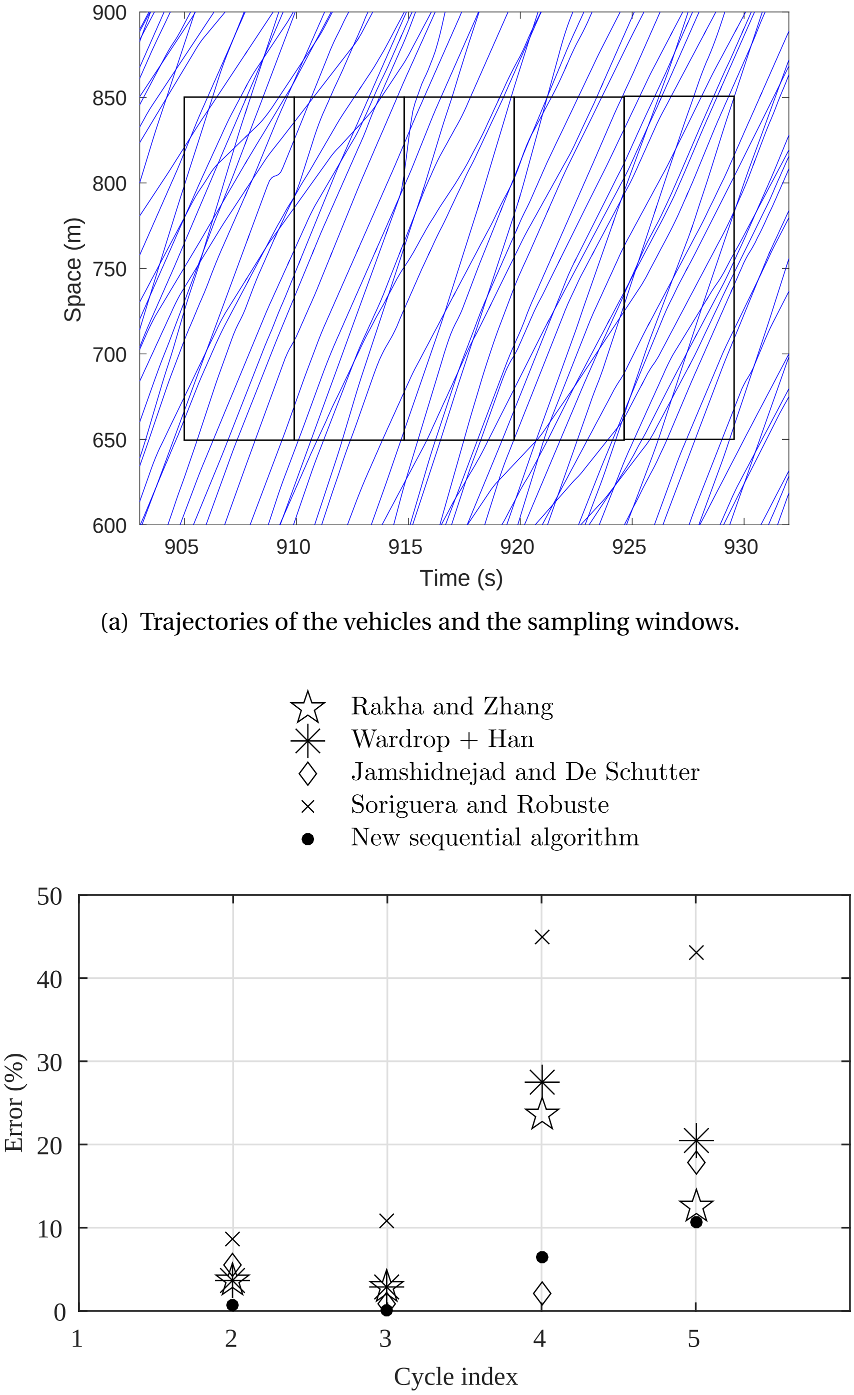
Figure 4.11: ***Dataset 1, fourth sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

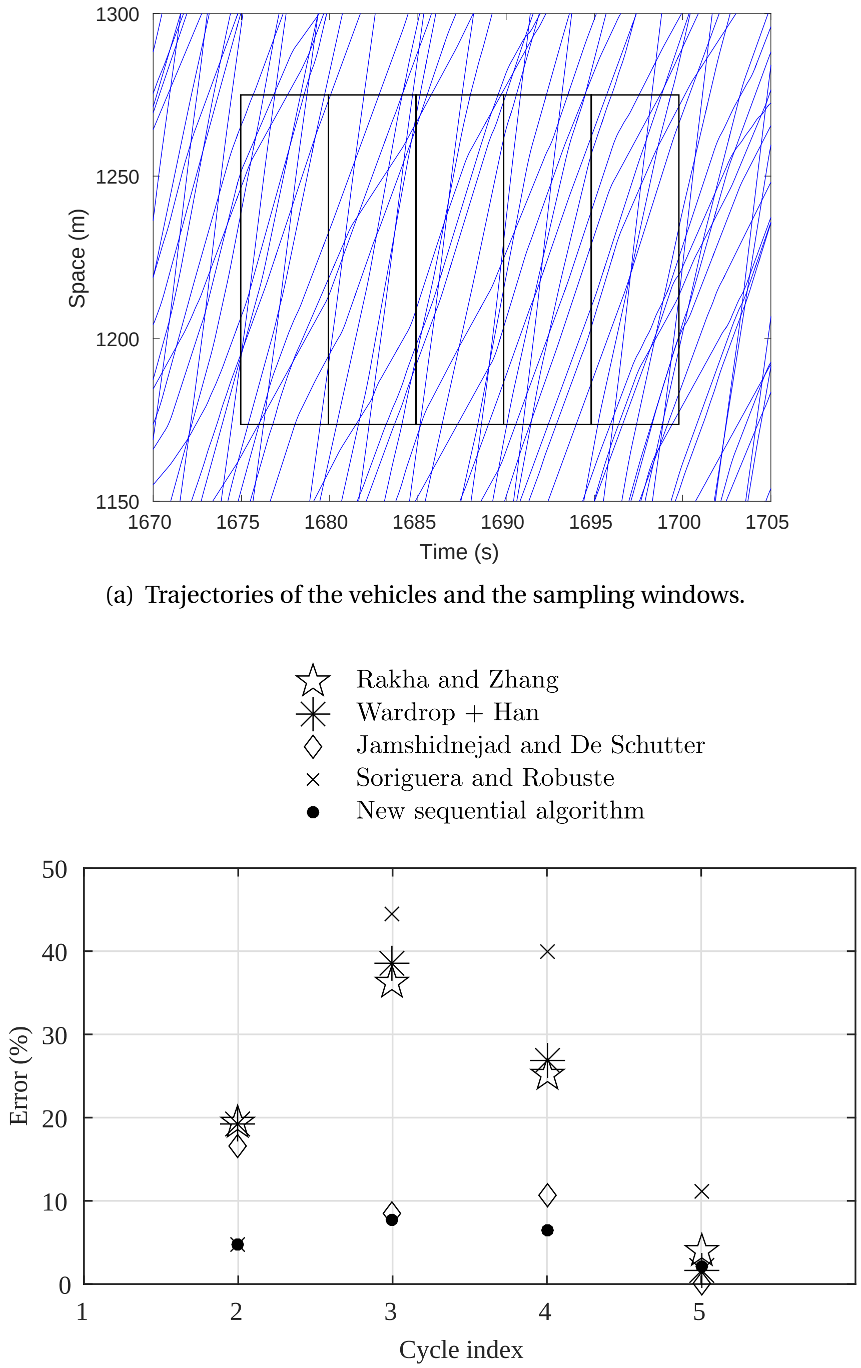
Figure 4.12: ***Dataset 2, first sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm (note that for the* 3^{rd} *window, the error of the formula by Soriguera and Robusté [141] exceeds* 50%*, but for the illustration purposes we have shown it at* 50%*).*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

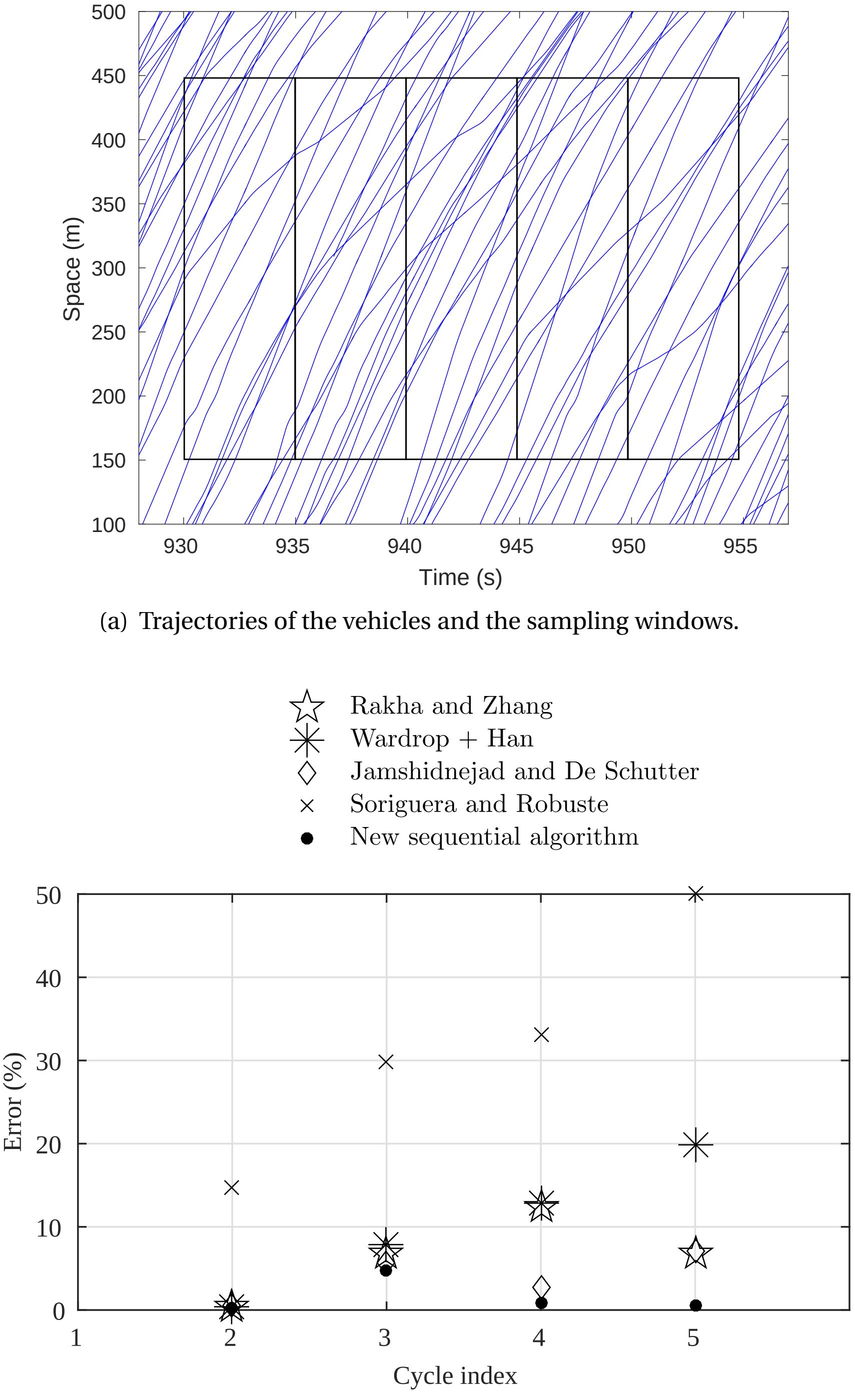
Figure 4.13: ***Dataset 2, second sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

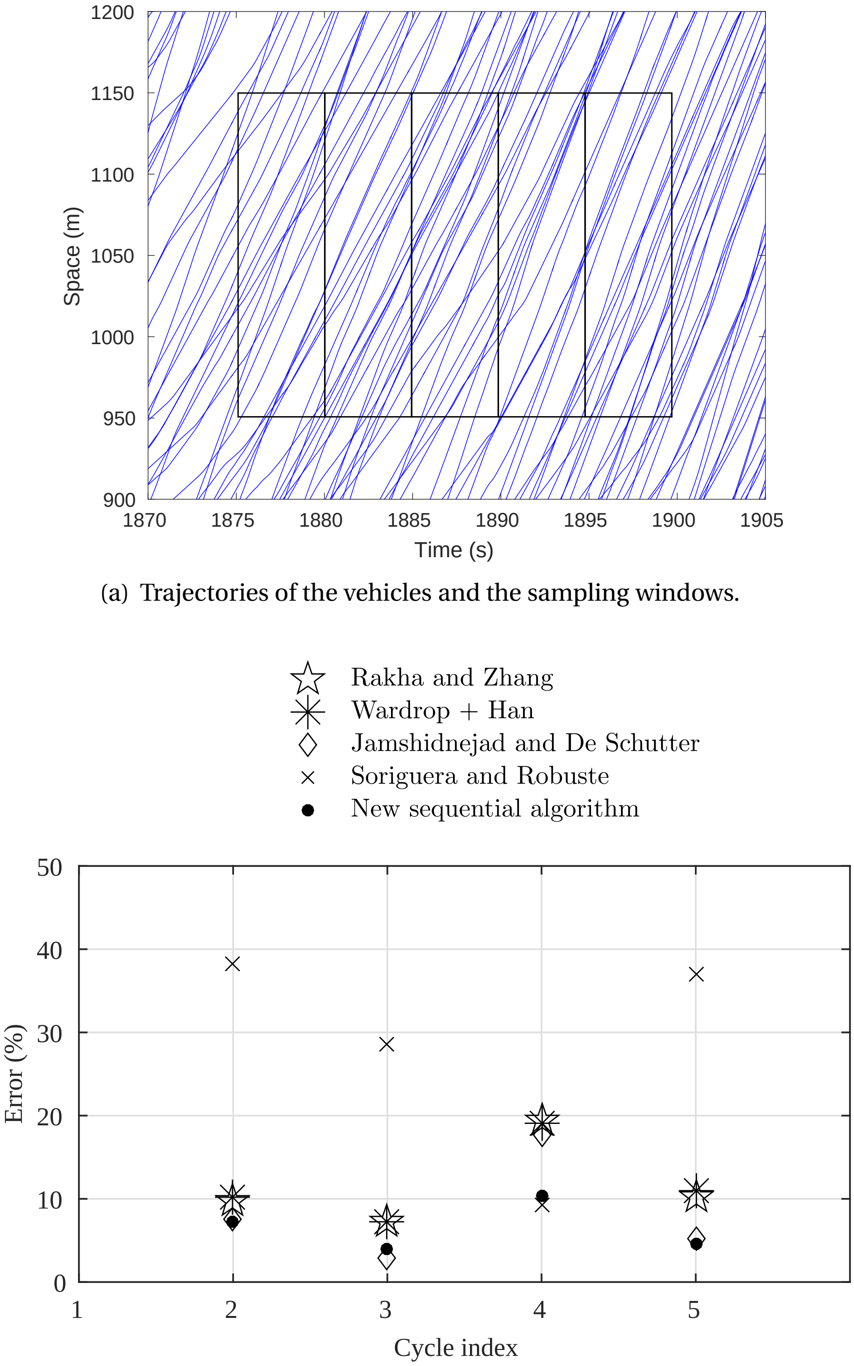
Figure 4.14: ***Dataset 2, third sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

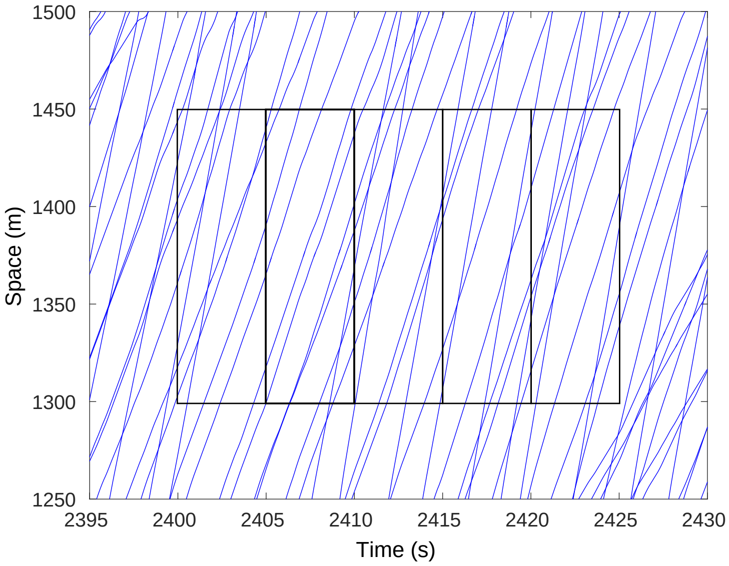
Figure 4.15: ***Dataset 2, fourth sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm (note that for the* 5^{th} *window, the error of the formula by Soriguera and Robusté [141] exceeds* 50%*, but for the illustration purposes we have shown it at* 50%*).*



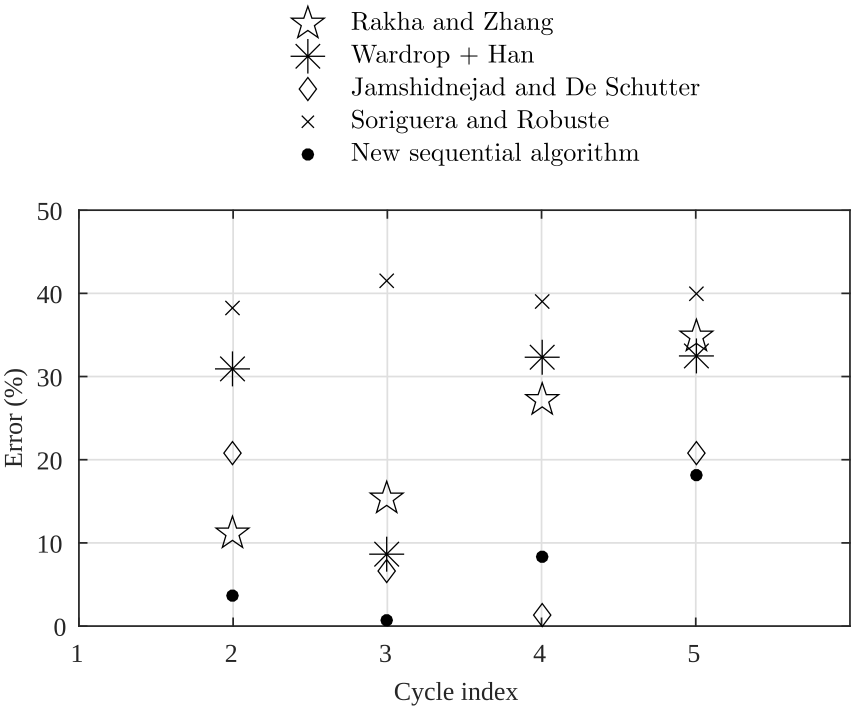
(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

Figure 4.16: ***Dataset 3, first sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*

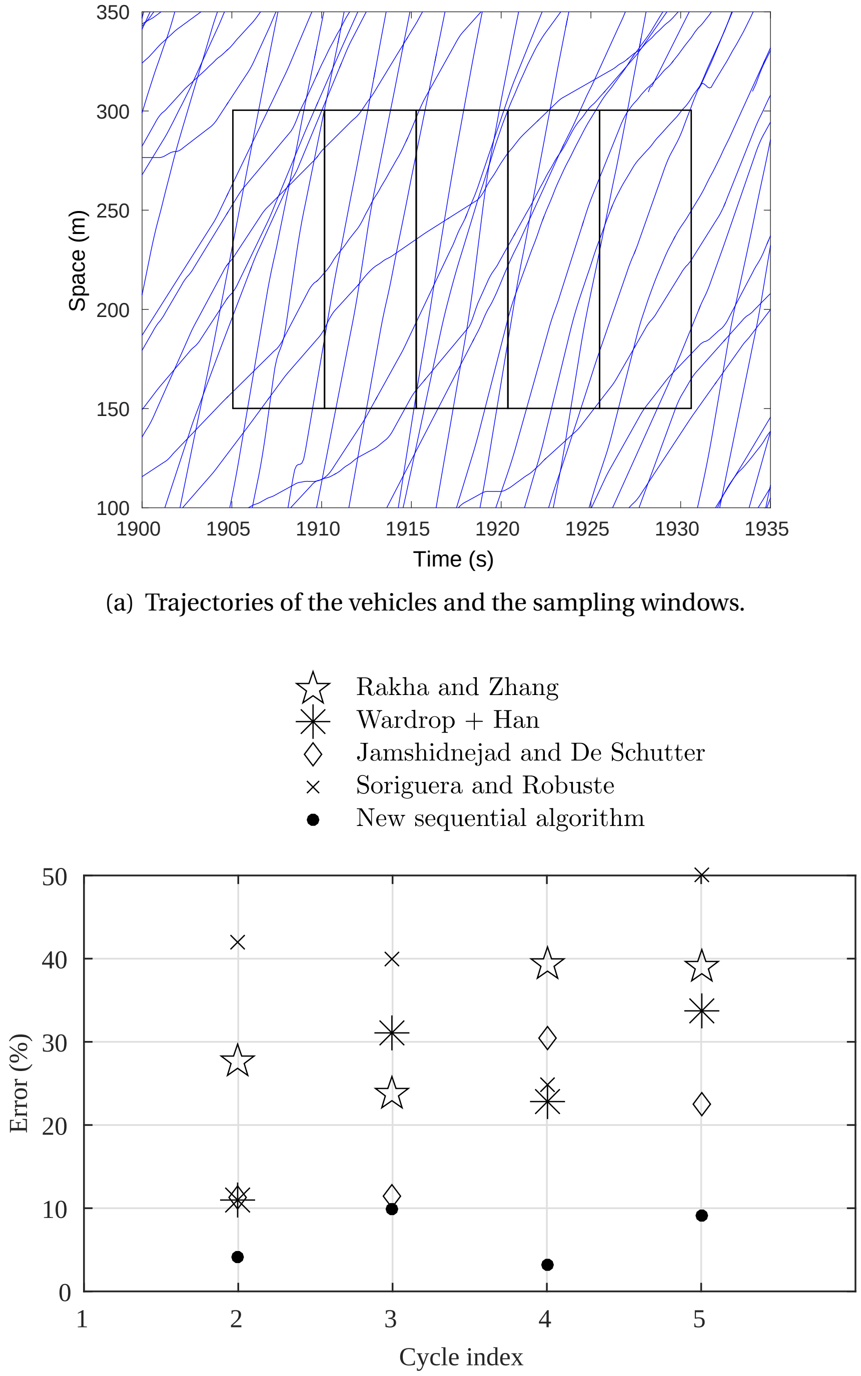


(a) Trajectories of the vehicles and the sampling windows.



(b) Relative errors in percentage.

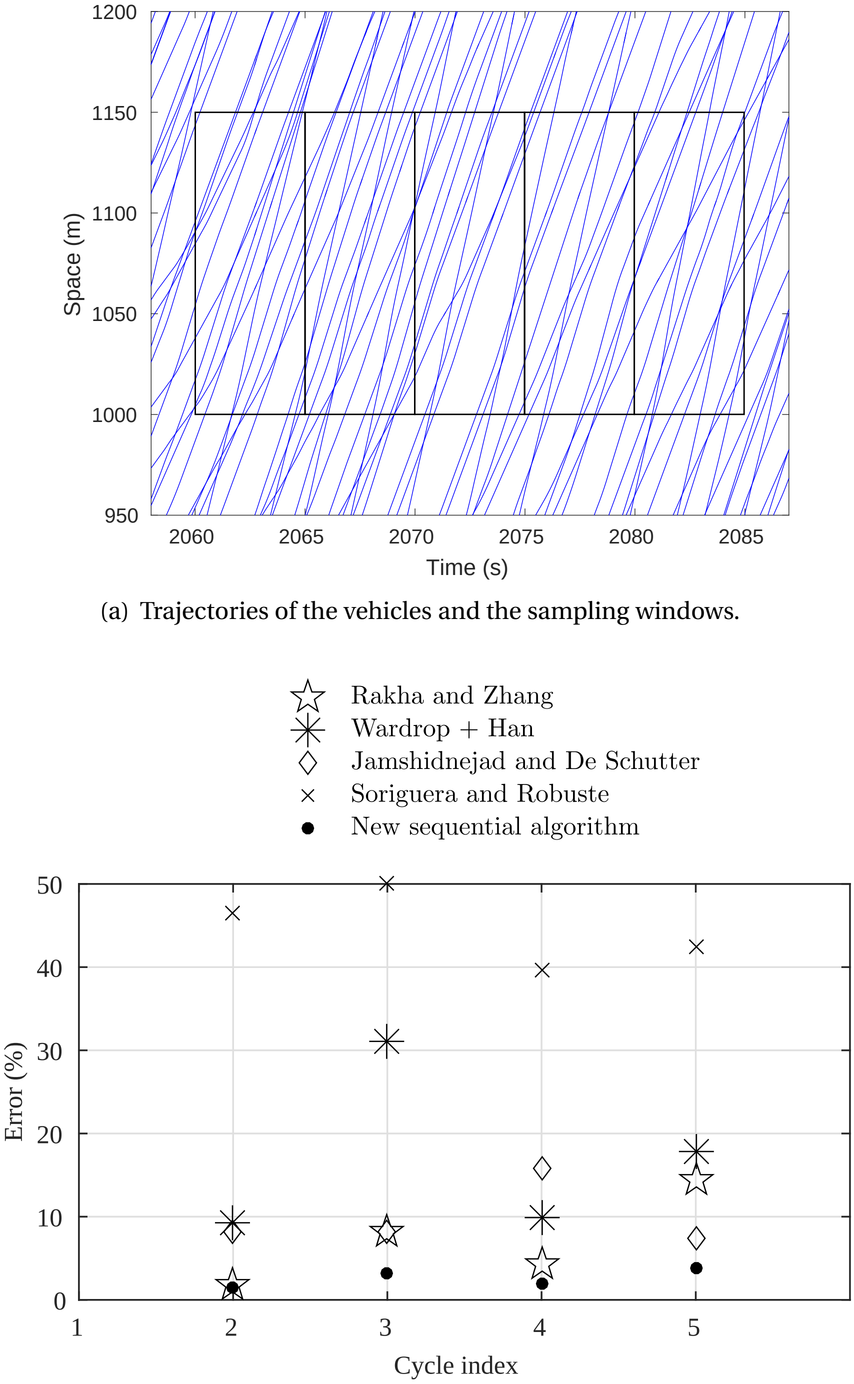
Figure 4.17: ***Dataset 3, second sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

Figure 4.18: ***Dataset 3, third sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*



(a) Trajectories of the vehicles and the sampling windows.

(b) Relative errors in percentage.

Figure 4.19: ***Dataset 3, fourth sampling road section:*** *trajectories and relative errors w.r.t. the real value of the generalized average speed for the formulas given by Rakha and Zhang [134], by Wardrop [153] and Han et al. [56], by Jamshidnejad and De Schutter [63], by Soriguera and Robusté [141], and by the new sequential algorithm.*

pling road section for more than one sampling cycle. The algorithm has been developed for both single-lane and multi-lane roads. In addition, we have also presented an approach that produces approximate trajectories of the vehicles in each sampling window using Newell's car-following model.

The results of the case study, which has used the real-life dataset NGSIM, show the excellent performance of the new sequential algorithm compared with other available formulas in the literature. In some situations, the microscopic formula by Jamshidnejad and De Schutter [63] (see also Chapter 3) performs better than the new sequential algorithm. Consequently, we propose as a topic for future work to consider the possibility of combining the microscopic formula given by Jamshidnejad and De Schutter [63] with the new sequential algorithm proposed in this chapter. As a second topic for future work, the formulas for estimating the fundamental traffic variables can be extracted for the approach proposed in Section 4.4.3 for trajectory approximation. We also suggest to combine the approach for finding approximate trajectories of vehicles based on Newell's car-following model proposed in this chapter with a lane-changing model to also take into account effects of lane change. Finally, we can consider situations where measurements of the loop detectors are noisy and see how efficient the proposed method is or what are the modifications we might need to make to improve the method for noisy measurements.

Table 4.1: Frequently used mathematical notations for Chapter 4.

A	area of a sampling region
n_A	total number of trajectories observed in the sampling region of area A
n_t	total number of trajectories observed on a stretch of the road at time t
T_j	sampling time for loop detector D_j
n_j^{cyc}	number of whole sampling cycles occurring within $\left[\bar{t}^{\mathrm{initial}}, \bar{t}^{\mathrm{final}}\right]$
$\mathrm{w}_{i,j}$	time-space sampling window corresponding to a single-lane road section j at time step i
$\mathrm{w}_{i,j,\ell}$	time-space sampling window corresponding to link ℓ of a multi-lane road section j at time step i
$\bar{t}_{i,j}$	starting time instant corresponding to the sampling window $\mathrm{w}_{i,j}$
$N_{g,i,j}$	number of vehicles belonging to group g in the sampling window $\mathrm{w}_{i,j}$
$G_{g,i,j}$	set of indices of the vehicles belonging to group g in the sampling window $\mathrm{w}_{i,j}$
ρ_A	generalized traffic density corresponding to the sampling area A
q_A	generalized traffic flow corresponding to the sampling area A
$\bar{v}_A$	generalized average traffic speed/TSMS corresponding to the sampling area A
$\rho_{i,j}$	generalized density corresponding to the sampling window $\mathrm{w}_{i,j}$
$q_{i,j}$	generalized flow corresponding to the sampling window $\mathrm{w}_{i,j}$
$\bar{v}_{i,j}$	generalized average speed corresponding to the sampling window $\mathrm{w}_{i,j}$
$A_{i,j}$	area of window $\mathrm{w}_{i,j}$
$\theta_{c,i,j}$	time instant at which the trajectory of vehicle c belonging to group 1 of the sampling window $\mathrm{w}_{i,j}$ enters the sampling window within the time-space plane
$L_{n^{\mathrm{loop}}}^{\mathrm{endpoint}}$	distance between the last loop detector and the endpoint of the road
$s_{(c,\bar{\imath}),g,i,j}^{\mathrm{start}}$	relative position at the start of the current sampling cycle and w.r.t. the beginning of the single-lane sampling road section j for vehicle c that has entered the road section in cycle $\bar{\imath}$ and that belongs to group g of the sampling window $\mathrm{w}_{i,j}$
$s_{(c,\bar{\imath}),g,i,j}^{\mathrm{end}}$	relative position at the end of the current sampling cycle and w.r.t. the beginning of the single-lane sampling road section j for vehicle c that has entered the road section in cycle $\bar{\imath}$ and that belongs to group g of the sampling window $\mathrm{w}_{i,j}$
$d_{(c,\bar{\imath}),g,i,j}$	traveled distance of vehicle c during one sampling cycle that has entered the single-lane sampling road section j in cycle $\bar{\imath}$ and that belongs to group g of the sampling window $\mathrm{w}_{i,j}$
$t_{(c,\bar{\imath}),g,i,j}$	travel time of vehicle c during one sampling cycle that has entered the single-lane sampling road section j in cycle $\bar{\imath}$ and belongs to group g of the sampling window $\mathrm{w}_{i,j}$
$s_{(c,\bar{\imath}),g,i,j,\ell}^{\mathrm{start}}$	relative position at the start of the current sampling cycle and w.r.t. the beginning of the multi-lane sampling road section j for vehicle c that has entered the road section in cycle $\bar{\imath}$ and belongs to group g of the sampling window $\mathrm{w}_{i,j}$ and moves on lane ℓ
$s_{(c,\bar{\imath}),g,i,j,\ell}^{\mathrm{end}}$	relative position at the end of the current sampling cycle and w.r.t. the beginning of the multi-lane sampling road section j for vehicle c that has entered the road section in cycle $\bar{\imath}$ and belongs to group g of the sampling window $\mathrm{w}_{i,j}$ and moves on lane ℓ
$d_{(c,\bar{\imath}),g,i,j,\ell}$	traveled distance of vehicle c that has entered the multi-lane sampling road section j in cycle $\bar{\imath}$ and belongs to group g of the sampling window $\mathrm{w}_{i,j}$ and moves on lane ℓ
$t_{(c,\bar{\imath}),g,i,j,\ell}$	travel time of vehicle c that has entered the multi-lane sampling road section j in cycle $\bar{\imath}$ and belongs to group g of the sampling window $\mathrm{w}_{i,j}$ and moves on lane ℓ

Part II

Efficient MPC:
Sustainable urban traffic control

Chapter 5

Background: Traffic modeling and control

5.1 Introduction

Traffic congestion in urban areas, especially when it becomes heavy and vehicles start to idle in long queues, increases the level of fuel consumption and causes waste of time and energy. Additionally, emissions produced by the vehicles significantly increase the distribution

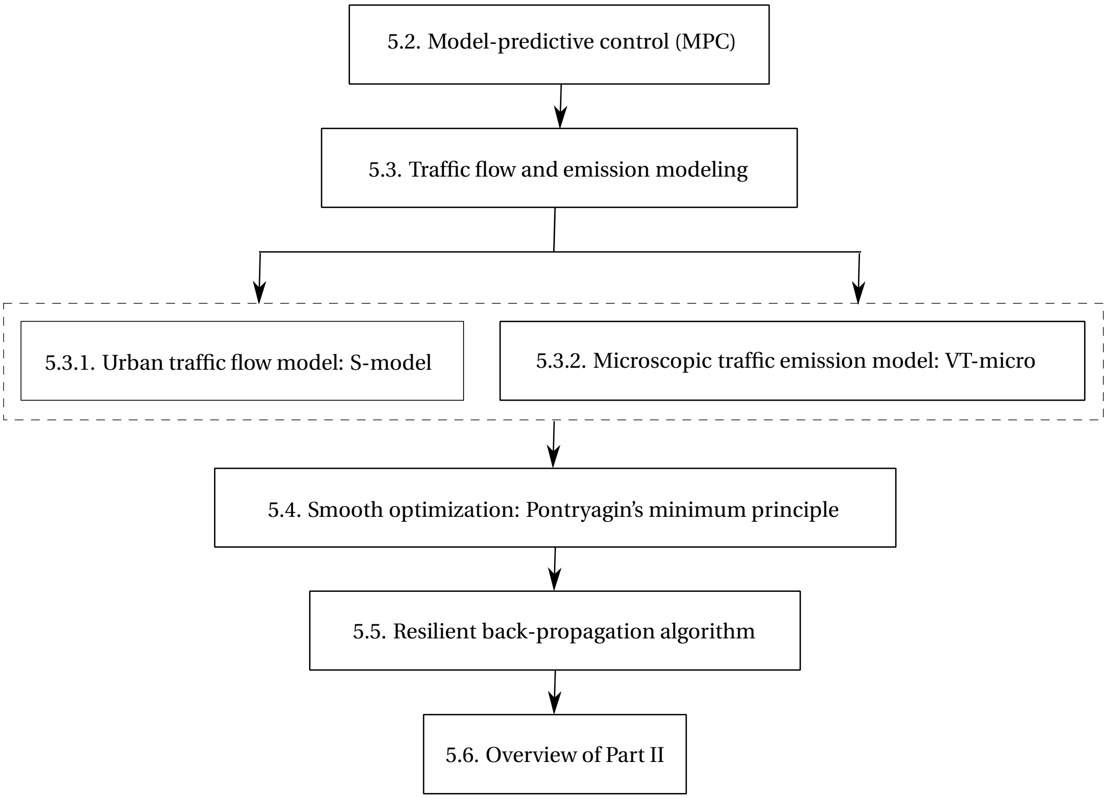

5.2. Model-predictive control (MPC)
5.3. Traffic flow and emission modeling
5.3.1. Urban traffic flow model: S-model
5.3.2. Microscopic traffic emission model: VT-micro
5.4. Smooth optimization: Pontryagin's minimum principle
5.5. Resilient back-propagation algorithm
5.6. Overview of Part II

Figure 5.1: Road map of Chapter 5.

Table 5.1: Frequently used mathematical notations for Chapter 5.

u	a node in the S-model corresponding to an upstream intersection
d	a node in the S-model corresponding to a downstream intersection
(u,d)	a link in the S-model with u as its upstream node and d as its downstream node
k_d	simulation time step counter for link (u,d)
$\mathbb{K}_d$	set of simulation time steps for the links with node d as the downstream node
c_d	sampling time of the traffic light located at the intersection corresponding to node d
$n_{u,d}(k_d)$	total number of vehicles on link (u,d) in between time steps k_d and k_d+1
$q_{u,d}(k_d)$	number of vehicles in the queue on link (u,d) in between time steps k_d and k_d+1
$q_{u,d,o}(k_d)$	number of vehicles in the queue on link (u,d) in between time steps k_d and k_d+1 that intend to move to the outgoing link (d,o) at time step k_d
$\alpha^{\text{enter,l}}_{u,d}(k_d)$	average entering flow rates of link (u,d) within the time interval $[k_dc_d,(k_d+1)c_d)$
$\alpha^{\text{leave,l}}_{u,d}(k_d)$	average exiting flow rates of link (u,d) within the time interval $[k_dc_d,(k_d+1)c_d)$
$\alpha^{\text{leave,l}}_{u,d,o}(k_d)$	average leaving flow rate of the sub-stream on link (u,d) that intends to move towards o within time interval $[k_dc_d,(k_d+1)c_d)$
$\alpha^{\text{arrive,q}}_{u,d,o}(k_d)$	average arriving flow rate at the tail of the waiting queue on link (u,d) of the sub-stream that intends to move towards o within $[k_dc_d,(k_d+1)c_d)$
$\beta_{u,d,o}(k_d)$	fraction of the vehicles on link (u,d) that intend to turn to node o
$\mu_{u,d}$	saturated leaving flow rate of the vehicles on link (u,d)
$g_{u,d,o}(k_d)$	green time length during the time interval $[k_dc_d,(k_d+1)c_d)$ for the traffic sub-stream that leaves link (u,d) towards node o
$C_{u,d}$	storage capacity of link (u,d)

of harmful substances in the air, e.g., carbon monoxide (CO) and dioxide (CO_2), hydrocarbon (HC), and nitrogen oxides (NO_x) [5, 9, 10, 140]. This is of course dangerous, especially in sensitive urban areas, such as in the neighborhood of hospitals, nursing homes for the elderly people, daycare centers, or schools. The mentioned consequences can be a source of huge economical and environmental costs in modern societies. The negative impacts of traffic congestion and emissions have been assessed for the US, the UK, Germany, and France in a report published by Center for Economics and Business Research [24]. According to this report, the expected increase of costs caused by traffic in these countries is up to 46 % by 2030 w.r.t. 2013. Hence, from both an economical and an environmental point-of-view, efficient actions should be undertaken to reduce traffic congestion and emissions in urban areas.

The main reason of congestion occurrence in urban traffic networks is inefficient use of the available roads [148]. Expanding the current road capacity requires long-term planning and can be highly costly. However, one solution that can be achieved in a relatively short time period is to make the best use of the existing road capacity by introducing more efficient traffic management and control systems. Real-time traffic-responsive control approaches are widely known for their high efficiency compared with other traffic management methods, and for their adaptability to dynamic changes in the traffic network [39, 40]. One of the real-time control categories that has been implemented extensively for different systems, including traffic networks, includes optimization-based control approaches such as model-predictive control (MPC) [105]. MPC has been used for freeway and urban traffic control and has proven to be very efficient (see, e.g., [2, 11, 37, 150]).

The rest of this chapter is organized as follows. In Section 5.2 we briefly discuss the idea of model-predictive control (MPC). Section 5.3 introduces two traffic models that will later be used in this thesis, i.e., an urban traffic flow model called the S-model [98] and a microscopic emission and fuel consumption model called VT-micro [3]. Section 5.4 explains the Pontryagin's minimum principle, which can be used for smooth optimization problems. A smooth optimization algorithm that can be used together with the Pontryagin's minimum principle, called the resilient back-propagation algorithm, is explained in Section 5.5. Finally, an overview of Part II of the thesis is given in Section 5.6. The road map of Chapter 5 is illustrated in Figure 5.1. Moreover, Table 5.1 gives the frequently used mathematical notations in Chapter 5.

5.2 Model-predictive control (MPC)

MPC is a model-based control approach that has proven to be efficient for traffic networks (see [2, 11, 39, 40, 150] for the application of MPC in urban traffic networks and freeways). In this section we explain MPC briefly.

Model-predictive control (MPC) [105] is an optimization-based control approach that minimizes a cost function within a prediction window of a finite length spanning N_p control time steps (see Figure 5.2). MPC is a dynamic optimization approach, i.e., at every control time step the MPC controller receives the state measurements as a feedback from the controlled system. These measurements are considered as the initial state for the dynamics of the controlled system to solve the optimization problem during the next N_p control time steps ahead. In Figure 5.2, $\boldsymbol{x}^{\mathrm{init}}(k_\mathrm{c})$ is the measured state at control time step k_c and is used as the initial state for solving the MPC optimization problem from time step k_c to $k_\mathrm{c} + N_\mathrm{p}$. The indicated characteristics make MPC different from optimal control strategies designed in an entirely open-loop scheme, where the optimization problem is solved in an open loop and once for the entire simulation period. Therefore, compared with classical optimal control approaches, the suboptimal solution determined by an MPC controller is more robust towards unexpected external disturbances [116].

MPC uses a prediction model to estimate the future states of the controlled system within the prediction window based on the initial states. This model should preferably provide a balanced trade-off between increasing the computation accuracy and reducing the computation time. Finally, a sequence of optimal control inputs within the prediction window is determined using (numerical) optimization. The first element of the sequence is implemented in practice for one control sampling time. At the subsequent control time step, the prediction window is shifted forward for one control time step, new state measurements are received from the controlled system, and the procedure of solving the optimization problem within

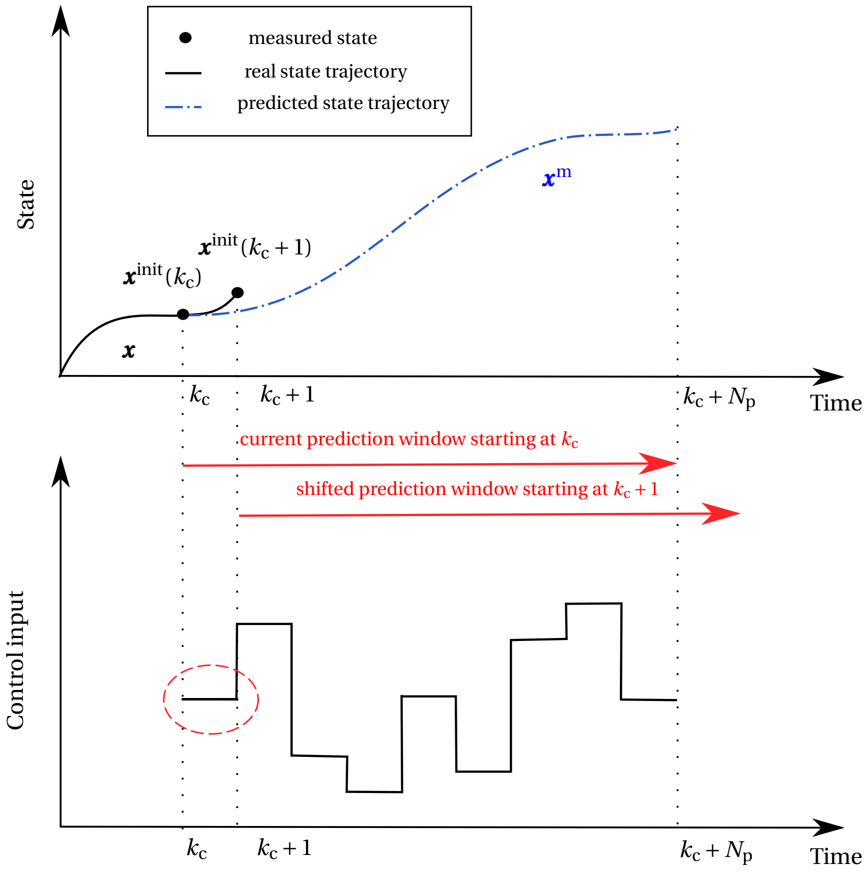


Figure 5.2: Illustration of the main idea of MPC.

the prediction window is repeated. In order to reduce the computational burden sometimes a control horizon N_c that is smaller than the prediction horizon is considered, which implies that the control input should remain constant from the control time step $k_c + N_c$ on. This way the number of optimization variables is reduced.

5.3 Traffic flow and emission modeling

To design an efficient MPC controller for sustainable urban traffic control, it is important to develop and to improve urban emission models that give an estimate of the current and possible future emission and fuel consumption levels, if integrated with an urban traffic flow model that can estimate and predict the current and future states of the traffic network.

Based on the level of detail, different traffic models (including flow and emission models) can be categorized as microscopic, mesoscopic, and macroscopic. If the model is focused on more details and takes into account the behavior of individual vehicles in the network, it is called a microscopic traffic model [49, 132, 133]. Some traffic models describe the average behavior of the vehicles as a fluid. These models are known as macroscopic traffic models (see, e.g., [115, 124]). There is also a third category for traffic models, known as mesoscopic mod-

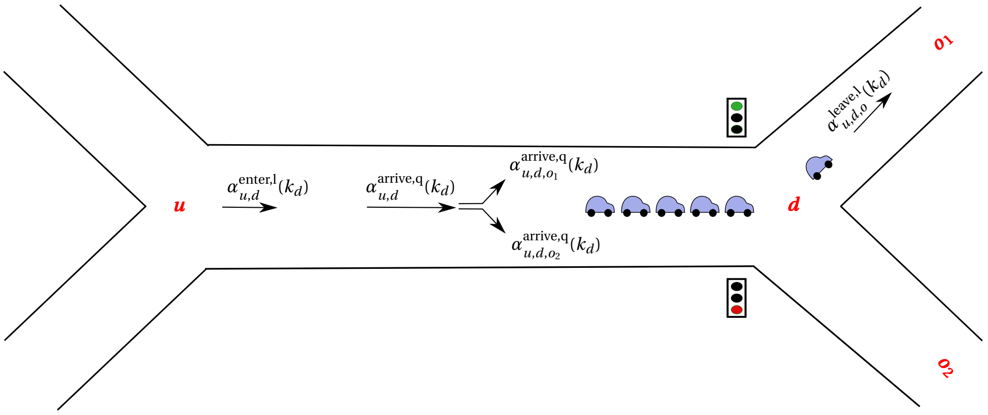

o_1
$\alpha^{\text{leave,l}}_{u,d,o}(k_d)$
u
$\alpha^{\text{enter,l}}_{u,d}(k_d)$
$\alpha^{\text{arrive,q}}_{u,d}(k_d)$
$\alpha^{\text{arrive,q}}_{u,d,o_1}(k_d)$
$\alpha^{\text{arrive,q}}_{u,d,o_2}(k_d)$
d
o_2

Figure 5.3: Illustration of a link in an urban traffic network at time step k_d, with the entering, arriving, and leaving flows.

els [59], which partly use the characteristics of microscopic and macroscopic models, i.e., the level of detail for a mesoscopic model is less than a microscopic and greater than a macroscopic traffic model. More specifically, mesoscopic models specify the behavior of traffic by groups of vehicles/drivers, where the interactions of these vehicles/drivers are described in a low level of detail.

5.3.1 Urban traffic flow model: S-model

Different models have been proposed for urban traffic flow modeling, such as the store-and-forward model [1], which has also been used in [39], the BLX model [97], the S-model [98], and a macroscopic urban traffic flow model developed by Barisone et al. [8]. The S-model is a nonlinear and discrete-time urban traffic flow model that provides an appropriate balance between low computation time and desired accuracy, and can deal well with the nonlinear behavior of traffic in urban areas. Hence, in this thesis we mainly use the S-model as the urban traffic flow model. The S-model is explained briefly in this section.

The S-model was introduced by Lin et al. [98], where the simulation sampling time of the S-model is one cycle time of the downstream intersection of a link. Therefore, the model updates the states less frequently than other macroscopic traffic models that typically have a sampling time of 1 s (such as the BLX model [97], the model of Kashani and Saridis [80], and the model proposed by van den Berg et al. [150]). This characteristic helps the S-model to be faster for model-based control applications, and to provide a trade-off between accuracy and computation time. Moreover, the simulation sampling time for each link may differ from other links.

In the S-model, a network is modeled as a collection of nodes and links, where each node represents an intersection and each link represents a road. We use the pair (u,d) to indicate a link with node u as its starting node (i.e., the upstream intersection of the corresponding road), and node d as its end node (i.e., the downstream intersection of the corresponding road). The set of all links and intersections within the network are denoted by, respectively,

$\mathcal{L}$ and $\mathcal{N}$, and the state variables of the model include the total number of vehicles $n_{u,d}(k_d)$ on link (u,d), and the number of vehicles $q_{u,d,o}(k_d)$ in the queue on link (u,d) that intend to move to the outgoing link (d,o) at time step k_d for all $(u,d) \in \mathcal{L}$. Both $n_{u,d}(k_d)$ and $q_{u,d,o}(k_d)$ are given in "number of vehicles", and admit real non-negative values instead of integers. The set of all downstream nodes of the outgoing links of link (u,d) is denoted by $\mathcal{O}_{u,d}$, and the set of all upstream nodes of the incoming links of link (u,d) is denoted by $\mathcal{I}_{u,d}$.

The state variables of the S-model are updated at every simulation time step k_d of the link (u,d) by

$$n_{u,d}(k_d+1) = n_{u,d}(k_d) + \left(\alpha_{u,d}^{\text{enter,l}}(k_d) - \alpha_{u,d}^{\text{leave,l}}(k_d)\right) c_d, \tag{5.1}$$

$$q_{u,d,o}(k_d+1) = q_{u,d,o}(k_d) + \left(\alpha_{u,d,o}^{\text{arrive,q}}(k_d) - \alpha_{u,d,o}^{\text{leave,l}}(k_d)\right) c_d, \tag{5.2}$$

with

$$q_{u,d}(k_d) = \textstyle\sum_{o\in\mathcal{O}_{u,d}} q_{u,d,o}(k_d), \tag{5.3}$$

where c_d is the cycle time of the downstream intersection d, $\alpha_{u,d}^{\text{enter,l}}(k_d)$ and $\alpha_{u,d}^{\text{leave,l}}(k_d)$ are the average entering and exiting flow rates of link (u,d) within the time interval $[k_d c_d, (k_d+1)c_d)$, and $\alpha_{u,d,o}^{\text{arrive,q}}(k_d)$ and $\alpha_{u,d,o}^{\text{leave,l}}(k_d)$ are the average arriving flow rate at the tail of the waiting queue and the average leaving flow rate of the sub-stream on link (u,d) that intends to move towards o within $[k_d c_d, (k_d+1)c_d)$. Note that these flow rates are computed for all the intermediate links of the S-model using the equations that will be given in this section, except for $\alpha^{\text{enter,l}}$ of the source links and $\alpha^{\text{leave,l}}$ of the exit links in the network. The entering flow rates $\alpha^{\text{enter,l}}$ of the source links are indeed the demand profiles that should either be given to the network as an input, or that should be determined by a prediction model. Similarly, the leaving flow rates $\alpha^{\text{leave,l}}$ of the exit links should be given as the boundary conditions of the network or should be predicted by a model. Moreover, $q_{u,d}(k_d)$ is the total number of vehicles waiting in the queue on link u,d at time step k_d. Moreover, we have (also see Figure 5.3):

$$\alpha_{u,d}^{\text{enter,l}}(k_d) = \textstyle\sum_{i\in\mathcal{I}_{u,d}} \alpha_{i,u,d}^{\text{enter,l}}(k_d), \tag{5.4}$$

$$\alpha_{u,d}^{\text{leave,l}}(k_d) = \textstyle\sum_{o\in\mathcal{O}_{u,d}} \alpha_{u,d,o}^{\text{leave,l}}(k_d), \tag{5.5}$$

$$\begin{aligned}\alpha_{u,d,o}^{\text{leave,l}}(k_d) = \min\Bigg(&\beta_{u,d,o}(k_d)\cdot\mu_{u,d}\cdot\frac{g_{u,d,o}(k_d)}{c_d},\\ &\frac{q_{u,d,o}(k_d)}{c_d} + \alpha_{u,d,o}^{\text{arrive,q}}(k_d), \frac{\beta_{u,d,o}(k_d)}{\sum\limits_{i\in\mathcal{I}_{d,o}}\beta_{i,d,o}(k_d)}\cdot\frac{C_{d,o}-n_{d,o}(k_d)}{c_d}\Bigg),\end{aligned} \tag{5.6}$$

$$\begin{aligned}\alpha_{u,d}^{\text{arrive,q}}(k_d) = &\frac{c_d-\gamma_{u,d}(k_d)}{c_d}\sum_{i\in\mathcal{I}_{u,d}}\alpha_{i,u,d}^{\text{leave,l}}(k_d-\delta_{u,d}(k_d)) +\\ &\frac{\gamma_{u,d}(k_d)}{c_d}\sum_{i\in\mathcal{I}_{u,d}}\alpha_{i,u,d}^{\text{leave,l}}(k_d-\delta_{u,d}(k_d)-1),\end{aligned} \tag{5.7}$$

$$\alpha_{u,d,o}^{\text{arrive,q}}(k_d) = \beta_{u,d,o}(k_d) \cdot \alpha_{u,d}^{\text{arrive,q}}(k_d), \tag{5.8}$$

where

$$\delta_{u,d}(k_d) = \left\lfloor \frac{\tau_{u,d}(k_d)}{c_d} \right\rfloor, \tag{5.9}$$

$$\gamma_{u,d}(k_d) = \text{rem}\{\tau_{u,d}(k_d), c_d\}, \tag{5.10}$$

with $\mu_{u,d}$ the saturated leaving flow rate of link (u,d), $g_{u,d,o}(k_d)$ the green time length during $[k_d c_d, (k_d+1)c_d)$ for the traffic sub-stream that leaves link (u,d) towards node o, $\beta_{u,d,o}(k_d)$ the fraction of vehicles within link (u,d) that intend to turn to o, $\alpha_{u,d}^{\text{arrive,q}}(k_d)$ the average within $[k_d c_d, (k_d+1)c_d)$ of the flow rate of vehicles arriving at the tail of the queue in link (u,d), and $\tau_{u,d}(k_d)$ the average delay time (from now on, we just call it the delay time) of the vehicles on link (u,d) within the interval $[k_d c_d, (k_d+1)c_d)$, i.e., the time vehicles entering the link need to reach the tail of the waiting queue. In Chapter 6, we will propose some extensions for the S-model with the aim of making this model more accurate.

In the S-model the simulation sampling time of different links might not be the same and hence, their time steps may not be synchronized automatically. However, especially for the neighboring links for which some variables are shared, synchronization of the joint variables is essential. For instance, the flow $\alpha_{i,u,d}^{\text{leave,l}}$ that leaves link (i,u) towards node d, at the same time forms a fraction of $\alpha_{u,d}^{\text{enter,l}}$ for link (u,d). However, $\alpha_{i,u,d}^{\text{leave,l}}$ is updated by the S-model at $k_u \in \mathbb{K}_u$ only, and $\alpha_{u,d}^{\text{enter,l}}$ is updated by the S-model at time steps $k_d \in \mathbb{K}_d$, where k_u and k_d may not be synchronized. Note that $\mathbb{K}_u$ and $\mathbb{K}_d$ are the set of simulation time steps for the links with, respectively, node u and node d as the downstream node.

Synchronization of the joint variables

Figure 5.4 illustrates $\alpha_{i,u,d}^{\text{leave,l}}(\cdot)$, which is updated at time steps $k_u \in \mathbb{K}_u$, while we need to find $\alpha_{i,u,d}^{\text{enter,l}}$ at time step k_d. We first make the discrete-time functions $\alpha_{i,u,d}^{\text{leave,l}}$ and $\alpha_{i,u,d}^{\text{enter,l}}$ continuous in time, using piecewise constant functions $\alpha_{i,u,d}^{\text{leave,l,c}}(\cdot)$ and $\alpha_{i,u,d}^{\text{enter,l,c}}(\cdot)$ for them. Hence, $\alpha_{i,u,d}^{\text{leave,l,c}}(\cdot)$ between any two consecutive time steps k_u and k_u+1 equals $\alpha_{i,u,d}^{\text{leave,l}}(k_u)$. Similarly, $\alpha_{i,u,d}^{\text{enter,l,c}}(\cdot)$ between any two consecutive time steps k_d and k_d+1 equals $\alpha_{i,u,d}^{\text{leave,l}}(k_d)$. Since the number of vehicles that leave link (i,u) and enter link (u,d) between any two consecutive time steps k_d and k_d+1 are equal, the highlighted surfaces in Figure 5.4 should have the same area, i.e.,[1]

$$\alpha_{i,u,d}^{\text{enter,l}}(k_d) = \frac{1}{c_d} \int_{k_d c_d}^{(k_d+1)c_d} \alpha_{i,u,d}^{\text{leave,l,c}}(t) \cdot \mathrm{d}t. \tag{5.11}$$

For more details about the S-model, we refer the readers to [98]. Moreover, some extensions to this model are proposed in this thesis in Chapter 6 to make the model more accurate.

[1] Note that (5.11) is a corrected version of the corresponding equation given in [98].

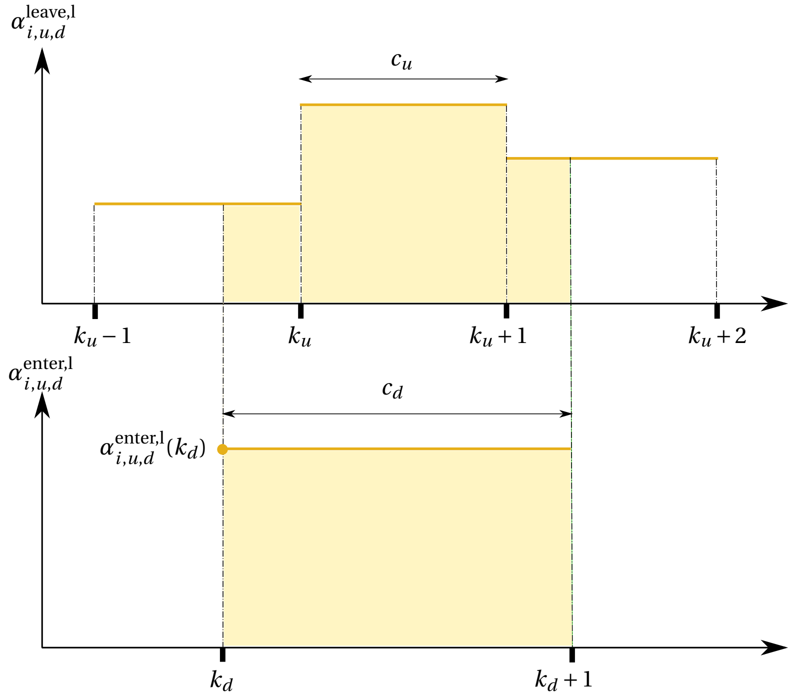

$\alpha_{i,u,d}^{\text{leave,l}}$
c_u
$k_u - 1$
k_u
$k_u + 1$
$k_u + 2$
$\alpha_{i,u,d}^{\text{enter,l}}$
c_d
$\alpha_{i,u,d}^{\text{enter,l}}(k_d)$
k_d
$k_d + 1$

Figure 5.4: Synchronization of joint variables for the neighboring links in the S-model.

5.3.2 Microscopic traffic emission model: VT-micro

In order to compute the emissions, several models from different classes (i.e., microscopic, mesoscopic, and macroscopic) have been developed (see, e.g., the macroscopic models by Ntziachristos et al. [124] and by Csikós et al. [32], the mesoscopic models by Rakha et al. [135] and by Gori et al. [51], and the microscopic models by Ahn et al. [3], by Ligterink et al. [95], and by Chen and Yu [27]). Some macroscopic models provide a high computation speed (such as COPERT [124], which is based on the average speed of the vehicles and the macroscopic model based on the total travel distance and the average speed of [32]), while they ignore the effect of acceleration and deceleration of the vehicles. This may bring issues for the accuracy of the results, especially in urban traffic areas with signalized intersections. More specifically, experiments show that in case of a positive acceleration, and in particular for the emissions of NO_x, the minimum of the emissions (in g/km) as a function of speed does not always correspond to the urban free-flow speed. Hence, minimizing the total delays of the vehicles does not necessarily lead to minimization of the emissions.

In this thesis, for computation of the emissions, we consider VT-micro [3]. VT-micro is a microscopic model that computes the instantaneous emissions of different substances such as CO, HC, and NO_x, and also the fuel consumption of vehicles based on their individual speed and acceleration. Suppose that for vehicle i, the observed values of the instantaneous speed and acceleration at time instant t are denoted by $v_i(t)$ and $a_i(t)$, respectively. First, we construct two vectors $\tilde{\boldsymbol{v}}(t)$ and $\tilde{\boldsymbol{a}}(t)$ of dimension 4 that are defined by

$$\begin{aligned} \tilde{\boldsymbol{v}}_i(t) &= \begin{bmatrix} 1 & v_i(t) & {v_i}^2(t) & {v_i}^3(t) \end{bmatrix}^\top, \\ \tilde{\boldsymbol{a}}_i(t) &= \begin{bmatrix} 1 & a_i(t) & {a_i}^2(t) & {a_i}^3(t) \end{bmatrix}^\top. \end{aligned} \tag{5.12}$$

The instantaneous emission/consumption of $e \in \{\text{CO, HC, NO}_\text{x}\text{, fuel}\}$ by vehicle i at time instant t is given by

$$E_p\left(\boldsymbol{v}_i(t), \boldsymbol{a}_i(t)\right) = \exp\left(\tilde{\boldsymbol{v}}_i^\top(t) \cdot \Pi_p \cdot \tilde{\boldsymbol{a}}_i(t)\right), \tag{5.13}$$

where Π_p are 4×4 precalibrated matrices, which are given by Zegeye [160], for computing the emissions in [kg/s] or the fuel consumption in [l/s]:

$$\Pi_\text{CO} = 0.01 \begin{bmatrix} -1292.8100 & 48.8324 & 32.8837 & -4.7675 \\ 23.2920 & 4.1656 & -3.2843 & 0.0000 \\ -0.8503 & 0.3291 & 0.5700 & -0.0532 \\ 0.0163 & -0.0082 & -0.0118 & 0.0000 \end{bmatrix},$$

$$\Pi_\text{HC} = 0.01 \begin{bmatrix} -1454.4000 & 0.0000 & 25.1563 & -0.3284 \\ 8.1857 & 10.9200 & -1.9423 & -1.2745 \\ -0.2260 & -0.3531 & 0.4356 & 0.1258 \\ 0.0069 & 0.0072 & -0.0080 & -0.0021 \end{bmatrix},$$

$$\Pi_{\text{NO}_\text{x}} = 0.01 \begin{bmatrix} -1488.3200 & 83.4524 & 9.5433 & -3.3549 \\ 15.2306 & 16.6647 & 10.1565 & -3.7076 \\ -0.1830 & -0.4591 & -0.6836 & 0.0737 \\ 0.0020 & 0.0038 & 0.0091 & -0.0016 \end{bmatrix},$$

$$\Pi_\text{fuel} = 0.01 \begin{bmatrix} -753.7000 & 44.3809 & 17.1641 & -4.2024 \\ 9.7326 & 5.1753 & 0.0109 & 0.0116 \\ 0.0053 & 0.0006 & -0.0010 & -0.0006 \end{bmatrix}.$$

5.4 Smooth optimization: Pontryagin's minimum principle

Pontryagin minimum principle (also known as Pontryagin maximum principle in some literature) is the main theory in optimal control that provides some necessary conditions on optimality. This theorem was developed and formulated by the Soviet mathematician Lev Semyonovich Pontryagin in 1956 [30]. Suppose that we want to find a suboptimal control input for a system that is described by the following discrete-time nonlinear equation:

$$\boldsymbol{x}(k+1) = \boldsymbol{f}\left(k, \boldsymbol{x}(k), \boldsymbol{u}(k)\right), \tag{5.14}$$

where $\boldsymbol{f}(\cdot)$ can in general be a nonlinear smooth function. If the formulation of the optimization problem is smooth, we can solve it using Pontryagin's minimum principle [92]. Next we explain the Pontryagin's minimum principle.

Define a (smooth) performance index for (5.14) during the prediction time interval[2] $[k_\mathrm{c}T_\mathrm{c},(k_\mathrm{c}+N_\mathrm{p})T_\mathrm{c})$ by

$$\mathcal{J}(k_\mathrm{c})=\mathcal{J}^\mathrm{t}(k_\mathrm{c})+\sum_{k=k_\mathrm{c}}^{k_\mathrm{c}+N_\mathrm{p}-1}\mathcal{J}^\mathrm{s}(k), \tag{5.15}$$

with $\mathcal{J}^\mathrm{s}(\cdot)$ the stage cost function, which at control time step $k\in\{k_\mathrm{c},\ldots,k_\mathrm{c}+N_\mathrm{p}-1\}$ can be given by an expression of the control time step k, the state vector $\boldsymbol{x}(k)$, and the control input $\boldsymbol{u}(k)$, i.e., $\mathcal{J}^\mathrm{s}(k)=J^\mathrm{s}(k,\boldsymbol{x}(k),\boldsymbol{u}(k))$, and with $\mathcal{J}^\mathrm{t}(\cdot)$ the terminal cost function, which at control time step k_c can be given by an expression of the terminal control time step $k_\mathrm{c}+N_\mathrm{p}$ and the terminal state vector $\boldsymbol{x}(k_\mathrm{c}+N_\mathrm{p})$, i.e., $\mathcal{J}^\mathrm{t}(k_\mathrm{c})=J^\mathrm{t}\left(k_\mathrm{c}+N_\mathrm{p},\boldsymbol{x}\left(k_\mathrm{c}+N_\mathrm{p}\right)\right)$. The Hamiltonian function [92] for minimizing (5.15) w.r.t. (5.14) at control time step $k\in\{k_\mathrm{c},\ldots,k_\mathrm{c}+N_\mathrm{p}-1\}$ is defined by

$$H(k,\boldsymbol{\lambda}(k+1),\boldsymbol{x}(k),\boldsymbol{u}(k))=J^\mathrm{s}(k,\boldsymbol{x}(k),\boldsymbol{u}(k))+\boldsymbol{\lambda}^\top(k+1)\cdot\boldsymbol{f}(k,\boldsymbol{x}(k),\boldsymbol{u}(k)), \tag{5.16}$$

where $\boldsymbol{\lambda}(\cdot)$ is called the costate. Pontryagin's minimum principle [79] states that for an input function $\bar{\boldsymbol{u}}(\cdot)$ to make the performance index (5.15) optimal, the following should hold for all $k\in\{k_\mathrm{c},\ldots,k_\mathrm{c}+N_\mathrm{p}-1\}$ at the same time [92]:

$$\boldsymbol{x}(k+1)=\frac{\partial H(k,\boldsymbol{\lambda}(k+1),\boldsymbol{x}(k),\bar{\boldsymbol{u}}(k))}{\partial\boldsymbol{\lambda}(k+1)}, \tag{5.17}$$

$$\boldsymbol{\lambda}(k)=\frac{\partial H(k,\boldsymbol{\lambda}(k+1),\boldsymbol{x}(k),\bar{\boldsymbol{u}}(k))}{\partial\boldsymbol{x}(k)}, \tag{5.18}$$

$$\boldsymbol{G}(k)=\frac{\partial H(k,\boldsymbol{\lambda}(k+1),\boldsymbol{x}(k),\bar{\boldsymbol{u}}(k))}{\partial\boldsymbol{u}(k)}=0, \tag{5.19}$$

where $\boldsymbol{G}(\cdot)$ is called the reduced gradient [128]. In order to numerically solve (5.17), we can start from the initial state of the system, $\boldsymbol{x}(k_\mathrm{c})$. To solve (5.18) via backward integration, we start from $\boldsymbol{\lambda}(k_\mathrm{c}+N_\mathrm{p})$, which is given by

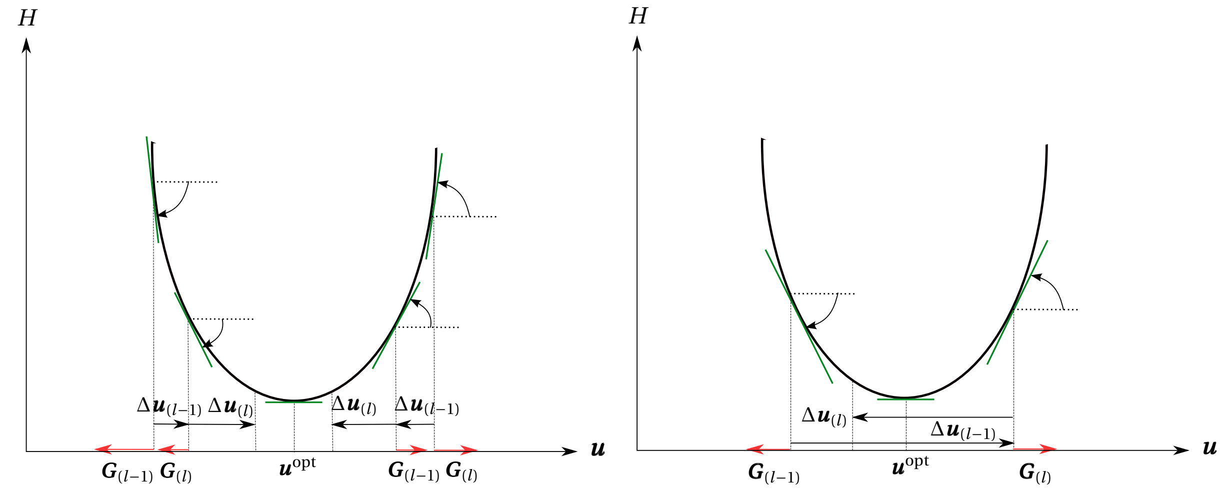
$$\boldsymbol{\lambda}(k_\mathrm{c}+N_\mathrm{p})=\frac{\partial J^\mathrm{t}\left(k_\mathrm{c}+N_\mathrm{p},\boldsymbol{x}(k_\mathrm{c}+N_\mathrm{p})\right)}{\partial\boldsymbol{x}(k_\mathrm{c}+N_\mathrm{p})}. \tag{5.20}$$

To find the optimal $\boldsymbol{u}(k)$ that makes the reduced gradient (5.19) zero, we can use different optimization algorithms, e.g., the resilient back-propagation algorithm (briefly called the RProp) [137], which is explained in the next section.

5.5 Resilient back-propagation algorithm

Using RProp, we can find the increment $\Delta\bar{\boldsymbol{u}}_{(l)}$ of the control vector at iteration l based on the values of the reduced gradients in the current and in the previous iterations, i.e., $\boldsymbol{G}_{(l)}$

[2] Note that for the sake of simplicity of the notations, here we assume that $T_\mathrm{c}=T_\mathrm{model}$, where T_c is the control sampling time and T_model is the model sampling time.


H
$\Delta\boldsymbol{u}_{(l-1)}$ $\Delta\boldsymbol{u}_{(l)}$
$\Delta\boldsymbol{u}_{(l)}$ $\Delta\boldsymbol{u}_{(l-1)}$
$\boldsymbol{u}$
$\boldsymbol{G}_{(l-1)}$ $\boldsymbol{G}_{(l)}$
$\boldsymbol{u}^{\mathrm{opt}}$
$\boldsymbol{G}_{(l-1)}$ $\boldsymbol{G}_{(l)}$
H
$\Delta\boldsymbol{u}_{(l)}$
$\Delta\boldsymbol{u}_{(l-1)}$
$\boldsymbol{u}$
$\boldsymbol{G}_{(l-1)}$
$\boldsymbol{u}^{\mathrm{opt}}$
$\boldsymbol{G}_{(l)}$

(a) RProp when the optimization variable for the previous and current iterations are on the same side of the optimum.

(b) RProp when the optimization variable for the previous and current iterations are on different sides of the optimum.

Figure 5.5: The resilient back-propagation (RProp) algorithm (note that in the figure it is assumed that u is a scalar, l is the iteration index, the green lines illustrate the tangent lines, and the red arrows are the gradients, i.e., they point in the direction of the increase of the function and their magnitudes represent the slope of the tangent line at the corresponding point on the graph of the function).

and $\boldsymbol{G}_{(l-1)}$. The elements of $\boldsymbol{G}$ and $\Delta\bar{\boldsymbol{u}}$ are indicated by, respectively, G_i and $\Delta\bar{u}_i$ for $i = 1,\ldots,|\bar{\boldsymbol{u}}|$, where $|\cdot|$ gives the number of entries of a vector. The elements of $\bar{\boldsymbol{u}}_{(l)}$ at time step $k \in \{k_{\mathrm{c}},\ldots,k_{\mathrm{c}}+N_{\mathrm{p}}-1\}$ are updated by

$$\bar{u}_{i,(l)}(k) = \mathrm{sat}\left(\bar{u}_{i,(l-1)}(k) + \Delta\bar{u}_{i,(l)}(k), \bar{u}_{\mathrm{min},i}, \bar{u}_{\mathrm{max},i}\right), \tag{5.21}$$

where

$$\mathrm{sat}(\bar{u}_i) = \begin{cases} \bar{u}_{\mathrm{max},i}, & \text{if } \bar{u}_i \geq \bar{u}_{\mathrm{max},i}, \\ \bar{u}_i, & \text{if } \bar{u}_{\mathrm{min},i} < \bar{u}_i < \bar{u}_{\mathrm{max},i}, \\ \bar{u}_{\mathrm{min},i}, & \text{if } \bar{u}_i \leq \bar{u}_{\mathrm{min},i}, \end{cases} \tag{5.22}$$

with $\bar{u}_{\mathrm{max},i}$ and $\bar{u}_{\mathrm{min},i}$ the upper and the lower bound for the element $\bar{u}_i$. Finally, for the elements of $\Delta\bar{\boldsymbol{u}}_{(\ell)}(k)$, for $i = 1,\ldots,|\bar{\boldsymbol{u}}|$ RProp gives

$$\Delta u_{i,(l)}(k) = \begin{cases} -\mathrm{sign}\left(G_{i,(l)}(k)\right)\eta^{+}|\Delta u_{i,(l-1)}(k)| & \text{for} \quad G_{i,(l-1)}(k)G_{i,(l)}(k) > 0, \\ -\eta^{-}\Delta u_{i,(l-1)}(k) & \text{otherwise}, \end{cases} \tag{5.23}$$

with $\eta^{+} > 1$ and $0 < \eta^{-} < 1$.

Figure 5.5 demonstrates the performance of the RProp algorithm for the two cases in (5.23) (note that $\boldsymbol{u}$ is considered to be a scalar in this figure). The x-axis corresponds to the control input and the y-axis corresponds to the Hamiltonian function. The green lines are the tangent lines at the corresponding point of $\boldsymbol{u}_{(l)}$ on the graph of the Hamiltonian function, and the red arrows are the reduced gradients computed by (5.19) at the corresponding point of $\boldsymbol{u}_{(l)}$, i.e., they point in the direction of the increase of the function and their magnitude equals the slope of the corresponding tangent line. In Figure 5.5(a), the red arrows on the left-hand side of the optimal control input $\boldsymbol{u}^{\text{opt}}$ correspond to a case where the reduced gradient has a negative sign for two consecutive iterations $l-1$ and l ,i.e., the resulting values of $\boldsymbol{u}_{(l-1)}$ and $\boldsymbol{u}_{(l)}$ for these two iterations are smaller than the optimal value of $\boldsymbol{u}$ (equivalently, $\boldsymbol{u}_{(l-1)}$ and $\boldsymbol{u}_{(l)}$ are located on the left-hand side of the optimum point on the x-axis). Hence, the first condition of (5.23) is satisfied and at iteration $l+1$, a control input increment in the same direction as that of iteration l, but with a magnitude larger than the control input increment at iteration l will be produced in order to speed up the movement towards the optimum point.

The red arrows on the right-hand side of the optimum point in Figure 5.5(a) correspond to a case where both reduced gradients have a positive sign for two consecutive iterations $l-1$ and l, i.e., the resulting values of $\boldsymbol{u}_{(l-1)}$ and $\boldsymbol{u}_{(l)}$ for these two iterations are larger than the optimal value of $\boldsymbol{u}$ (equivalently, $\boldsymbol{u}_{(l-1)}$ and $\boldsymbol{u}_{(l)}$ are located on the right-hand side of the optimum point on the x-axis). Hence the first condition of (5.23) is again satisfied and at iteration $l+1$, the control input increment computed by (5.23) will be in the same direction as the control input increment at iteration l, but its magnitude will be larger than the control input increment at iteration l. Therefore, a larger step towards the optimum point will be taken at iteration $l+1$.

Figure 5.5(b) represents a case where the reduced gradient at iteration $l-1$ is negative and hence the resulting value of $\boldsymbol{u}_{(l-1)}$ is smaller than the optimal value of $\boldsymbol{u}$ (i.e., $\boldsymbol{u}_{(l-1)}$ is located on the left-hand side of the optimum point on the x-axis). Then a jump from the left-hand side of the optimum point to its right-hand side occurs from iteration $l-1$ to iteration l. Hence, the reduced gradient at iteration l becomes positive and the resulting value of $\boldsymbol{u}_{(l)}$ at iteration l is larger than the optimal value of $\boldsymbol{u}$. Therefore, the second condition of (5.23) is satisfied and the control input increment at iteration $l+1$ will be in the opposite direction of the control input increment at iteration l so that it gets back closer to the optimum point. Note that the second relationship in (5.23) results in a magnitude for the control input increment at iteration $l+1$ that is smaller than that of iteration l. Hence, we can make sure that the resulting value of $\boldsymbol{u}_{(l+1)}$ will never be smaller than the value of $\boldsymbol{u}_{(l-1)}$ (i.e., the algorithm does not allow the resulting $\boldsymbol{u}_{(l+1)}$ to get farther from the optimum point than it was before).

5.6 Overview of Part II

The aim of Part II of the thesis is to propose methods for fast and accurate modeling of urban traffic networks, and also efficient approaches for urban traffic control. Hence, in Chapter 6, we introduce a novel integrating and interfacing framework for macroscopic urban traffic flow models and microscopic emission and fuel consumption models. The resulting integrated model has a mesoscopic nature, which provides a balance between accuracy and computation time of the model. In Chapter 7, we propose general smoothening methods to make nonsmooth optimization problems smooth. Hence, efficient gradient-based optimization approaches can be used to solve the resulting smooth optimization problem. We also design a smooth MPC controller for an urban traffic network that aims to find a balanced trade-off between reducing the total time spent and total emissions produced by the vehicles in the traffic network. Finally, in Chapter 8, we propose to include endpoint penalties in the formulation of the MPC optimization problem in order to approximate the solution of an infinite-horizon MPC problem using a finite-horizon MPC optimization problem. The efficiency of the approaches proposed in Chapters 6-8 are evaluated via case studies.

Chapter 6

A mesoscopic framework for integrating traffic flow and emission models

6.1 Introduction

The focus of this chapter is on the introduction and development of a new framework for integrating and interfacing any macroscopic urban traffic flow model that updates the total number of vehicles in the links and the number of vehicles standing in the queues as the states of the traffic network, e.g., the S-model by Lin et al. [98] (see Section 5 for details), with any microscopic fuel consumption and emission model that uses both the speed and the acceleration of the individual vehicles, e.g., VT-micro by Ahn et al. [3] (see Section 5 for details) and VERSIT+ by Ligterink et al. [95]. The proposed approach will result in a new mesoscopic integrated urban traffic flow-emission model that provides a balanced trade-off between high accuracy and low computation time. The resulting model belongs to the mesoscopic category considering the following characteristic given for mesoscopic traffic models by Hoogendoorn and Bovy [59]: mesoscopic models specify the behavior of traffic by groups of vehicles/drivers, where the interactions of these vehicles/drivers are described in a low level of detail.Hence, in our proposed framework, we take into account the effect of the acceleration and deceleration of the vehicles to improve the accuracy of the results for urban traffic networks.

The mesoscopic emission model developed by Rakha et al. [135] provides promising results w.r.t. the microscopic model VT-micro (the errors are within the range of 10-27%). In this chapter, we aim to improve the accuracy of these results to an even higher level for a mesoscopic model that is resulting from our proposed integrating framework. Hence, we start from a microscopic point-of-view, by considering the time-speed trajectories of individual vehicles in the traffic network. Then we distinguish some groups of vehicles with similar traffic behaviors and we define a (virtual) representative vehicle for each group. Afterwards, we use a microscopic emission model (such as VT-micro or VERSIT+) to compute the instantaneous emissions of the representative vehicle for each specific traffic behavior (free-flow, idling, decelerating, accelerating). By multiplying the resulting emissions by the total number of vehicles in each group and by the average time of the given behavior, we obtain a mesoscopic emission model.

The simulation results show that the relative error of the computed emissions by our proposed mesoscopic approach are less than 6%. Hence, we will successfully improve the results w.r.t. [135]. Compared with the mesoscopic model proposed by Gori et al. [51], which is limited to signalized intersections and which ignores the decelerating behavior of the vehicles, our framework is more general and can be used for both signalized and non-signalized traffic networks. As mentioned earlier, we do not ignore the decelerating behavior by the vehicles. Gori et al. [51] consider two different traffic scenarios in an urban traffic network, i.e., the under-saturated and the saturated scenarios. We add a third scenario, i.e., the over-saturated scenario, which helps us to provide more accuracy.

Chen and Yu [27] develop a microscopic simulation platform by integrating the microscopic traffic simulator VISSIM and the microscopic modal emission model CMEM. The main aim of their work is to provide a platform to assess the effect of various factors of a traffic network on the amount of emissions. Our focus in this chapter is mostly on providing a mathematical model for computation of emissions that can be applied in model-based analysis and control of traffic.

Previous work on integrating traffic flow and emission models include the work by Zegeye et al. [161], where METANET [115], a macroscopic freeway flow model, and VT-micro are integrated. For urban traffic, an integrated flow and emission model has been developed by Lin et al. [99], where the S-model is integrated with VT-micro to form a simple integrated model that suits real-time control applications. The proposed macroscopic model has a low computation time and in a case study given in [99] shows satisfactory results when used as the internal model of a model-predictive controller. Inspired by the macroscopic integrated model given by Lin et al. [99] and by Zegeye et al. [161], in this chapter we develop a general mesoscopic framework to integrate macroscopic traffic flow models and microscopic emission models (i.e., our proposed framework is not limited to specific flow and emission models). The aim is to provide even more accuracy, while keeping the computation speed still high. Therefore, similar to the work by Lin et al. [99], we divide the possible traffic states in an urban traffic network into three different scenarios, i.e., under-saturated, saturated, and over-saturated. In addition to that, we distinguish different groups of vehicles that may be observed for each traffic scenario. Hence, compared with the macroscopic model proposed by Lin et al. [99], we build up a mesoscopic framework that takes into account more details than a macroscopic one. Moreover, we define a (virtual) representative vehicle that should reflect an average behavior for all the vehicles in a group. In summary, our proposed framework provides more details compared with the previous work, while it still guarantees a trade-off between high accuracy and low computation time.

Contributions and organization of the chapter

The main contributions of the chapter include:

1. We propose some extensions for the S-model to make it more accurate. We also develop a general formulation for transforming a time-delayed system with a time-varying delay given in the continuous-time domain into an equivalent time-delayed system in

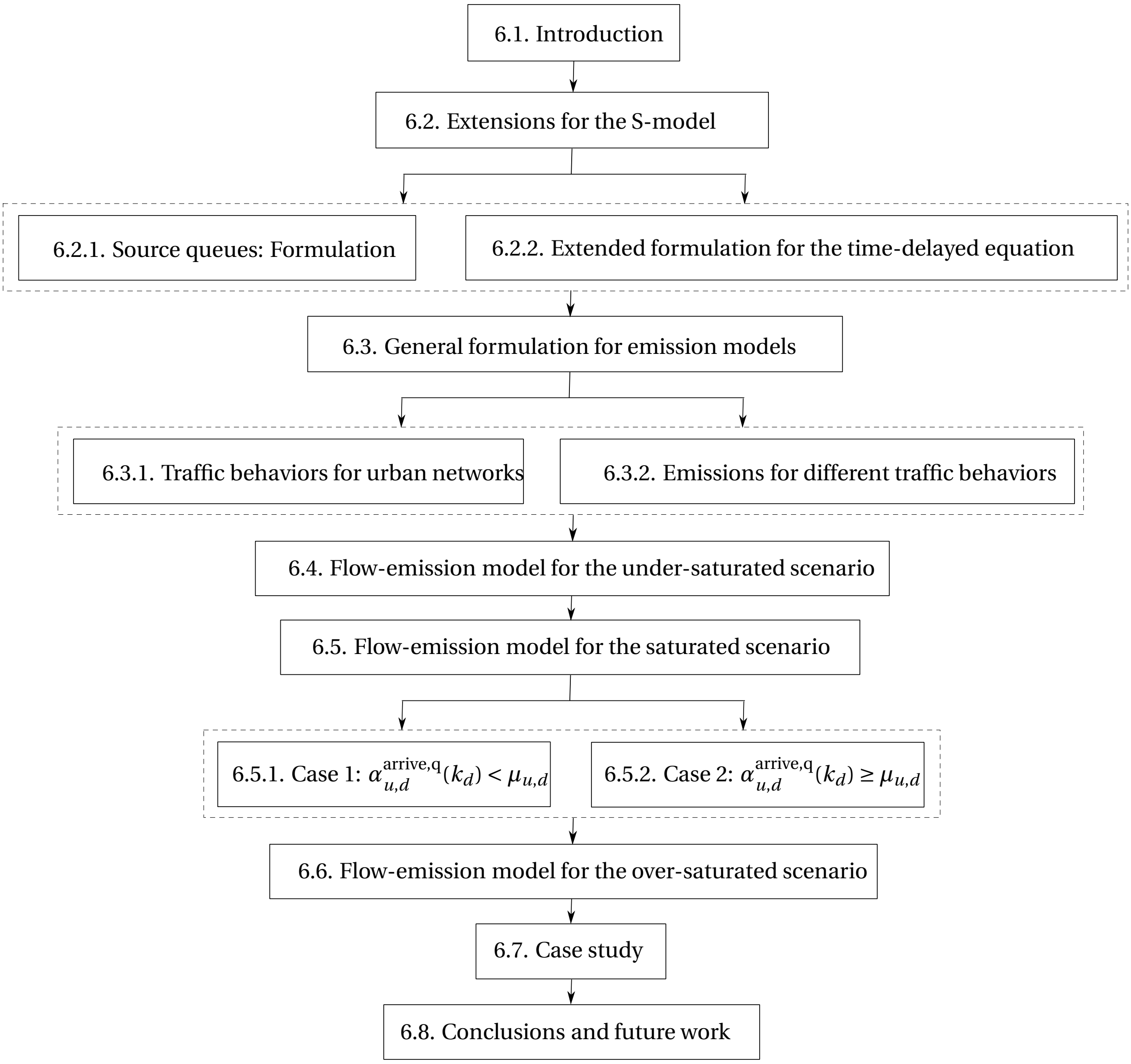

6.1. Introduction
6.2. Extensions for the S-model
6.2.1. Source queues: Formulation
6.2.2. Extended formulation for the time-delayed equation
6.3. General formulation for emission models
6.3.1. Traffic behaviors for urban networks
6.3.2. Emissions for different traffic behaviors
6.4. Flow-emission model for the under-saturated scenario
6.5. Flow-emission model for the saturated scenario
6.5.1. Case 1: $\alpha_{u,d}^{\text{arrive,q}}(k_d) < \mu_{u,d}$
6.5.2. Case 2: $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$
6.6. Flow-emission model for the over-saturated scenario
6.7. Case study
6.8. Conclusions and future work

Figure 6.1: Road map of Chapter 6.

the discrete-time domain (see Section 6.2.2). The resulting formulation can be used in discrete-time physical models of systems that involve time-varying time delays, and can give much more accurate results than a time-independent delay approximation.

2. We give a general framework for constructing mesoscopic emission models including formulations for computing the instantaneous emissions for different traffic behaviors observed in a traffic network (e.g., uniform speed, accelerating, decelerating, etc.), applying microscopic models that use speed and/or acceleration of vehicles.

3. For urban traffic networks, we distinguish different possible traffic scenarios (including

Table 6.1: Frequently used mathematical notations for Chapter 6.

u	a node in the S-model corresponding to an upstream intersection
d	a node in the S-model corresponding to a downstream intersection
(u,d)	a link in the S-model with u as its upstream node and d as its downstream node
$N^{\text{lane}}_{u,d}$	number of lanes in link (u,d)
k_d	simulation time step counter for link (u,d)
c_d	sampling time of the traffic light located at the intersection corresponding to node d
$n_{u,d}(k_d)$	total number of vehicles on link (u,d) in between time steps k_d and k_d+1
$q_{u,d}(k_d)$	number of vehicles in the queue on link (u,d) in between time steps k_d and k_d+1
$q_{u,d,o}(k_d)$	number of vehicles in the queue on link (u,d) in between time steps k_d and k_d+1 that intend to move to the outgoing link (d,o) at time step k_d
$\mathcal{O}_{u,d}$	set of all downstream nodes of the outgoing links of link (u,d)
$\mathcal{I}_{u,d}$	set of all upstream nodes of the incoming links of link (u,d)
$\alpha^{\text{enter,l}}_{u,d}(k_d)$	average entering flow rates of link (u,d) within the time interval $[k_d c_d,(k_d+1)c_d)$
$\alpha^{\text{leave,l}}_{u,d}(k_d)$	average exiting flow rates of link (u,d) within the time interval $[k_d c_d,(k_d+1)c_d)$
$\alpha^{\text{leave,l}}_{u,d,o}(k_d)$	average leaving flow rate of the sub-stream on link (u,d) that intends to move towards o within time interval $[k_d c_d,(k_d+1)c_d)$
$\alpha^{\text{arrive,q}}_{u,d,o}(k_d)$	average arriving flow rate at the tail of the waiting queue on link (u,d) of the sub-stream that intends to move towards o within $[k_d c_d,(k_d+1)c_d)$
$\beta_{u,d,o}(k_d)$	fraction of the vehicles on link (u,d) that intend to turn to node o
$\mu_{u,d}$	saturated leaving flow rate of the vehicles on link (u,d)
$g_{u,d,o}(k_d)$	green time length during the time interval $[k_d c_d,(k_d+1)c_d)$ for the traffic sub-stream that leaves link (u,d) towards node o
$C_{u,d}$	storage capacity of link (u,d)
$\tau_{u,d}(k_d)$	average delay time of the vehicles on link (u,d) within the time interval $[k_d c_d,(k_d+1)c_d)$ that is needed by the entering vehicles to the link to reach the tail of the waiting queue
$a^{\text{acc}}_{u,d}$	acceleration rate of the vehicles on link (u,d)
$a^{\text{dec}}_{u,d}$	deceleration rate of the vehicles on link (u,d)
$v^{\text{free}}_{u,d}$	free-flow speed of the vehicles moving on link (u,d)
$v^{\text{idle}}_{u,d}$	idling speed of the vehicles in the queue on link (u,d)
$\bar{E}^{b}_{e,u,d}$	total emissions of pollutant e on link (u,d) caused by those vehicles that show behavior b (free-flow, idling, decelerating, accelerating) on this link
l^{veh}	average length of the vehicles

under-saturated, saturated, and over-saturated) within a mesoscopic structure. Then we separate different groups of vehicles that may be observed for each scenario. The distinction is based on the traffic behavior each group of vehicles show within the network. Time-speed curves are extracted for different groups of vehicles for each traffic scenario, where the curve represents the average behavior of the vehicles in that group for a (possibly virtual) representative vehicle.

4. We extract the formulas that compute the time spent and the emissions of the representative vehicle of each group, which, multiplied by the total number of vehicles within the group, give the total time spent and the total emissions of the vehicles in that group.

5. The interfacing and integrating framework proposed in this chapter results in a mesoscopic flow and emission model, where the proposed integrating approach is applicable to any macroscopic flow model that updates at every simulation time step the total number of vehicles observed in a link and the number of vehicles waiting in the queue in a link, and also any microscopic emission model that computes the emissions and/or fuel consumption of the vehicles in a network based on their individual instantaneous speed and acceleration.

The rest of the chapter is organized as follows. Section 6.2 proposes some extensions for the original S-model in order to make this model more accurate. In Section 6.3, we give general formulations to compute the emissions using the instantaneous speed and the instantaneous acceleration of the vehicles for different traffic behaviors in urban traffic networks. Sections 6.4, 6.5, and 6.6 present formulations for adapting the equations given in Section 6.3 for different urban traffic scenarios, i.e., the under-saturated, saturated, and over-saturated scenarios. Section 6.7 presents the results of some case studies, where the traffic microsimulator SUMO is used together with VT-micro as the evaluation platform for estimating the emission levels. These estimated emission values are compared with the emissions computed by the new mesoscopic integrated flow-emission model. Section 6.8 concludes the work and gives topics for future work. The road map of this chapter is shown in Figure 6.1. Moreover, Table 6.1 gives the mathematical notations that are frequently used in Chapter 6.

This chapter of the thesis is based on [73].

6.2 Extensions for the S-model

One possible option for an urban traffic flow model that can be used in our proposed integrated framework is the S-model (see Section 5 for details regarding the original model). In the original S-model, only the dynamics of the system within the network of interest is considered. However, the dynamics of the source nodes, which are located on the boundaries of the network, is missing. This may not bring issues when the source links of the network are under-saturated. However, for larger demands at the source nodes, this may result in saturated and over-saturated links; hence, defining the appropriate dynamics for the boundaries of the network is essential. This is because the additional vehicles that cannot immediately enter the network should not be injected into the entering links. However, such an issue occurs with the original S-model and may lead to negative states with large absolute values. In contrast, those vehicles should be kept in queues that will gradually form at the sources of the network, and they should be injected into the network only when the available free space of the source links allows it.

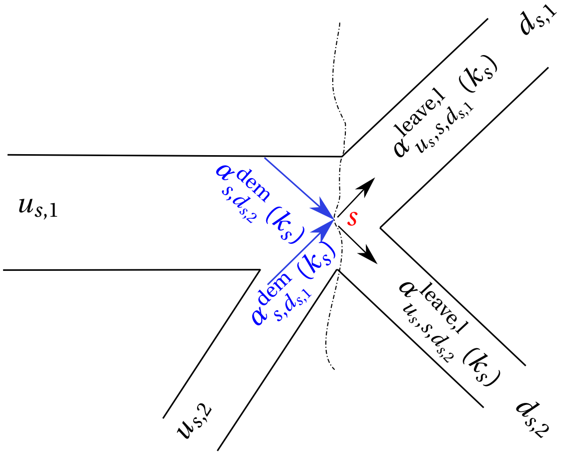

$d_{s,1}$
$\alpha^{\text{leave},1}_{u_s,s,d_{s,1}}(k_s)$
$u_{s,1}$
$\alpha^{\text{dem}}_{s,d_{s,2}}(k_s)$
s
$\alpha^{\text{dem}}_{s,d_{s,1}}(k_s)$
$\alpha^{\text{leave},1}_{u_s,s,d_{s,2}}(k_s)$
$u_{s,2}$
$d_{s,2}$

Figure 6.2: Two source links $(u_{s,1}, s)$ *and* $(u_{s,2}, s)$ *that feed links* $(s, d_{s,1})$ *and* $(s, d_{s,2})$ *of the network via source node* s.

In addition, in the original S-model the formula for the computation of the arriving flow $\alpha^{\text{arrive,q}}_{u,d,o}(k_d)$ (see (5.7)) is extracted based on the (hidden) assumption that the delay times for the vehicles are time-independent[1] (see [96] for more details). Numerical experiments show that with this assumption and by using (5.7), in some cases the error in computation of the updated states can grow up to 50% (this error is larger when the difference between the previous and current delay times is prominent, e.g., when the traffic scenario on the link changes from under-saturated to saturated or over-saturated in between two consecutive time steps).

Therefore, in this section, we first propose some extensions for the S-model to make it more accurate. To solve the first issue, we propose an additional network element for the S-model called the source link, which is a link that has one of the source nodes of the network as its downstream node (see Figure 6.2), and that feeds the network with new demand at every time step. Moreover, we develop a general formulation for discrete-time systems with time-varying delays, which produces accurate results compared with the results produced by the continuous-time formulation. We later use this discrete-time representation to compute $\alpha^{\text{arrive,q}}_{u,d,o}(k_d)$.

6.2.1 Source queues

The original S-model allows any number of vehicles that is dictated by the demands to enter the sources of the urban traffic network. This causes the model to fail in reliable estimation of the traffic states, especially for high demands at the sources of the network, which may result in saturated and over-saturated scenarios. For instance, the total number of the vehicle on the entrance links of the urban traffic network may grow exponentially as the demand grows, and hence can easily exceed the link's capacity. Therefore, in this section we define additional formulations for the S-model, which allow some origin queues to be formed at the entrances

[1] We call it a hidden assumption since in the formulas given by Lin et al. [98], the delay time is expressed as a function of time, but later on it is treated as a time-independent variable.

of the urban traffic network in case there is no available space for the vehicles that intend to enter the traffic network.

Suppose that $\mathcal{S}$ is the set of all source nodes of the network, $\mathcal{U}(s)$ and $\mathcal{D}(s)$ are the sets of all upstream nodes of $s \in \mathcal{S}$ outside of the network and all downstream nodes of $s \in \mathcal{S}$ inside the network, $u_{s,i} \in \mathcal{U}(s),\ i \in \{1,\ldots,\mathrm{card}(\mathcal{U}(s))\}$ denotes the upstream nodes of source node $s \in \mathcal{S}$ outside of the network, and $d_{s,j} \in \mathcal{D}(s),\ j \in \{1,\ldots,\mathrm{card}(\mathcal{D}(s))\}$ denotes the downstream nodes of source node s inside of the network. A queue will be formed within a source link $(u_{s,i}, s)$, if there is not enough free space for the given demand on those links of the network that are connected to this source link via node s (i.e., links $(s, d_{s,j}),\ j \in \{1,\ldots,\mathrm{card}(\mathcal{D}(s))\}$). We define $q_s^{\text{source}}(k_s)$, called the source queue, as the overall queue length in [veh/s] at source node s at time step k_s. Thus the source queue q_s^{source} is the summation of all queue lengths for the source links for node s. We then have

$$q_s^{\text{source}}(k_s+1) = q_s^{\text{source}}(k_s) + \left(\alpha_s^{\text{dem}}(k_s) - \alpha_s^{\text{leave,l}}(k_s)\right) c_s, \tag{6.1}$$

where $\alpha_s^{\text{dem}}(k_s)$ is the cumulative demand flow in [veh/s] at source node s at time step k_s, given by

$$\alpha_s^{\text{dem}}(k_s) = \sum\nolimits_{j=1}^{\mathrm{card}(\mathcal{D}(s))} \alpha_{s,d_{s,j}}^{\text{dem}}(k_s), \tag{6.2}$$

and $\alpha_s^{\text{leave,l}}(k_s)$ is the cumulative leaving flow in [veh/s] at source node s at time step k_s (i.e., the total flow of vehicles that can enter the network via source node s during $[k_s c_s, (k_s+1)c_s)$). We have

$$\alpha_s^{\text{leave,l}}(k_s) = \min\left\{\mu_s, \alpha_s^{\text{dem}}(k_s) + \frac{q_s^{\text{source}}(k_{d_s})}{c_s}, \sum\nolimits_{j=1}^{\mathrm{card}(\mathcal{D}(s))} \frac{C_{s,d_{s,j}} - n_{s,d_{s,j}}(k_{d_s})}{c_s}\right\},$$

with $\mu_s = \sum_{i=1}^{\mathrm{card}(\mathcal{U}(s))} \mu_{u_{s,i},s}$ the cumulative saturated leaving flow rate at source node s. Then $\alpha_{s,d_{s,j}}^{\text{enter,l}} = \beta_{s,d_{s,j}}^{\text{source}} \cdot \alpha_s^{\text{leave,l}}(k_s)$, where $\beta_{s,d_{s,j}}^{\text{source}} = \frac{\sum_{i=1}^{\mathrm{card}(\mathcal{U}(s))} \beta_{u_{s,i},s,d_{s,j}}}{\sum_{i=1}^{\mathrm{card}(\mathcal{U}(s))} \sum_{j=1}^{\mathrm{card}(\mathcal{D}(s))} \beta_{u_{s,i},s,d_{s,j}}}$ and $j \in \{1,\ldots,\mathrm{card}(\mathcal{D}(s))\}$.

6.2.2 Extended formulation of the time-delayed equation for arriving flow

To compute $\alpha_{u,d}^{\text{arrive,q}}(k_d)$, we should consider the time the entering vehicles in link (u,d) need to reach the tail of the waiting queue, i.e., the delay time $\tau_{u,d}(k_d)$, and the entering rate of the vehicles in the link $\tau_{u,d}(k_d)$ time units ago. Hence, for the queue length $q_{u,d,o}(k_d)$, we actually have a time-delayed difference equation in the discrete-time domain. We first consider a general case and propose and prove a lemma that can also be used to formulate the discrete-time delayed equation of the S-model.

Consider the time-delayed differential equation

$$\dot{x}(t) = Ax(t) + Bu(t - \tau(t)) + Ce(t), \tag{6.3}$$

with $u(\cdot)$ and $e(\cdot)$ two continuous-time functions representing the inflow and outflow functions of the dynamic system. We should first discretize the model to find $\tilde{x}(k+1)$ from the current state $\tilde{x}(k)$, where the discrete-time variables are identified by a tilde, and the time step counter is denoted by k. Additionally, we define

$$\tau(t) = \delta(t)h + \gamma(t), \tag{6.4}$$

$$\delta(t) = \left\lfloor \tau(t)/h \right\rfloor, \tag{6.5}$$

$$\gamma(t) = \text{rem}\{\tau(t), h\}, \tag{6.6}$$

where h is the sampling time. Note that for the discrete form of $\tau(t)$, $\delta(t)$, and $\gamma(t)$, we will use $\tilde{\tau}(k)$, $\tilde{\delta}(k)$, and $\tilde{\gamma}(k)$, respectively.

Lemma 6.1 *The equivalent difference equation in the discrete-time domain of the delayed differential equation* (6.3)*, and with an input function* $u(\cdot)$ *that is piece-wise constant in the interval* $[ch, (c+1)h)$, $c \in \mathbb{Z}$, *is*

$$\tilde{x}(k+1) = \tilde{A}\tilde{x}(k) + \sum_{i=0}^{|\tilde{\delta}(k-1)-\tilde{\delta}(k)+1|} \tilde{B}_i(k)\tilde{u}\big(k-1-\max\big(\tilde{\delta}(k),\tilde{\delta}(k-1)+1\big)+i\big) + \tilde{C}\tilde{e}(k), \tag{6.7}$$

with

$$\begin{aligned}
\tilde{B}_0(k) &= \begin{cases} \tilde{\gamma}(k-1)/h, & \text{if} \quad \tilde{\delta}(k-1)+1 \geq \tilde{\delta}(k), \\ \tilde{\gamma}(k)/h, & \text{if} \quad \tilde{\delta}(k-1)+1 < \tilde{\delta}(k), \end{cases} \\
\tilde{B}_i(k) &= 1, \qquad\qquad\qquad \text{for} \qquad i \in \left\{1, \ldots, \left|\tilde{\delta}(k-1) - \tilde{\delta}(k) + 1\right| - 1\right\}, \\
\tilde{B}_{|\tilde{\delta}(k-1)-\tilde{\delta}(k)+1|}(k) &= \begin{cases} \big(h-\tilde{\gamma}(k)\big)/h, & \text{if} \quad \tilde{\delta}(k-1)+1 \geq \tilde{\delta}(k), \\ \big(h-\tilde{\gamma}(k-1)\big)/h, & \text{if} \quad \tilde{\delta}(k-1)+1 < \tilde{\delta}(k). \end{cases}
\end{aligned}$$

Proof: We should find the effective inflow $\bar{u}_h(t)$ during $(t-h, t]$ for the delayed differential equation (6.3) with a time-varying delay function. From (6.3), the effective inflow at time instant t is the delayed inflow $u(t-\tau(t))$. Therefore, the effective inflow within the interval $(t-h, t]$ is indeed the cumulative value of the inflows from time instant $t-h-\tau(t)$ until time instant $t-\tau(t)$. Correspondingly, we have

$$\bar{u}_h(t) = \frac{1}{h}\int_{t-h-\tau(t-h)}^{t-\tau(t)} u(\theta)\mathrm{d}\theta. \tag{6.8}$$

Remark 6.1 The inequality $\tilde{\delta}(k) \leq \tilde{\delta}(k-1)$ in the discrete-time domain is equivalent to the inequality $t-h-\tau(t-h) < t-\tau(t)$ in the continuous-time domain as explained next:

$$\tilde{\delta}(k) \leq \tilde{\delta}(k-1) \Rightarrow \tilde{\delta}(k) < \tilde{\delta}(k-1)+1$$

Since both sides of the inequality are integer numbers, by adding two values smaller than unity to both sides, the strict inequality still holds. Hence,

$$\tilde{\delta}(k) + \frac{\tilde{\gamma}(k)}{h} < \tilde{\delta}(k-1) + \frac{\tilde{\gamma}(k-1)}{h} + 1 \Rightarrow$$

$$\begin{aligned}
&\tilde{\delta}(k)h + \tilde{\gamma}(k) < \tilde{\delta}(k-1)h + \tilde{\gamma}(k-1) + h \Rightarrow \\
&\tau(t) < \tau(t-h) + h \Rightarrow \\
&t - h - \tau(t-h) < t - \tau(t).
\end{aligned}$$

Similarly, we can show that the inequality $\tilde{\delta}(k) > \tilde{\delta}(k-1)$ in the discrete-time domain is equivalent to the inequality $t-h-\tau(t-h) \geq t-\tau(t)$ in the continuous-time domain. □

Assume a piece-wise constant inflow function $u(\cdot)$ in $[st, st+h)$, $s \in \mathbb{Z}$, (see Figure 6.3), and $\tau(\theta) = \delta(\theta)h + \gamma(\theta)$ (with θ a continuous-time variable). We separate the two cases $\tilde{\delta}(k) \leq \tilde{\delta}(k-1)$ or equivalently in the continuous-time domain $t-h-\tau(t-h) < t-\tau(t)$ (see Remark 6.1), and $\tilde{\delta}(k) > \tilde{\delta}(k-1)$ or equivalently in the continuous-time domain $t-h-\tau(t-h) \geq t-\tau(t)$ (see Remark 6.1). These two cases are represented in Figures 6.3 and 6.4).

Case 1. $\tilde{\delta}(k) \leq \tilde{\delta}(k-1)$: (6.8) can be expanded as follows (also see Figure 6.3):

$$\begin{aligned}
\bar{u}_h(t) = {} & \frac{1}{h}\int_{t-h-\delta(t-h)h-\gamma(t-h)}^{t-h-\delta(t-h)h} u(\theta)\mathrm{d}\theta \\
& + \frac{1}{h}\int_{t-h-\delta(t-h)h}^{t-h-(\delta(t-h)-1)h} u(\theta)\mathrm{d}\theta \\
& + \ldots \\
& + \frac{1}{h}\int_{t-h-(\delta(t)+1)h}^{t-h-\delta(t)h} u(\theta)\mathrm{d}\theta \\
& + \frac{1}{h}\int_{t-h-\delta(t)h}^{t-\delta(t)h-\gamma(t)} u(\theta)\mathrm{d}\theta \\
= {} & \frac{1}{h}\gamma(t-h)u(t-h-(\delta(t-h)+1)h) \\
& + \frac{1}{h}\left(h\sum_{i=1}^{\delta(t-h)-\delta(t)} u(t-h-(\delta(t-h)-i+1)h)\right) \\
& + \frac{h-\gamma(t)}{h}u(t-h-\delta(t)h).
\end{aligned} \tag{6.9}$$

Knowing that $t-h$ and t in the continuous-time domain correspond, respectively, to $k-1$ and k in the discrete-time domain, the above equation can be written as

$$\tilde{\bar{u}}_h(k) = \sum_{i=0}^{\tilde{\delta}(k-1)-\tilde{\delta}(k)+1} \tilde{B}_i(k)\tilde{u}\big(k-1-\tilde{\delta}(k-1)-1+i\big), \tag{6.10}$$

with

$$\begin{aligned}
&\tilde{B}_0(k) = \tilde{\gamma}(k-1)/h, \\
&\tilde{B}_i(k) = 1, \qquad\qquad \text{for} \quad i \in \big\{1,\ldots,\tilde{\delta}(k-1)-\delta(\tilde{k})\big\}, \\
&\tilde{B}_{\tilde{\delta}(k-1)-\tilde{\delta}(k)+1}(k) = \big(h-\tilde{\gamma}(k)\big)/h.
\end{aligned}$$

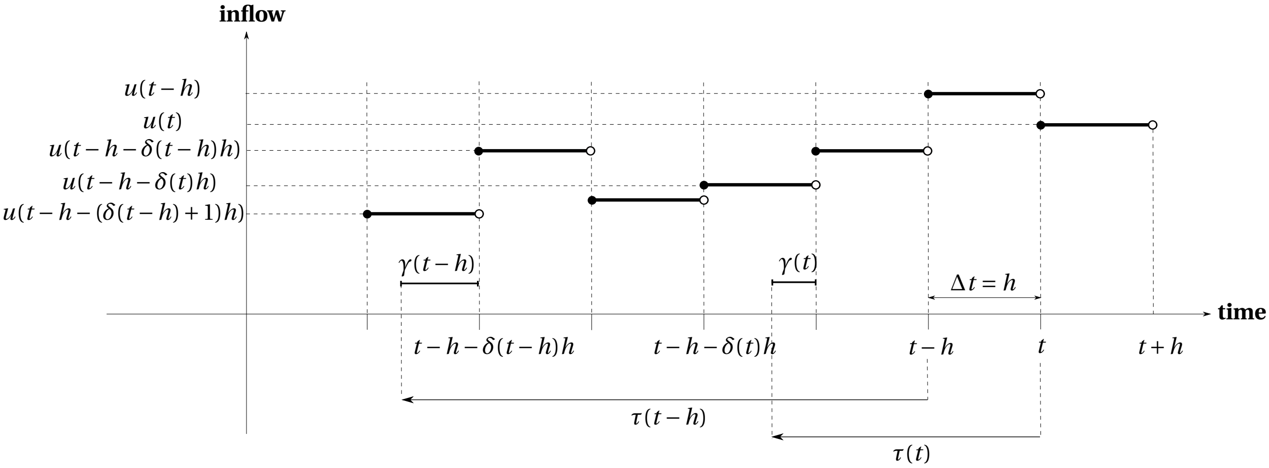


*Figure 6.3: **Case 1.** $\tilde{\delta}(k) \leq \tilde{\delta}(k-1)$: the effective inflow during $(t-h, t]$ for a delayed-time differential equation with a time-varying delay.*

Case 2. $\tilde{\delta}(k) > \tilde{\delta}(k-1)$: (6.8) can be expanded as follows (also see Figure 6.4):

$$\begin{aligned}\bar{u}_h(t) &= \frac{1}{h}\int_{t-\delta(t)h-\gamma(t)}^{t-\delta(t)h} u(\theta)\mathrm{d}\theta \\ &+ \frac{1}{h}\int_{t-\delta(t)h}^{t-\delta(t)h+h} u(\theta)\mathrm{d}\theta \\ &+ \ldots \\ &+ \frac{1}{h}\int_{t-h-\delta(t-h)h-h}^{t-h-\delta(t-h)h-\gamma(t-h)} u(\theta)\mathrm{d}\theta \\ &= \frac{1}{h}\gamma(t)\,u\left(t-(\delta(t)+1)h\right) \\ &+ \frac{1}{h}\left(h\sum_{i=1}^{\delta(t)-\delta(t-h)-2} u\left(t-(\delta(t)-i+1)\,h\right)\right) \\ &+ \frac{h-\gamma(t-h)}{h}\,u\left(t-h-\delta(t-h)h-h\right).\end{aligned} \tag{6.11}$$

Knowing that $t-h$ and t in the continuous-time domain correspond, respectively, to $k-1$ and k in the discrete-time domain, the above equation can be written as

$$\tilde{\bar{u}}_h(k) = \sum_{i=0}^{\tilde{\delta}(k)-\tilde{\delta}(k-1)-1} \tilde{B}_i(k)\,\tilde{u}\left(k-1-\tilde{\delta}(k)+i\right), \tag{6.12}$$

with

$$\begin{aligned}&\tilde{B}_0(k) = \tilde{\gamma}(k)/h, \\ &\tilde{B}_i(k) = 1, \qquad\qquad \text{for} \quad i \in \left\{1,\ldots,\tilde{\delta}(k)-\tilde{\delta}(k-1)-2\right\},\end{aligned}$$

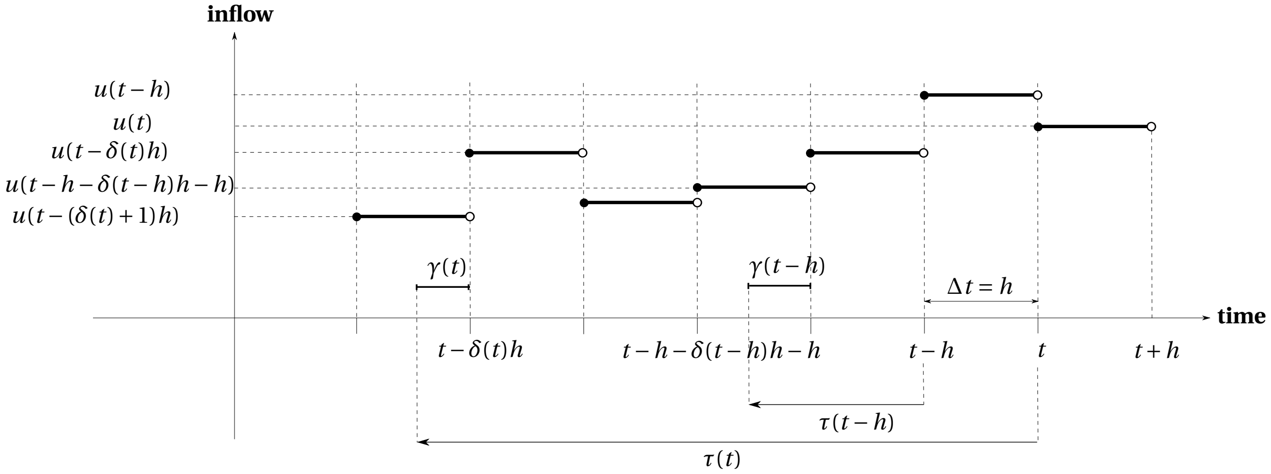


*Figure 6.4: **Case 2.** $\delta(\tilde{k}) > \tilde{\delta}(k-1)$: the effective inflow during $(t-h, t]$ for a delayed-time differential equation with a time-varying delay.*

$$\tilde{B}_{\tilde{\delta}(k)-\tilde{\delta}(k-1)-1}(k) = \left(h - \tilde{\gamma}(k-1)\right)/h.$$

From (6.10) and (6.12), we can obtain (6.7). □

Using Lemma 6.1, $\tilde{x}(k)$ will be substituted by $q_{u,d,o}(k_d)$, and the factors $\tilde{A}$ and $\tilde{C}$ by 1 and -1. We obtain

$$\begin{aligned} q_{u,d,o}(k_d+1) = q_{u,d,o}(k_d) + \\ \beta_{u,d,o}(k_d) \cdot \sum_{i=0}^{\substack{|\tilde{\delta}_{u,d}(k_d-1)- \\ \tilde{\delta}_{u,d}(k_d)+1|}} \tilde{B}_{u,d,i}(k_d)\alpha_{u,d}^{\text{enter,l}}\left(k_d - 1 - \max\left(\tilde{\delta}_{u,d}(k_d), \tilde{\delta}_{u,d}(k_d-1)+1\right) + i\right) - \\ \alpha_{u,d,o}^{\text{leave,l}}(k_d)c_d, \end{aligned} \tag{6.13}$$

where

$$\begin{aligned} &\tilde{B}_{u,d,0}(k_d) = \begin{cases} \tilde{\gamma}_{u,d}(k_d-1)/c_d, & \text{if } \tilde{\delta}_{u,d}(k_d-1)+1 \geq \tilde{\delta}_{u,d}(k_d), \\ \tilde{\gamma}_{u,d}(k_d)/c_d, & \text{if } \tilde{\delta}_{u,d}(k_d-1)+1 < \tilde{\delta}_{u,d}(k_d), \end{cases} \\ &\tilde{B}_{u,d,i}(k_d) = 1, \qquad \text{for} \quad i \in \left\{1, \ldots, \left|\tilde{\delta}_{u,d}(k_d-1) - \tilde{\delta}_{u,d}(k_d)+1\right| - 1\right\}, \\ &\tilde{B}_{u,d,|\tilde{\delta}_{u,d}(k_d-1)-\tilde{\delta}_{u,d}(k_d)+1|}(k_d) = \begin{cases} \left(c_d - \tilde{\gamma}_{u,d}(k_d)\right)/c_d, & \text{if } \tilde{\delta}_{u,d}(k_d-1)+1 \geq \tilde{\delta}_{u,d}(k_d), \\ \left(c_d - \tilde{\gamma}_{u,d}(k_d-1)\right)/c_d, & \text{if } \tilde{\delta}_{u,d}(k_d-1)+1 < \tilde{\delta}_{u,d}(k_d). \end{cases} \end{aligned} \tag{6.14}$$

and

$$\delta_{u,d}(t) = \left\lfloor \frac{\tau_{u,d}(t)}{c_d} \right\rfloor, \tag{6.15}$$

$$\gamma_{u,d}(t) = \text{rem}\left\{\tau_{u,d}(t), c_d\right\}. \tag{6.16}$$

Comparing (6.13) with (5.2) and (5.8), we conclude that

$$\alpha_{u,d}^{\text{arrive,q}}(k_d) = \sum_{i=0}^{\substack{|\tilde{\delta}_{u,d}(k_d-1)- \\ \tilde{\delta}_{u,d}(k_d)+1|}} \tilde{B}_{u,d,i}(k_d)\alpha_{u,d}^{\text{enter,l}}\left(k_d - 1 - \max\left(\tilde{\delta}_{u,d}(k_d), \tilde{\delta}_{u,d}(k_d-1)+1\right) + i\right). \tag{6.17}$$

Moreover, the delay time $\tilde{\tau}_{u,d}(k_d)$ at time step k_d (assumed to be constant within $[k_d c_d, (k_d + 1)c_d)$) is computed as follows. Vehicles are assumed to enter the link with[2] the free-flow speed and will move for $T_{u,d}^{\text{free}}(k_d)$ time units with this speed. If there is an idling queue in front of them, after a while they will decelerate to reach the idling speed of the vehicles in the queue. During this period, the distance between the upstream intersection and the tail of the waiting queue will be traveled by these vehicles (note that in the original S-model [98], it is assumed that this distance is traveled with a constant speed $v_{u,d}^{\text{free}}$).

The following lemma, which is trivial to prove, will be used to determine the delay time:

Lemma 6.2 *For a particle that starts moving with the initial speed "v_0" with a constant acceleration "a" the displacement, $\Delta x(t)$, after t time units is*

$$\Delta x(t) = \frac{1}{2}at^2 + v_0 t.$$

Therefore, for the vehicles that enter the link and then reach the idling queue after a while, we can write

$$\frac{\left(C_{u,d} - q_{u,d}^{\text{ave}}(k_d)\right) l^{\text{veh}}}{N_{u,d}^{\text{lane}}} = v_{u,d}^{\text{free}} \cdot T_{u,d}^{\text{free}}(k_d) + \frac{1}{2}a_{u,d}^{\text{dec}}\left(\frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}\right)^2 + v_{u,d}^{\text{free}}\left(\frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}\right), \tag{6.18}$$

where $C_{u,d}$ is the storage capacity of link (u,d), $q_{u,d}^{\text{ave}}(k_d)$ is the average queue length on link (u,d) within $[k_d c_d, (k_d + 1)c_d)$, l^{veh} is the average length of the vehicles, $N_{u,d}^{\text{lane}}$ is the number of lanes in link (u,d), $v_{u,d}^{\text{free}}$ and $v_{u,d}^{\text{idle}}$ are the free-flow and idling speed on link (u,d), $T_{u,d}^{\text{free}}(k_d)$ is the time that the vehicles on link (u,d) move with a free-flow speed before they reach the tail of the waiting queue, and $a_{u,d}^{\text{dec}}$ is the deceleration of the vehicles on link (u,d). Note that $\frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}$ is the time needed for vehicles to reach $v_{u,d}^{\text{idle}}$ from $v_{u,d}^{\text{free}}$ by the constant deceleration $a_{u,d}^{\text{dec}}$. Then the delay time $\tilde{\tau}_{u,d}(k_d)$ of the vehicles is obtained by

$$\tilde{\tau}_{u,d}(k_d) = T_{u,d}^{\text{free}}(k_d) + \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}, \tag{6.19}$$

[2] The typical values of the free-flow speeds and accelerations/decelerations can be determined via identifying the S-model's parameters w.r.t. real-life data or data from a traffic microsimulator. The difference between the output produced by the model and the data is then minimized by solving an optimization problem offline [131].

where the value of $T^{\text{free}}_{u,d}(k_d)$ is computed from (6.18). Hence, the delay time is

$$\tilde{\tau}_{u,d}(k_d) = \frac{\left(C_{u,d} - q^{\text{ave}}_{u,d}(k_d)\right) l^{\text{veh}}}{N^{\text{lane}}_{u,d} v^{\text{free}}_{u,d}} - \frac{\left(v^{\text{idle}}_{u,d} - v^{\text{free}}_{u,d}\right)^2}{2a^{\text{dec}}_{u,d} v^{\text{free}}_{u,d}}. \tag{6.20}$$

In order to compute the average queue length $q^{\text{ave}}_{u,d}(k_d)$, we propose the following three options:

Option 1. The average queue length is approximated by the queue length at k_d (computed by (5.2) and (5.3)), i.e.,

$$q^{\text{ave}}_{u,d}(k_d) = q_{u,d}(k_d). \tag{6.21}$$

Option 2. The average queue length is approximated by the average of the queue lengths at k_d and at $k_d + 1$, where to estimate $q_{u,d}(k_d + 1)$ at k_d we can use extrapolation[3], i.e.,

$$q_{u,d}(k_d + 1) = q_{u,d}(k_d) + \left(q_{u,d}(k_d) - q_{u,d}(k_d - 1)\right).$$

Then we have:

$$\begin{aligned} q^{\text{ave}}_{u,d}(k_d) &= \frac{q_{u,d}(k_d) + q_{u,d}(k_d + 1)}{2} \\ &= \frac{3}{2} q_{u,d}(k_d) - \frac{1}{2} q_{u,d}(k_d - 1). \end{aligned} \tag{6.22}$$

Option 3. The output of a predictor-corrector procedure, such as the procedure explained below:

step 1. Use either (6.21) or (6.22) to initialize the value of $q^{\text{ave}}_{u,d}(k_d)$.

step 2. Apply this $q^{\text{ave}}_{u,d}(k_d)$ to (6.20) and estimate $\tilde{\tau}_{u,d}(k_d)$.

step 3. Use the computed value of $\tilde{\tau}_{u,d}(k_d)$ in (6.15), (6.16), and (6.14).

step 4. Substitute the values that are determined in step 3 in (6.13) and (5.3) to get an update for $q_{u,d}(k_d + 1)$.

step 5. Apply (6.22) to estimate an updated value for $q^{\text{ave}}_{u,d}(k_d)$.

step 6. If a predefined convergence criterion or maximum number of iterations is reached, then stop the procedure. Otherwise, go to step 2.

6.3 General framework for emission models

In this section, we integrate the S-model with the VT-micro emission model, which produces the instantaneous fuel consumption and the instantaneous emissions of CO, HC, and NO_x

[3] Note that for computing $q_{u,d}(k_d + 1)$ using (5.2) and (5.3), we should already know $\tilde{\tau}_{u,d}(k_d)$. Hence, we use extrapolation first to estimate $q_{u,d}(k_d + 1)$ from the known values of $q_{u,d}(k_d - 1)$ and $q_{u,d}(k_d)$. Then we can use this estimated value in (6.18) to compute $\tilde{\tau}_{u,d}(k_d)$.

for a vehicle based on its instantaneous speed and acceleration. In the end, we will see that the developed mesoscopic integrated model can admit any urban traffic flow model that produces the number of vehicles and the queue lengths on different links, and any microscopic emission model that computes the instantaneous fuel consumption and the instantaneous emissions of $e \in \mathbb{E}$, with $\mathbb{E}$ the set of all polluting substances that are taken into account by the model, from the instantaneous speed $v_i(k)$ and the instantaneous acceleration $a_i(k)$ of individual vehicles. Hence, we should extract $v_i(k)$ and $a_i(k)$ from the urban traffic flow model for groups of vehicles (also called the traffic streams) that show a similar behavior. We first categorize different driving behaviors observed in urban traffic networks for different traffic scenarios.

6.3.1 Traffic behaviors for urban networks

We simplify the possible driving behaviors that might be observed for vehicles in parts of the simulation time slot and in different urban traffic scenarios via the following categories:

1. **Free-flow behavior**, for which the traffic stream moves with the free-flow speed $v_{u,d}^{\text{free}}$ on link (u, d), and $a_{u,d}^{\text{free}} = 0$.

2. **Idling behavior**, for which the traffic stream is waiting in a queue; we assume that the vehicles have a constant idling speed $v_{u,d}^{\text{idle}}$, and $a_{u,d}^{\text{idle}} = 0$.

3. **Decelerating behavior**, for which the traffic stream decreases its speed from $v_{u,d}^{b_1}$ to $v_{u,d}^{b_2}$ with a constant deceleration $a_{u,d}^{\text{dec}}$, with $b_1 \in \{\text{free}, \text{middle}\}$ and $b_2 \in \{\text{middle}, \text{idle}\}$, and $b_1 \neq b_2$. Note that $v_{u,d}^{\text{middle}}$ is a speed higher than the idling speed $v_{u,d}^{\text{idle}}$ and lower than the free-flow speed $v_{u,d}^{\text{free}}$. The traffic stream that already has an accelerating (a decelerating) behavior, would change its behavior to a decelerating (an accelerating) one as its speed reaches $v_{u,d}^{\text{middle}}$ (more details will be given later in this chapter).

4. **Accelerating behavior**, for which the traffic stream increases its speed from $v_{u,d}^{b_2}$ to $v_{u,d}^{b_1}$ with a constant acceleration[4] $a_{u,d}^{\text{acc}}$, with $b_1 \in \{\text{free}, \text{middle}\}$ and $b_2 \in \{\text{middle}, \text{idle}\}$, and $b_1 \neq b_2$.

5. **Nonstop behavior**, for which the traffic stream that has had a free-flow behavior during the current cycle, faces a green light at the end of the cycle. For a nonstop traffic behavior, two different reactions by the drivers might be observed:

   - In case the number of vehicles on the road is relatively low, as the traffic stream approaches the green light it might increase its speed from $v_{u,d}^{\text{free}}$ to $v_{u,d}^{\text{free+}}$ by a constant acceleration $a_{u,d}^{\text{acc}}$. This situation actually indicates that the ratio of the arriving flow at the end of the link, $\alpha_{u,d}^{\text{arrive,q}}(k_d)$, and the saturated average leaving flow,

[4] The acceleration and deceleration in this thesis adopt signed numbers, i.e., $a_{u,d}^{\text{dec}} < 0$ and $a_{u,d}^{\text{acc}} > 0$.

$\mu_{u,d}$, is less than or equal to a predefined threshold, $\lambda_{u,d}$. Hence,

$$\frac{\alpha_{u,d}^{\text{arrive,q}}(k_d)}{\mu_{u,d}} \leq \lambda_{u,d},$$

with $\lambda_{u,d} \in [0,1]$. The parameter $\lambda_{u,d}$ could be determined via offline model identification using real-life datasets or data from a traffic microsimulator [131].

- In case the number of vehicles on the road is relatively high, as the traffic stream approaches the green light it might decrease its speed from $v_{u,d}^{\text{free}}$ to $v_{u,d}^{\text{free}-}$ by a constant deceleration $a_{u,d}^{\text{dec}}$. Mathematically, this situation can be formulated by

$$\frac{\alpha_{u,d}^{\text{arrive,q}}(k_d)}{\mu_{u,d}} > \lambda_{u,d}.$$

In the following section, we formulate the instantaneous emissions for each of the above behaviors, using a microscopic emission model that uses the instantaneous acceleration and the instantaneous speed of the vehicles. Note that since the nature of a nonstop behavior is either an accelerating or a decelerating behavior, we do not distinguish the nonstop behavior from the accelerating/decelerating behaviors in the following discussions.

6.3.2 Emissions for different traffic behaviors

In the previous section, we distinguished different traffic behaviors that may be observed in an urban traffic network. Next, we will formulate the emissions that are produced by the vehicles that show these behaviors.

- Total emissions of $e \in \mathbb{E}$ caused by free-flow behavior on link (u,d) within the interval $\square_{u,d}^{\text{free}}(k_d)$, with $\square_{u,d}^{\text{free}}(k_d)$ the union of all sub-intervals $\square_{u,d}^{\text{free},j}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$ for which at least one vehicle on the link shows a free-flow behavior, is given by

$$\bar{E}_{e,u,d}^{\text{free}}(k_d) = \sum_{j=1}^{N_{u,d}^{\text{free}}(k_d)} \left(n_{u,d}^{\text{free},j}(k_d) T_{u,d}^{\text{free},j}(k_d)\right) \cdot E_e\left(v_{u,d}^{\text{free}}, 0\right), \tag{6.23}$$

where $N_{u,d}^{\text{free}}(k_d)$ denotes the number of all sub-intervals $\square_{u,d}^{\text{free},j}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$, during which free-flow behavior is observed (i.e., $\square_{u,d}^{\text{free}}(k_d) = \bigcup_{j=1}^{N_{u,d}^{\text{free}}(k_d)} \square_{u,d}^{\text{free},j}(k_d)$), and $n_{u,d}^{\text{free},j}(k_d)$ is the total number of vehicles that show a free-flow behavior within $\square_{u,d}^{\text{free},j}(k_d)$, and $T_{u,d}^{\text{free},j}(k_d)$ is the length of $\square_{u,d}^{\text{free},j}(k_d)$.

- Total emissions of $e \in \mathbb{E}$ caused by idling behavior on link (u,d) within the interval $\square_{u,d}^{\text{idle}}(k_d)$, with $\square_{u,d}^{\text{idle}}(k_d)$ the union of all sub-intervals $\square_{u,d}^{\text{idle},j}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$ for which at least one vehicle on the link shows an idling behavior, is given by

$$\bar{E}^{\text{idle}}_{e,u,d}(k_d) = \sum_{j=1}^{N^{\text{idle}}_{u,d}(k_d)} \left(n^{\text{idle},j}_{u,d}(k_d) T^{\text{idle},j}_{u,d}(k_d) \right) \cdot E_e \left(v^{\text{idle}}_{u,d}, 0 \right), \tag{6.24}$$

where $N^{\text{idle}}_{u,d}(k_d)$ denotes the number of all sub-intervals $\mathbb{I}^{\text{idle},j}_{u,d}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$, during which idling behavior is observed, and $n^{\text{idle},j}_{u,d}(k_d)$ is the total number of vehicles that show an idling behavior within $\mathbb{I}^{\text{idle},j}_{u,d}(k_d)$, and $T^{\text{idle},j}_{u,d}(k_d)$ is the length of $\mathbb{I}^{\text{idle},j}_{u,d}(k_d)$.

- Total emissions of $e \in \mathbb{E}$ caused by decelerating behavior on link (u,d) within the interval $\mathbb{I}^{\text{dec}}_{u,d}(k_d)$, with $\mathbb{I}^{\text{dec}}_{u,d}(k_d)$ the union of all sub-intervals $\mathbb{I}^{\text{dec},j}_{u,d}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$ for which at least one vehicle on the link shows a decelerating behavior, is given by

$$\bar{E}^{\text{dec}}_{e,u,d}(k_d) = \sum_{j=1}^{N^{\text{dec}}_{u,d}(k_d)} n^{\text{dec},j}_{u,d}(k_d) \cdot \int\limits_{\mathbb{I}^{\text{dec},j}_{u,d}(k_d)} E_e \left(v(t), a^{\text{dec}}_{u,d} \right) \cdot \mathrm{d}t, \tag{6.25}$$

where $N^{\text{dec}}_{u,d}(k_d)$ denotes the number of all sub-intervals $\mathbb{I}^{\text{dec},j}_{u,d}(k_d)$ of $[k_d c_d, (k_d+1)c_d)$, during which a decelerating behavior is observed, $n^{\text{dec},j}_{u,d}(k_d)$ is the number of vehicles that show a decelerating behavior within $\mathbb{I}^{\text{dec},j}_{u,d}(k_d)$, and $T^{\text{dec},j}_{u,d}(k_d)$ is the length of $\mathbb{I}^{\text{dec},j}_{u,d}(k_d)$. Suppose that

$$\mathbb{I}^{\text{dec},j}_{u,d}(k_d) = \left[t^{b_1,j}_{u,d}, t^{b_2,j}_{u,d} \right], \qquad j \in \{1, \ldots, N^{\text{dec}}_{u,d}(k_d)\},$$

then by changing the integration variable from time t to speed v (knowing that $\mathrm{d}v = a^{\text{dec}}_{u,d} \cdot \mathrm{d}t$), we obtain:

$$\begin{aligned} \bar{E}^{\text{dec}}_{e,u,d}(k_d) &= \sum_{j=1}^{N^{\text{dec}}_{u,d}(k_d)} n^{\text{dec},j}_{u,d}(k_d) \cdot \int\limits_{t^{b_1,j}_{u,d}}^{t^{b_2,j}_{u,d}} E_e \left(v(t), a^{\text{dec}}_{u,d} \right) \cdot \mathrm{d}t \\ &= \sum_{j=1}^{N^{\text{dec}}_{u,d}(k_d)} \frac{n^{\text{dec},j}_{u,d}(k_d)}{a^{\text{dec}}_{u,d}} \cdot \int\limits_{v^{b_1,j}_{u,d}}^{v^{b_2,j}_{u,d}} E_e \left(v, a^{\text{dec}}_{u,d} \right) \cdot \mathrm{d}v, \end{aligned} \tag{6.26}$$

with $v^{b_1,j}_{u,d}$ and $v^{b_2,j}_{u,d}$ the speed of the decelerating vehicles in $\mathbb{I}^{\text{dec},j}_{u,d}(k_d)$ at, respectively, $t^{b_1,j}_{u,d}$ and $t^{b_2,j}_{u,d}$, and with the following possible combinations for b_1 and b_2:

$$b_1 = \text{free} \quad \wedge \quad b_2 = \text{middle},$$

$$b_1 = \text{middle} \quad \wedge \quad b_2 = \text{idle},$$
$$b_1 = \text{free} \quad \wedge \quad b_2 = \text{idle}.$$

- Total emissions caused by accelerating behavior on link (u, d) can be computed using a similar approach as we used for the decelerating behavior. We have

$$\bar{E}^{\mathrm{acc}}_{e,u,d}(k_d) = \sum_{j=1}^{N^{\mathrm{acc}}_{u,d}(k_d)} \frac{n^{\mathrm{acc},j}_{u,d}(k_d)}{a^{\mathrm{acc}}_{u,d}} \cdot \int\limits_{v^{b_1,j}_{u,d}}^{v^{b_2,j}_{u,d}} E_e\left(v, a^{\mathrm{acc}}_{u,d}\right) \cdot \mathrm{d}v, \tag{6.27}$$

where $N^{\mathrm{acc}}_{u,d}(k_d)$ denotes the number of all sub-intervals $\text{▯}^{\mathrm{acc},j}_{u,d}(k_d)$ of $[k_d c_d, (k_d + 1)c_d)$, during which an accelerating behavior is observed, $n^{\mathrm{acc},j}_{u,d}(k_d)$ is the number of vehicles that show an accelerating behavior within $\text{▯}^{\mathrm{acc},j}_{u,d}(k_d)$, and $T^{\mathrm{acc},j}_{u,d}(k_d)$ is the length of $\text{▯}^{\mathrm{acc},j}_{u,d}(k_d)$. Moreover, the following combinations can be used for b_1 and b_2:

$$b_1 = \text{middle} \quad \wedge \quad b_2 = \text{free},$$
$$b_1 = \text{idle} \quad \wedge \quad b_2 = \text{middle},$$
$$b_1 = \text{idle} \quad \wedge \quad b_2 = \text{free}.$$

In the next sections (i.e., Sections 6.4, 6.5, and 6.6), we consider the following three traffic scenarios that might happen in an urban traffic network:

- under-saturated scenario,
- saturated scenario,
- over-saturated scenario.

For each of the above scenarios, we distinguish different groups of vehicles based on the traffic behaviors observed (see Section 6.3.1 for potential traffic behaviors). The traffic behavior of each group of vehicles is then illustrated via a time-speed curve, where this curve represents the behavior of a single representative vehicle, which shows the average collective behavior of that group. We define this average behavior as follows: since vehicles in each group will enter the link at different time instants, the starting time of the average time-speed curve is considered as the midpoint of the starting times of the curves corresponding to the first and the last vehicle in that group. Therefore, the time-speed curve corresponding to the representative vehicle illustrates the average time spent on a specific behavior that might be observed for each vehicle in that group. Later on, by computing the total number of vehicles in each group, we can find an estimate of the total emission of the vehicles in that group.

For the sake of simplicity, we suppose that all vehicles waiting in a queue will instantaneously react to the traffic light as it turns green, and that there will be no delay for the following vehicles to imitate the speed of the leading vehicles. Therefore, for the vehicles that idle

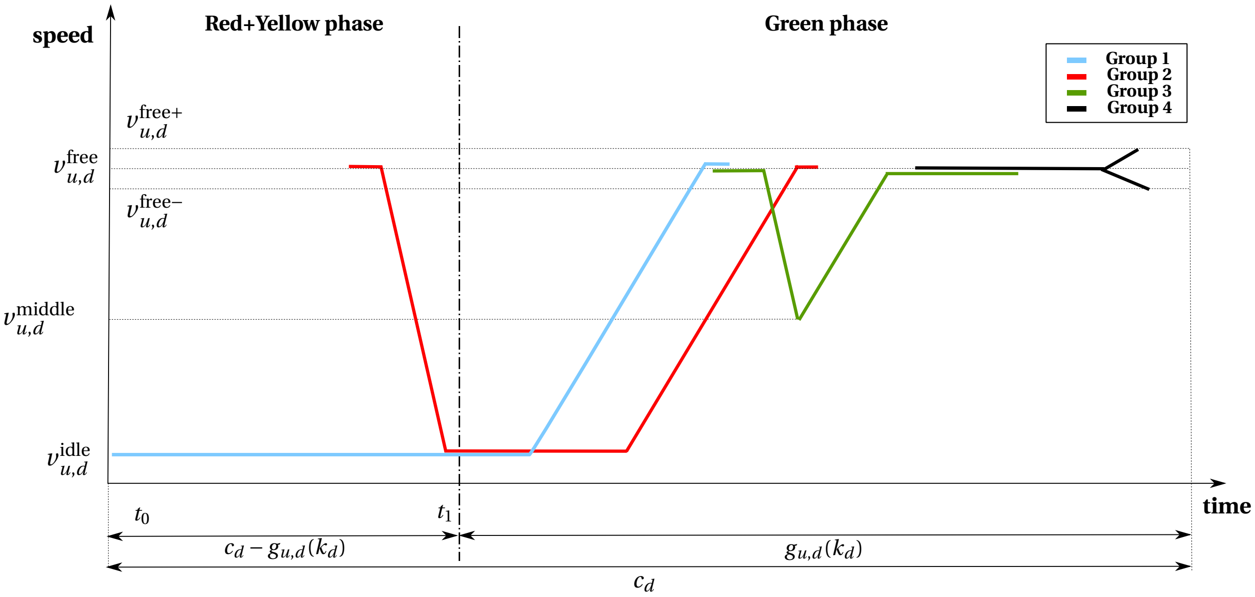


*Figure 6.5: Traffic behaviors on link (u,d) within one cycle for the **under-saturated** urban traffic scenario.*

in front of a red light, after the light turns green the time-speed curves will overlap. Vehicles in each group will show the same combination of behaviors given in Section 6.3.1. For the representative vehicle, the time duration of each of these behaviors is considered to be the average of the time durations with that specific behavior for the first and the last vehicle in the group.

6.4 Flow-emission model for the under-saturated scenario

The under-saturated scenario is observed when the demand (i.e., the number of vehicles waiting in the queue and the number of the arriving vehicles at the tail of the queue) is less than the number of vehicles that can be discharged by the saturated average leaving flow rate in one cycle. Hence, for the under-saturated scenario, the queue within the link will be completely dissolved during one cycle. Mathematically, this is expressed as follows:

$$q_{u,d}(k_d) + c_d \cdot \alpha_{u,d}^{\text{arrive,q}}(k_d) \leq \sum_{o \in \mathcal{O}_{u,d}} \beta_{u,d,o} \cdot \mu_{u,d} \cdot g_{u,d,o}(k_d). \tag{6.28}$$

Suppose that each cycle starts by a red phase; then the vehicles observed within one cycle with an under-saturated condition are divided into the following four groups:

Group 1: composed of those vehicles that are already in the link standing in a queue at the beginning of the cycle (see the blue curve in Figure 6.5 for the average behavior of this group); they idle with a constant speed $v_{u,d}^{\text{idle}}$ from the beginning of the cycle till the traffic light turns green (at t_1 in Figure 6.5); then they accelerate[5] with $a_{u,d}^{\mathrm{acc}}$ and move ahead till they leave the link. Looking at the average behavior of the representative vehicle of group 1 (the blue curve in Figure 6.5), the time instant at which the representative vehicle starts to accelerate is to the right of the instant t_1. This is because when the traffic light turns green, the vehicles in the queue can accelerate only one-by-one, i.e., the first vehicle in the queue starts to accelerate at t_1, while the other vehicles can accelerate only when the rest of the vehicles in the queue front of them have already accelerated (also see Figure 6.6(a), which represents the time-speed curve of the first and the last vehicle of group 1). As shown in Figure 6.5, this group of vehicles might also show a free-flow behavior at the end before they leave the link.

Group 2: composed of those vehicles that arrive at the tail of the waiting queue in the current cycle, after they decelerate from $v_{u,d}^{\mathrm{free}}$ to $v_{u,d}^{\mathrm{idle}}$. They idle for a while together with the waiting queue, and then start to accelerate as the traffic light turns green. The average behavior of this group is shown by the red time-speed curve in Figure 6.5.

Group 3: composed of those vehicles that arrive at the tail of the waiting queue in the current cycle, when the queue has already started to move. The representative of this group will decelerate with $a_{u,d}^{\mathrm{dec}}$ to the speed $v_{u,d}^{\mathrm{middle}}$. For $v_{u,d}^{\mathrm{middle}}$ we have (this formula will be motivated later in this chapter)

$$v_{u,d}^{\mathrm{middle}} = \frac{v_{u,d}^{\mathrm{free}} + v_{u,d}^{\mathrm{idle}}}{2}. \tag{6.29}$$

The green curve in Figure 6.5 shows the average behavior of this group. These vehicles might show a free-flow behavior at the end of their trip in the link.

Group 4: composed of those vehicles that arrive in the link with $v_{u,d}^{\mathrm{free}}$, and continue moving ahead with this speed until they approach the end of the link and show a nonstop behavior (see the black curve in Figure 6.5 for the average behavior of this group).

We first give the formulas that represent the emission model corresponding to the urban under-saturated scenario. Then we explain in more detail how the time-speed curves and the following formulas are obtained. For the number of vehicles in each group, we have

$$n_{u,d}^{\mathrm{G_1,usat}}(k_d) = q_{u,d}(k_d), \tag{6.30}$$

$$n_{u,d}^{\mathrm{G_2,usat}}(k_d) = \alpha_{u,d}^{\mathrm{arrive,q}}(k_d) \cdot T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d), \tag{6.31}$$

$$n_{u,d}^{\mathrm{G_3,usat}}(k_d) = \alpha_{u,d}^{\mathrm{arrive,q}}(k_d) \cdot T_{u,d}^{\mathrm{arrive,G_3,usat}}(k_d), \tag{6.32}$$

[5] For the sake of simplicity in the notations, for all the three scenarios under-saturated, saturated, and over-saturated, we have used the same acceleration $a_{u,d}^{\mathrm{acc}}$ and deceleration $a_{u,d}^{\mathrm{dec}}$. This can easily be extended by considering different acceleration rates $a_{u,d}^{\mathrm{acc,usat}}$, $a_{u,d}^{\mathrm{acc,sat}}$, and $a_{u,d}^{\mathrm{acc,osat}}$, and different deceleration rates $a_{u,d}^{\mathrm{dec,usat}}$, $a_{u,d}^{\mathrm{dec,sat}}$, and $a_{u,d}^{\mathrm{dec,osat}}$ for different traffic scenarios, where "usat" refers to the under-saturated scenario, "sat" refers to the saturated scenario, and "osat" refers to the over-saturated scenario.

$$n_{u,d}^{\mathrm{G_4,usat}}(k_d) = n_{u,d}^{\mathrm{usat}}(k_d) - \sum_{i=1}^{3} n_{u,d}^{\mathrm{G}_i\mathrm{,usat}}(k_d), \tag{6.33}$$

with

$$T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d) = \frac{\mu_{u,d}}{\mu_{u,d} - \alpha_{u,d}^{\mathrm{arrive,q}}(k_d)} \left(c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,usat}}(k_d)}{\mu_{u,d}} - \tau_{u,d}^{\mathrm{usat,G_2}}(k_d) \right), \tag{6.34}$$

$$T_{u,d}^{\mathrm{arrive,G_3,usat}}(k_d) = \tau_{u,d}^{\mathrm{usat,G_2}}(k_d) + \frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}, \tag{6.35}$$

where $\tau_{u,d}^{\mathrm{usat,G_2}}(k_d)$ is the average time that vehicles in group 2 take to reach the tail of the waiting queue in front of them from the time they enter the link (note that although in Section 6.2.2, we have distinguished the discrete-time variables from their corresponding continuous-time ones by a tilde symbol, for the sake of simplicity for the notations, we do not use a tilde symbol for the discrete-time delay in the rest of the thesis). To compute $\tau_{u,d}^{\mathrm{usat,G_2}}(k_d)$ within the time interval $[k_d c_d, (k_d+1)c_d)$, we use (6.18) with[6] $q_{u,d}^{\mathrm{G_2,ave}}(k_d) = n_{u,d}^{\mathrm{G_1,usat}}(k_d)$, which gives

$$\tau_{u,d}^{\mathrm{usat,G_2}}(k_d) = \frac{\left(C_{u,d} - n_{u,d}^{\mathrm{G_1,usat}}(k_d)\right) l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} - \frac{\left(v_{u,d}^{\mathrm{idle}} - v_{u,d}^{\mathrm{free}}\right)^2}{2a_{u,d}^{\mathrm{dec}} v_{u,d}^{\mathrm{free}}}. \tag{6.36}$$

In order to compute the emissions for the under-saturated scenario, we need to determine the number of the vehicles that show each of the driving behaviors "free", "idling", "decelerating", "accelerating", and "non-stop", and also the time duration of the free and idling behaviors during one cycle. These variables can be computed as follows:

- The number of vehicles in each of the groups 1-4 that show a free-flow behavior within the given cycle is

$$\begin{aligned} n_{u,d}^{\mathrm{free,G_1,usat}}(k_d) &= n_{u,d}^{\mathrm{G_1,usat}}(k_d),\\ n_{u,d}^{\mathrm{free,G_2,usat}}(k_d) &= n_{u,d}^{\mathrm{G_2,usat}}(k_d),\\ n_{u,d}^{\mathrm{free,G_3,usat}}(k_d) &= n_{u,d}^{\mathrm{G_3,usat}}(k_d),\\ n_{u,d}^{\mathrm{free,G_4,usat}}(k_d) &= n_{u,d}^{\mathrm{G_4,usat}}(k_d), \end{aligned} \tag{6.37}$$

  and the average time that a vehicle in each of the groups spends showing the free-flow behavior is

$$T_{u,d}^{\mathrm{free,G_1,usat}}(k_d) = \frac{\left(0.5 n_{u,d}^{\mathrm{G_1,usat}}(k_d)\right) l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} - \frac{\left(v_{u,d}^{\mathrm{free}}\right)^2 - \left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}} v_{u,d}^{\mathrm{free}}},$$

[6] Note that based on the definitions given above for different groups of vehicles, the queue in front of the vehicles in group 2 is formed by the vehicles in group 1 (also see Figure 6.5).

$$\begin{aligned}
T_{u,d}^{\text{free},\text{G}_2,\text{usat}}(k_d) &= \tau_{u,d}^{\text{usat},\text{G}_2}(k_d) - \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}} + \\
&\qquad \frac{\left(n_{u,d}^{\text{G}_1,\text{usat}}(k_d) + 0.5 n_{u,d}^{\text{G}_2,\text{usat}}(k_d)\right) l^{\text{veh}}}{N_{u,d}^{\text{lane}} v_{u,d}^{\text{free}}} - \frac{\left(v_{u,d}^{\text{free}}\right)^2 - \left(v_{u,d}^{\text{idle}}\right)^2}{2 a_{u,d}^{\text{acc}} v_{u,d}^{\text{free}}}, \\
T_{u,d}^{\text{free},\text{G}_3,\text{usat}}(k_d) &= \frac{C_{u,d} l^{\text{veh}}}{N_{u,d}^{\text{lane}} v_{u,a}^{\text{free}}} + \left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right) \frac{a_{u,d}^{\text{acc}} - a_{u,d}^{\text{dec}}}{2 a_{u,d}^{\text{dec}} a_{u,d}^{\text{acc}}}, \\
T_{u,d}^{\text{free},\text{G}_4,\text{usat}}(k_d) &= \frac{n_{u,d}^{\text{G}_4,\text{usat}}(k_d)}{\alpha_{u,d}^{\text{arrive,q}}(k_d)} - T_{u,d}^{\text{nonstop}}(k_d),
\end{aligned} \tag{6.38}$$

with

$$T_{u,d}^{\text{nonstop}}(k_d) = \begin{cases} \frac{v_{u,d}^{\text{free}+} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{acc}}}, & \text{if} \quad \frac{\alpha_{u,d}^{\text{arrive,q}}(k)}{\mu_{u,d}} \leq \lambda_{u,d}, \\ \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{free}-}}{a_{u,d}^{\text{dec}}}, & \text{if} \quad \frac{\alpha_{u,d}^{\text{arrive,q}}(k)}{\mu_{u,d}} > \lambda_{u,d}, \end{cases} \tag{6.39}$$

with $\lambda_{u,d}$ defined in Section 6.3.1 as a threshold for detecting heavy or light traffic conditions in the link (u,d).

- The number of vehicles that show an idling behavior within the given cycle is

$$\begin{aligned}
n_{u,d}^{\text{idle},\text{G}_1,\text{usat}}(k_d) &= n_{u,d}^{\text{G}_1,\text{usat}}(k_d), \\
n_{u,d}^{\text{idle},\text{G}_2,\text{usat}}(k_d) &= n_{u,d}^{\text{G}_2,\text{usat}}(k_d), \\
n_{u,d}^{\text{idle},\text{G}_3,\text{usat}}(k_d) &= 0, \\
n_{u,d}^{\text{idle},\text{G}_4,\text{usat}}(k_d) &= 0,
\end{aligned} \tag{6.40}$$

and the average time that a vehicle in each of the groups spends showing the idling behavior is

$$\begin{aligned}
T_{u,d}^{\text{idle},\text{G}_1,\text{usat}}(k_d) &= c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\text{G}_1,\text{usat}}(k_d)}{2\mu_{u,d}}, \\
T_{u,d}^{\text{idle},\text{G}_2,\text{usat}}(k_d) &= T_{u,d}^{\text{idle},\text{G}_1,\text{usat}}(k_d) - \tau_{u,d}^{\text{usat},\text{G}_2}(k_d), \\
T_{u,d}^{\text{idle},\text{G}_3,\text{usat}}(k_d) &= 0, \\
T_{u,d}^{\text{idle},\text{G}_4,\text{usat}}(k_d) &= 0.
\end{aligned} \tag{6.41}$$

- The number of vehicles that show a decelerating behavior within the given cycle is

$$\begin{aligned}
n_{u,d}^{\text{dec},\text{G}_1,\text{usat}}(k_d) &= 0, \\
n_{u,d}^{\text{dec},\text{G}_2,\text{usat}}(k_d) &= n_{u,d}^{\text{G}_2,\text{usat}}(k_d), \\
n_{u,d}^{\text{dec},\text{G}_3,\text{usat}}(k_d) &= n_{u,d}^{\text{G}_3,\text{usat}}(k_d), \\
n_{u,d}^{\text{dec},\text{G}_4,\text{usat}}(k_d) &= \begin{cases} 0, & \text{for light traffic,} \\ n_{u,d}^{\text{G}_4,\text{usat}}(k_d), & \text{for heavy traffic,} \end{cases}
\end{aligned} \tag{6.42}$$

and the average time that a vehicle spends showing the decelerating behavior is

$$
\begin{aligned}
&T_{u,d}^{\text{dec},\text{G}_1,\text{usat}}(k_d) = 0,\\
&T_{u,d}^{\text{dec},\text{G}_2,\text{usat}}(k_d) = \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}},\\
&T_{u,d}^{\text{dec},\text{G}_3,\text{usat}}(k_d) = \frac{v_{u,d}^{\text{middle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}},\\
&T_{u,d}^{\text{dec},\text{G}_4,\text{usat}}(k_d) = \begin{cases} 0, & \text{for light traffic,}\\ \frac{v_{u,d}^{\text{free}-} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}, & \text{for heavy traffic.} \end{cases}
\end{aligned}
\tag{6.43}
$$

- The number of vehicles that show an accelerating behavior within the given cycle is

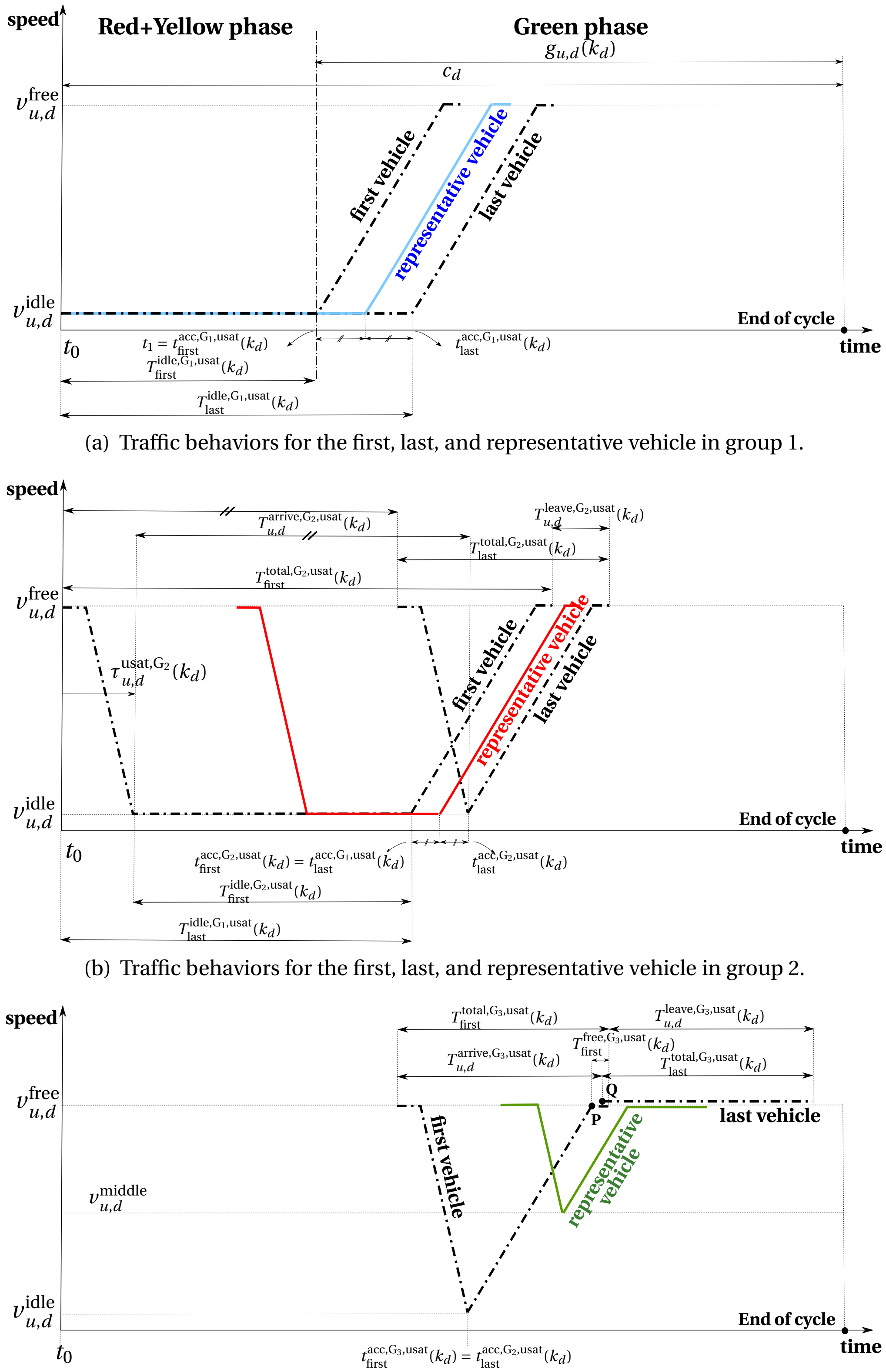
$$
\begin{aligned}
&n_{u,d}^{\text{acc},\text{G}_1,\text{usat}}(k_d) = n_{u,d}^{\text{G}_2,\text{usat}}(k_d),\\
&n_{u,d}^{\text{acc},\text{G}_2,\text{usat}}(k_d) = n_{u,d}^{\text{G}_2,\text{usat}}(k_d),\\
&n_{u,d}^{\text{acc},\text{G}_3,\text{usat}}(k_d) = n_{u,d}^{\text{G}_3,\text{usat}}(k_d),\\
&n_{u,d}^{\text{acc},\text{G}_4,\text{usat}}(k_d) = \begin{cases} n_{u,d}^{\text{G}_4,\text{usat}}(k_d), & \text{for light traffic,}\\ 0, & \text{for heavy traffic,} \end{cases}
\end{aligned}
\tag{6.44}
$$

and the average time that a vehicle spends showing the accelerating behavior is

$$
\begin{aligned}
&T_{u,d}^{\text{acc},\text{G}_1,\text{usat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc},\text{G}_2,\text{usat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc},\text{G}_3,\text{usat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{middle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc},\text{G}_4,\text{usat}}(k_d) = \begin{cases} \frac{v_{u,d}^{\text{free}+} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{acc}}}, & \text{for light traffic,}\\ 0, & \text{for heavy traffic.} \end{cases}
\end{aligned}
\tag{6.45}
$$

Next we explain in more detail how the time-speed curves represented in Figure 6.5 and correspondingly the given equations for the under-saturated scenario are obtained. Since for the under-saturated scenario all vehicles that are initially waiting in the queue will have enough time to leave the link within one cycle, we consider them as one group, and illustrate the behavior of the representative of this group (shown in blue in Figure 6.5) as the average of the time-speed curves of the first and the last vehicle in the group.

In Figure 6.6(a), the time-speed curves of the first, the last, and the representative vehicle in group 1 are shown. The blue (solid) curve in this figure, which demonstrates the behavior of the representative vehicle, is the same as the blue curve in Figure 6.5. The first vehicle of group 1 will immediately start to accelerate as the traffic light turns green at t_1 (see Figure 6.6(a)). After its speed reaches $v_{u,d}^{\text{free}}$, this vehicle might move with $v_{u,d}^{\text{free}}$ (depending on the


speed
Red+Yellow phase
Green phase
$g_{u,d}(k_d)$
c_d
$v_{u,d}^{\text{free}}$
first vehicle
representative vehicle
last vehicle
$v_{u,d}^{\text{idle}}$
End of cycle
t_0
$t_1 = t_{\text{first}}^{\text{acc,G}_1\text{,usat}}(k_d)$
$t_{\text{last}}^{\text{acc,G}_1\text{,usat}}(k_d)$
time
$T_{\text{first}}^{\text{idle,G}_1\text{,usat}}(k_d)$
$T_{\text{last}}^{\text{idle,G}_1\text{,usat}}(k_d)$
speed
$T_{u,d}^{\text{arrive,G}_2\text{,usat}}(k_d)$
$T_{u,d}^{\text{leave,G}_2\text{,usat}}(k_d)$
$T_{\text{last}}^{\text{total,G}_2\text{,usat}}(k_d)$
$T_{\text{first}}^{\text{total,G}_2\text{,usat}}(k_d)$
$v_{u,d}^{\text{free}}$
$\tau_{u,d}^{\text{usat,G}_2}(k_d)$
first vehicle
representative vehicle
last vehicle
$v_{u,d}^{\text{idle}}$
End of cycle
t_0
$t_{\text{first}}^{\text{acc,G}_2\text{,usat}}(k_d) = t_{\text{last}}^{\text{acc,G}_1\text{,usat}}(k_d)$
$t_{\text{last}}^{\text{acc,G}_2\text{,usat}}(k_d)$
time
$T_{\text{first}}^{\text{idle,G}_2\text{,usat}}(k_d)$
$T_{\text{last}}^{\text{idle,G}_1\text{,usat}}(k_d)$
speed
$T_{\text{first}}^{\text{total,G}_3\text{,usat}}(k_d)$
$T_{u,d}^{\text{leave,G}_3\text{,usat}}(k_d)$
$T_{\text{first}}^{\text{free,G}_3\text{,usat}}(k_d)$
$T_{u,d}^{\text{arrive,G}_3\text{,usat}}(k_d)$
$T_{\text{last}}^{\text{total,G}_3\text{,usat}}(k_d)$
$v_{u,d}^{\text{free}}$
Q
P
last vehicle
first vehicle
representative vehicle
$v_{u,d}^{\text{middle}}$
$v_{u,d}^{\text{idle}}$
End of cycle
t_0
$t_{\text{first}}^{\text{acc,G}_3\text{,usat}}(k_d) = t_{\text{last}}^{\text{acc,G}_2\text{,usat}}(k_d)$
time

(a) Traffic behaviors for the first, last, and representative vehicle in group 1.

(b) Traffic behaviors for the first, last, and representative vehicle in group 2.

(c) Traffic behaviors for the first, last, and representative vehicle in group 3.

Figure 6.6: Traffic behavior of different groups for the ***under-saturated*** *urban traffic scenario.*

length of the link, the free-flow speed, etc.) for a while before it leaves the link. The last vehicle of group 1 will start to accelerate only after all other vehicles in group 1 have done so.

Looking at the representative vehicle of group 1, this vehicle should travel the distance of $\frac{0.5 n_{u,d}^{\mathrm{G_1,usat}}(k_d)\, l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}}$ partly with the constant acceleration $a_{u,d}^{\mathrm{acc}}$ and partly by the constant speed $v_{u,d}^{\mathrm{free}}$. Hence, for the representative vehicle of group 1 we can write

$$\frac{0.5 n_{u,d}^{\mathrm{G_1,sat}}(k_d)\, l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}} = v_{u,d}^{\mathrm{free}} \cdot T_{\mathrm{rep}}^{\mathrm{free,G_1}}(k_d) + \frac{a_{u,d}^{\mathrm{acc}}}{2}\left(\frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right)^2 + v_{u,d}^{\mathrm{idle}}\left(\frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right), \tag{6.46}$$

which results in the first equation of (6.38) when solved for $T_{\mathrm{rep,G_1}}^{\mathrm{free}}(k_d)$.

Since for the under-saturated scenario, the leaving rate of the vehicles in the queue is the saturated leaving flow rate (i.e., $\mu_{u,d}$), we can write

$$t_{\mathrm{last}}^{\mathrm{acc,G_1,usat}}(k_d) - t_{\mathrm{first}}^{\mathrm{acc,G_1,usat}}(k_d) = \frac{n_{u,d}^{\mathrm{G_1,usat}}(k_d)}{\mu_{u,d}}, \tag{6.47}$$

where $t_{\mathrm{first}}^{\mathrm{acc,usat,G_1}}(k_d)$ and $t_{\mathrm{last}}^{\mathrm{acc,usat,}G_1}(k_d)$ denote, respectively, the time instant the first and the last vehicle in group 1 start to accelerate. Hence, the idling times of the first and the last vehicle in group 1 are

$$T_{\mathrm{first}}^{\mathrm{idle,G_1,usat}}(k_d) = c_d - g_{u,d}(k_d), \tag{6.48}$$

$$T_{\mathrm{last}}^{\mathrm{idle,G_1,usat}}(k_d) = c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,usat}}(k_d)}{\mu_{u,d}}. \tag{6.49}$$

The idling time of the representative vehicle of group 1 for the under-saturated scenario is the average of the idling times of the first and the last vehicle (see the first equation of (6.41)), i.e.,

$$T_{\mathrm{rep}}^{\mathrm{idle,G_1,usat}}(k_d) = c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,usat}}(k_d)}{2\mu_{u,d}}. \tag{6.50}$$

Figure 6.6(b) illustrates the time-speed curves of the first, the last, and the representative vehicle in group 2 of the under-saturated scenario, where the red curve (corresponding to the representative vehicle of group 2) is identical to the red (solid) curve in Figure 6.5. Similar to the vehicles in group 1, a free-flow behavior might or might not be observed at the end of the trip for vehicles in group 2. After the traffic light turns green, vehicles in group 2 will start to accelerate with a delay: the first vehicle will accelerate just after the last vehicle in group 1 does (i.e., at $t_{\mathrm{last}}^{\mathrm{acc,G_1,usat}}(k_d)$ in Figure 6.6(a), which is the same as $t_{\mathrm{first}}^{\mathrm{acc,G_2,usat}}(k_d)$ in Figure 6.6(b)). The last vehicle of group 2 will move ahead such that it starts to accelerate immediately after it reaches $v_{u,d}^{\mathrm{idle}}$ (see Figure 6.6(b)). Note that since the next arriving vehicle decelerates to a speed higher than $v_{u,d}^{\mathrm{idle}}$ (i.e., it does not idle at all), it will be put in a separate group, i.e., group 3 discussed later.

All vehicles in group 2 become a part of the waiting queue before they start to accelerate, and they will also leave the link with rate $\mu_{u,d}$. The total travel time of the first vehicle in group 2 is composed of the following terms:

- free-flow (start) + decelerating term:

$$\tau_{u,d}^{\text{usat},\text{G}_2}(k_d), \tag{6.51}$$

where $\tau_{u,d}^{\text{usat},\text{G}_2}(k_d)$ is the average time duration for the vehicles in group 2 to reach the tail of the waiting queue from the instant they enter the link (see Figure 6.6(b)). Note that the time period of the free-flow behavior (at start) alone is

$$\tau_{u,d}^{\text{usat},\text{G}_2}(k_d) - \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}; \tag{6.52}$$

- idling (see Figure 6.6(b)) term:

$$T_{\text{last}}^{\text{idle},\text{G}_1,\text{usat}}(k_d) - \tau_{u,d}^{\text{usat},\text{G}_2}(k_d); \tag{6.53}$$

- accelerating term:

$$\frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}}; \tag{6.54}$$

- free-flow (end) term:

$$\frac{n_{u,d}^{\text{G}_1,\text{usat}}(k_d)\, l^{\text{veh}}}{N_{u,d}^{\text{lane}}\, v_{u,d}^{\text{free}}} - \frac{\left(v_{u,d}^{\text{free}}\right)^2 - \left(v_{u,d}^{\text{idle}}\right)^2}{2a_{u,d}^{\text{acc}}\, v_{u,d}^{\text{free}}}. \tag{6.55}$$

We should notice that the number of the vehicles in front of the first vehicle in group 2 is $n_{u,d}^{\text{G}_1,\text{usat}}(k_d)$ when the traffic light turns green. This means that a distance of $\frac{n_{u,d}^{\text{G}_1,\text{usat}}(k_d)\, l^{\text{veh}}}{N_{u,d}^{\text{lane}}}$ should be traveled by the first vehicle of this group to cross the traffic light (partly with the constant acceleration $a_{u,d}^{\text{acc}}$ and partly with the constant speed $v_{u,d}^{\text{free}}$). From Lemma 6.2, we obtain

$$\frac{n_{u,d}^{\text{G}_1,\text{sat}}(k_d)\, l^{\text{veh}}}{N_{u,d}^{\text{lane}}} = v_{u,d}^{\text{free}} \cdot T_{\text{first}}^{\text{free},\text{G}_2}(k_d) + \frac{a_{u,d}^{\text{acc}}}{2}\left(\frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}}\right)^2 + v_{u,d}^{\text{idle}}\left(\frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}}\right), \tag{6.56}$$

which reduces to (6.55) when solved for $T_{\text{first}}^{\text{free},\text{G}_2}(k_d)$.

Similarly, we can find the total travel time of the last vehicle of group 2, which is the same as that of the first vehicle (resulting in the second equations of (6.41), (6.43), and (6.45)) except that it does not include the idling time, and that the free-flow (end) time using Lemma 6.2 is[7]

$$\frac{\left(n_{u,d}^{\mathrm{G_1,usat}}(k_d)+n_{u,d}^{\mathrm{G_2,usat}}(k_d)\right)l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}\,v_{u,d}^{\mathrm{free}}}-\frac{\left(v_{u,d}^{\mathrm{free}}\right)^2-\left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}}\,v_{u,d}^{\mathrm{free}}}. \tag{6.57}$$

Therefore, for the representative vehicle of group 2, the time duration of the free-flow behavior at the end of its presence in the link (see the first two equations of (6.38)) is:

$$\frac{\left(n_{u,d}^{\mathrm{G_1,usat}}(k_d)+0.5n_{u,d}^{\mathrm{G_2,usat}}(k_d)\right)l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}\,v_{u,d}^{\mathrm{free}}}-\frac{\left(v_{u,d}^{\mathrm{free}}\right)^2-\left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}}\,v_{u,d}^{\mathrm{free}}}. \tag{6.58}$$

Finally, for the vehicles in group 2 we can write

$$\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)\,T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d)=\mu_{u,d}\,T_{u,d}^{\mathrm{leave,G_2,usat}}(k_d), \tag{6.59}$$

where $T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d)$ is the duration of the time period during which vehicles in group 2 reach the tail of the queue, and $T_{u,d}^{\mathrm{leave,G_2,usat}}(k_d)$ is the duration of the time period during which vehicles in group 2 leave the link. Moreover, from Figure 6.6(b) we have

$$T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d)+T_{\mathrm{last}}^{\mathrm{total,G_2,usat}}(k_d)=T_{u,d}^{\mathrm{leave,G_2,usat}}(k_d)+T_{\mathrm{first}}^{\mathrm{total,G_2,usat}}(k_d), \tag{6.60}$$

which results in (6.34) when solved for $T_{u,d}^{\mathrm{arrive,G_2,usat}}(k_d)$. Note that from (6.28), $\alpha_{u,d}^{\mathrm{arrive,q}}(k_d) < \mu_{u,d}$, and hence the denominator of (6.34) will never become zero.

As indicated before the arriving vehicles are divided into three groups for the under-saturated scenario, and since the vehicles that are already in the link at the beginning of the current cycle are put in group 1, then these three groups extracted from the arriving vehicles are put in group 2–group 4 of the under-saturated scenario. The vehicles in group 3 are different from the vehicles in group 2 in the sense that they will also decelerate, but to speed values larger than $v_{u,d}^{\mathrm{idle}}$ (since the queue composed of groups 1 and 2 has already started to move and there is no need anymore for the arriving vehicles to idle). Figure 6.6(c) shows the time-speed curves of the first, the last, and the representative vehicle in group 3. Comparing this figure with Figure 6.5, the green curves, i.e., the time-speed curves of the representative vehicle in group 3, are identical.

The first vehicle in group 3 arrives just after the last vehicle in group 2 (see Figure 6.6(c)). In the limit, the time-speed curves of the last vehicle in group 2 and the first vehicle in group 3 become exactly the same (compare Figures 6.6(b) and 6.6(c)). The next arriving vehicles will show the same behavior as the first vehicle, but each of them decelerates to a speed (we call it the *transition speed*) that is in general larger than the speed of the previous vehicle (i.e., the transition speed increases gradually for successive vehicles). The time-speed curve corresponding to the last vehicle in group 3 becomes a straight line in the limit (see Figure 6.6(c)).

[7] Note that the number of the vehicles in front of the last vehicle in group 2 is $n_{u,d}^{\mathrm{G_1,usat}}(k_d)+n_{u,d}^{\mathrm{G_2,usat}}(k_d)$.

Assuming that the changes observed in the transition speed of the vehicles in group 3 occurs gradually and linearly, then the middle speed that corresponds to the representative vehicle of group 3 is the average of the transition speed of the first vehicle in this group (i.e., $v_{u,d}^{\text{idle}}$) and the transition speed of the last vehicle in this group (which in the limit is $v_{u,d}^{\text{free}}$. This explains (6.29). Note that the point "P" in this figure separates groups 3 and 4, w.r.t. the arrival times. The point "Q" is the starting point of the time-speed curve of the last vehicle in group 3. The transition speed corresponding to the representative vehicle in group 3, i.e., the middle speed $v_{u,d}^{\text{middle}}$ is the mean value of the transition speed of the first vehicles (i.e., $v_{u,d}^{\text{idle}}$) and the last vehicle (i.e., $v_{u,d}^{\text{free}}$). Therefore, we obtain (6.29).

Moreover, since the vehicles in group 3 will not idle at all during the entire cycle, the leaving flow rate for them is the same as their arriving flow rate, i.e.,

$$\alpha_{u,d}^{\text{leave,l,G3}}(k_d) = \alpha_{u,d}^{\text{arrive,l,G3}}(k_d). \tag{6.61}$$

From Figure 6.6(c), we can write

$$\alpha_{u,d}^{\text{arrive,q}}(k_d)\,T_{u,d}^{\text{arrive,G3,usat}}(k_d) = \alpha_{u,d}^{\text{leave,l}}(k_d)\,T_{u,d}^{\text{leave,G3,usat}}(k_d), \tag{6.62}$$

which in combination with (6.61) gives

$$T_{u,d}^{\text{arrive,G3,usat}}(k_d) = T_{u,d}^{\text{leave,G3,usat}}(k_d). \tag{6.63}$$

In addition, from (6.63) and Figure 6.6(c), we obtain

$$T_{\text{last}}^{\text{total,G3,usat}}(k_d) = T_{\text{first}}^{\text{total,G3,usat}}(k_d). \tag{6.64}$$

The last vehicle in group 3 travels through the entire link with a constant speed $v_{u,d}^{\text{free}}$. Hence,

$$T_{\text{last}}^{\text{G3,usat}}(k_d) = \frac{C_{u,d}\,l^{\text{veh}}}{N_{u,d}^{\text{lane}}\,v_{u,d}^{\text{free}}}, \tag{6.65}$$

where $\dfrac{C_{u,d}\,l^{\text{veh}}}{N_{u,d}^{\text{lane}}}$ is the total length of the link. For the first vehicle in group 3 we have:

$$T_{\text{first}}^{\text{total,G3,usat}}(k_d) = \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}} + \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}} + T_{\text{first}}^{\text{free,G3,usat}}(k_d), \tag{6.66}$$

and from (6.64) and (6.65), we finally have

$$T_{\text{first}}^{\text{free,G3,usat}}(k_d) = \frac{C_{u,d}\,l^{\text{veh}}}{N_{u,d}^{\text{lane}}\,v_{u,d}^{\text{free}}} - \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}} - \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}}, \tag{6.67}$$

which is indeed the third equation of (6.38). Note that in the limit, the points P and Q will coincide, and therefore, (6.35) will be obtained.

The rest of the equations of the under-saturated model that are not explicitly explained in detail can easily be obtained with a similar approach (also see Figure 6.5).

6.5 Flow-emission model for the saturated scenario

The saturated scenario is observed when the demand is larger than the number of vehicles that can be discharged by the saturated average leaving flow rate in one cycle. For the saturated traffic scenario, the initial queue in the link will completely be dissolved during one cycle, but a new queue will be formed by a part of the arriving vehicles (i.e., not all the arriving vehicles in a saturated traffic scenario can leave the link during the green phase). Mathematically, this is formulated as follows:

$$q_{u,d}(k_d) \leq \sum_{o\in\mathcal{O}_{u,d}} \beta_{u,d,o} \cdot \mu_{u,d} \cdot g_{u,d,o}(k_d) < q_{u,d}(k_d) + c_d \cdot \alpha_{u,d}^{\text{arrive,q}}(k_d). \tag{6.68}$$

For a saturated urban traffic scenario, the average leaving flow rate during the entire cycle is the saturated leaving flow rate, $\mu_{u,d}$. Two cases are possible, i.e., $\alpha_{u,d}^{\text{arrive,q}}(k_d) < \mu_{u,d}$ and $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$, which result in different model equations. We first develop the model for case 1, i.e., $\alpha_{u,d}^{\text{arrive,q}}(k_d) < \mu_{u,d}$, and later we extend the model for case 2, i.e., $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$.

6.5.1 Case 1: $\alpha_{u,d}^{\textbf{arrive,q}}(k_d) < \mu_{u,d}$

First, we give the equations, and then we explain how the equations are obtained. Some of the equations may not be explained explicitly, as they can be easily found with a similar approach as for the under-saturated scenario considering Figure 6.8. The vehicles observed during one cycle are divided into the following three groups:

Group 1: composed of those vehicles that are already in the link standing in a queue at the beginning of the cycle (see the blue curve in Figure 6.7 for the average behavior of this group); the average behavior of this group is identical to that of the first group for an under-saturated scenario (see Section 6.4).

Group 2: composed of those vehicles that arrive at the tail of the waiting queue during the current cycle after they decelerate from $v_{u,d}^{\text{free}}$ to $v_{u,d}^{\text{idle}}$. They idle for a while behind the initial queue, and then accelerate just after the queue in front of them has been dissolved. They finally leave the link during the current cycle (see the red curve in Figure 6.7 for the average behavior of this group). The observed behavior for this group of vehicles, is the same as the observed behavior for group 2 in the under-saturated scenario.

Group 3: composed of those vehicles that arrive at the tail of the queue in the current cycle, when the queue in front of them has already started to move. Therefore, they should

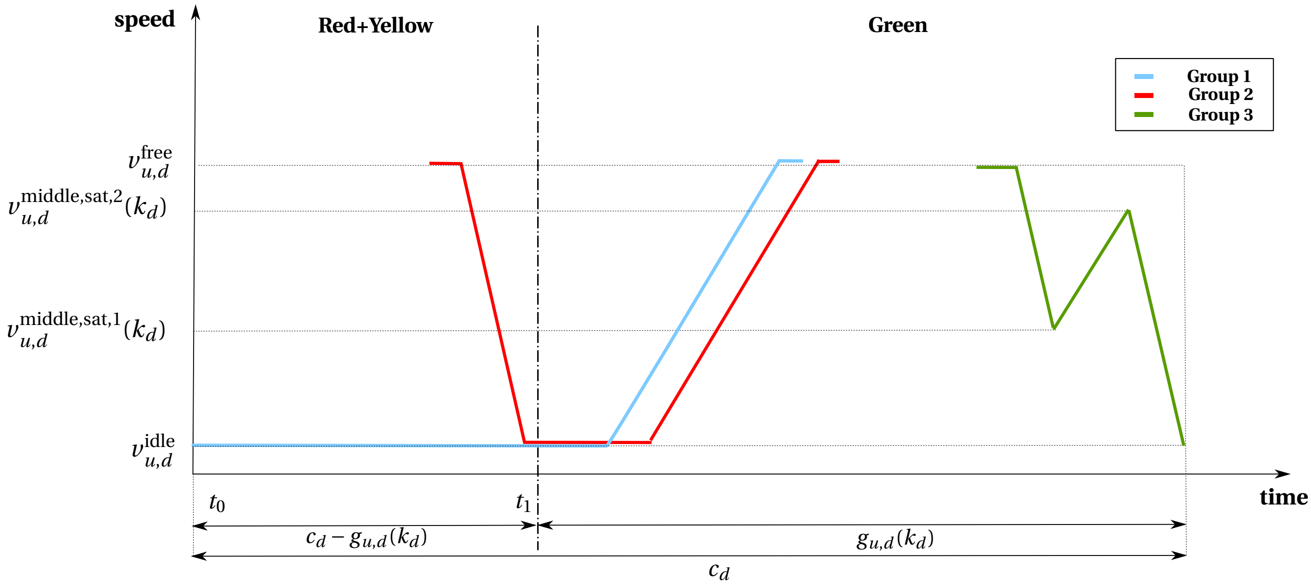


*Figure 6.7: Traffic behaviors on link (u,d) within one cycle for the **saturated** urban traffic scenario, case 1: $\alpha^{\text{arrive,q}}_{u,d}(k_d) < \mu_{u,d}$.*

first decelerate from $v^{\text{free}}_{u,d}$ to a speed value larger than $v^{\text{idle}}_{u,d}$, i.e., to $v^{\text{middle,sat,1}}_{u,d}(k_d)$ (see the green curve in Figure 6.7 for the average behavior of this group). Then, with the moving traffic they also accelerate to reach the speed $v^{\text{middle,sat,2}}_{u,d}(k_d)$, when they notice that there is not enough time to reach and pass the traffic light, and they decelerate again to reach $v^{\text{idle}}_{u,d}$ by the time the traffic light turns red. The values of $v^{\text{middle,sat,1}}_{u,d}(k_d)$ and $v^{\text{middle,sat,2}}_{u,d}$ are computed by

$$v^{\text{middle,sat,1}}_{u,d}(k_d) = \frac{v^{\text{idle}}_{u,d} + v^{\text{middle,sat,2}}_{u,d}(k_d)}{2}, \tag{6.69}$$

$$v^{\text{middle,sat,2}}_{u,d}(k_d) = v^{\text{idle}}_{u,d} + \frac{n^{\text{G}_3\text{,sat}}_{u,d}(k_d)}{\alpha^{\text{arrive,q}}_{u,d}(k_d)} \cdot \frac{a^{\text{acc}}_{u,d} a^{\text{dec}}_{u,d}}{a^{\text{dec}}_{u,d} - a^{\text{acc}}_{u,d}}. \tag{6.70}$$

For the number of vehicles in each of the groups, we have

$$n^{\text{G}_1\text{,sat}}_{u,d}(k_d) = q_{u,d}(k_d), \tag{6.71}$$

$$\begin{aligned} n^{\text{G}_2\text{,sat}}_{u,d}(k_d) = \min\Bigg\{ & n^{\text{sat}}_{u,d}(k_d) - n^{\text{G}_1\text{,sat}}_{u,d}(k_d), \max\Bigg\{ \alpha^{\text{arrive,q}}_{u,d}(k_d) \cdot T^{\text{arrive,G}_2\text{,sat}}_{u,d}(k_d), \\ & n^{\text{sat}}_{u,d}(k_d) - n^{\text{G}_1\text{,sat}}_{u,d}(k_d) - \alpha^{\text{arrive,q}}_{u,d}(k_d)\left(v^{\text{free}}_{u,d} - v^{\text{idle}}_{u,d}\right)\left(\frac{1}{a^{\text{acc}}_{u,d}} - \frac{1}{a^{\text{dec}}_{u,d}}\right)\Bigg\}\Bigg\}, \end{aligned} \tag{6.72}$$

$$n_{u,d}^{\mathrm{G_3,sat}}(k_d) = n_{u,d}^{\mathrm{sat}}(k_d) - \sum_{i=1}^{2} n_{u,d}^{\mathrm{G}_i\mathrm{,sat}}(k_d), \tag{6.73}$$

with $T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)$ computed by

$$\begin{aligned} T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d) =& \frac{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}} + \alpha_{u,d}^{\mathrm{arrive,q}}(k_d) l^{\mathrm{veh}}} \cdot \\ & \left(c_d - \frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d) l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} + \frac{\left(v_{u,d}^{\mathrm{free}}\right)^2 - \left(v_{u,d}^{\mathrm{idle}}\right)^2}{2 v_{u,d}^{\mathrm{free}} a_{u,d}^{\mathrm{acc}}} - \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) \right), \end{aligned} \tag{6.74}$$

and $\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)$ computed similarly to $\tau_{u,d}^{\mathrm{usat,G_2}}(k_d)$, see (6.36).

The number of vehicles in each group that show a specific behavior, and also the time that the representative vehicle shows the behavior are given next. These values can be used to compute the emissions in the saturated scenario using the equations introduced in Section 6.3.2.

- For the free-flow behavior, we have

$$\begin{aligned} n_{u,d}^{\mathrm{free,G_1,sat}}(k_d) &= n_{u,d}^{\mathrm{G_1,sat}}(k_d), \\ n_{u,d}^{\mathrm{free,G_2,sat}}(k_d) &= n_{u,d}^{\mathrm{G_2,sat}}(k_d), \\ n_{u,d}^{\mathrm{free,G_3,sat}}(k_d) &= n_{u,d}^{\mathrm{G_3,sat}}(k_d), \end{aligned} \tag{6.75}$$

and

$$\begin{aligned} T_{u,d}^{\mathrm{free,G_1,sat}}(k_d) &= \frac{\left(n_{u,d}^{\mathrm{G_1,sat}}(k_d) + 0.5 n_{u,d}^{\mathrm{G_2,sat}}(k_d)\right) l^{\mathrm{veh}}}{v_{u,d}^{\mathrm{free}}} - \frac{\left(v_{u,d}^{\mathrm{free}}\right)^2 - \left(v_{u,d}^{\mathrm{idle}}\right)^2}{2 a_{u,d}^{\mathrm{acc}} v_{u,d}^{\mathrm{free}}}, \\ T_{u,d}^{\mathrm{free,G_2,sat}}(k_d) &= \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) - \frac{v_{u,d}^{\mathrm{idle}} - v_{u,d}^{\mathrm{free}}}{a_{u,d}^{\mathrm{dec}}} + T_{u,d}^{\mathrm{free,G_1,sat}}(k_d), \\ T_{u,d}^{\mathrm{free,G_3,sat}}(k_d) &= \tau_{u,d}^{\mathrm{sat,G_3}}(k_d) - \frac{v_{u,d}^{\mathrm{middle,sat,1}}(k_d) - v_{u,d}^{\mathrm{free}}}{a_{u,d}^{\mathrm{dec}}}, \end{aligned} \tag{6.76}$$

with $\tau_{u,d}^{\mathrm{sat,G_3}}(k_d)$ computed by[8]

$$\tau_{u,d}^{\mathrm{sat,G_3}}(k_d) = \frac{C_{u,d} - n_{u,d}^{\mathrm{G_1,usat}}(k_d) - n_{u,d}^{\mathrm{G_2,usat}}(k_d)}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} l^{\mathrm{veh}} - \frac{\left(v_{u,d}^{\mathrm{middle,sat,1}}(k_d) - v_{u,d}^{\mathrm{free}}\right)^2}{2 a_{u,d}^{\mathrm{dec}} v_{u,d}^{\mathrm{free}}}. \tag{6.77}$$

[8] Note that the number of vehicles already in the queue in front of group 3 is $n_{u,d}^{\mathrm{G_1,usat}}(k_d) + n_{u,d}^{\mathrm{G_2,usat}}(k_d)$, and the queue is already moving forward with $v_{u,d}^{\mathrm{middle,sat,1}}(k_d)$.

- For the idling behavior, we have

$$\begin{aligned}
&n_{u,d}^{\text{idle},\text{G}_1,\text{sat}}(k_d) = n_{u,d}^{\text{G}_1,\text{sat}}(k_d),\\
&n_{u,d}^{\text{idle},\text{G}_2,\text{sat}}(k_d) = n_{u,d}^{\text{G}_2,\text{sat}}(k_d),\\
&n_{u,d}^{\text{idle},\text{G}_3,\text{sat}}(k_d) = 0,
\end{aligned} \tag{6.78}$$

and

$$\begin{aligned}
&T_{u,d}^{\text{idle},\text{G}_1,\text{sat}}(k_d) = c_d - g_{u,d}(k_d) + \frac{0.5 n_{u,d}^{\text{G}_1,\text{sat}}(k_d)}{\mu_{u,d}},\\
&T_{u,d}^{\text{idle},\text{G}_2,\text{sat}}(k_d) = T_{u,d}^{\text{idle},\text{G}_1,\text{sat}}(k_d) - \tau_{u,d}^{\text{sat},\text{G}_2}(k_d),\\
&T_{u,d}^{\text{idle},\text{G}_3,\text{sat}}(k_d) = 0.
\end{aligned} \tag{6.79}$$

- For the decelerating behavior, we have

$$\begin{aligned}
&n_{u,d}^{\text{dec},\text{G}_1,\text{sat}}(k_d) = 0,\\
&n_{u,d}^{\text{dec},\text{G}_2,\text{sat}}(k_d) = n_{u,d}^{\text{G}_2,\text{sat}}(k_d),\\
&n_{u,d}^{\text{dec},\text{G}_3,\text{sat}}(k_d) = n_{u,d}^{\text{G}_3,\text{sat}}(k_d),
\end{aligned} \tag{6.80}$$

and

$$\begin{aligned}
&T_{u,d}^{\text{dec},\text{G}_1,\text{sat}}(k_d) = 0,\\
&T_{u,d}^{\text{dec},\text{G}_2,\text{sat}}(k_d) = \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}},\\
&T_{u,d}^{\text{dec},\text{G}_3,\text{sat}}(k_d) = \frac{v_{u,d}^{\text{middle},\text{sat},1}(k_d) - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}} + \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{middle},\text{sat},2}(k_d)}{a_{u,d}^{\text{dec}}}.
\end{aligned} \tag{6.81}$$

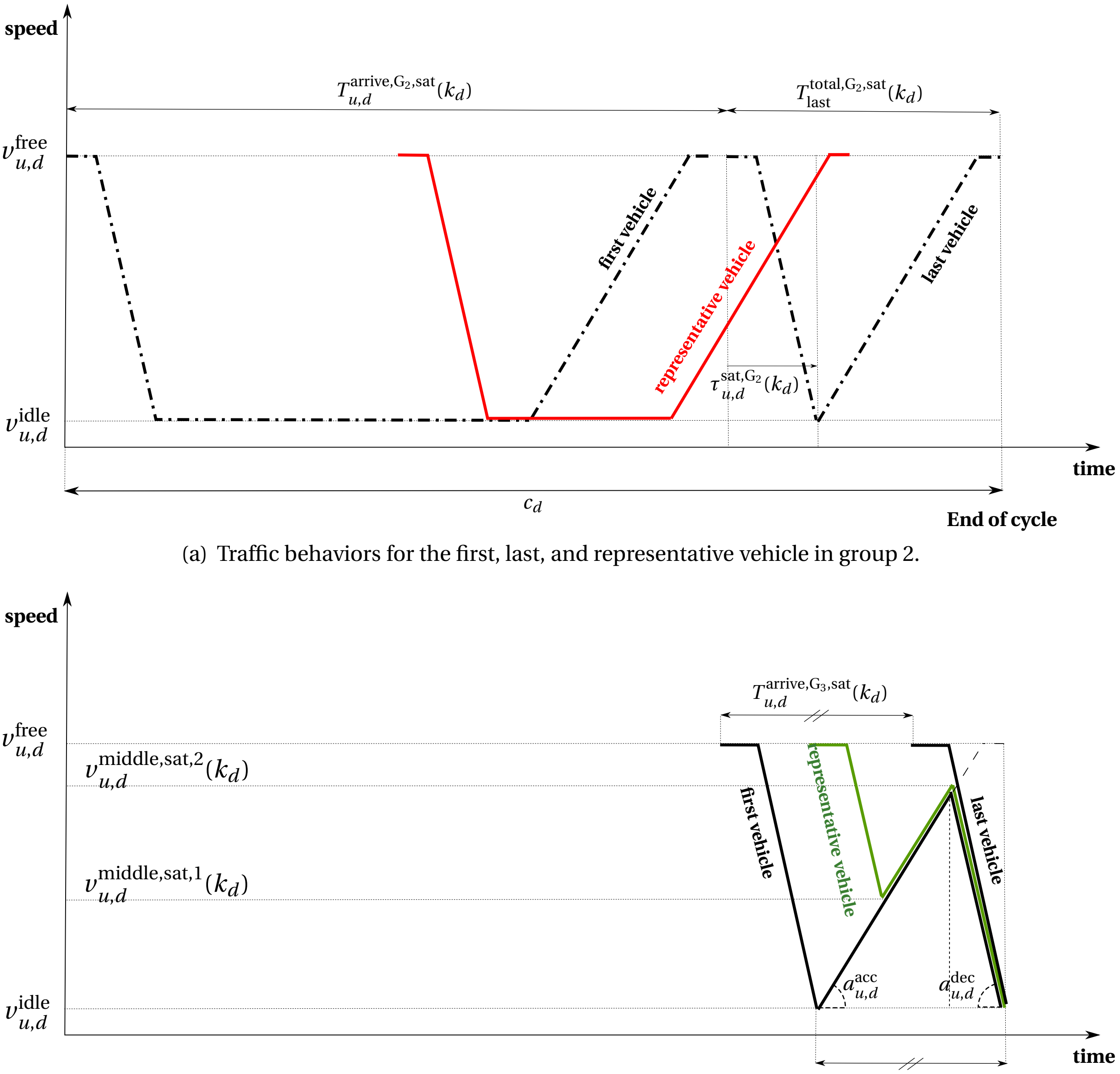
- For the accelerating behavior, we have

$$\begin{aligned}
&n_{u,d}^{\text{acc},\text{G}_1,\text{sat}}(k_d) = n_{u,d}^{\text{G}_1,\text{sat}}(k_d),\\
&n_{u,d}^{\text{acc},\text{G}_2,\text{sat}}(k_d) = n_{u,d}^{\text{G}_2,\text{sat}}(k_d),\\
&n_{u,d}^{\text{acc},\text{G}_3,\text{sat}}(k_d) = n_{u,d}^{\text{G}_3,\text{sat}}(k_d),
\end{aligned} \tag{6.82}$$

and

$$\begin{aligned}
&T_{u,d}^{\text{acc},\text{G}_1,\text{sat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc},\text{G}_2,\text{sat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc},\text{G}_3,\text{sat}}(k_d) = \frac{v_{u,d}^{\text{middle},\text{sat},2}(k_d) - v_{u,d}^{\text{middle},\text{sat},1}(k_d)}{a_{u,d}^{\text{acc}}}.
\end{aligned} \tag{6.83}$$

Now, we explain in more detail how the equations presented above are obtained. Since $\alpha_{u,d}^{\text{arrive},\text{q}}(k_d) < \mu_{u,d}$, i.e., the arriving flow of the vehicles is less than the leaving flow rate, the temporal distance between the arrival times of two successive vehicles in group 2 should be less than the temporal distance between their leaving times. This implies that for each vehicle



(a) Traffic behaviors for the first, last, and representative vehicle in group 2.

(b) Traffic behaviors for the first, last, and representative vehicle in group 3.

*Figure 6.8: Traffic behavior of different groups for the **saturated** urban traffic scenario, **case 1**.*

in group 2 compared with its predecessor, the length of the idling behavior will be reduced in the time-speed curve. Figure 6.8(a), illustrating the time-speed curves of the first, last, and representative vehicle of group 2, is obtained based on this fact. From this figure, the arrival time of the vehicles in group 2 (i.e., the time period between the arrival instant of the first and the last vehicle) for the saturated scenario is

$$T_{u,d}^{\text{arrive,G}_2\text{,sat}}(k_d) = c_d - T_{\text{last}}^{\text{total,G}_2\text{,sat}}(k_d). \tag{6.84}$$

Moreover, for the last vehicle in group 2, from Lemma 6.2 for the accelerating and for the free-flow behavior at the end, we can write

$$\begin{aligned}\frac{\left(n_{u,d}^{\mathrm{G_1,sat}}(k_d)+n_{u,d}^{\mathrm{G_2,sat}}(k_d)\right)l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}} &= v_{u,d}^{\mathrm{free}}\cdot T^{\mathrm{free,end}}(k_d)+\\ &\frac{a_{u,d}^{\mathrm{acc}}}{2}\left(\frac{v_{u,d}^{\mathrm{free}}-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right)^2+v_{u,d}^{\mathrm{idle}}\left(\frac{v_{u,d}^{\mathrm{free}}-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right),\end{aligned} \tag{6.85}$$

from which $T^{\mathrm{free,end}}(k_d)$ is found. Moreover,

$$T_{\mathrm{last}}^{\mathrm{total,G_2,sat}}(k_d)=\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)+\frac{v_{u,d}^{\mathrm{free}}-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}+T^{\mathrm{free,end}}(k_d), \tag{6.86}$$

and

$$n_{u,d}^{\mathrm{G_2,sat}}(k_d)=\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)\,T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d). \tag{6.87}$$

By substituting (6.86) into (6.84) and (6.87) into the expression for $T^{\mathrm{free,end}}(k_d)$ resulting from (6.85), (6.74) is obtained.

Next, we explain how Equations (6.69) and (6.70) are obtained. We consider the time-speed curves of the first, the last, and the representative vehicle in group 3 (see Figure 6.8(b)). Since $v_{u,d}^{\mathrm{middle,sat,1}}(k_d)$ corresponds to the average behavior of group 3, we consider it as the average of $v_{u,d}^{\mathrm{idle}}$ (corresponding to the first vehicle) and $v_{u,d}^{\mathrm{middle,sat,2}}(k_d)$ (corresponding to the last vehicle (see (6.69))). To obtain $v_{u,d}^{\mathrm{middle,sat,2}}(k_d)$, we can write the following relationships:

$$n_{u,d}^{\mathrm{sat}}(k_d)=\sum_{i=1}^{3}n_{u,d}^{\mathrm{G}_i\mathrm{,sat}}(k_d), \tag{6.88}$$

where $n_{u,d}^{\mathrm{sat}}(k_d)$ is computed by the flow model. We can write

$$T_{u,d}^{\mathrm{arrive,G_3,sat}}(k_d)=\frac{n_{u,d}^{\mathrm{G_3,sat}}(k_d)}{\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)}. \tag{6.89}$$

Moreover, from Figure 6.8(b) we have

$$\begin{aligned}T_{u,d}^{\mathrm{arrive,G_3,sat}}(k_d)&=\frac{v_{u,d}^{\mathrm{middle,sat,2}}(k_d)-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}+\frac{v_{u,d}^{\mathrm{idle}}-v_{u,d}^{\mathrm{middle,sat,2}}(k_d)}{a_{u,d}^{\mathrm{dec}}}\\ &=\left(v_{u,d}^{\mathrm{middle,sat,2}}(k_d)-v_{u,d}^{\mathrm{idle}}\right)\left(\frac{1}{a_{u,d}^{\mathrm{acc}}}-\frac{1}{a_{u,d}^{\mathrm{dec}}}\right).\end{aligned} \tag{6.90}$$

Therefore, we obtain (6.70), i.e.,

$$v_{u,d}^{\text{middle,sat,2}}(k_d) = v_{u,d}^{\text{idle}} + \frac{n_{u,d}^{\text{G}_3\text{,sat}}(k_d)}{\alpha_{u,d}^{\text{arrive,q}}(k_d)} \left(\frac{a_{u,d}^{\text{acc}} a_{u,d}^{\text{dec}}}{a_{u,d}^{\text{dec}} - a_{u,d}^{\text{acc}}} \right). \tag{6.91}$$

However, there is no guarantee yet that $v_{u,d}^{\text{middle,sat,2}}(k_d)$ given by (6.91) does never exceed $v_{u,d}^{\text{free}}$. The definition of $n_{u,d}^{\text{G}_2\text{,sat}}(k_d)$ given by (6.72) gives us this guarantee, since $v_{u,d}^{\text{middle,sat,2}}(k_d) \leq v_{u,d}^{\text{free}}$ implies that

$$\left(v_{u,d}^{\text{middle,sat,2}}(k_d) - v_{u,d}^{\text{idle}} \right) \left(\frac{1}{a_{u,d}^{\text{acc}}} - \frac{1}{a_{u,d}^{\text{dec}}} \right) \leq \left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}} \right) \left(\frac{1}{a_{u,d}^{\text{acc}}} - \frac{1}{a_{u,d}^{\text{dec}}} \right), \tag{6.92}$$

with the left-hand side of (6.92) equal to $T_{u,d}^{\text{arrive,G}_3\text{,sat}}(k_d)$ (see (6.90)). Hence, from (6.89) we obtain

$$n_{u,d}^{\text{G}_3\text{,sat}}(k_d) \leq \alpha_{u,d}^{\text{arrive,q}}(k_d) \left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}} \right) \left(\frac{1}{a_{u,d}^{\text{acc}}} - \frac{1}{a_{u,d}^{\text{dec}}} \right). \tag{6.93}$$

Finally, from (6.88) and (6.93) we obtain (6.72). Moreover, (6.72) together with (6.88) results in (6.93); (6.93) yields (6.92), which itself implies that $v_{u,d}^{\text{middle,sat,2}}(k_d) \leq v_{u,d}^{\text{free}}$ should hold.

6.5.2 Case 2: $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$

For the second case of the saturated urban traffic scenario, i.e., for $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$, the vehicles observed during one cycle are divided into four groups:

Group 1: composed of those vehicles that are already in the link standing in a queue at the beginning of the cycle (see the blue curve in Figure 6.9 for the average behavior of this group); the average behavior of this group is exactly similar to that of the first group for case 1 of the saturated scenario (see Section 6.5.1).

Group 2: composed of those vehicles that arrive at the tail of the waiting queue during the current cycle after they decelerate from $v_{u,d}^{\text{free}}$ to $v_{u,d}^{\text{idle}}$. They idle for a while behind the initial queue, and then accelerate just after the queue in front of them has been dissolved. They finally leave the link during the current cycle (see the red curve in Figure 6.9 for the average behavior of this group). The observed behavior for this group of vehicles is similar to the observed behavior for group 2 for case 1 of the saturated scenario (see Section 6.5.1), except that for the same queue number and green time values, the representative vehicle of group 2 shows an idling behavior for a longer period compared with case 1 (compare Figures 6.7 and 6.9).

Group 3: composed of those vehicles that should decelerate after a while as they approach the idling queue in front of them consisting of the vehicles in group 1 and group 2 (see the green curve in Figure 6.9 for the average behavior of this group). They reach $v_{u,d}^{\text{idle}}$, idle for a while, and then accelerate to reach $v_{u,d}^{\text{middle,sat,2}}(k_d)$ when they notice that there is not enough time to reach and pass the traffic light. Hence, they decelerate to $v_{u,d}^{\text{idle}}$. The value of $v_{u,d}^{\text{middle,sat,2}}$ is given by

$$v_{u,d}^{\text{middle,sat,2}}(k_d) = v_{u,d}^{\text{idle}} + \frac{a_{u,d}^{\text{acc}} a_{u,d}^{\text{dec}}}{a_{u,d}^{\text{dec}} - a_{u,d}^{\text{acc}}} \cdot \left(c_d - \frac{n_{u,d}^{\text{G}_2\text{,sat}}(k_d) + n_{u,d}^{\text{G}_3\text{,sat}}(k_d)}{\alpha_{u,d}^{\text{arrive,q}}(k_d)} - \right.$$
$$\left. \frac{C_{u,d} - n_{u,d}^{\text{G}_1\text{,sat}}(k_d) - n_{u,d}^{\text{G}_2\text{,sat}}(k_d) - n_{u,d}^{\text{G}_3\text{,sat}}(k_d)}{N_{u,d}^{\text{lane}} v_{u,d}^{\text{free}}} l^{\text{veh}} + \frac{\left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right)^2}{2 a_{u,d}^{\text{acc}} v_{u,d}^{\text{free}}} \right). \tag{6.94}$$

Group 4: composed of those vehicles that arrive at the tail of the queue consisting of the vehicles in groups 1, 2, and 3, when the queue has already started to move forward (see the black curve in Figure 6.9 for the average behavior of vehicles in group 4). Hence, as they approach the moving queue they decelerate to reach $v_{u,d}^{\text{middle,sat,1}}(k_d)$ given by

$$v_{u,d}^{\text{middle,sat,1}}(k_d) = \frac{v_{u,d}^{\text{idle}} + v_{u,d}^{\text{middle,sat,2}}(k_d)}{2}. \tag{6.95}$$

Then they accelerate with the queue until they reach $v_{u,d}^{\text{middle,sat,2}}(k_d)$. At this moment, since the vehicles in group 3 in front of them start to decelerate, the vehicles in group 4 will follow the same behavior and decelerate to $v_{u,d}^{\text{idle}}$.

For the number of vehicles in each group, we have

$$n_{u,d}^{\text{G}_1\text{,sat}}(k_d) = q_{u,d}(k_d), \tag{6.96}$$

$$n_{u,d}^{\text{G}_2\text{,sat}}(k_d) = \min\left\{ \alpha_{u,d}^{\text{arrive,q}}(k_d) \cdot T_{u,d}^{\text{arrive,G}_2\text{,sat}}(k_d), n_{u,d}^{\text{sat}}(k_d) - n_{u,d}^{\text{G}_1\text{,sat}}(k_d) \right\}, \tag{6.97}$$

$$n_{u,d}^{\text{G}_3\text{,sat}}(k_d) = \min\left\{ \alpha_{u,d}^{\text{arrive,q}}(k_d) \cdot T_{u,d}^{\text{arrive,G}_3\text{,sat}}(k_d), \right.$$
$$\left. \frac{\left(v_{u,d}^{\text{free}}\right)^2 - \left(v_{u,d}^{\text{idle}}\right)^2}{l^{\text{veh}}} N_{u,d}^{\text{lane}} \left(\frac{1}{a_{u,d}^{\text{acc}}} - \frac{1}{a_{u,d}^{\text{dec}}} \right) - n_{u,d}^{\text{G}_1\text{,sat}}(k_d) - n_{u,d}^{\text{G}_2\text{,sat}}(k_d) \right\}, \tag{6.98}$$

$$n_{u,d}^{\text{G}_4\text{,sat}}(k_d) = n_{u,d}^{\text{sat}}(k_d) - \sum_{i=1}^{3} n_{u,d}^{\text{G}_i\text{,sat}}(k_d), \tag{6.99}$$

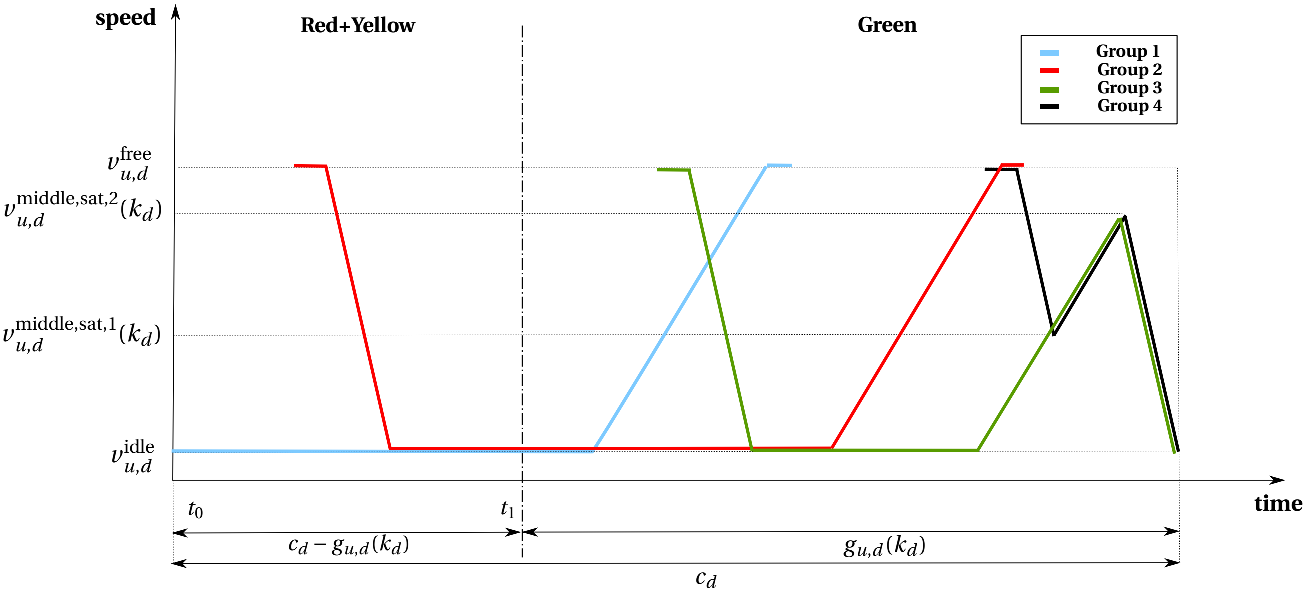


Figure 6.9: Traffic behaviors on link (u,d) within one cycle for the ***saturated*** *urban traffic scenario, case 2: $\alpha^{\text{arrive,q}}_{u,d}(k_d) \geq \mu_{u,d}$.*

with $T^{\text{arrive,G}_2\text{,sat}}_{u,d}(k_d)$ and $T^{\text{arrive,G}_3\text{,sat}}_{u,d}(k_d)$ computed by

$$T^{\text{arrive,G}_2\text{,sat}}_{u,d}(k_d) = \frac{N^{\text{lane}}_{u,d}\, v^{\text{free}}_{u,d}}{\dfrac{2\alpha^{\text{arrive,q}}_{u,d}(k_d)}{\mu_{u,d}} N^{\text{lane}}_{u,d}\, v^{\text{free}}_{u,d} + \alpha^{\text{arrive,q}}_{u,d}(k_d)\, l^{\text{veh}}} \cdot \left(g_{u,d}(k_d) - \frac{n^{\text{G}_1\text{,sat}}_{u,d}(k_d)\, l^{\text{veh}}}{N^{\text{lane}}_{u,d}\, v^{\text{free}}_{u,d}} - \frac{\left(v^{\text{free}}_{u,d} - v^{\text{idle}}_{u,d}\right)^2}{2\, v^{\text{free}}_{u,d}\, a^{\text{acc}}_{u,d}} \right), \tag{6.100}$$

$$\begin{aligned} T^{\text{arrive,G}_3\text{,sat}}_{u,d}(k_d) = {} & \frac{\left(v^{\text{middle,sat,2}}_{u,d}(k_d)\right)^2 - \left(v^{\text{idle}}_{u,d}\right)^2}{\alpha^{\text{arrive,q}}_{u,d}(k_d)\, l^{\text{veh}}} N^{\text{lane}}_{u,d} \left(\frac{1}{a^{\text{acc}}_{u,d}} - \frac{1}{a^{\text{dec}}_{u,d}} \right) \\ & - T^{\text{arrive,G}_2\text{,sat}}_{u,d}(k_d) - \frac{n^{\text{G}_1\text{,sat}}_{u,d}(k_d)}{\alpha^{\text{arrive,q}}_{u,d}(k_d)}, \end{aligned} \tag{6.101}$$

and $\tau^{\text{sat,G}_2}_{u,d}(k_d)$ by (6.36). Note that we should use the minimum function in (6.97) and (6.98) to make sure that the number of vehicles is non-negative and also that $v^{\text{middle,sat,2}}_{u,d}(k_d)$ does not exceed $v^{\text{free}}_{u,d}$ (for more details we refer the readers to the explanations given in Section 6.5.1 for case 1).

The number of vehicles in each group that show a specific behavior and the time duration of each behavior for case 2 of the urban saturated scenario is given next:

- For the free-flow behavior, we have

$$\begin{aligned}
n_{u,d}^{\mathrm{free,G_1,sat}}(k_d) &= n_{u,d}^{\mathrm{G_1,sat}}(k_d),\\
n_{u,d}^{\mathrm{free,G_2,sat}}(k_d) &= n_{u,d}^{\mathrm{G_2,sat}}(k_d),\\
n_{u,d}^{\mathrm{free,G_3,sat}}(k_d) &= n_{u,d}^{\mathrm{G_3,sat}}(k_d),\\
n_{u,d}^{\mathrm{free,G_4,sat}}(k_d) &= n_{u,d}^{\mathrm{G_4,sat}}(k_d),
\end{aligned} \tag{6.102}$$

and

$$\begin{aligned}
T_{u,d}^{\mathrm{free,G_1,sat}}(k_d) &= \frac{\left(n_{u,d}^{\mathrm{G_1,sat}}(k_d) + 0.5 n_{u,d}^{\mathrm{G_2,sat}}(k_d)\right) l^{\mathrm{veh}}}{v_{u,d}^{\mathrm{free}}} - \frac{\left(v_{u,d}^{\mathrm{free}}\right)^2 - \left(v_{u,d}^{\mathrm{idle}}\right)^2}{2 a_{u,d}^{\mathrm{acc}} v_{u,d}^{\mathrm{free}}},\\
T_{u,d}^{\mathrm{free,G_2,sat}}(k_d) &= \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) - \frac{v_{u,d}^{\mathrm{idle}} - v_{u,d}^{\mathrm{free}}}{a_{u,d}^{\mathrm{dec}}} + T_{u,d}^{\mathrm{free,G_1,sat}}(k_d),\\
T_{u,d}^{\mathrm{free,G_3,sat}}(k_d) &= \tau_{u,d}^{\mathrm{sat,G_3}}(k_d) - \frac{v_{u,d}^{\mathrm{idle}} - v_{u,d}^{\mathrm{free}}}{a_{u,d}^{\mathrm{dec}}},\\
T_{u,d}^{\mathrm{free,G_4,sat}}(k_d) &= \tau_{u,d}^{\mathrm{sat,G_4}}(k_d) - \frac{v_{u,d}^{\mathrm{middle,sat,1}}(k_d) - v_{u,d}^{\mathrm{free}}}{a_{u,d}^{\mathrm{dec}}},
\end{aligned} \tag{6.103}$$

with $\tau_{u,d}^{\mathrm{sat,G_3}}(k_d)$ computed by (6.77).

- For the idling behavior, we have

$$\begin{aligned}
n_{u,d}^{\mathrm{idle,G_1,sat}}(k_d) &= n_{u,d}^{\mathrm{G_1,sat}}(k_d),\\
n_{u,d}^{\mathrm{idle,G_2,sat}}(k_d) &= n_{u,d}^{\mathrm{G_2,sat}}(k_d),\\
n_{u,d}^{\mathrm{idle,G_3,sat}}(k_d) &= n_{u,d}^{\mathrm{G_3,sat}}(k_d),\\
n_{u,d}^{\mathrm{idle,G_4,sat}}(k_d) &= 0,
\end{aligned} \tag{6.104}$$

and

$$\begin{aligned}
T_{u,d}^{\mathrm{idle,G_1,sat}}(k_d) &= c_d - g_{u,d}(k_d) + \frac{0.5 n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{\mu_{u,d}},\\
T_{u,d}^{\mathrm{idle,G_2,sat}}(k_d) &= c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{\mu_{u,d}} - \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) + \\
&\qquad 0.5 n_{u,d}^{\mathrm{idle,G_2,sat}}(k_d)\left(\frac{1}{\mu_{u,d}} - \frac{1}{\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)}\right),\\
T_{u,d}^{\mathrm{idle,G_3,sat}}(k_d) &= 0.5\left(c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{\mu_{u,d}} - \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) + \right.\\
&\qquad \left. n_{u,d}^{\mathrm{idle,G_2,sat}}(k_d)\left(\frac{1}{\mu_{u,d}} - \frac{1}{\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)}\right)\right),\\
T_{u,d}^{\mathrm{idle,G_4,sat}}(k_d) &= 0.
\end{aligned} \tag{6.105}$$

- For the decelerating behavior, we have

$$\begin{aligned}
&n_{u,d}^{\text{dec,G}_1\text{,sat}}(k_d) = 0,\\
&n_{u,d}^{\text{dec,G}_2\text{,sat}}(k_d) = n_{u,d}^{\text{G}_2\text{,sat}}(k_d),\\
&n_{u,d}^{\text{dec,G}_3\text{,sat}}(k_d) = n_{u,d}^{\text{G}_3\text{,sat}}(k_d),\\
&n_{u,d}^{\text{dec,G}_4\text{,sat}}(k_d) = n_{u,d}^{\text{G}_4\text{,sat}}(k_d),
\end{aligned} \tag{6.106}$$

and

$$\begin{aligned}
&T_{u,d}^{\text{dec,G}_1\text{,sat}}(k_d) = 0,\\
&T_{u,d}^{\text{dec,G}_2\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}},\\
&T_{u,d}^{\text{acc,G}_3\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{idle}}(k_d) - v_{u,d}^{\text{free}}(k_d)}{a_{u,d}^{\text{dec}}} + \frac{v_{u,d}^{\text{idle}}(k_d) - v_{u,d}^{\text{middle,sat,2}}(k_d)}{a_{u,d}^{\text{dec}}},\\
&T_{u,d}^{\text{dec,G}_4\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{middle,sat,1}}(k_d) - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}} + \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{middle,sat,2}}(k_d)}{a_{u,d}^{\text{dec}}}.
\end{aligned} \tag{6.107}$$

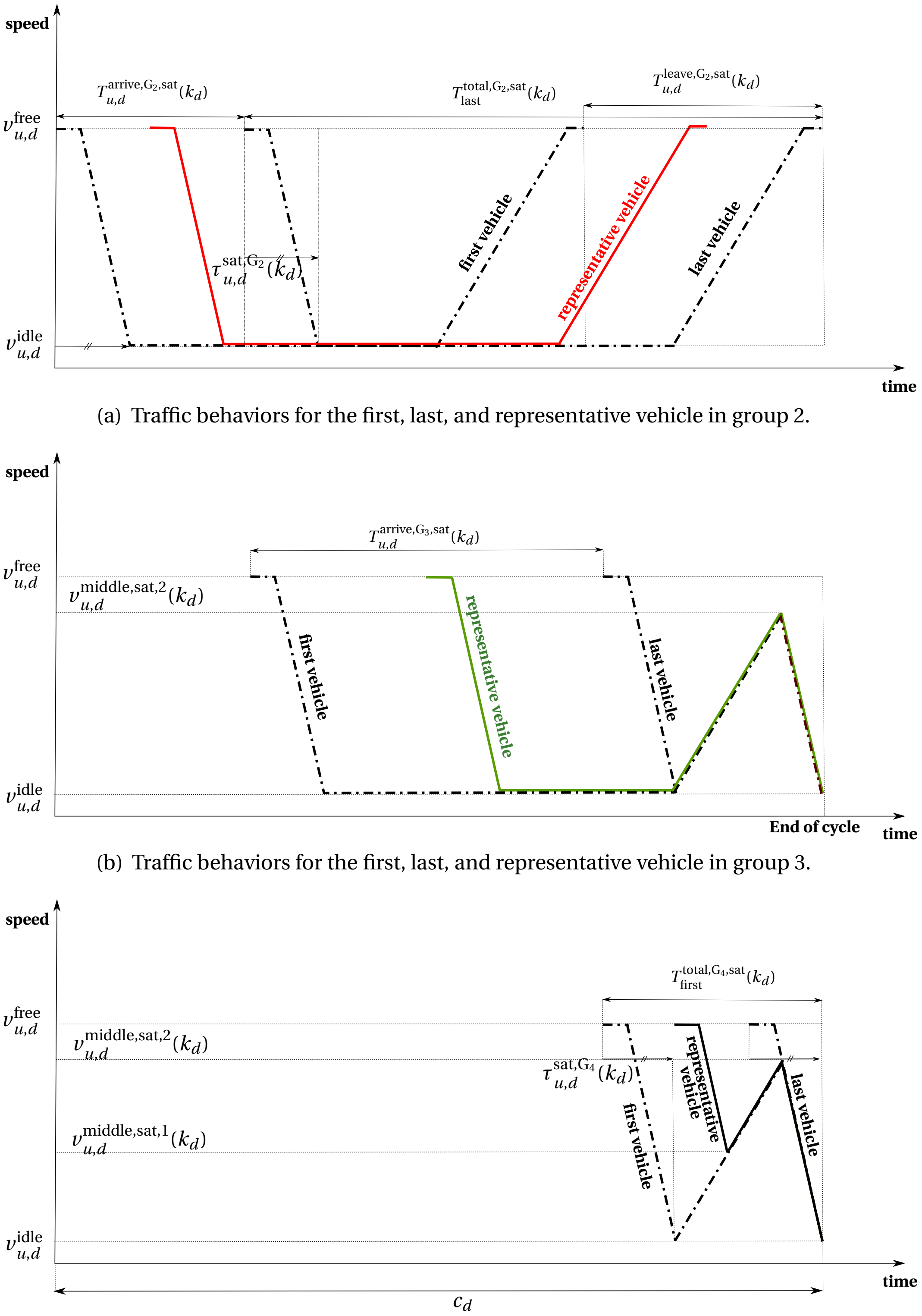
- For the accelerating behavior, we have

$$\begin{aligned}
&n_{u,d}^{\text{acc,G}_1\text{,sat}}(k_d) = n_{u,d}^{\text{G}_1\text{,sat}}(k_d),\\
&n_{u,d}^{\text{acc,G}_2\text{,sat}}(k_d) = n_{u,d}^{\text{G}_2\text{,sat}}(k_d),\\
&n_{u,d}^{\text{acc,G}_3\text{,sat}}(k_d) = n_{u,d}^{\text{G}_3\text{,sat}}(k_d),\\
&n_{u,d}^{\text{acc,G}_4\text{,sat}}(k_d) = n_{u,d}^{\text{G}_4\text{,sat}}(k_d),
\end{aligned} \tag{6.108}$$

and

$$\begin{aligned}
&T_{u,d}^{\text{acc,G}_1\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc,G}_2\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc,G}_3\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{idle}}(k_d) - v_{u,d}^{\text{middle,sat,2}}(k_d)}{a_{u,d}^{\text{acc}}},\\
&T_{u,d}^{\text{acc,G}_4\text{,sat}}(k_d) = \frac{v_{u,d}^{\text{middle,sat,2}}(k_d) - v_{u,d}^{\text{middle,sat,1}}(k_d)}{a_{u,d}^{\text{acc}}}.
\end{aligned} \tag{6.109}$$

Next we explain in more detail how the equations given in this section are derived considering the time-speed curves of each group of vehicles for case 2 of the saturated scenario. Figure 6.10 illustrates the time-speed curves of the first, last, and representative vehicle in groups 2, 3, and 4 for case 2 of the saturated urban traffic scenario (the time-speed curves of the vehicles in group 1 are not shown, since they are the same as for case 1). Since for case 2 we have $\alpha_{u,d}^{\text{arrive,q}}(k_d) \geq \mu_{u,d}$, the temporal distance between the starting points of the time-speed curves of every two successive vehicles in group 2 should be greater than or equal to


speed
$T_{u,d}^{\text{arrive},G_2,\text{sat}}(k_d)$
$T_{\text{last}}^{\text{total},G_2,\text{sat}}(k_d)$
$T_{u,d}^{\text{leave},G_2,\text{sat}}(k_d)$
$v_{u,d}^{\text{free}}$
$\tau_{u,d}^{\text{sat},G_2}(k_d)$
first vehicle
representative vehicle
last vehicle
$v_{u,d}^{\text{idle}}$
time

(a) Traffic behaviors for the first, last, and representative vehicle in group 2.

speed
$T_{u,d}^{\text{arrive},G_3,\text{sat}}(k_d)$
$v_{u,d}^{\text{free}}$
$v_{u,d}^{\text{middle},\text{sat},2}(k_d)$
first vehicle
representative vehicle
last vehicle
$v_{u,d}^{\text{idle}}$
End of cycle
time

(b) Traffic behaviors for the first, last, and representative vehicle in group 3.

speed
$T_{\text{first}}^{\text{total},G_4,\text{sat}}(k_d)$
$v_{u,d}^{\text{free}}$
$v_{u,d}^{\text{middle},\text{sat},2}(k_d)$
$\tau_{u,d}^{\text{sat},G_4}(k_d)$
representative vehicle
last vehicle
first vehicle
$v_{u,d}^{\text{middle},\text{sat},1}(k_d)$
$v_{u,d}^{\text{idle}}$
time
c_d

(c) Traffic behaviors for the first, last, and representative vehicle in group 4.

*Figure 6.10: Traffic behavior of different groups for the **saturated** urban traffic scenario, **Case 2**.*

the temporal distance between the endpoints of the curves. This implies that the idling time of a vehicle in group 2 compared with its predecessor increases.

For the first vehicle in group 2 (see Figure 6.10(a)), the time-speed curve includes the following behaviors for the given time durations:

- free-flow (start) + decelerating term:

$$\tau_{u,d}^{\mathrm{sat,G_2}}(k_d), \tag{6.110}$$

where $\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)$ is computed similar to (6.36).

- idling term (the idling time of the first vehicle in group 2 is in the limit equal to the idling time of the last vehicle in group 1 minus $\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)$):

$$c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{\mu_{u,d}} - \tau_{u,d}^{\mathrm{sat,G_2}}(k_d); \tag{6.111}$$

- accelerating term:

$$\frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}; \tag{6.112}$$

- free-flow (end)[9] term:

$$\frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} l^{\mathrm{veh}} - \frac{\left(v_{u,d}^{\mathrm{free}}\right)^2 - \left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}} v_{u,d}^{\mathrm{free}}}. \tag{6.113}$$

Similarly, for the last vehicle in group 2, we can see that the time duration of each behavior is the same as for the first vehicle, except for the idling and for the free-flow behavior at the end, where we have

- idling term:

$$c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\mathrm{G_1,sat}}(k_d)}{\mu_{u,d}} - \tau_{u,d}^{\mathrm{sat,G_2}}(k_d) + n_{u,d}^{\mathrm{G_2,sat}}(k_d)\left(\frac{1}{\mu_{u,d}} - \frac{1}{\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)}\right). \tag{6.114}$$

Note that the last term in (6.114) shows that for case 2, the idling time of the vehicles in group 2 increases gradually for each two successive vehicles. To obtain this term, from the fact that the leaving flow rate of the vehicles in a saturated urban traffic scenario is $\mu_{u,d}$, we obtain $\alpha_{u,d}^{\mathrm{arrive,q}}(k_d) = \dfrac{n_{u,d}^{\mathrm{G_2,sat}}(k_d)}{T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)}$ and $\mu_{u,d} = \dfrac{n_{u,d}^{\mathrm{G_2,sat}}(k_d)}{T_{u,d}^{\mathrm{leave,G_2,sat}}(k_d)}$, with $T_{u,d}^{\mathrm{leave,G_2,sat}}(k_d)$ the time that vehicles in group 2 need to leave the link (see Figure 6.10(a)). Moreover, the difference between $T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)$ and $T_{u,d}^{\mathrm{leave,G_2,sat}}(k_d)$ shows

[9] The number of vehicles in front of the first vehicle in group 2 is equal to $n_{u,d}^{\mathrm{G_1,sat}}(k_d)$.

the excess time that the last vehicle in group 2 spends idling within the queue w.r.t. the first vehicle. Therefore, the last term in (6.114) is obtained.

- free-flow (end) term:

$$\frac{\left(n_{u,d}^{\mathrm{G}_1,\mathrm{sat}}(k_d)+n_{u,d}^{\mathrm{G}_2,\mathrm{sat}}(k_d)\right)}{N_{u,d}^{\mathrm{lane}}v_{u,d}^{\mathrm{free}}}l^{\mathrm{veh}}-\frac{\left(v_{u,d}^{\mathrm{free}}\right)^2-\left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}}v_{u,d}^{\mathrm{free}}}. \tag{6.115}$$

Now we can also find an expression for $T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)$, $T_{u,d}^{\mathrm{arrive,G_3,sat}}(k_d)$, and $v_{u,d}^{\mathrm{middle,sat,2}}(k_d)$. From Figures 6.10(a) for $T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)$ we can write

$$T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)=c_d-T_{\mathrm{last}}^{\mathrm{total,G_2,sat}}(k_d). \tag{6.116}$$

Hence,

$$\begin{aligned}T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)=&c_d-\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)-c_d+g_{u,d}(k_d)-\\&\frac{n_{u,d}^{\mathrm{G_2,sat}}(k_d)}{\mu_{u,d}}+\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)-n_{u,d}^{\mathrm{G_2,sat}}(k_d)\left(\frac{1}{\mu_{u,d}}-\frac{1}{\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)}\right)-\\&\frac{v_{u,d}^{\mathrm{free}}-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}-\frac{\left(n_{u,d}^{\mathrm{G_1,sat}}(k_d)+n_{u,d}^{\mathrm{G_2,sat}}(k_d)\right)}{N_{u,d}^{\mathrm{lane}}v_{u,d}^{\mathrm{free}}}l^{\mathrm{veh}}+\frac{\left(v_{u,d}^{\mathrm{free}}\right)^2-\left(v_{u,d}^{\mathrm{idle}}\right)^2}{2a_{u,d}^{\mathrm{acc}}v_{u,d}^{\mathrm{free}}},\end{aligned} \tag{6.117}$$

with

$$n_{u,d}^{\mathrm{G_2,sat}}(k_d)=\alpha_{u,d}^{\mathrm{arrive,q}}(k_d)\cdot T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d). \tag{6.118}$$

From (6.116), (6.117), and (6.118), (6.100) is found.

From Figures 6.10(a), 6.10(b), and 6.10(c), we have

$$T_{u,d}^{\mathrm{arrive,G_2,sat}}(k_d)+T_{u,d}^{\mathrm{arrive,G_3,sat}}(k_d)+T_{u,d}^{\mathrm{first,G_4,sat}}(k_d)=c_d. \tag{6.119}$$

Moreover, the first vehicle of group 4 should travel a distance of $\dfrac{C_{u,d}\cdot l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}}$ in total during the current cycle. Hence, from Lemma 6.2 we can write

$$\begin{aligned}&\left(n_{u,d}^{\mathrm{G_1,sat}}(k_d)+n_{u,d}^{\mathrm{G_2,sat}}(k_d)+n_{u,d}^{\mathrm{G_3,sat}}(k_d)\right)\frac{l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}}}=\\&\frac{1}{2}a_{u,d}^{\mathrm{acc}}\left(\frac{v_{u,d}^{\mathrm{middle,sat,2}}(k_d)-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right)^2+v_{u,d}^{\mathrm{idle}}\left(\frac{v_{u,d}^{\mathrm{middle,sat,2}}(k_d)-v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right)+\\&\frac{1}{2}a_{u,d}^{\mathrm{dec}}\left(\frac{v_{u,d}^{\mathrm{idle}}-v_{u,d}^{\mathrm{middle,sat,2}}(k_d)}{a_{u,d}^{\mathrm{dec}}}\right)^2+v_{u,d}^{\mathrm{middle,sat,2}}(k_d)\left(\frac{v_{u,d}^{\mathrm{idle}}-v_{u,d}^{\mathrm{middle,sat,2}}(k_d)}{a_{u,d}^{\mathrm{dec}}}\right).\end{aligned} \tag{6.120}$$

Substituting $n_{u,d}^{\mathrm{G_2,sat}}(k_d)$ by (6.118) and $n_{u,d}^{\mathrm{G_3,sat}}(k_d)$ by

$$n_{u,d}^{\mathrm{G_3,sat}}(k_d) = \alpha_{u,d}^{\mathrm{arrive,q}}(k_d) \cdot T_{u,d}^{\mathrm{arrive,G_3,sat}}(k_d), \tag{6.121}$$

we obtain (6.101).

Since for the first vehicle in group 4, we have

$$T_{u,d}^{\mathrm{first,G_4,sat}}(k_d) = \tau_{u,d}^{\mathrm{sat,G_4}}(k_d) + \left(v_{u,d}^{\mathrm{middle,sat,2}}(k_d) - v_{u,d}^{\mathrm{idle}}\right)\left(\frac{1}{a_{u,d}^{\mathrm{acc}}} - \frac{1}{a_{u,d}^{\mathrm{dec}}}\right), \tag{6.122}$$

with $\tau_{u,d}^{\mathrm{sat,G_4}}(k_d)$ found similar to $\tau_{u,d}^{\mathrm{sat,G_2}}(k_d)$ by

$$\tau_{u,d}^{\mathrm{sat,G_4}}(k_d) = \frac{\left(C_{u,d} - n_{u,d}^{\mathrm{G_1,sat}}(k_d) - n_{u,d}^{\mathrm{G_2,sat}}(k_d) - n_{u,d}^{\mathrm{G_3,sat}}(k_d)\right) l^{\mathrm{veh}}}{N_{u,d}^{\mathrm{lane}} v_{u,d}^{\mathrm{free}}} - \frac{\left(v_{u,d}^{\mathrm{idle}} - v_{u,d}^{\mathrm{free}}\right)^2}{2a_{u,d}^{\mathrm{dec}} v_{u,d}^{\mathrm{free}}}. \tag{6.123}$$

From (6.119)-(6.123), (6.94) is obtained.

6.6 Flow-emission model for the over-saturated scenario

For the over-saturated urban traffic scenario, not all the vehicles that are initially in the queue can leave the link within one cycle. Mathematically, this is expressed as follows:

$$\sum_{o \in \mathcal{O}_{u,d}} \beta_{u,d,o} \cdot \mu_{u,d} \cdot g_{u,d,o}(k_d) < q_{u,d}(k_d). \tag{6.124}$$

Therefore, for this scenario we divide the vehicles that are observed in a link in the following three groups:

Group 1: composed of those vehicles in the queue that can leave the link during the current cycle (see the blue curve in Figure 6.11 for the average behavior of this group).

Group 2: composed of those vehicles in the queue that do not have enough time during the current cycle to pass through the traffic light (see the dark blue curve in Figure 6.11 for the average behavior of this group). These vehicles first accelerate as soon as the last vehicle in the queue in front of them composed of vehicles of group 1 starts to move forward. However, since there is not enough time left for them to leave the link, after they reach $v_{u,d}^{\mathrm{middle,osat}}(k_d)$ they decelerate to $v_{u,d}^{\mathrm{idle}}$ and stop in front of the red light for at least another cycle time. We set

$$v_{u,d}^{\mathrm{middle,osat}}(k_d) = v_{u,d}^{\mathrm{idle}} + T_{u,d}^{\mathrm{leave,osat,G_1}}(k_d)\frac{a_{u,d}^{\mathrm{dec}} - a_{u,d}^{\mathrm{acc}}}{2a_{u,d}^{\mathrm{dec}} a_{u,d}^{\mathrm{acc}}}, \tag{6.125}$$

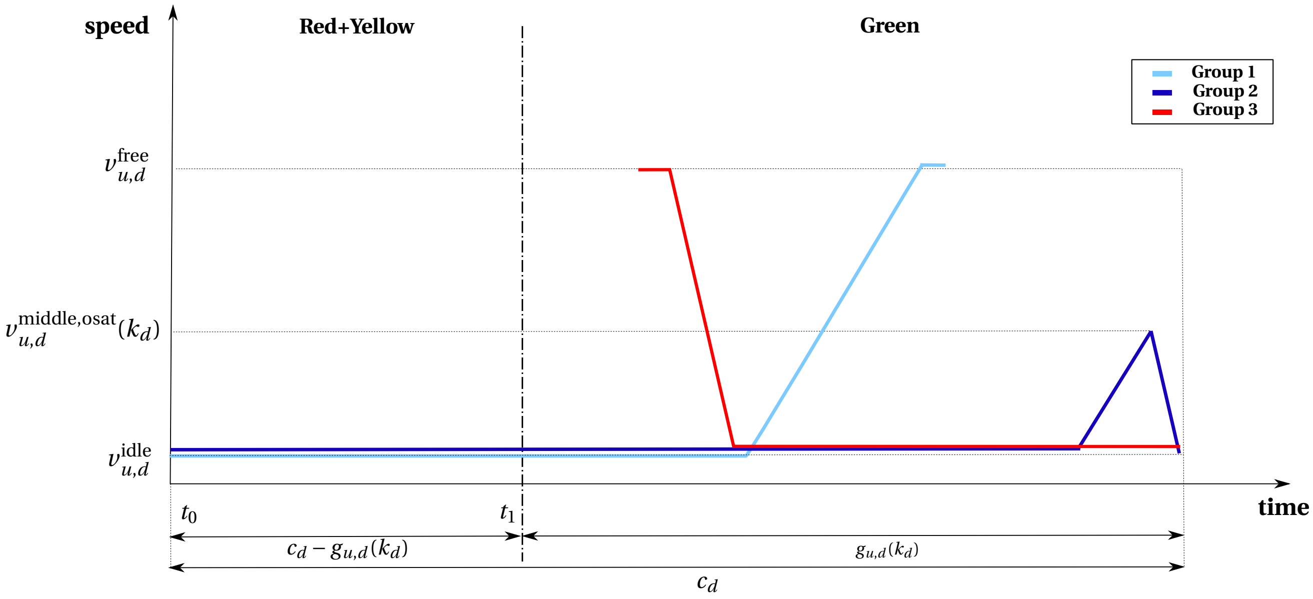


Figure 6.11: Traffic behaviors on link (u,d) within one cycle for the ***over-saturated*** *urban traffic scenario.*

with

$$T_{u,d}^{\text{leave,osat,G}_1}(k_d) = \frac{\mu_{u,d}\, l^{\text{veh}} g_{u,d}(k_d)}{2v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}} + \frac{\left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right)^2}{a_{u,d}^{\text{acc}}\left(2v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}\right)}. \tag{6.126}$$

Group 3: composed of the vehicles that arrive at the tail of the queue during the current cycle. These vehicles first decelerate and then idle for the rest of the cycle (see the red curve in Figure 6.11 for the average behavior of this group).

The leaving flow rate of the vehicles for the over-saturated scenario is the saturated leaving flow rate, $\mu_{u,d}$. Hence, for the number of vehicles in each of these three groups, we can write

$$n_{u,d}^{\text{G}_1\text{,osat}}(k_d) = \mu_{u,d}\left(g_{u,d}(k_d) - T_{u,d}^{\text{leave,osat,G}_1}(k_d)\right), \tag{6.127}$$

$$n_{u,d}^{\text{G}_2\text{,osat}}(k_d) = q_{u,d}(k_d) - n_{u,d}^{\text{G}_1\text{,osat}}(k_d), \tag{6.128}$$

$$n_{u,d}^{\text{G}_3\text{,osat}}(k_d) = n_{u,d}^{\text{osat}}(k_d) - \sum_{i=1}^{2} n_{u,d}^{\text{G}_i\text{,osat}}(k_d). \tag{6.129}$$

Next, we determine the number of vehicles in each group that show a specific behavior, and also the average time that the representative vehicle shows the behavior.

- For the free-flow behavior, we have

$$\begin{aligned} n_{u,d}^{\text{free,G}_1\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_1\text{,osat}}(k_d), \\ n_{u,d}^{\text{free,G}_2\text{,osat}}(k_d) &= 0, \\ n_{u,d}^{\text{free,G}_3\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_3\text{,osat}}(k_d), \end{aligned} \tag{6.130}$$

and similarly to the under-saturated scenario,

$$\begin{aligned}
T_{u,d}^{\text{free,G}_1\text{,osat}}(k_d) &= \frac{\mu_{u,d} l^{\text{veh}} g_{u,d}(k_d)}{2v_{u,d}^{\text{free}} + \mu_{u,d} l^{\text{veh}}} - \frac{\left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right)\left(\mu_{u,d} l^{\text{veh}} + v_{u,d}^{\text{free}} + v_{u,d}^{\text{idle}}\right)}{a_{u,d}^{\text{acc}}\left(2v_{u,d}^{\text{free}} + \mu_{u,d} l^{\text{veh}}\right)},\\
T_{u,d}^{\text{free,G}_2\text{,osat}}(k_d) &= 0,\\
T_{u,d}^{\text{free,G}_3\text{,osat}}(k_d) &= \tau_{u,d}^{\text{osat,G}_3}(k_d) - \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}},
\end{aligned} \tag{6.131}$$

with

$$\tau_{u,d}^{\text{osat,G}_3}(k_d) = \frac{\left(C_{u,d} - n_{u,d}^{\text{G}_1\text{,osat}}(k_d) - n_{u,d}^{\text{G}_2\text{,osat}}(k_d)\right) l^{\text{veh}}}{N_{u,d}^{\text{lane}} v_{u,d}^{\text{free}}}. \tag{6.132}$$

- For the idling behavior, we have

$$\begin{aligned}
n_{u,d}^{\text{idle,G}_1\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_1\text{,osat}}(k_d),\\
n_{u,d}^{\text{idle,G}_2\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_2\text{,osat}}(k_d),\\
n_{u,d}^{\text{idle,G}_3\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_3\text{,osat}}(k_d),
\end{aligned} \tag{6.133}$$

and

$$\begin{aligned}
T_{u,d}^{\text{idle,G}_1\text{,osat}}(k_d) &= c_d - g_{u,d}(k_d) + \frac{n_{u,d}^{\text{G}_1\text{,osat}}(k_d)}{2\mu_{u,d}},\\
T_{u,d}^{\text{idle,G}_2\text{,osat}}(k_d) &= c_d - \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}},\\
T_{u,d}^{\text{idle,G}_3\text{,osat}}(k_d) &= \frac{c_d - \tau_{u,d}^{\text{osat,G}_3}(k_d)}{2}.
\end{aligned} \tag{6.134}$$

- For the decelerating behavior, we have

$$\begin{aligned}
n_{u,d}^{\text{dec,G}_1\text{,osat}}(k_d) &= 0,\\
n_{u,d}^{\text{dec,G}_2\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_2\text{,osat}}(k_d),\\
n_{u,d}^{\text{dec,G}_3\text{,osat}}(k_d) &= n_{u,d}^{\text{G}_3\text{,osat}}(k_d),
\end{aligned} \tag{6.135}$$

and

$$\begin{aligned}
T_{u,d}^{\text{dec,G}_1\text{,osat}}(k_d) &= 0,\\
T_{u,d}^{\text{dec,G}_2\text{,osat}}(k_d) &= \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{middle,osat}}(k_d)}{a_{u,d}^{\text{dec}}},\\
T_{u,d}^{\text{dec,G}_3\text{,osat}}(k_d) &= \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}.
\end{aligned} \tag{6.136}$$

- For the accelerating behavior, we have

$$
\begin{aligned}
n_{u,d}^{\mathrm{acc,G_1,osat}}(k_d) &= n_{u,d}^{\mathrm{G_2,osat}}(k_d),\\
n_{u,d}^{\mathrm{acc,G_2,osat}}(k_d) &= n_{u,d}^{\mathrm{G_3,osat}}(k_d),\\
n_{u,d}^{\mathrm{acc,G_3,osat}}(k_d) &= 0,
\end{aligned}
\tag{6.137}
$$

and

$$
\begin{aligned}
T_{u,d}^{\mathrm{acc,G_1,osat}}(k_d) &= \frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}},\\
T_{u,d}^{\mathrm{acc,G_2,osat}}(k_d) &= \frac{v_{u,d}^{\mathrm{middle,osat}}(k_d) - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}},\\
T_{u,d}^{\mathrm{acc,G_3,osat}}(k_d) &= 0.
\end{aligned}
\tag{6.138}
$$

Now we explain in more detail how the main equations given in this section are obtained. Figure 6.12 illustrates the time-speed curves of the first, last, and representative vehicle of groups 2 and 3 for the over-saturated urban scenario. The dashed line on the left-hand side in Figure 6.12(a) shows the time-speed curve of the first vehicle in group 1, and the dashed line on the right-hand side in this figure shows the time-speed curve of the last vehicle in group 1 that can still accelerate and leave the link. However, the next vehicle in the queue (i.e., the first vehicle in group 2) is the first vehicle in the queue that should necessarily decelerate after its speed reaches $v_{\mathrm{first},u,d}^{\mathrm{middle,osat}}(k_d)$, so that it can stop in front of the red light at the end of the cycle.

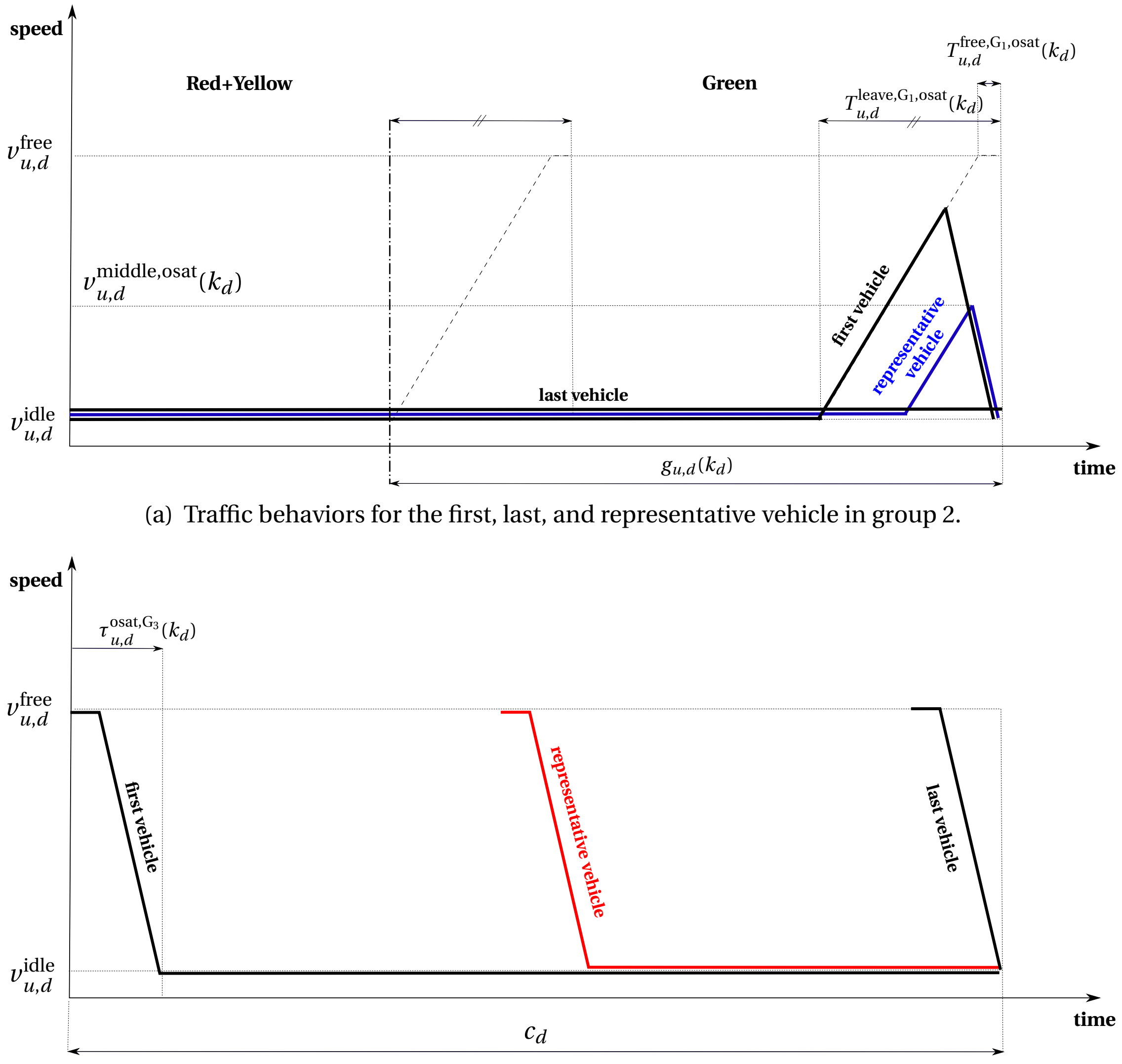
Next, we compute $T_{u,d}^{\mathrm{leave,osat,G_1}}(k_d)$ (i.e., the time that each vehicle in group 1 needs after idling to leave the link), $T_{u,d}^{\mathrm{free,G_1,osat}}(k_d)$ (i.e., the time duration of the free-flow behavior for each vehicle in group 1), $v_{\mathrm{first},u,d}^{\mathrm{middle,osat}}(k_d)$ corresponding to the first vehicle of group 2, and $v_{u,d}^{\mathrm{middle,osat}}(k_d)$ corresponding to the representative vehicle of group 2. For the sake of simplicity, we assume that the traveled distance to the end of the link for all vehicles in group 1 is the same and is equal to $0.5n_{u,d}^{\mathrm{G_1,osat}}(k_d)l^{\mathrm{veh}}$ (i.e., we assume to have a vertical queue composed of vehicles in group 1).

For the last vehicle in group 1 that will show an accelerating and (possibly a free-flow behavior at the end), using Lemma 6.2 we obtain

$$
\frac{n_{u,d}^{\mathrm{G_1,osat}}(k_d)\,l^{\mathrm{veh}}}{2} = \frac{1}{2}a_{u,d}^{\mathrm{acc}}\left(\frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right)^2 + v_{u,d}^{\mathrm{idle}}\left(\frac{v_{u,d}^{\mathrm{free}} - v_{u,d}^{\mathrm{idle}}}{a_{u,d}^{\mathrm{acc}}}\right) + v_{u,d}^{\mathrm{free}}\cdot T_{u,d}^{\mathrm{free,G_1,osat}}(k_d). \tag{6.139}
$$

Moreover, since the leaving flow rate of group 1 in the over-saturated scenario is $\mu_{u,d}$, we can write

$$
n_{u,d}^{\mathrm{G_1,osat}}(k_d) = \mu_{u,d}\left(g_{u,d}(k_d) - T_{u,d}^{\mathrm{leave,G_1,osat}}(k_d)\right), \tag{6.140}
$$



(a) Traffic behaviors for the first, last, and representative vehicle in group 2.

(b) Traffic behaviors for the first, last, and representative vehicle in group 3.

Figure 6.12: Traffic behavior of different groups for the ***over-saturated*** *urban traffic scenario.*

where $g_{u,d}(k_d) - T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d)$ denotes the temporal distance between the two dashed curves in Figure 6.12(a). Additionally, from this figure we have

$$T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d) = \frac{v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}} + T_{u,d}^{\text{free},\text{G}_1,\text{osat}}(k_d), \tag{6.141}$$

where from (6.139), (6.140), and (6.141) we obtain

$$T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d) = \frac{\mu_{u,d}\, l^{\text{veh}}\, g_{u,d}(k_d)}{2 v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}} + \frac{\left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right)^2}{a_{u,d}^{\text{acc}}\left(2 v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}\right)}, \tag{6.142}$$

and

$$T_{u,d}^{\text{free},\text{G}_1,\text{osat}}(k_d) = \frac{\mu_{u,d}\, l^{\text{veh}} g_{u,d}(k_d)}{2 v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}} - \frac{\left(v_{u,d}^{\text{free}} - v_{u,d}^{\text{idle}}\right)\left(\mu_{u,d}\, l^{\text{veh}} + v_{u,d}^{\text{free}} + v_{u,d}^{\text{idle}}\right)}{a_{u,d}^{\text{acc}}\left(2 v_{u,d}^{\text{free}} + \mu_{u,d}\, l^{\text{veh}}\right)}. \tag{6.143}$$

For $v_{u,d}^{\text{middle,osat}}(k_d)$, from the time-speed curve of the first vehicle (see Figure 6.12(a)), we can write

$$T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d) = \frac{v_{\text{first},u,d}^{\text{middle,osat}}(k_d) - v_{u,d}^{\text{idle}}}{a_{u,d}^{\text{acc}}} + \frac{v_{u,d}^{\text{idle}} - v_{\text{first},u,d}^{\text{middle,osat}}(k_d)}{a_{u,d}^{\text{dec}}}, \tag{6.144}$$

which gives

$$v_{\text{first},u,d}^{\text{middle,osat}}(k_d) = v_{u,d}^{\text{idle}} + T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d) \frac{a_{u,d}^{\text{dec}} - a_{u,d}^{\text{acc}}}{a_{u,d}^{\text{dec}} a_{u,d}^{\text{acc}}}, \tag{6.145}$$

and since $v_{u,d}^{\text{middle,osat}}(k_d)$ is the mean value of $v_{\text{first},u,d}^{\text{middle,osat}}(k_d)$ and $v_{u,d}^{\text{idle}}$, we obtain

$$v_{u,d}^{\text{middle,osat}}(k_d) = v_{u,d}^{\text{idle}} + T_{u,d}^{\text{leave},\text{G}_1,\text{osat}}(k_d) \frac{a_{u,d}^{\text{dec}} - a_{u,d}^{\text{acc}}}{2 a_{u,d}^{\text{dec}} a_{u,d}^{\text{acc}}}. \tag{6.146}$$

Figure 6.12(b) shows the time-speed curves of the first, last, and representative vehicle of group 3 in the over-saturated scenario. From this figure the free-flow, decelerating, and idling time durations of the representative vehicle are obtained as follows:

- free-flow time duration:

$$\tau_{u,d}^{\text{osat},\text{G}_3}(k_d) - \frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}; \tag{6.147}$$

- decelerating time duration:

$$\frac{v_{u,d}^{\text{idle}} - v_{u,d}^{\text{free}}}{a_{u,d}^{\text{dec}}}; \tag{6.148}$$

- idling time duration:

$$\tau_{u,d}^{\text{osat},\text{G}_3}(k_d) - \frac{c_d - \tau_{u,d}^{\text{osat},\text{G}_3}(k_d)}{2}, \tag{6.149}$$

where $\tau_{u,d}^{\text{osat},\text{G}_3}(k_d)$ is given by (6.132).

6.7 Case study

In this section, we present some results of a simulation-based case study to evaluate the performance of the proposed mesoscopic integrated flow-emission model. As an evaluation platform, we use the traffic microsimulator SUMO (Simulation of Urban MObility) [85, 86], which

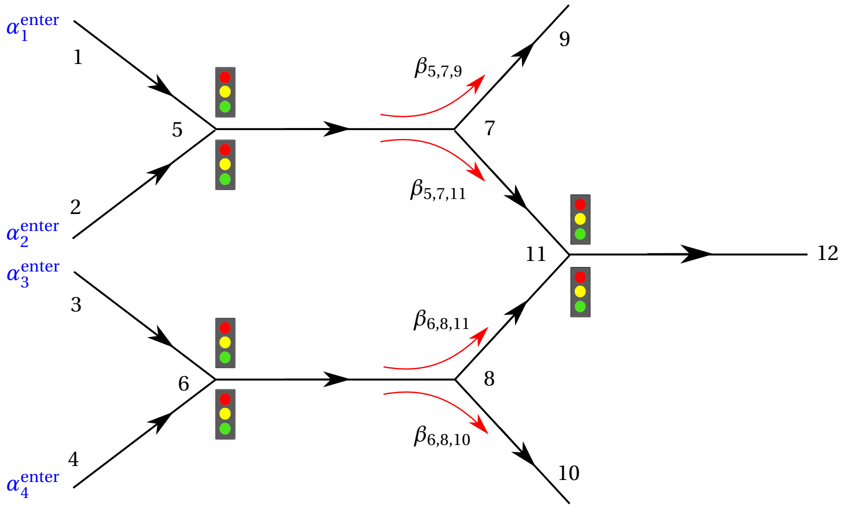

α_1^{enter}
1
α_2^{enter}
2
α_3^{enter}
3
α_4^{enter}
4
5
6
$\beta_{5,7,9}$
$\beta_{5,7,11}$
$\beta_{6,8,11}$
$\beta_{6,8,10}$
7
8
9
10
11
12

Figure 6.13: Urban traffic network used for the case study.

is an open-source microscopic traffic flow simulator (continuous-space and discrete-time) developed by the Institute for Transportation Research, German Aerospace Center (Deutsches Zentrum für Luft- und Raumfahrt, DLR). Since its initial release in 2002, SUMO has evolved into a reliable and versatile platform that provides full-featured simulation and analysis tools for traffic networks (see [86] for more details). Moreover, a remote traffic control interface, called "TraCI" [154], is available, which provides an interface between SUMO and different programming languages including MATLAB for various online analysis and control scenarios.

The focus of this chapter and the case study presented in this section is on the interface of the macroscopic flow models and microscopic emission models. For this case study, in order to assess the accuracy of the interface, we use SUMO and VT-micro to generate both the microscopic and the mesoscopic data. More specifically, for evaluation of the mesoscopic integrated flow-emission model developed via the integrating and interfacing approach proposed in this chapter, the urban traffic network represented in Figure 6.13 is considered. This traffic network includes 12 nodes, 11 single-lane links of length 500 m, four entrances (i.e., links $(1,5)$, $(2,5)$, $(3,6)$, and $(4,6)$), three exits (i.e., links $(7,9)$, $(8,10)$, and $(11,12)$), and three traffic lights located at nodes 3, 8, and 11. The entering flows to the network are denoted by α_1^{enter}, α_2^{enter}, α_3^{enter}, and α_4^{enter}, and the turning rates (i.e., the percentage of vehicles on a link that intend to turn to a specific downstream link) $\beta_{5,7,9}$, $\beta_{5,7,11}$, $\beta_{6,8,10}$, and $\beta_{6,8,11}$ are assumed to be 0.4, 0.6, 0.4, and 0.6, respectively. Note that $\beta_{5,7,9}$ and $\beta_{5,7,11}$ are the turning rates for the vehicles on link $(5,7)$ to, respectively, link 9 and link 11, and that $\beta_{6,8,10}$ and $\beta_{6,8,11}$ are the turning rates for the vehicles on link $(6,8)$ to, respectively, link 10 and link 11 (see Figure 6.13).

The traffic network shown in Figure 6.13 was modeled in SUMO, where the measurements obtained from SUMO were coupled with the microscopic emission model VT-micro explained (see Chapter 5 for details about VT-micro). VT-micro is selected here because it can provide the instantaneous emissions of CO, HC, and NO_x based on both the speed and the acceleration of individual vehicles in the traffic network. Before we start to implement the proposed model, we should identify the parameters of the model by solving an offline identification optimization problem. For identifying the parameters and for computation of the emissions by the proposed mesoscopic integrated flow-emission model, VT-micro was used as the microscopic emission model. An extensive dataset was collected by running various traffic scenarios in SUMO. The demand profiles, the cycle time and the green times of the traffic lights, and the turning rates of the links are the inputs of the identification procedure. The data that should be collected includes the total number of vehicles on each link within the network and the number of vehicles waiting in the queue on each link. The total emissions of CO, HC, and $\mathrm{NO_x}$ computed by VT-micro based on the flow data captured by SUMO are the outputs of the identification procedure. Moreover, the set of parameters that should be identified include the free-flow and the idling speed, the acceleration and deceleration, the link capacities, the saturated leaving flow rates (i.e., the maximum possible discharge rates) of the links, and the average vehicle length in the network. Then an optimization problem was formulated that minimizes the relative error of the produced emissions via the proposed model w.r.t. the computed emissions by SUMO and VT-micro, with the parameter vector considered as the optimization variable. We used the "lsqnonlin" solver with the "trust-region-reflective" algorithm of the MATLAB optimization toolbox to find the parameters. The parameter values found via model identification for the emissions of CO, HC, and $\mathrm{NO_x}$ are given [10] in Tables 6.2, 6.3, and 6.4.

Remark 6.2 Note that the identified values of the traffic parameters may differ from their real physical values as a result of the assumptions and simplifications made for developing the mathematical model. In particular, in addition to the dynamic variables such as the speeds and the acceleration and deceleration, we should also identify other parameters such as the link capacities to minimize the mismatch between the two mesoscopic and microscopic models. □

To evaluate the performance of the proposed model, new traffic scenarios that are different from those used for model identification were simulated in SUMO. The total number of vehicles in each link and the number of vehicles standing in the queue in each link were extracted from SUMO. Then for each link, the average value of the total number of vehicles and the average value of the queue length were computed per model time step (i.e., per cycle time of the traffic lights, which is 60 s in our simulations).

Figures 6.14 and 6.15 show the demand profiles (i.e., α_i^{enter}, $i = 1,2,3,4$) for two different traffic scenarios, where the first scenario corresponds to a low demand and the second scenario corresponds to a high demand. Note that for this scenario the green times of the three traffic lights of the urban traffic network are assumed to be 30 s. The instantaneous speed and the instantaneous acceleration profiles of the individual vehicles within the network were extracted from SUMO for the two given cases, and the emissions of CO, HC, and

[10] To identify the parameters of the model for the case study, we have considered the same acceleration and deceleration rates for the under-saturated, saturated, and over-saturated traffic scenarios (in order to reduce the number of parameters that should be determined via the identification optimization problem). However, since this optimization problem is solved offline, one could consider different acceleration and deceleration rates, which may possibly result in more accurate results.

Table 6.2: Identified traffic parameters for the emissions of CO using SUMO.

v^{free} [m/s]	v^{idle} [m/s]	a^{acc} [m/s^2]	a^{dec} [m/s^2]	C [veh]	μ [veh/s]	l^{veh} [m]
11.8	0.1	1.60	-1.80	14.3	0.9	5.4

Table 6.3: Identified traffic parameters for the emissions of HC using SUMO.

v^{free} [m/s]	v^{idle} [m/s]	a^{acc} [m/s^2]	a^{dec} [m/s^2]	C [veh]	μ [veh/s]	l^{veh} [m]
16.9	0.1	1.99	-1.25	71.6	4.9	3.5

Table 6.4: Identified traffic parameters for the emissions of NO_x *using SUMO.*

v^{free} [m/s]	v^{idle} [m/s]	a^{acc} [m/s^2]	a^{dec} [m/s^2]	C [veh]	μ [veh/s]	l^{veh} [m]
13.8	0.1	1.36	-1.05	62.7	3.8	5.7

NO_x produced by each vehicle were computed using VT-micro. Summing up these individual values, the total emissions of CO, HC, and NO_x by the vehicles for the entire simulated period were obtained. These resulting values were divided into two set to provide a training dataset and a validation dataset. The training dataset has been used to train the parameters of the model, and the validation dataset has been used for evaluation of the emission values that are computed via the proposed mesoscopic flow-emission model. The identified values of the model parameters are given in Tables 6.2, 6.3, and 6.4 for, respectively, the emissions of CO, the emissions of HC, and the emissions of NO_x.

Figures 6.17(a), 6.18(a), and 6.19(a) (which correspond to case 1 with low demands) show the relative error of the instantaneous emissions computed by the proposed integrated flow-emission model w.r.t. the instantaneous emissions captured from SUMO combined with VT-micro. The relative error is given in percentage and as a function of time step for a one-hour simulation. The relative error for the pollutant $e \in \{\text{CO}, \text{HC}, \text{NO}_\text{x}\}$ is defined by

$$\epsilon_e^{\text{rel}} = \frac{\left|\text{TE}_e^{\text{model}} - \text{TE}_e^{\text{microsimulator}}\right|}{\text{TE}_e^{\text{microsimulator}}}. \tag{6.150}$$

Table 6.5 summarizes the results of the simulations for the instantaneous emissions. From this table, the relative error of the instantaneous emissions of CO has a mean value of 0.152%, a standard deviation of 0.149, and a maximum of 0.593%. The mean value, the standard deviation, and the maximum value of the instantaneous emissions of HC are, respectively, 0.226%, 0.189, and 0.884%. For the instantaneous emissions of NO_x, the mean value of the relative error is 0.226%, the standard deviation of the relative error is 0.223, and its maximum value is 1.13%.

Figures 6.17(b), 6.18(b), and 6.19(b) show the relative error in percentage of the cumulative total emissions of CO, HC, and NO_x (which is the sum of the instantaneous emissions from the first time step till the current time step) for case 1 w.r.t. the emissions computed via SUMO and VT-micro. From these figures, Table 6.6 was extracted, which shows that the relative error of the cumulative emissions of CO during a 1-hour simulation does not exceed 2%, where this relative error has a mean value and a standard deviation of, respectively, 0.483%

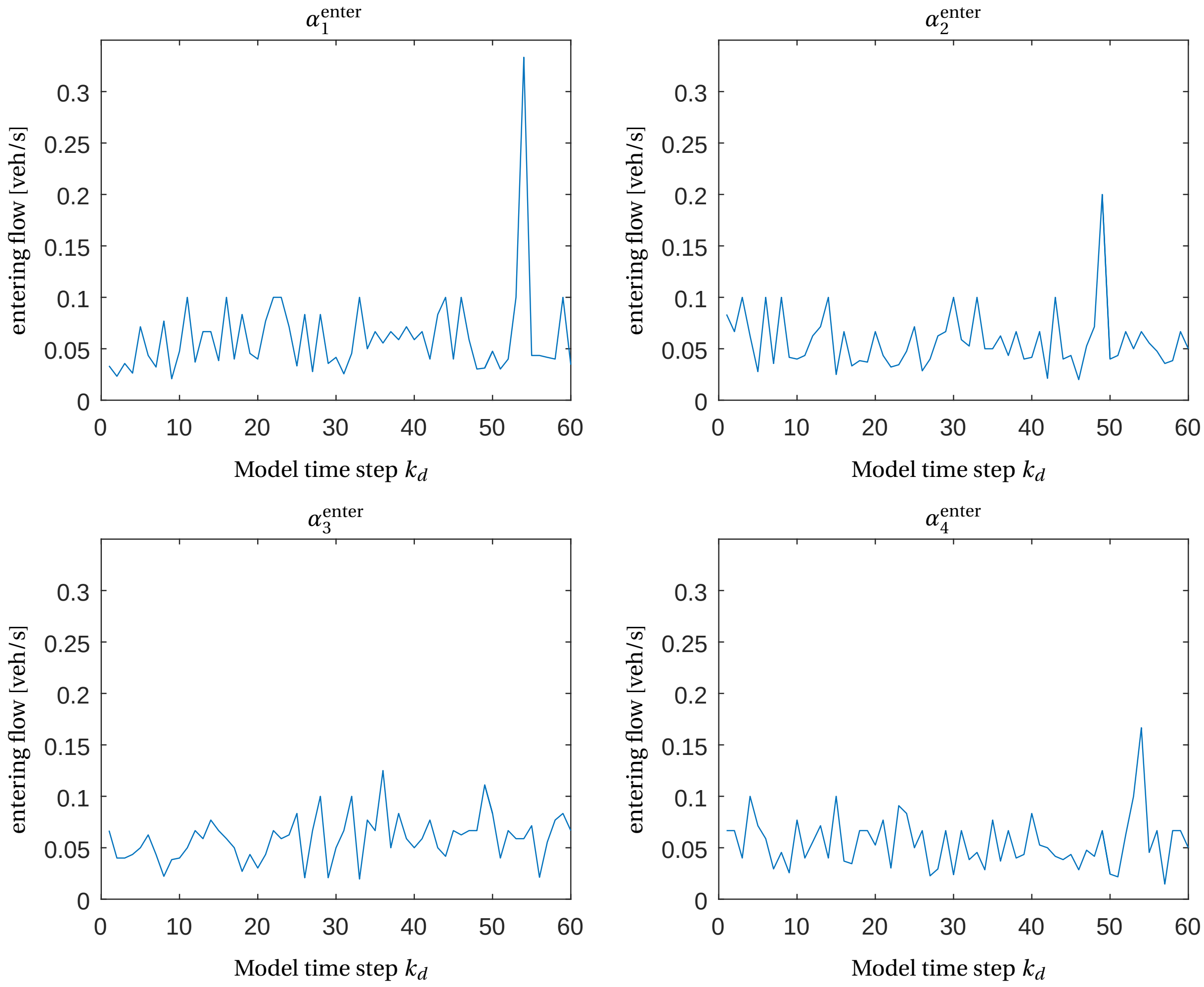


Figure 6.14: Demand profiles for the case study for case 1 (low demand) for a 1-hour simulation (the green times are 30 s).

and 0.368. For the cumulative total emissions of HC, the mean value and the standard deviation of the relative error are, respectively, 0.848% and 0.617, while the relative error is always less than 2.5%. Finally, from Figure 6.19(b), for the cumulative total emissions of NO_x, the mean value is 0.887%, the standard deviation is 0.617, and the relative error does not go higher than 2.5%. These results prove the excellent accuracy provided by the mesoscopic integrated flow-emission model, which is based on the integrating and interfacing approach proposed in this chapter.

Moreover, the results corresponding to case 2 are shown in Figures 6.20(a), 6.21(a), and 6.22(a), which show the relative error in percentage of the instantaneous emissions of CO, HC, and NO_x as a function of the time step, and also in Figures 6.20(b), 6.21(b), and 6.22(b), which show the relative error in percentage of the cumulative total emissions of CO, HC, and NO_x. These results are summarized in Tables 6.7 and 6.8, where we can see that the relative error

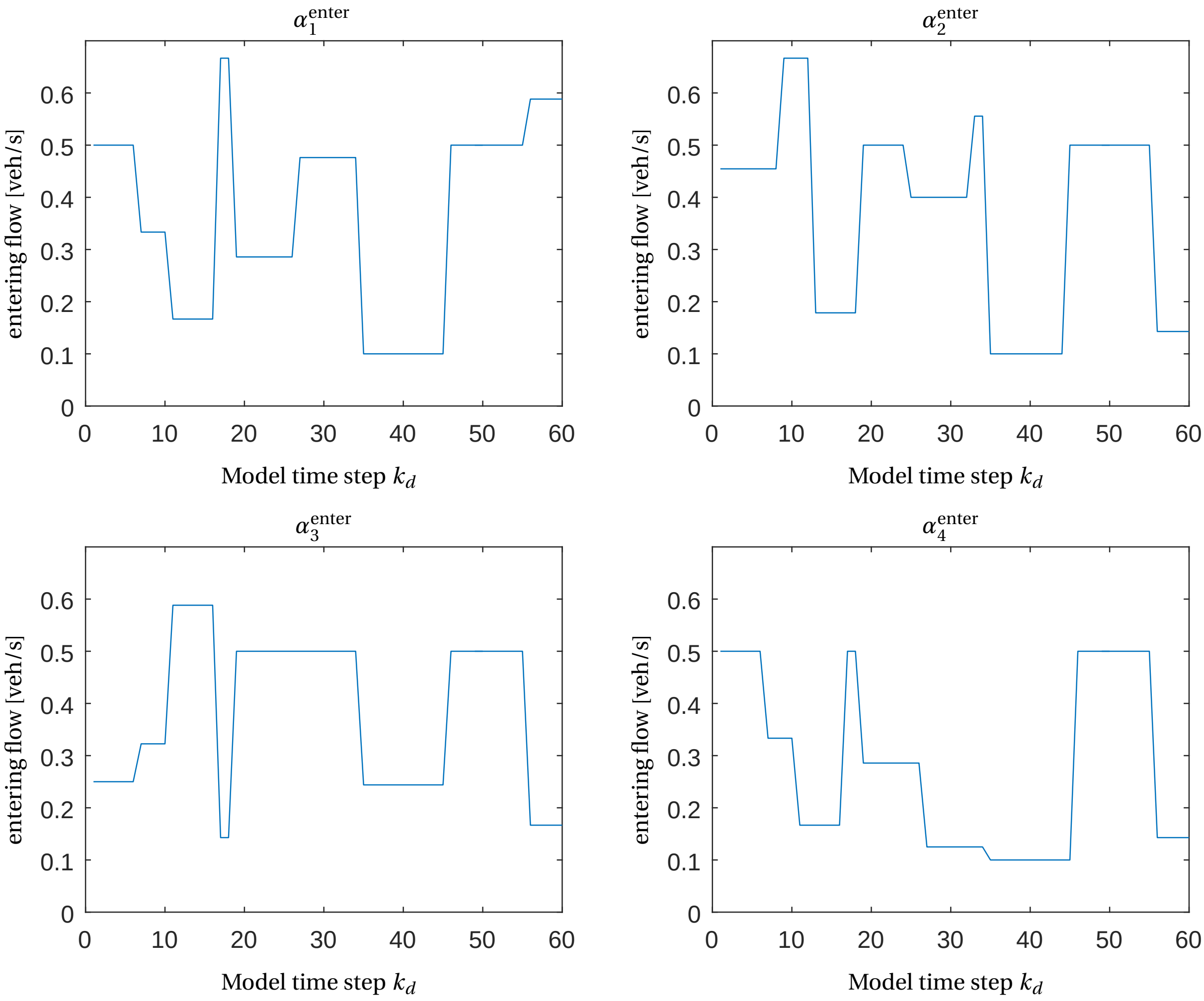


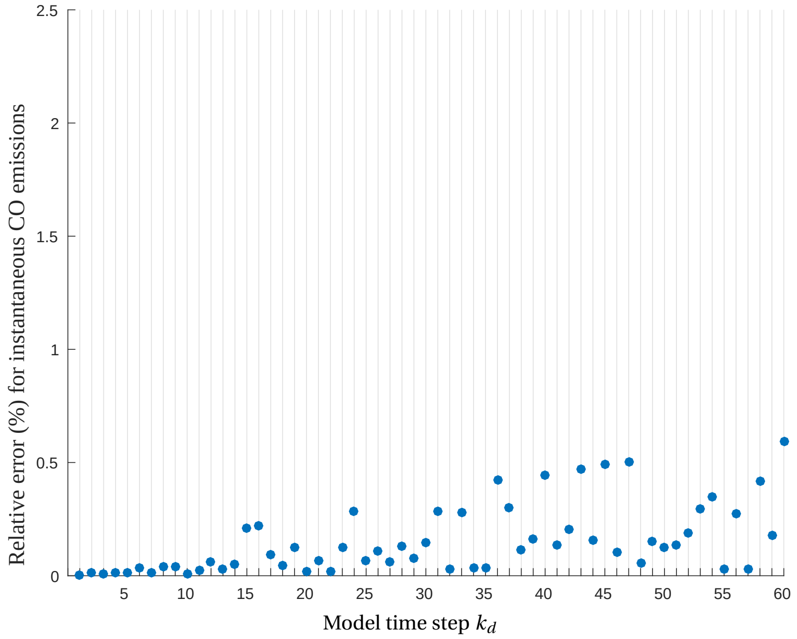
Figure 6.15: Demand profiles for the case study for case 2 (high demand) for a 1-hour simulation (the green times are 30 s).

Figure 6.16: Demand profiles for the case study (case 1: low demand, case 2: high demand).

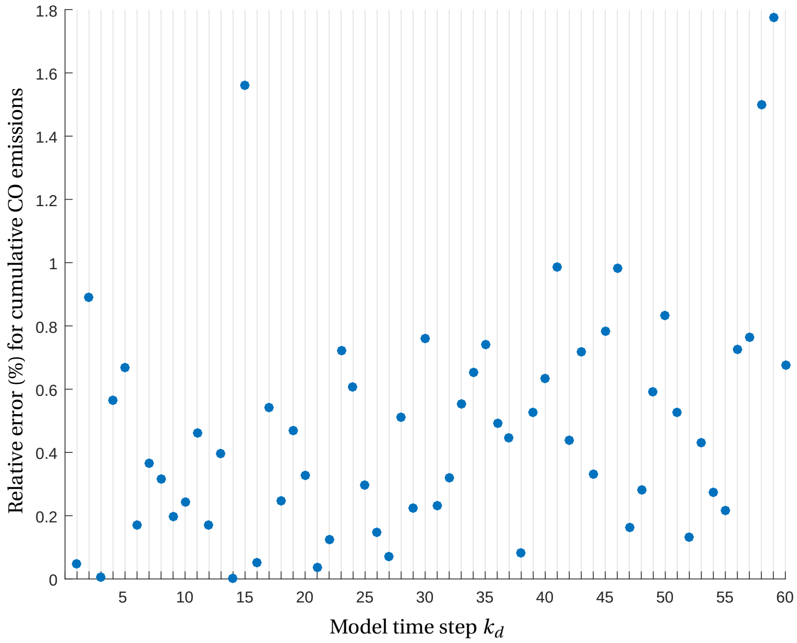
of the instantaneous emissions of CO, HC, and NO_x is less than, respectively, 3.233%, 5.310%, and 2.778%. Additionally, the standard deviations of the relative errors are always less than 1.6. For the relative error of the cumulative emissions of CO, HC, and NO_x, the relative error does not exceed 2.3%, the mean value of the relative error is less than 1.4%, and the standard deviation is below 0.32.

6.8 Conclusions and future work

In this chapter, we have proposed a general framework to integrate and interface traffic flow and emission models. Using the proposed approach, a mesoscopic integrated flow-emission

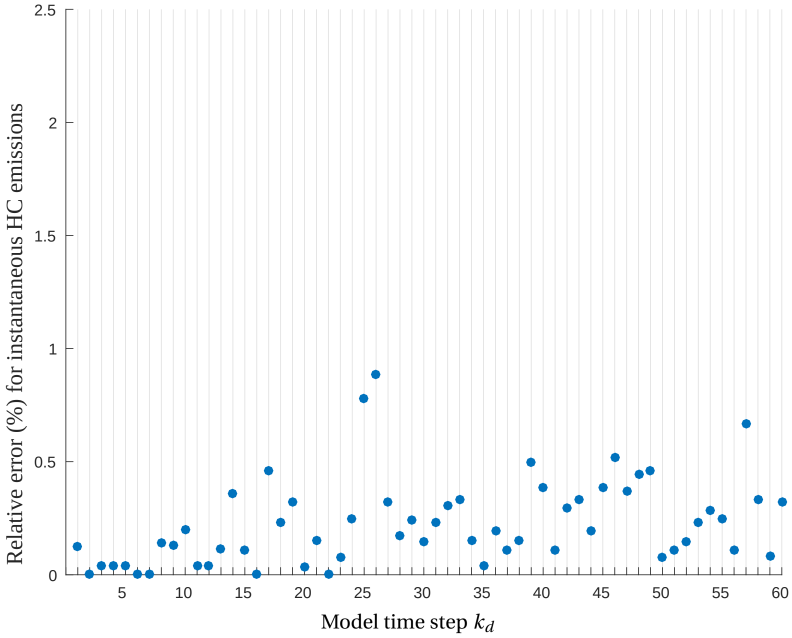


(a) Relative errors (%) of the instantaneous CO emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous CO emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

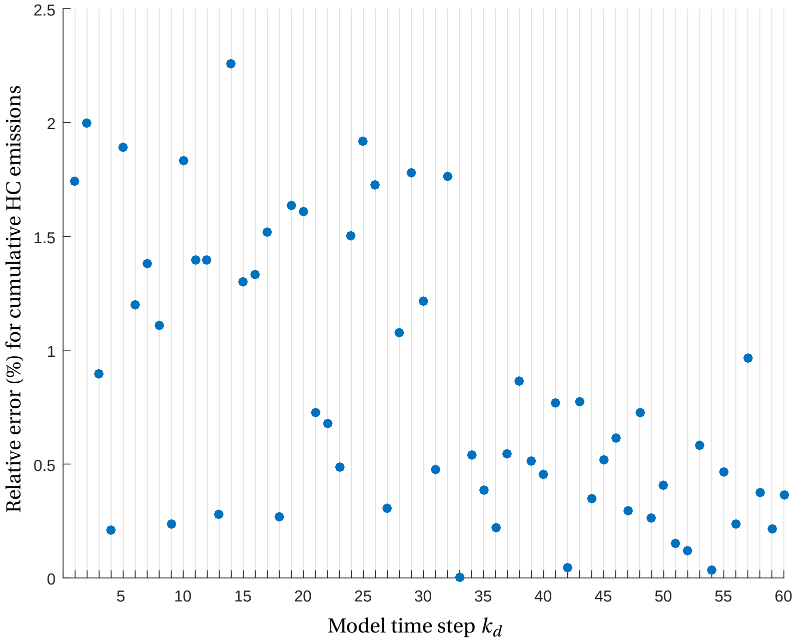


(b) Relative errors (%) of the cumulative CO emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative CO emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

Figure 6.17: Results for CO emissions for case 1 (low demand).

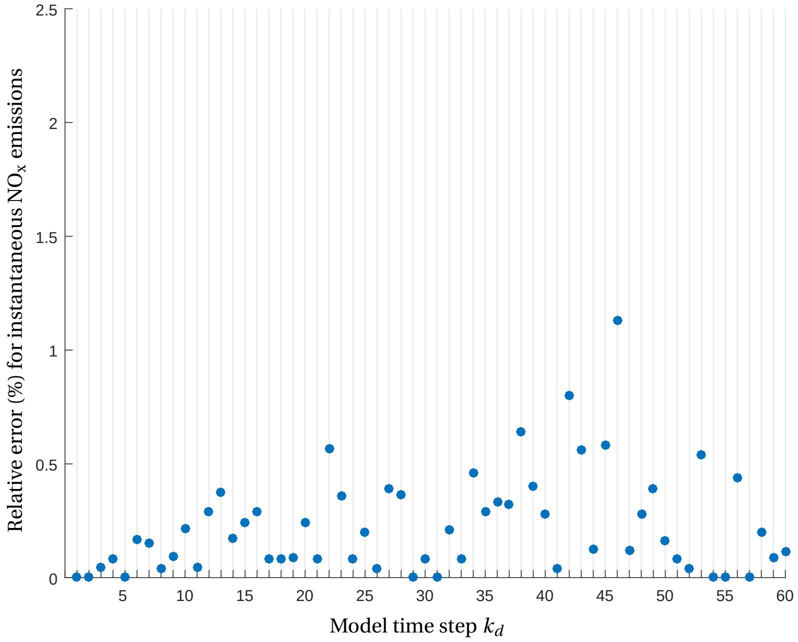


(a) Relative errors (%) of the instantaneous HC emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous HC emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

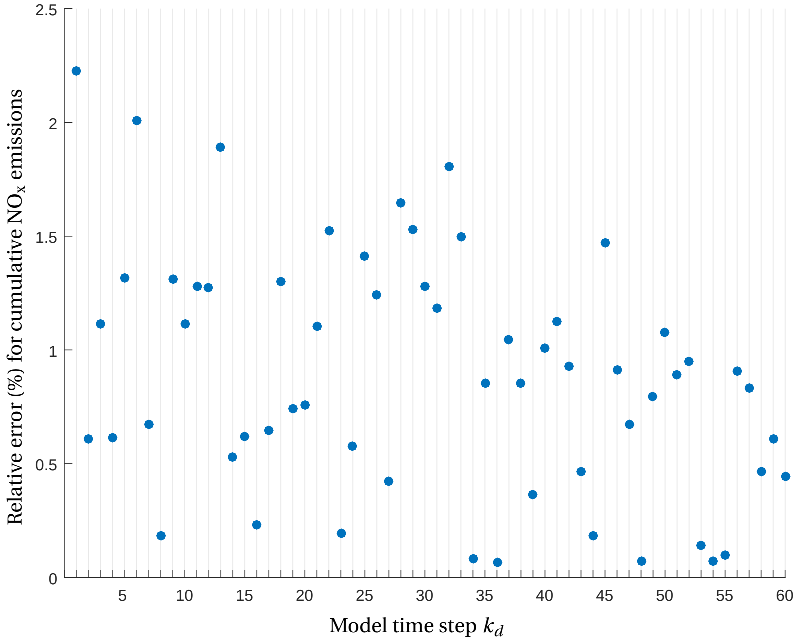


(b) Relative errors (%) of the cumulative HC emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative HC emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

Figure 6.18: Results for HC emissions for case 1 (low demand).

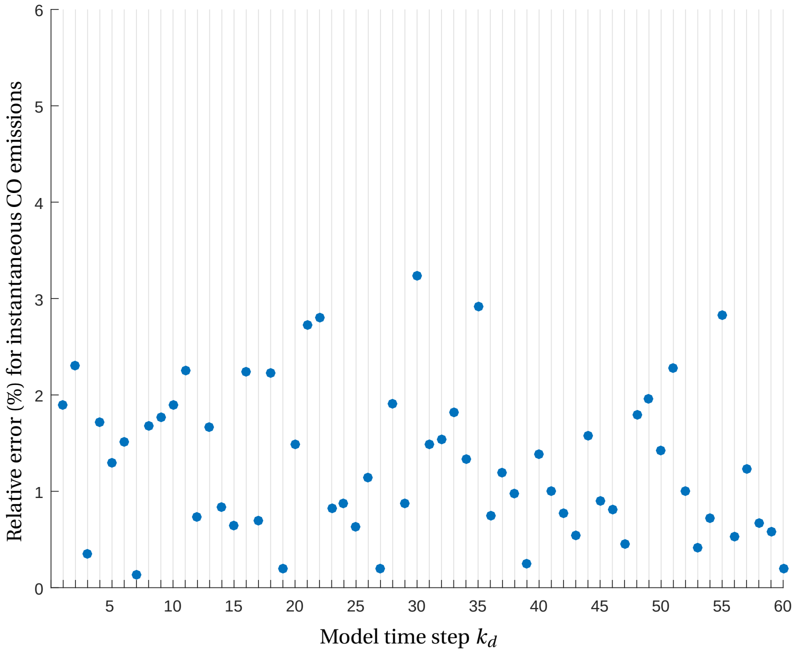


(a) Relative errors (%) of the instantaneous NO_x emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous NO_x emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

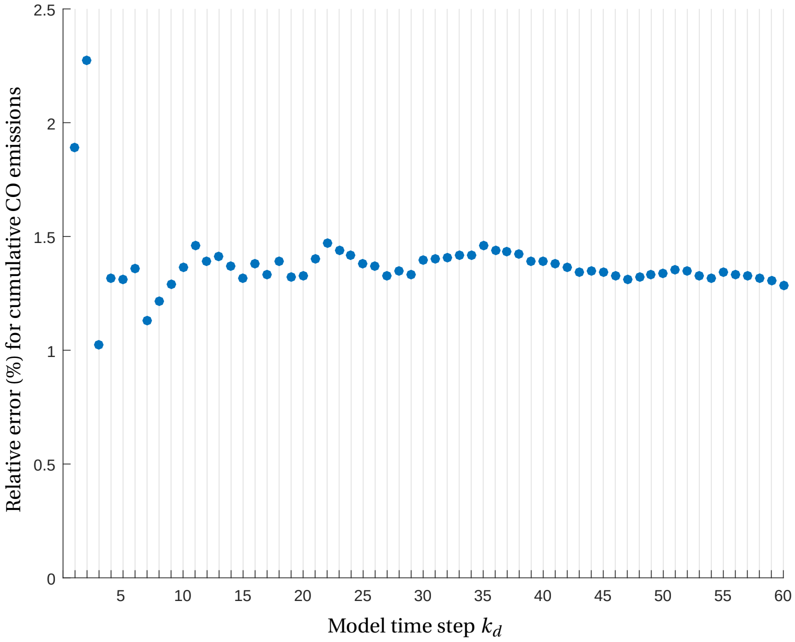


(b) Relative errors (%) of the cumulative NO_x emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative NO_x emissions by the combined SUMO and VT-micro model for the demand profiles of case 1.

Figure 6.19: Results for $\mathrm{NO_x}$ *emissions for case 1 (low demand).*

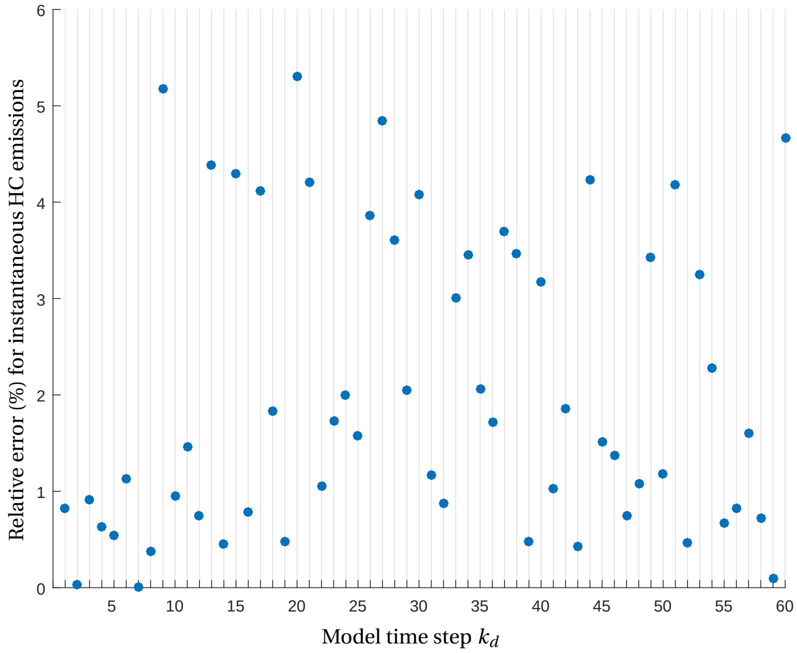


(a) Relative errors (%) of the instantaneous CO emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous CO emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.

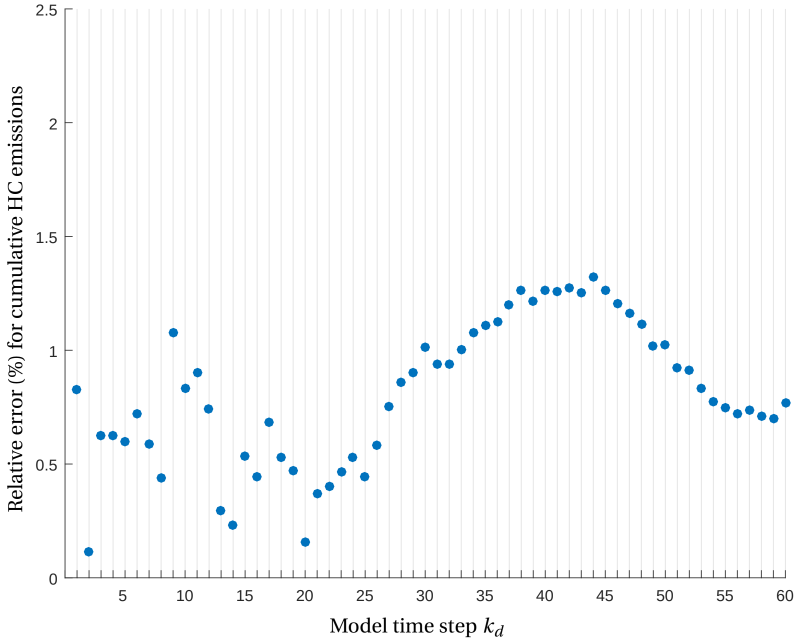


(b) Relative errors (%) of the cumulative CO emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative CO emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.

Figure 6.20: Results for CO emissions for case 2 (high demand).

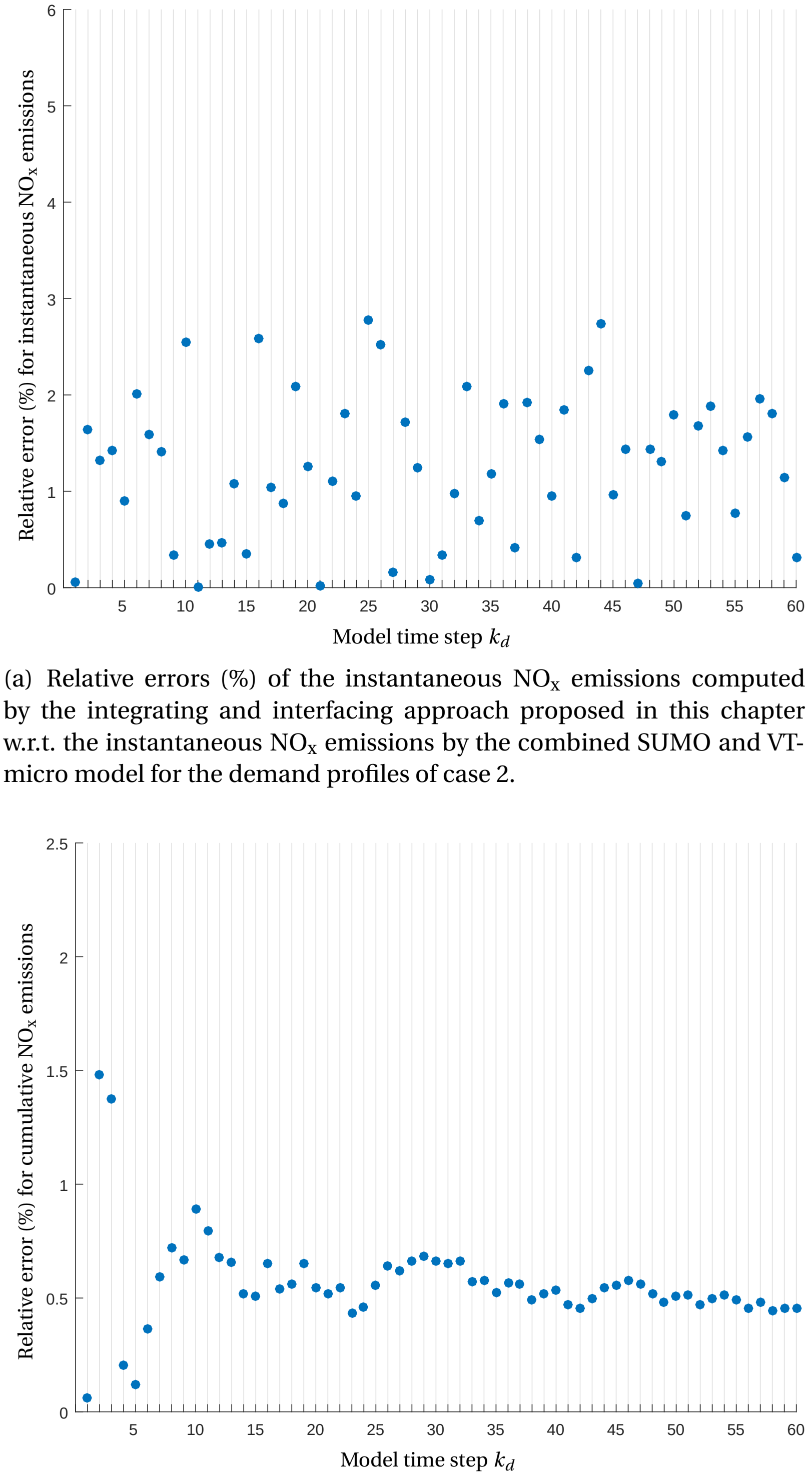


(a) Relative errors (%) of the instantaneous HC emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous HC emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.



(b) Relative errors (%) of the cumulative HC emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative HC emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.

Figure 6.21: Results for HC emissions for case 2 (high demand).



(a) Relative errors (%) of the instantaneous NO$_x$ emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the instantaneous NO$_x$ emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.

(b) Relative errors (%) of the cumulative NO$_x$ emissions computed by the integrating and interfacing approach proposed in this chapter w.r.t. the cumulative NO$_x$ emissions by the combined SUMO and VT-micro model for the demand profiles of case 2.

Figure 6.22: Results for NO$_x$ *emissions for case 2 (high demand).*

Table 6.5: Mean and standard deviation of the relative error of the instantaneous emissions for case 1 (low demand).

	mean of the relative error (%)	standard deviation of the relative error	maximum of the relative error (%)
TE_{CO}	0.152	0.149	0.593
TE_{HC}	0.226	0.189	0.884
TE_{NO_x}	0.226	0.223	1.130

Table 6.6: Mean and standard deviation of the relative error of the cumulative emissions for case 1 (low demand).

	mean of the relative error (%)	standard deviation of the relative error	maximum of the relative error (%)
TE_{CO}	0.483	0.368	1.776
TE_{HC}	0.848	0.617	2.260
TE_{NO_x}	0.887	0.525	2.226

Table 6.7: Mean and standard deviation of the relative error of the instantaneous emissions for case 2 (high demand).

	mean of the relative error (%)	standard deviation of the relative error	maximum of the relative error (%)
TE_{CO}	1.303	0.772	3.233
TE_{HC}	2.036	1.542	5.310
TE_{NO_x}	1.255	0.746	2.778

Table 6.8: Mean and standard deviation of the relative error of the cumulative emissions for case 2 (high demand).

	mean of the relative error (%)	standard deviation of the relative error	maximum of the relative error (%)
TE_{CO}	1.377	0.154	2.273
TE_{HC}	0.810	0.313	1.322
TE_{NO_x}	0.566	0.209	1.481

traffic model is generated that considers an aggregated behavior for groups of vehicles instead of considering the behavior of individual vehicles. The proposed approach can integrate any traffic flow model that updates the total number of vehicles in the links and the number of vehicles in the queues on the links at every model time step, and any microscopic emission model that takes into account the individual speed and acceleration of vehicles in the network. In this chapter, we have also introduced a general formulation for transforming a time-delayed differential equation in the continuous-time domain into an equivalent discrete-time difference equation.

The resulting mesoscopic integrated flow-emission models are, on the one hand, fast enough for real-time computations due to their mesoscopic nature. On the other hand, the resulting models provide a high level of accuracy for computation of the total emissions and the total fuel consumption of the vehicles compared with microscopic traffic models, based on the results of a case study presented in this chapter. The results of the case study show that the relative error (in percentage) of the instantaneous total emissions of the vehicles computed by the mesoscopic integrated flow-emission model generated via the proposed approach w.r.t. the instantaneous total emissions found via the microsimulation traffic software SUMO and the microscopic emission model VT-micro does not exceed 5.3% for CO, HC, and NO_x. Moreover, the mean value and the standard deviation of the relative error are, respectively, less than 2% and 1.6 for CO, HC, and NO_x. Moreover, the relative error (%) of the cumulative total emissions of the vehicles for the mesoscopic integrated flow-emission model w.r.t. SUMO and VT-micro does not exceed 2.3% for CO, HC, and NO_x. The mean value of the relative error is less than 1.4%, and the standard deviation is less than 0.7.

As a topic for future work, due to the resulting low computation time and high accuracy, the proposed integrating and interfacing approach can be used to generate mesoscopic integrated flow-emission models suited for (real-time) model-based traffic applications. A more extensive assessment of the proposed integrating and interfacing approach can be done for further set-ups, scenarios, and traffic signal control strategies. An extensive and in-depth sensitivity analysis of the results produced by the model w.r.t. the parameters used in the model is another topic for future work. Moreover, one could also consider combinations of other macroscopic traffic flow models and microscopic emission models.

Chapter 7

Efficient optimization for MPC: Comparison between smooth and nonsmooth methods

7.1 Introduction

The focus of this chapter is on the development of MPC strategies for urban traffic networks in order to reduce both congestion and emissions. The MPC controller needs a model of the system that is accurate, and at the same time simple and fast for real-time applications. Hence, as the prediction model of the MPC controller we will use the mesoscopic integrated model developed in Chapter 6, where for the flow model we consider the S-model [98] (see Chapter 5 for details regarding the S-model), and for the emission model, we consider VT-micro[3] (see also Chapter 6).

Moreover, to solve the optimization problem of the MPC controller, it will be useful if we can apply efficient gradient-based approaches. However, many of the available mathematical models, including traffic models, have nonsmooth functions or discrete variables in their formulations. Applying these models as prediction models of MPC controllers results in a nonsmooth MPC optimization problem, which restricts application of gradient-based approaches (e.g., methods that are based on Pontryagin's principle [79, 92]). Hence, we propose general smoothening methods for mathematical models of physical systems, and we apply them for the S-model to obtain a smooth model and hence, a smooth optimization problem.

One of the efficient gradient-based approaches that has been widely used [127, 130] is the feasible-direction algorithm proposed by Papageorgiou and Marinaki [128]. To identify an efficient search direction for the optimization, we will apply the latest version of a well-known approach called the resilient back-propagation (RProp) given by Bailey [7], while the main approach has initially been introduced by Riedmiller and Braun [137] (see Chapter 5 for details on the RProp approach).

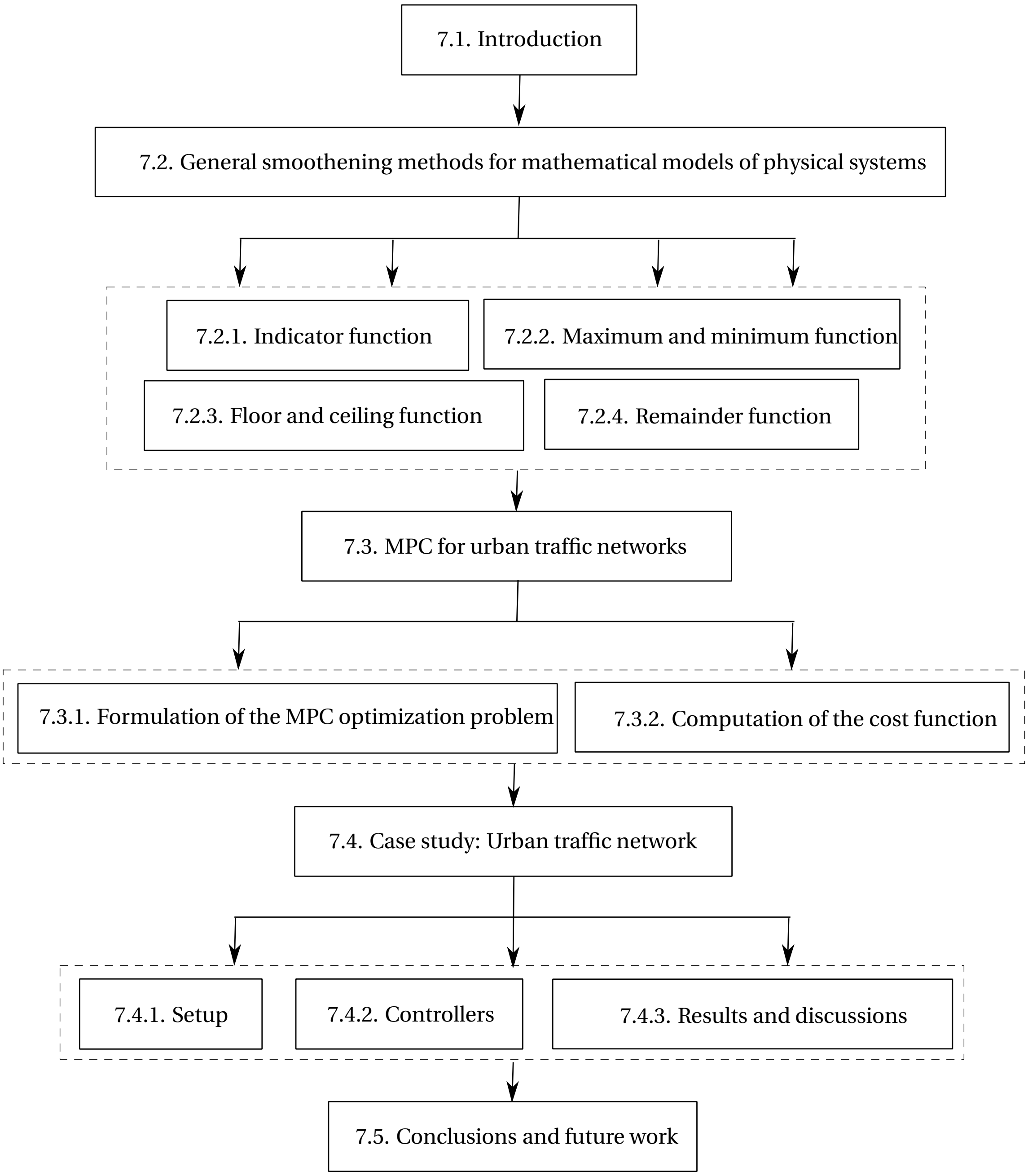

7.1. Introduction
7.2. General smoothening methods for mathematical models of physical systems
7.2.1. Indicator function
7.2.2. Maximum and minimum function
7.2.3. Floor and ceiling function
7.2.4. Remainder function
7.3. MPC for urban traffic networks
7.3.1. Formulation of the MPC optimization problem
7.3.2. Computation of the cost function
7.4. Case study: Urban traffic network
7.4.1. Setup
7.4.2. Controllers
7.4.3. Results and discussions
7.5. Conclusions and future work

Figure 7.1: Road map of Chapter 7.

Contributions and organization of the chapter

The main contributions of the chapter include:

- We propose general smoothening methods for mathematical models of physical systems, where the proposed methods can be applied to various models with nonsmooth functions, including transportation models (see Section 7.2). The resulting smooth models can be used in model-based optimal control techniques, e.g., in MPC, where the optimization problem can be solved by efficient smooth and gradient-based solvers.

- Implementing the proposed general smoothening methods, we show that even for nonsmooth and nonlinear flow models, we can easily apply an efficient gradient-based optimization approach that uses resilient back-propagation (RProp) and solves the optimization problem of the MPC controller much faster than nonsmooth optimization approaches. Finally, all these methods are combined to develop a novel approach to urban traffic network control aiming at optimizing a trade-off of congestion mitigation versus emission reduction (see Section 7.4).

The structure of this chapter is as follows. In Section 7.2 we introduce general smoothening approaches for mathematical models of physical systems that involve nonsmooth functions. When these nonsmooth models are used to formulate an optimization problem, these smoothening methods can transform the resulting nonsmooth problem into a smooth one, which can then be solved by efficient gradient-based optimization methods. In Section 7.3 an MPC controller for urban traffic networks is developed that finds a balanced trade-off between reduction of the total time spent and total emissions of the vehicles. We discuss how to formulate the optimization problem of the MPC controller and how to compute the corresponding cost function. For application, we can use the smoothening methods proposed in Section 7.2 to make the optimization problem smooth. Correspondingly, we consider a gradient-based optimization solver that benefits from the resilient back-propagation (RProp) method. Section 7.4 presents the results of a case study and the gradient-based approach for solving the MPC optimization problem. Finally, Section 7.5 concludes the work and proposes topics for future work. The road map of Chapter 7 is illustrated in Figure 7.1. Moreover, Table 7.1 gives the frequently used mathematical notations of this chapter.

This chapter of the thesis is based on [72].

7.2 General smoothening methods for mathematical models of physical systems

Many of the available mathematical models of physical systems, including transportation networks, involve nonsmooth functions in their formulations. Some examples of such transportation models include the cell transmission model (CTM) [33], the dynamic IFTN model [93] applied in modeling the container flows, and also max-plus-linear models, e.g., see [52], that are used to model railway transportation systems. It is very beneficial if we can use these models as the prediction model of an MPC controller, and still apply efficient gradient-based approaches (e.g., Pontryagin's approach [79]) to solve the optimization problem of the MPC. For this aim, we first need to find general approaches to render these nonsmooth functions smooth and hence, differentiable. In this section, we develop smooth approximate functions for the nonsmooth indicator, maximum, minimum, floor, ceiling, and remainder functions that appear frequently in mathematical models of physical systems.

Table 7.1: Frequently used mathematical notations for Chapter 7.

α	smoothening parameter
I_L	indicator function for the logical statement L
$\lfloor\cdot\rfloor$	floor function
$\lceil\cdot\rceil$	ceiling function
$\mathrm{rem}\{\cdot,\cdot\}$	remainder function
$\mathcal{J}(\cdot)$	cost function
$\mathcal{J}^{\mathrm{t}}(\cdot)$	terminal cost function
$\mathcal{J}^{\mathrm{s}}(\cdot)$	stage cost function
$\mathcal{J}^{\mathrm{s,cum}}(\cdot)$	cumulative stage cost function
k_{c}	control time step counter
T_{c}	control sampling time
k_d	simulation time step counter
c_d	simulation sampling time
N_{p}	prediction horizon of the MPC controller
$\boldsymbol{u}(k_{\mathrm{c}})$	control input at control time step k_{c}
$\tilde{\boldsymbol{u}}(k_{\mathrm{c}})$	a vector that contains control inputs $\boldsymbol{u}(k_{\mathrm{c}}),\ldots,\boldsymbol{u}\left(k_{\mathrm{c}}+N_{\mathrm{p}}-1\right)$
$\mathrm{TTS}(k)$	total time spent of the vehicles in between time steps k and $k+1$
$\mathrm{TE}_p(k)$	total emissions of the pollutant p by the vehicles in between time steps k and $k+1$
$\mathrm{TTS}^{\mathrm{t}}$	typical value of the total time spent of the vehicles within one prediction window
$\mathrm{TE}_p^{\mathrm{t}}$	typical value of the total emissions of the vehicles within one prediction window

7.2.1 Indicator function

The indicator function $I_{x\geq a}$ is defined by

$$I_{x\geq a}=\begin{cases}1, & x\geq a\\ 0, & x<a\end{cases}, \tag{7.1}$$

and hence is nonsmooth at zero. In [25] the smooth form of the indicator function is approximated by a sigmoid function (see Figure 7.2, which shows the indicator function with a solid black curve and its smooth approximate with a dashed red curve):

$$I_{x\geq 0}\approx\left(1+e^{-\alpha x}\right)^{-1}, \qquad \alpha>0, \tag{7.2}$$

with α a smoothening parameter.

7.2.2 Maximum and minimum functions

In this section, we consider the maximum and minimum functions, which both are nonsmooth and which are also easily transformable into one another. In general, we are looking

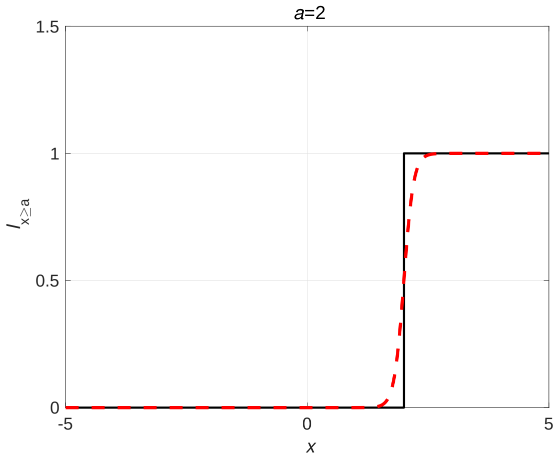


Figure 7.2: Indicator function of $I_{x \geq a}$ and its smooth approximation.

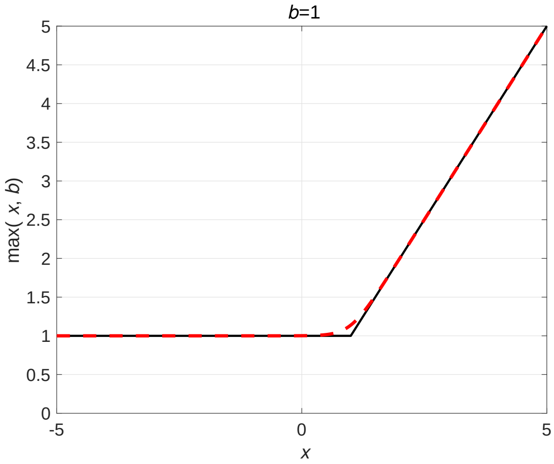


Figure 7.3: Maximum function and its smooth approximation.

for the smooth form of $\max_{i=1,\dots,n} \{x_i\}$ and $\min_{i=1,\dots,n} \{x_i\}$. Figure 7.3 illustrates the maximum function $\max\{x,b\}$ for $b=1$ with a solid black curve and its smooth form with a dashed red curve. We have

$$\max\{x_1, x_2\} \approx \frac{1}{\alpha}\left(e^{\alpha x_1} + e^{\alpha x_2}\right). \tag{7.3}$$

From the fact that

$$\max\{x_1, x_2, x_3\} = \max\{x_1, \max\{x_2, x_3\}\},$$

and by mathematical induction, we can easily obtain the following general form of the smooth maximum function:

$$\max_{i=1,\dots,n} \{x_i\} \approx \frac{1}{\alpha} \sum_{i=1}^{n} e^{\alpha x_i}. \tag{7.4}$$

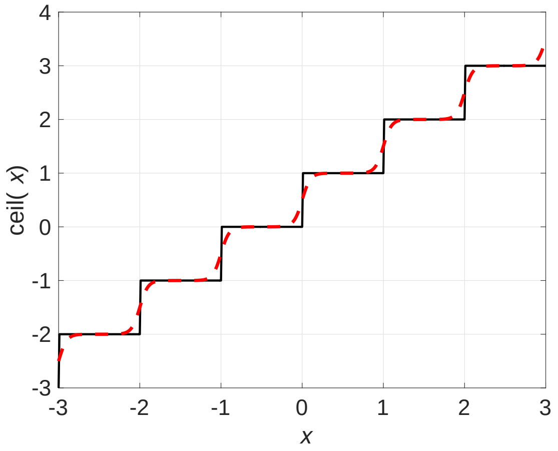


Figure 7.4: Ceiling function and its approximation.

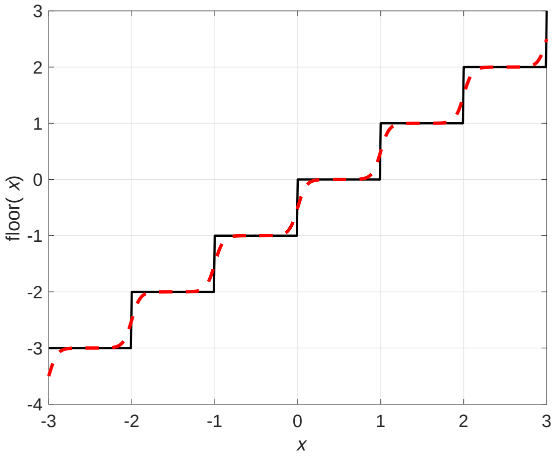


Figure 7.5: Floor function and its approximation.

For the minimum function, we have

$$\min_{i=1,\dots,n}\{x_i\} = -\max_{i=1,\dots,n}\{-x_i\},$$

which together with (7.4) gives

$$\min_{i=1,\dots,n}\{x_i\} \approx -\frac{1}{\alpha}\sum_{i=1}^{n} e^{-\alpha x_i}. \tag{7.5}$$

7.2.3 Floor and ceiling functions

The ceiling and the floor functions may appear in mathematical models of different physical systems, especially for discrete-time models. The ceiling function $\lceil x \rceil$, which operates on $x \in \mathbb{R}$, is defined as the least integer that is not less than x. The floor function $\lfloor x \rfloor$, which also operates on $x \in \mathbb{R}$, is defined as the largest integer that is not larger than x. Figures 7.4 and 7.5 illustrate the ceiling and the floor functions by solid black curves and their smooth approximates by dashed red curves. From these figures, it is observed that at the integer points on the horizontal axis, both the ceiling and the floor function show a nonsmooth behavior.

Consider the indicator function (see Section 7.2.1) together with an infinite number of transformed indicator functions, with the first transformed function shifted for one unit to the right, the second transformed function shifted for two units to the right, and so on. The ceiling function on the right-hand side of the vertical axis (see Figure 7.4), i.e., for non-negative integers $k \in \{0, 1, \ldots\}$, is obtained from the summation of the indicator function and all these transformed indicator functions over the range of the non-negative integers. Next consider an infinite number of transformed indicator functions, where the first transformed function is shifted for one unit to the left, the second transformed function is shifted for two units to the left, and so on. Moreover, all the transformed indicator functions are shifted down for one unit. The summation of these functions within the range of negative integers produces the ceiling function for the negative integer values. Hence, to approximate the ceiling function with a smooth function, we can consider the summation of the sigmoid function (see Section 7.2.1) and the corresponding transformed sigmoid functions for the appropriate integer ranges as explained above.

In summary, for the smooth approximation of the ceiling function we have

$$\lceil x \rceil \approx \sum_{k\in\{0,1,2,\ldots\}} \left(1+e^{-\alpha(x-k)}\right)^{-1} + \sum_{k\in\{\ldots,-2,-1\}} \left(\left(1+e^{-\alpha(x-k)}\right)^{-1} - 1\right), \tag{7.6}$$

or equivalently

$$\lceil x \rceil \approx 0.5 + \sum_{k\in\mathbb{Z}} \left(\left(1+e^{-\alpha(x-k)}\right)^{-1} - 0.5\right). \tag{7.7}$$

Since $\lfloor x \rfloor = \lceil x \rceil - 1$, we can write

$$\lfloor x \rfloor \approx -0.5 + \sum_{k\in\mathbb{Z}} \left(\left(1+e^{-\alpha(x-k)}\right)^{-1} - 0.5\right). \tag{7.8}$$

7.2.4 Remainder function

The remainder function $\text{rem}\{a, b\}$ yields the remainder value of the division of a by b, i.e.,

$$\text{rem}\{a, b\} = a - qb, \tag{7.9}$$

where q is the quotient. Figure 7.6 illustrates the graph representing $\text{rem}\{a, b\}$ for $b = 2$. We consider the concept of Euclidean division:

- for a positive divisor $b > 0$, we have $q = \left\lfloor \frac{a}{b} \right\rfloor$,
- for a negative divisor $b < 0$, we have $q = \left\lceil \frac{a}{b} \right\rceil$.

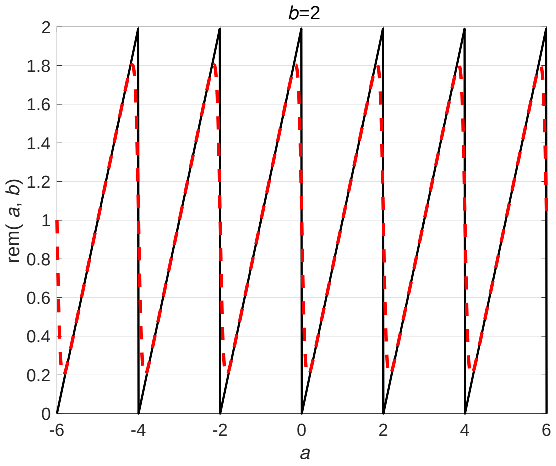


Figure 7.6: Remainder function and its smooth approximation.

The above expressions can equivalently be expressed by

$$q = \frac{1+\text{sign}(b)}{2}\left\lfloor \frac{a}{b} \right\rfloor + \frac{1-\text{sign}(b)}{2}\left\lceil \frac{a}{b} \right\rceil, \tag{7.10}$$

with sign(·) indicating the sign function:

$$\text{sign}(x) = \begin{cases} 1 & x > 0, \\ -1 & x < 0. \end{cases}$$

We can write $\text{sign}(x) = I_{x\geq 0} - I_{x\leq 0}$. Therefore,

$$\text{sign}(x) \approx \left(1+e^{-\alpha x}\right)^{-1} - \left(1+e^{\alpha x}\right)^{-1}, \tag{7.11}$$

and for the remainder function we obtain

$$\text{rem}\{a,b\} \approx a - b\sum_{k\in\mathbb{Z}}\left(\left(1+e^{-\alpha\left(\frac{a}{b}-k\right)}\right)^{-1} - 0.5\right) + \frac{b}{2}\left(\left(1+e^{-\alpha b}\right)^{-1} - \left(1+e^{\alpha b}\right)^{-1}\right). \tag{7.12}$$

7.3 MPC for urban traffic networks

Taking into account the positive characteristics of an MPC controller (i.e., it performs a feedback-based and to some extent robust control approach), which fit well the requirements that are usually expected for management of traffic networks with highly dynamic behaviors, we consider the MPC approach for urban traffic control. In this section, the aim of the controller is to find a balanced trade-off between prevention/reduction of congestion and reduction of emissions. Therefore, for the prediction model of the MPC controller, the integrated flow and emission framework introduced in Chapter 6 including the extended S-model (see Section 6.2) and VT-micro (see Section 5.3.2) is used.

7.3.1 Formulation of the MPC optimization problem

$$
\begin{aligned}
&\min_{\tilde{\boldsymbol{u}}(k_\mathrm{c})} \mathcal{J}(k_\mathrm{c}) = \min_{\boldsymbol{u}(\tilde{k}_\mathrm{c})} \left(\mathcal{J}^\mathrm{t}(k_\mathrm{c}) + \mathcal{J}^\mathrm{s,cum}(k_\mathrm{c})\right),\\
&\text{s.t.}\\
&\qquad \text{(7.15), (7.16), and (7.20),} \qquad\qquad (7.13)\\
&\qquad \text{Integrated flow and emission model (see Chapter 6),}\\
&\qquad \mathcal{U}(\tilde{\boldsymbol{u}}(k_\mathrm{c})) = 0 \text{ (e.g., see (7.14)),}\\
&\qquad \boldsymbol{u}_\mathrm{min} \leq \tilde{\boldsymbol{u}}(k_\mathrm{c}) \leq \boldsymbol{u}_\mathrm{max},
\end{aligned}
$$

where $\mathcal{J}(k_\mathrm{c})$ denotes the summation of the terminal objective $\mathcal{J}^\mathrm{t}(k_\mathrm{c})$ and the cumulative stage objective function $\mathcal{J}^\mathrm{s,cum}(k_\mathrm{c})$ that is computed within one prediction window starting at control time step k_c, i.e., within $[k_\mathrm{c}T_\mathrm{c}, \left(k_\mathrm{c}+N_\mathrm{p}\right)T_\mathrm{c})$ with T_c the control sampling cycle. Furthermore, the optimization variable $\tilde{\boldsymbol{u}}(k_\mathrm{c})$ is a vector that includes $\boldsymbol{u}^\top(k_\mathrm{c}),\ldots,\boldsymbol{u}^\top(k_\mathrm{c}+N_\mathrm{p}-1)$, $\boldsymbol{u}(k_\mathrm{c})$ is a vector that includes all control inputs of the system (the green time lengths for an urban traffic network along the prediction window) at control time step k_c, $\boldsymbol{u}_\mathrm{min}$ and $\boldsymbol{u}_\mathrm{max}$ are vectors of the same size as $\boldsymbol{u}(k_\mathrm{c})$ that include element-wise the minimum and maximum allowed values for the control inputs within $\boldsymbol{u}(k_\mathrm{c})$, and $\mathcal{U}(\boldsymbol{u}(k_\mathrm{c}))=0$ indicates the equality constraints on the control vector. For example, $\mathcal{U}(\boldsymbol{u}(k_\mathrm{c}))=0$ may indicate that the summation of the green and yellow times for each intersection equals the cycle time for that intersection. More specifically, suppose that $u_{d,i}(l)$ for all $d\in\mathcal{I}^\mathrm{ctrl}$ (with $\mathcal{I}^\mathrm{ctrl}$ the set of all intersections of the urban traffic network that are controlled by traffic signals) and $i\in\left\{1,\ldots,n_d^\mathrm{green}\right\}$ (with n_d^green the number of green signals[1] of intersection d) indicates the i^th traffic signal at intersection d. Then for all $d\in\mathcal{I}^\mathrm{ctrl}, \mathcal{U}(\boldsymbol{u}(k_\mathrm{c}))=0$ is a relationship of the form

$$
\sum_{i=1}^{n_d^\mathrm{green}} u_{d,i}(l) = c_d - y_d,\ l\in\left\{k_\mathrm{c},\ldots,k_\mathrm{c}+N_\mathrm{p}-1\right\}, \tag{7.14}
$$

where y_d indicates the total yellow or all-red time within one cycle of the intersection corresponding to node d.

The main aim of the urban traffic control system is to find a balanced trade-off between reducing the congestion level, reducing the total emissions, and preventing high fluctuations in time for the control inputs. Hence, the stage cost function of the MPC controller is formulated as a weighted combination of the total time spent (which quantifies the congestion level), the total emissions, and the absolute difference of two temporally successive control vectors. We have

$$
\mathcal{J}^\mathrm{s,cum}(k_\mathrm{c}) = \sum_{k=k_\mathrm{c}}^{k_\mathrm{c}+N_\mathrm{p}-1} \mathcal{J}^\mathrm{s}(k), \tag{7.15}
$$

$$
\mathcal{J}^\mathrm{s}(k) = w_\mathrm{TTS}\frac{\mathrm{TTS}(k)}{\mathrm{TTS}^\mathrm{t}} + \sum_{p\in\mathbb{P}} w_{\mathrm{TE}_p}\frac{\mathrm{TE}_p(k)}{\mathrm{TE}_p^\mathrm{t}} + w_v\frac{\mathcal{V}(\boldsymbol{u}(k))}{\mathcal{V}^\mathrm{n}}, \tag{7.16}
$$

[1] Note that at intersection d, for each right-of-way, we consider a traffic signal.

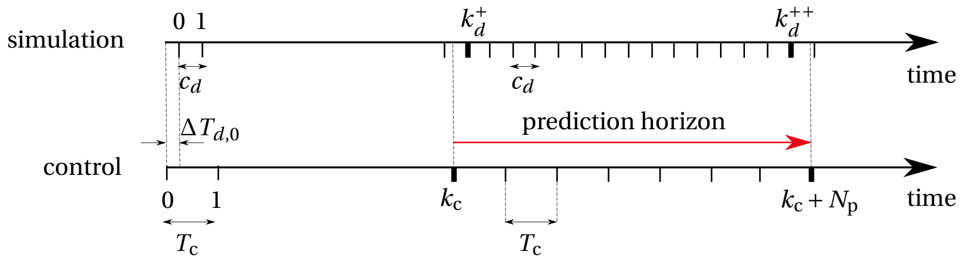


Figure 7.7: Simulation and control time step counters.

with $\text{TTS}(k)$ and $\text{TE}_p(k)$ the total time spent and total emissions of $p \in \mathbb{P}$ (where $\mathbb{P}$ is a set of pollutants, e.g., $\mathbb{P} = \{\text{CO, HC, NO}_\text{x}\}$) within one control sampling time (i.e., within $[kT_c, (k+1)T_c)$ for $k \in \{k_c, \ldots, k_c + N_p - 1\}$), TTS^t and TE_p^t typical values of $\text{TTS}(\cdot)$ and $\text{TE}_p(\cdot)$ within one prediction window, $\mathcal{V}(\boldsymbol{u}(k_c)) = \|\boldsymbol{u}(k_c) - \boldsymbol{u}(k_c - 1)\|$ for some norm function $\|\cdot\|$, and $\mathcal{V}^\text{n}$ a nominal value for $\mathcal{V}(\cdot)$ in one control sampling time that may be computed by $\|\boldsymbol{u}_\text{max} - \boldsymbol{u}_\text{min}\|$.

In our problem, for the terminal cost function, we have considered the following formulation:

$$\mathcal{J}^\text{t}(k_c + N_p) = \alpha \left\|\boldsymbol{x}(k_c + N_p)\right\|, \tag{7.17}$$

with α a positive constant and $\boldsymbol{x}$ the state vector (including the total number of vehicles in a link and the number of vehicles waiting in the queue on a link). The aim of this terminal cost is to reduce the final queue lengths on the links. Moreover, for the stage cost function (7.16), we should determine $\text{TTS}_{u,d}(k)$ and $\text{TE}_{p,u,d}(k)$ for all links (u,d) within the traffic network and sum them up to obtain $\text{TTS}(k)$ and $\text{TE}_{p,u,d}(k)$ of all the links in the traffic network for control sampling cycle k. First the governing traffic scenario on each link is detected using (6.28), (6.68), and (6.124). Then for the under-saturated scenario, we use (6.37)-(6.45), for case 1 of the saturated scenario, we use (6.75)-(6.83), for case 2 of the saturated scenario, we use (6.102)-(6.109), and for the over-saturated scenario, we use (6.130)-(6.138) to obtain the TTS of each link. Moreover, using (6.23)-(6.27) together with the mentioned equations, we can obtain the TE on each link.

7.3.2 Computation of the cost function

In general, the system's sampling time and the control sampling time might not be equal and therefore, the time instants $k_d c_d$ and $k_c T_c$ (correspondingly the simulation time step k_d of link (u,d) and the control time step k_c of the network) may not coincide. Hence, we should find a relationship between k_d and k_c. Suppose that $k_c = 0$ corresponds to the absolute time instant $t = 0$ (see Figure 7.7). Then, the first upcoming simulation time step $k_d^+(k_c)$ for link (u,d) at time instant $k_c T_c$ is computed by

$$k_d^+(k_c) = \left\lceil \frac{k_c T_c - \Delta T_{d,0}}{c_d} \right\rceil, \tag{7.18}$$

where $\Delta T_{d,0}$ is the offset between the first simulation time instant of link (u,d) (corresponding to the simulation time step $k_d = 0$ in Figure 7.7), and the first control time instant (corresponding to $k_{\text{c}} = 0$ or $t = 0$ in Figure 7.7). For $k_d^{++}(k_{\text{c}})$, i.e., the last simulation time step that occurs during the current prediction time interval (see Figure 7.7), we can write

$$k_d^{++}(k_{\text{c}}) = \left\lfloor \frac{(k_{\text{c}} + N_{\text{p}})T_{\text{c}} - \Delta T_{d,0}}{c_d} \right\rfloor. \tag{7.19}$$

In (7.16), $\text{TTS}(\cdot)$ and $\text{TE}_p(\cdot)$ are the summation of the total time spent and total emissions for all links in the traffic network. For the total time spent by the vehicles on link (u,d) at time step k_{c}, we have

$$\begin{aligned}
\text{TTS}_{u,d}(k_{\text{c}}) = & \sum_{i=1}^{N_{\text{G}}^s} \sum_{b\in\mathbb{B}} \left(n_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})-1\right) \min\left\{ T_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})-1\right), k_d^+(k_{\text{c}})c_d - k_{\text{c}}T_{\text{c}} \right\} \right) \\
& + \sum_{j=k_d^+(k_{\text{c}})}^{k_d^{++}(k_{\text{c}})-1} \sum_{i=1}^{N_{\text{G}}^s} \sum_{b\in\mathbb{B}} n_{u,d}^{b,i,s}(j)\, T_{u,d}^{b,i,s}(j) \\
& + \sum_{i=1}^{N_{\text{G}}^s} \sum_{b\in\mathbb{B}} \left(n_{u,d}^{b,i,s}\left(k_d^{++}(k_{\text{c}})\right) \min\left\{ T_{u,d}^{b,i,s}\left(k_d^{++}(k_{\text{c}})\right), (k_{\text{c}}+N_{\text{p}})T_{\text{c}} - k_d^{++}(k_{\text{c}})c_d \right\} \right),
\end{aligned} \tag{7.20}$$

where s adopts under-saturated, saturated, and over-saturated based on the ongoing traffic scenario on the link, $\mathbb{B}$={free, idling, dec, acc}, N_{G}^s shows the number of different groups for the ongoing traffic state, and $T_{u,d}^{b,\text{G}_i,s}(k_d)$ is computed by the equations given in Chapter 6 (Sections 6.4), 6.5, and 6.6), recalling that $n_{u,d}^{b,i,s}(k_d)$ and $T_{u,d}^{b,i,s}(k_d)$ refer to the total number of vehicles and the average travel time for each of these vehicles on link (u,d) that belong to group i, and that show the traffic behavior b during $[k_d c_d, (k_d+1)c_d)$, while the governing traffic scenario on the link is scenario s. The total emissions E_p of p for each link of the network can be computed via (6.23), (6.24), (6.26), and (6.27).

Note that in (7.20), we have assumed that $n_{u,d}^{b,i,s}(\cdot)$ follows a piecewise constant trajectory, i.e., its value is fixed in the intervals $[k_d c_d, (k_d+1)c_d)$. Therefore,

$$n_{u,d}^{b,i,s}(k_{\text{c}}) = n_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})-1\right), \qquad \text{for} \quad k_d^+(k_{\text{c}})-1 \le k_{\text{c}} < k_d^+(k_{\text{c}}).$$

One may assume a linear extrapolation approximation for the state instead, which is based on the two consecutive system's time steps, i.e.,

$$n_{u,d}^{\text{linear},b,i,s}(k_{\text{c}}) = n_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})\right) + \left(k_{\text{c}}\frac{T_{\text{c}}}{c_d} - k_d^+(k_{\text{c}})\right)\cdot\left(n_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})\right) - n_{u,d}^{b,i,s}\left(k_d^+(k_{\text{c}})-1\right)\right). \tag{7.21}$$

7.4 Case study

In this section, we consider a case study to evaluate the designed smooth MPC controller. We focus on both the performance and the computation speed of the proposed control approach.

Table 7.2: Network parameters

N^{lane} [-]	ℓ^{link} [m]	μ [veh/s]	c [s]	$\beta_{5,7,9}$	$\beta_{5,7,11}$	$\beta_{6,8,11}$	$\beta_{6,8,10}$
1	500	0.8	60	0.6	0.4	0.6	0.4

Table 7.3: Model parameters

v^{free} [m/s]	v^{idle} [m/s]	a^{acc} [m/s^2]	a^{dec} [m/s^2]	l^{veh} [m]
14	0.4	2	-2	7

In order to evaluate the performance of the MPC controller, we compare the values of total time spent, total emissions, and the value of the cost function computed by (7.15)-(7.17) for the MPC controller with those of the no-control case and an efficient fixed-time traffic control approach that has been precomputed offline. We also evaluate the controller from the point of computational efficiency (i.e., the CPU time).

7.4.1 Setup

The urban traffic network we use for the case study is the one used in Chapter 6 shown in Figure 6.13. The network consists of 11 links, where all links have the same characteristics, i.e., the same number of lanes N^{lane}, length ℓ^{link}, and saturated leaving flow rate μ. Moreover, the traffic lights of all the three intersections that are controlled (see Figure 6.13) have the same cycle time c. The parameters β_1, β_2, β_3, and β_4 are the turning rates (i.e., the percentage of vehicles on a link that turn to a specific downstream link) of the vehicles at the corresponding intersection. We use the integrated flow and emission model proposed in Chapter 6 with the traffic parameters listed in Tables 7.2 and 7.3 to simulate the traffic in this network.

We run the simulations for three different demand profiles shown in Figure 7.8, where α_1^{enter}, α_2^{enter}, α_3^{enter}, and α_4^{enter} are the demands (i.e., the entering flow rates) of, respectively, origin 1, 2, 3, and 4 in Figure 6.13. These profiles have been selected in such a way that they highlight specific features, and such that they result in various traffic scenarios (e.g., undersaturated, saturated, and over-saturated) on different links of the traffic network. More specifically, 'Demand profile 1' corresponds to a relatively balanced case for the four demands $\alpha_1^{\text{enter}}, \ldots, \alpha_4^{\text{enter}}$. 'Demand profile 2' is a case where the demands at origin 2 (α_2^{enter}) and 3 (α_3^{enter}) are medium. For 'Demand profile 3', the demand at origin 3 is very high, while at the other origins the demand is low to medium. Comparing 'Demand profile 2' and 'Demand profile 3', we see a more irregular pattern for the latter case. Note that for each of the three demand profiles, at some periods during the simulation congestion occurs, more specifically on links (1,5), (2,5), (3,6), (4,6), (7,11), and (8,11). Moreover, on the intermediate links (5,7) and (6,8), a moderately congested traffic is sometimes observed. Each simulation run covers an entire hour. For each demand profile, we repeat the simulations 10 times and compute the CPU time and the realized values of the objective function, the total time spent, and the total emissions of CO, HC, and NO$_\text{x}$ for each run. We will consider and compare the performance of different controllers that are described in Section 7.4.2 with the no-control case, which corresponds to a case where there are no traffic lights at all at the intersections (i.e., all links are always open to the vehicles, unless the downstream road is blocked).

7.4.2 Controllers

State-feedback controller

At every control time step k, the state-feedback controller divides the green times $g_{u_1,d}$ and $g_{u_2,d}$ between two incoming links (u_1, d) and (u_2, d) at an intersection d taking into account the total number of vehicles $n_{u_1,d}$ and $n_{u_2,d}$ on the links, and the number of vehicles in the queues $q_{u_1,d}$ and $q_{u_2,d}$, i.e., division of the green times between the links is based on the ratios $\frac{n_{u_1,d}(k)+\rho q_{u_1,d}(k)}{n_{u_1,d}(k)+n_{u_2,d}(k)+\rho\left(q_{u_1,d}(k)+q_{u_2,d}(k)\right)}$ and $\frac{n_{u_2,d}(k)+\rho q_{u_2,d}(k)}{n_{u_1,d}(k)+n_{u_2,d}(k)+\rho\left(q_{u_1,d}(k)+q_{u_2,d}(k)\right)}$, respectively. Note that ρ is a parameter that can be tuned.

Optimal fixed-time controller

This controller has constant signal settings that have been optimized offline for each demand class (Demand profiles 1, 2, 3) separately. For this aim, the objective function (i.e., the weighted sum of the total time spent and total emissions) was minimized for each demand class in a 1-h simulation using brute force with a grid size of 0.1, and a fixed traffic signal setting was obtained.

MPC controller with RProp

This controller uses a gradient-based optimization approach based on the resilient backpropagation algorithm (see Sections 5.4 and 5.5), and is designed as explained in Section 7.3, with the smooth and extended S-model (see Sections 7.2 and 6.2) as the prediction flow model and VT-micro [3] as the prediction emission model of the controller. Note that in the S-model, the minimum, floor, and remainder functions that appear respectively in (5.6), (5.9), and (5.10), will be smoothened using (7.5), (7.8), and (7.12). These two models are integrated using the mesoscopic framework proposed in Chapter 6. The control sampling time is equal to the cycle time of the traffic lights in the traffic network (60 s), and the length of the prediction horizon is 7 time steps (note that according to the tuning rules proposed by Hegyi et al. [58], the horizon length is selected such that it is longer than the time needed for vehicles to cross the network).

MPC controller with pattern search

For evaluation of the CPU time, we compare the smooth MPC controller with a nonsmooth MPC controller. In general, several nonsmooth optimization algorithms exist, among which pattern search and genetic algorithm are the most frequently used algorithms. We noticed from several experiments that the performance and the computation speed for pattern search are more satisfactory than those of the genetic algorithm. Hence, we used pattern search for the case study that is implemented in the Global Optimization Toolbox of MATLAB, version R2015B.

The experiments were run on a PC with an Intel Xeon Quad core E5-1620 V3 CPU with a clock speed of 3.5 GHz.

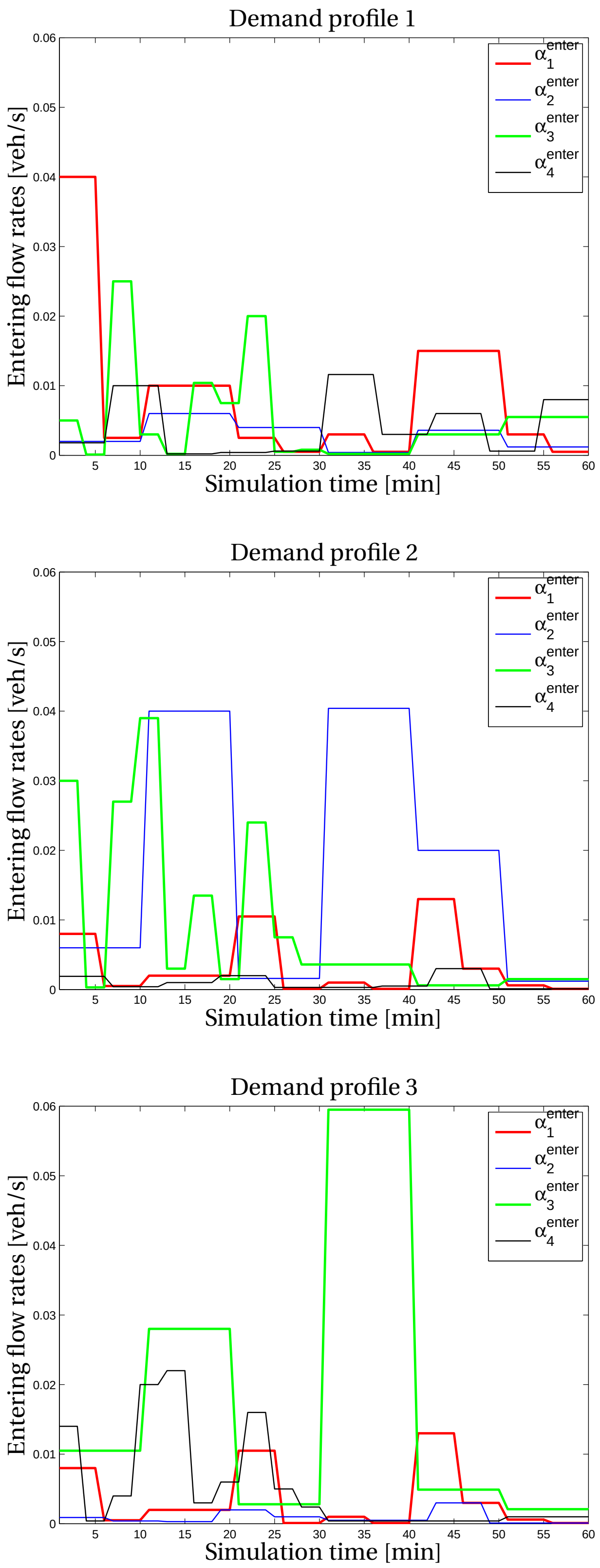


Figure 7.8: Different demand profiles used for 3 simulations cases.

7.4.3 Results and discussion

Since the MPC optimization problem may in general be a nonconvex one, in order to prevent the gradient-based approach from giving a solution that is only locally optimal and that may give a much worse performance than the global optimal value, we first ran a set of offline experiments (see [129]). The aim was to determine how selection of different starting points may play a role in getting a good approximation for the global optimum, and that what is the best choice for the selection of the starting points. These points have been selected in a structured way, i.e., we have three sets of deterministic starting points and two sets of random ones. The deterministic starting points include the shifted suboptimal solution of the previous control time step, the average of the shifted suboptimal solutions determined in the two previous control time steps, and the average of all the shifted suboptimal solutions found in the previous control time steps. The random starting points are uniformly distributed over the feasible region. The results of the offline optimization did not show a noticeable difference in the overall performance of the controlled system for the different choices of the starting points. Hence, for the online application of the optimization-based control approaches in this chapter, we used one set of starting points at every control time step, in particular, the shifted suboptimal solution of the previous control time step. For pattern search, we used a similar approach.

Tables 7.4-7.8 show the resulting value of the objective function, the total time spent, and the total emissions of CO, HC, $\mathrm{NO_x}$ for the no-control case, and for the state-feedback, optimal fixed-time, and smooth and nonsmooth MPC controllers applied to the urban traffic network illustrated in Figure 6.13. We have considered the following values for the parameters given in (7.15): $w_T = 0.3$, $w_{E_{\mathrm{CO}}} = w_{E_{\mathrm{HC}}} = w_{E_{\mathrm{NO_x}}} = 0.2$, $T^{\mathrm{t}} = 10^5$ [s], $E^{\mathrm{t}}_{\mathrm{CO}} = E^{\mathrm{t}}_{\mathrm{HC}} = E^{\mathrm{t}}_{\mathrm{NO_x}} = 1$ [kg], $w_v = 0$. These results show that compared with the no-control case, the overall performance of the system is improved significantly with the state-feedback and optimal feedback controllers. The reason that the fixed-time traffic signal setting performs much worse than the state-feedback controller for Demand profile 3, is the irregular pattern of this demand profile compared with the other two profiles (see Figure 7.8). A constant setting is of course not always expected to be the best choice for both a very high and a very low demand, which occurs in Demand profile 3 for origin 3. However, a state-feedback controller that considers the queue lengths can adapt its traffic signal setting w.r.t. the current traffic state, and hence performs better than the fixed-time controller.

From Table 7.7, the improvement of the system's performance for the smooth MPC controller is higher than for the other controllers w.r.t. the no-control case. Therefore, we can conclude that the online smooth MPC approach for the given case study is highly beneficial compared with the other given controllers.

Next, we compare both the computation time and the performance of the smooth MPC controller that uses RProp with those of a nonsmooth MPC controller that uses pattern search. The corresponding results showing the performance of pattern search are given in Table 7.8 (note that the average CPU time per control time step is indeed the total CPU time for all iterations divided by the total number of control time steps). We see that for all the given demand profiles, the CPU time for the MPC controller with RProp is 12-45 times less than the

Table 7.4: Value of the objective function, total time spent, and total emissions over the entire simulation period for the no-control case

	Objective function	TTS [veh.s]	$\mathrm{TE}_{\mathrm{CO}}$ [kg]	$\mathrm{TE}_{\mathrm{HC}}$ [kg]	$\mathrm{TE}_{\mathrm{NO_x}}$ [kg]
Demand profile 1	214.2	$2.9016 \cdot 10^7$	540.2	34.2	61.4
Demand profile 2	91.0	$1.2279 \cdot 10^7$	230.3	14.6	26.0
Demand profile 3	61.8	$8.3106 \cdot 10^6$	156.8	9.9	17.7

Table 7.5: Value of the objective function, total time spent, and total emissions over the entire simulation period for the state-feedback controller

	Objective function	TTS [veh.s]	$\mathrm{TE}_{\mathrm{CO}}$ [kg]	$\mathrm{TE}_{\mathrm{HC}}$ [kg]	$\mathrm{TE}_{\mathrm{NO_x}}$ [kg]
Demand profile 1	4.7	$6.2404 \cdot 10^5$	12.0	0.8	1.2
Demand profile 2	5.3	$7.0179 \cdot 10^5$	13.6	0.9	1.4
Demand profile 3	6.2	$8.1954 \cdot 10^5$	16.0	1.0	1.7

Table 7.6: Value of the objective function, total time spent, and total emissions over the entire simulation period for the optimal fixed-time controller

	Objective function	TTS [veh.s]	$\mathrm{TE}_{\mathrm{CO}}$ [kg]	$\mathrm{TE}_{\mathrm{HC}}$ [kg]	$\mathrm{TE}_{\mathrm{NO_x}}$ [kg]
Demand profile 1	4.9	$6.2580 \cdot 10^5$	12.9	0.8	1.3
Demand profile 2	1.2	$1.2192 \cdot 10^5$	3.6	0.3	0.3
Demand profile 3	54.9	$7.4177 \cdot 10^5$	138.8	8.8	15.7

Table 7.7: Value of the average CPU time, objective function, total time spent, and total emissions over the entire simulation period for the smooth MPC controller with RProp

	Average CPU time per control time step [s]	Objective function	TTS [veh.s]	$\mathrm{TE}_{\mathrm{CO}}$ [kg]	$\mathrm{TE}_{\mathrm{HC}}$ [kg]	$\mathrm{TE}_{\mathrm{NO_x}}$ [kg]
Demand profile 1	1697	4.4	$6.4377 \cdot 10^5$	10.7	0.7	1.1
Demand profile 2	5466	0.8	$1.0326 \cdot 10^5$	2.0	0.1	0.2
Demand profile 3	13241	3.3	$4.4603 \cdot 10^5$	8.4	0.5	0.9

Table 7.8: Value of the average CPU time, objective function, total time spent, and total emissions over the entire simulation period for the nonsmooth MPC controller with pattern search

	Average CPU time per control time step [s]	Objective function	TTS [veh.s]	$\mathrm{TE}_{\mathrm{CO}}$ [kg]	$\mathrm{TE}_{\mathrm{HC}}$ [kg]	$\mathrm{TE}_{\mathrm{NO_x}}$ [kg]
Demand profile 1	76479	14.0	$1.8726 \cdot 10^6$	35.8	2.3	3.9
Demand profile 2	152631	7.0	$9.4632 \cdot 10^5$	17.9	1.2	1.9
Demand profile 3	151867	28.2	$3.7954 \cdot 10^6$	71.6	4.5	8.0

Table 7.9: Percentage of improvement of the objective value w.r.t. the state-feedback controller: $\frac{J_{\text{state-feedback}}-J_{\text{MPC}}}{J_{\text{state-feedback}}}$ *(+: improved, -: became worse)*

	MPC with RProp	MPC with pattern search
Demand profile 1	+6%	-66%
Demand profile 2	+84%	-32%
Demand profile 3	+47%	-354%

Table 7.10: Percentage of improvement of the objective value w.r.t. the optimal fixed-time controller: $\frac{J_{\text{fixed-time}}-J_{\text{MPC}}}{J_{\text{fixed-time}}}$ *(+: improved, -: became worse)*

	MPC with RProp	MPC with pattern search
Demand profile 1	+10%	-185%
Demand profile 2	+34%	-483%
Demand profile 3	+94%	+48.6%

CPU time for pattern search. This indicates that the gradient-based optimization approach performs significantly better than pattern search considering the computation speed. Moreover, the realized value of the objective function for a 1-hour simulation for the gradient-based optimization approach is almost 31.5% of the realized value of the objective function for pattern search for ‘Demand profile 1’, 11% for ‘Demand profile 2’, and 11.5% for ‘Demand profile 3’ (compare Tables 7.7 and 7.8). In addition to that, by looking at the realized values of the total time spent and total emissions of CO, HC, and NO_x individually, we see that all these quantities are prominently smaller for the gradient-based optimization compared with pattern search. Note that for pattern search the optimization procedure takes long and hence, sometimes the maximum number of iterations is reached before the optimum values are found.

We have listed the percentages of improvement for the MPC controllers w.r.t. the state-feedback and optimal fixed-time controllers in Tables 7.9 and 7.10. These results show that for the traffic network shown in Figure 6.13, the proposed smooth MPC controller that uses RProp is the most efficient controller among the other given controllers. Additionally, from the smooth and nonsmooth MPC controllers, both the performance and the CPU time of the smooth one are much better.

7.5 Conclusions and future work

In this chapter, we have introduced general smoothening methods for nonsmooth mathematical models. Therefore, when the resulting models are used together with an optimization-based controller, e.g., an MPC controller, the resulting smooth optimization problem can be solved using available efficient gradient-based optimization approaches. Then we have formulated a highly efficient smooth model-predictive controller for urban traffic networks. The proposed MPC controller has been made smooth using the proposed approaches in this chapter. The aim of the controller is to find a balanced trade-off between reduction of the total time spent by the vehicles and the total emissions. We have applied a gradient-based optimization approach based on the resilient back-propagation (RProp) algorithm to find the suboptimal solution of the MPC controller.

The simulation results for the given case study have shown that the smooth MPC controller improves the performance of the network significantly w.r.t. the no-control case, and state-feedback, optimal fixed-time, and nonsmooth MPC-based controlled cases. Moreover, the smooth (gradient-based) optimization method is much faster than the nonsmooth one (the CPU time of the smooth method is 12-45 times less than the CPU time of the nonsmooth method). Note that although the resulting CPU time for the smooth optimization-based controller shows a significant decrease compared to that of the nonsmooth optimization-based controller, it is currently not yet suited for real-time control. Therefore, further improvements in computation speed of the smooth optimization-based controller should be obtained. This improvement may be achieved by using dedicated software and hardware, distributed MPC, fast MPC techniques, parameterized control approaches, etc.

Topics for future work include implementing the smooth optimization approach to a large-scale network considering a multi-level and/or multi-agent control architecture. Additionally, to make the MPC controller faster, we can consider parameterized control laws. We also suggest an extensive validation of the proposed control approach for various networks with real-life datasets.

Chapter 8

MPC with endpoint penalties for urban traffic control

8.1 Introduction

In this chapter, we propose to include endpoint penalties in formulating the optimization problem of the MPC controller designed in Chapter 7 for an urban traffic network. Endpoint penalties can be added to the formulation of a finite-horizon MPC optimization problem to approximate the solution of an infinite-horizon MPC problem, without resulting in additional computational burdens. Endpoint penalties provide a set of equations that are extracted based on some simplifying assumptions and a simple static model of the controlled system. This set of equations is used to approximate the cost function of the optimization problem beyond the prediction horizon.

Endpoint penalties have been used before in formulating the optimization problem of MPC controllers for stability guarantees, and in order to assure that the states beyond the prediction interval adopt a desired value (see, e.g., [14, 28, 81, 87, 118, 136]). In addition to stability guarantee reasons, endpoint penalties may also be added to the cost function of an MPC controller with the aim of improving the performance of the controller. For example Liu et al. [100] consider endpoint penalties for freeway traffic control and show via simulations that the performance of the resulting MPC controller improves compared with an MPC controller that does not consider these endpoint penalties.

Contributions and organization of the chapter

The main contributions of the chapter include:

- We propose an MPC controller for urban traffic networks that considers the effect of endpoint penalties in its optimization problem. Hence, in addition to the total time spent and total emissions within one prediction window (the cumulative stage cost), we consider the approximate total time spent and the approximate total emissions of

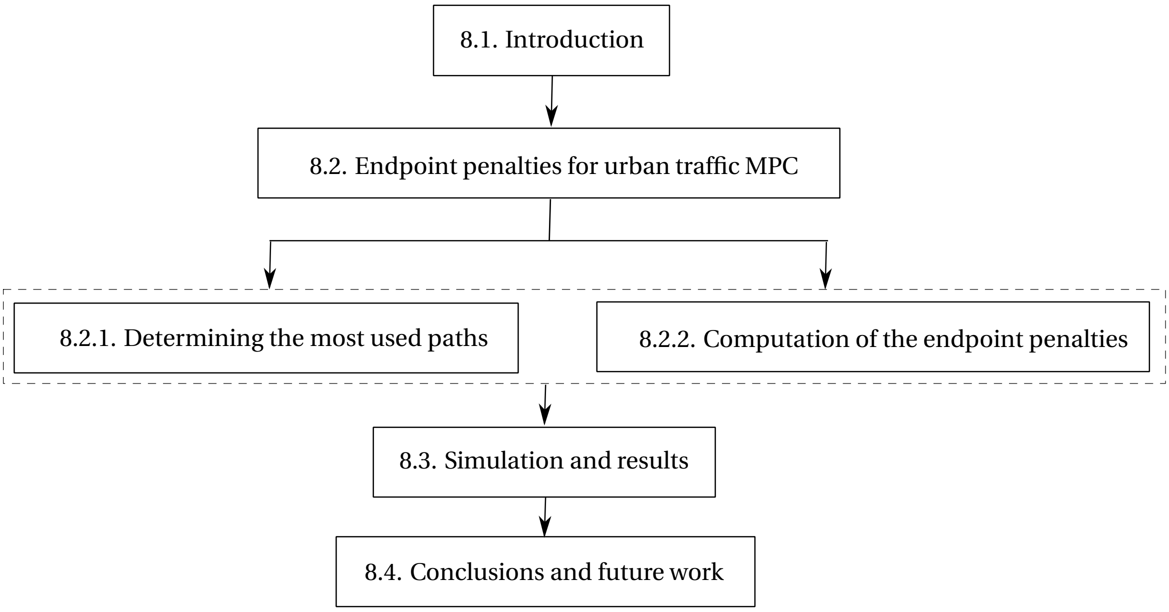

8.1. Introduction
8.2. Endpoint penalties for urban traffic MPC
8.2.1. Determining the most used paths
8.2.2. Computation of the endpoint penalties
8.3. Simulation and results
8.4. Conclusions and future work

Figure 8.1: Road map of Chapter 8.

those vehicles that are already in the link at the end of the current prediction interval until they leave the network in formulating the optimization problem.

- In order to estimate the endpoint penalties, we introduce a novel and efficient approach that is based on a K shortest path algorithm.

The rest of the chapter is organized as follows. In Section 8.2 we propose an approach for estimating the endpoint penalties (i.e., the total time spent and the total emissions of the vehicles that are already in the traffic network at the end of the prediction interval until they leave the network) for an urban traffic network. We first propose a method that is based on a K shortest-path algorithm in order to find the most used paths by the vehicles. Then we develop the formulas for computation of the endpoint penalties. In Section 8.3 we perform a simulation considering the endpoint penalties and the total time spent of the vehicles as the cost function of the MPC controller. Finally, Section 8.4 concludes the chapters and gives some topics for future work. The road map of Chapter 8 is illustrated in Figure 8.1. Moreover, Table 8.1 gives the frequently used mathematical notations of this chapter.

This chapter of the thesis is based on [67–70].

8.2 Endpoint penalties for urban traffic MPC

Estimation of the endpoint penalties for the MPC optimization problem of the urban traffic network described in this chapter is based on the following two assumptions:

Assumption 1. In estimating the endpoint penalties for the urban traffic network, we take

Table 8.1: Frequently used mathematical notations for Chapter 8.

$P_j(d)$	a path between node d and the virtual node v in the graph that corresponds to the traffic network
N_d^{path}	the number of all possible paths beginning at node d and ending at node v (excluding the cyclic paths)
$d_i\left(P_j(d)\right)$	the downstream node of the i^{th} link within path $P_j(d)$
$\bar{n}_j$	the number of real links in path $P_j(d)$
π_u^{veh}	the percentage of vehicles that a path receives
$\chi_u^{\mathrm{cum}}\left(P_j(d)\right)$	the cumulative cost of path $P_j(d)$
$\chi_{d_i(P_j(d)),d_{i+1}(P_j(d))}\left(P_j(d)\right)$	the cost of link $\left(d_i\left(P_j(d)\right), d_{i+1}\left(P_j(d)\right)\right)$ corresponding to path $P_j(d)$ in the recast shortest-path problem
$\zeta_u\left(P_j(d)\right)$	the percentage of vehicles on link (u,d) that intend to leave the traffic network via path $P_j(d)$
$\mathrm{TTS}^{\mathrm{endpoint}}(k_{\mathrm{c}})$	the total time spent for those vehicles that are already in the traffic network at the end of the prediction interval, from time instant $\left(k_{\mathrm{c}}+N_{\mathrm{p}}\right)T_{\mathrm{c}}$ until they leave the traffic network
$\mathrm{TE}_p^{\mathrm{endpoint}}(k_{\mathrm{c}})$	the total emissions of the pollutant $p \in \mathbb{P}$ for those vehicles that are already in the traffic network at the end of the prediction interval, from time instant $\left(k_{\mathrm{c}}+N_{\mathrm{p}}\right)T_{\mathrm{c}}$ until they leave the traffic network
$\mathrm{TTS}^{\mathrm{path}}\left(P_j(d)\right)$	the total time spent by the vehicles that are traveling via path $P_j(d)$
$\mathrm{TE}_p^{\mathrm{path}}\left(P_j(d)\right)$	the total emissions of the pollutant p by the vehicles that are traveling via path $P_j(d)$
$\mathcal{N}$	the set of all nodes in the graph that corresponds to the traffic network
$\mathcal{U}_d$	the set of all upstream nodes of node d in the graph that corresponds to the traffic network

into account only those vehicles that are already in the traffic network at the end of the prediction interval until they all leave the traffic network.

Assumption 2. We assume that the speed of the vehicles is constant beyond the prediction interval, and is equal to their own speed or the average speed of all the vehicles on the same link at the end of the prediction interval (i.e., at time instant $\left(k_{\mathrm{c}}+N_{\mathrm{p}}\right)T_{\mathrm{c}}$). Note that this is different from considering fixed states for the traffic network (reminding that the states of the traffic network include the total number of vehicles in the links and the number of vehicles waiting in the queues).

Now, suppose that for the time interval beyond the prediction interval, i.e., from time instant $\left(k_{\mathrm{c}}+N_{\mathrm{p}}\right)T_{\mathrm{c}}$ until the vehicles that are in the traffic network at the end of the prediction interval leave the traffic network, we use a static model of the traffic network that is destination-independent. Note that in a static mathematical model the equilibrium/steady-state condition of the modeled system is considered. We should highlight it here that by static we refer

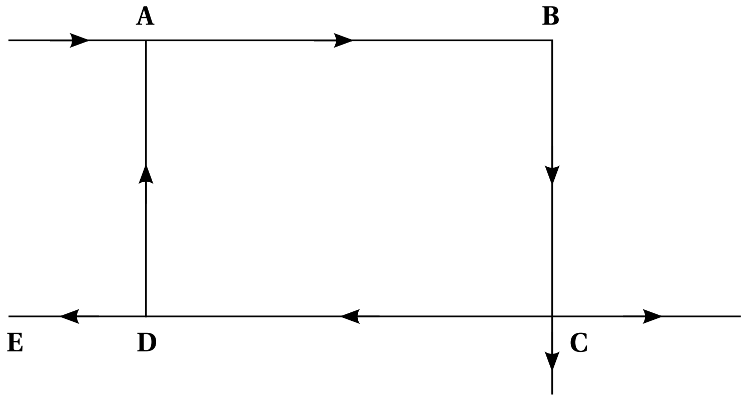


Figure 8.2: A grid-shaped network that creates an infinite number of path choices (because of the cyclic path A-B-C-D-A) for a vehicle at node A to exit via node E.

to the steady dynamics of the traffic network.

In order to estimate the endpoint penalties for the urban traffic network, we should first allocate a path to each of the existing vehicles in the network at time instant $\left(k_\mathrm{c} + N_\mathrm{p}\right) T_\mathrm{c}$ (the main reason for allocating these paths becomes more clear in Section 8.2.2, where (8.12) and (8.13) are developed for computing the endpoint penalties). This path should lead the vehicle to one of the exits of the traffic network. Based on the geometry and the traffic rules of the network, there might be multiple path choices per vehicle. However, if we consider all the possible paths to the exits of the traffic network, the number of path choices for some vehicles may become infinite. For instance, for a grid-shaped network, some vehicles might move within cyclic paths forever and this makes the number of possible paths to become infinity (see Figure 8.2, for which a vehicle that is located at node A may turn several (up to an infinite number of) times within the cyclic path A-B-C-D-A before it exits the network via node E). To prevent this situation, for the vehicles that are located on a link (u, d) at time instant $\left(k_\mathrm{c} + N_\mathrm{p}\right) T_\mathrm{c}$, we determine a limited number $K_{u,d}$ of the most likely used paths from node d to the exit nodes of the network by those vehicles that are located on link (u, d) at the end of the prediction interval. The number $K_{u,d}$ is indeed a design parameter for the endpoint penalty approach that should be determined beforehand. After $K_{u,d}$ is known, the paths can be found using an existing shortest-path algorithm, e.g., Yen's algorithm (see [16]).

8.2.1 Determining the mostly used paths in the traffic network

In the setup of a shortest-path algorithm, each link that connects two consecutive nodes corresponds to a predefined value called the link cost. A path between a starting node s and an end node e is defined as a set of consecutive links, where s is the upstream node of the first link in the path and e is the downstream node of the last link in the path. A shortest-path algorithm solves a problem known as the shortest-path problem, i.e., it selects one path among all the possible paths that connect s and e, where for the selected path the cumulative cost for all the involved links is the minimum compared with the cumulative cost of the links within the other paths between these two nodes. Next we explain how our problem can be translated as a shortest-path problem.

Although for the illustration purposes and to clarify the discussions, we use the traffic network shown in Figure 8.3, the proposed approach in this chapter can be used for various traffic networks modeled as a collection of nodes and links. Since a shortest-path problem is usually formulated for a single end node, we first consider a single virtual end node (indicated by “v” in Figure 8.3) and connect all the exit nodes of the traffic network by virtual links to node “v”. Note that by virtual links we mean those links that do not really exist in the traffic network and are added to the graph representation of the network to formulate the endpoint penalties. These virtual links are represented by dashed lines in Figure 8.3. By using the term real links, we exclude the virtual links and consider only the links that really exist in the graph representation of the traffic network. Real links are illustrated by solid lines in Figure 8.3).

Finally, we obtain a point-to-point representation of our problem, which suits the formulation of a shortest-path problem. Then, a path $P_j(d)$ between node d and node v, with $j \in \left\{1,\dots,N_d^{\text{path}}\right\}$ and N_d^{path} the number of all possible paths beginning at node d and ending at node v (excluding the cyclic paths), is defined by

$$P_j(d) = \left\{\left(d, d_1\left(P_j(d)\right)\right), \left(d_1\left(P_j(d)\right), d_2\left(P_j(d)\right)\right), \dots, \left(d_{\bar{n}_j}\left(P_j(d)\right), \text{v}\right)\right\}, \tag{8.1}$$

with $d_i\left(P_j(d)\right)$ for $i \in \{1,\dots,\bar{n}_j\}$ the downstream node of the i^{th} link within the path $P_j(d)$, $d_0\left(P_j(d)\right) := d$, and $\bar{n}_j$ the number of real links in the path. The links colored in red in Figure 8.3 demonstrate a path $P_j(d)$ from node d to node v.

We should also redefine some of the concepts for the recast problem. To define a cost value for each link, we make use of the turning rates, which are also used in the S-model explained in Section 5.3.1. Note that all the virtual links are assumed to have a cost value equal to zero. Therefore, finding the $K_{u,d}$ shortest paths from node d to the virtual node v for those vehicles that are on link (u,d) is equivalent to finding the $K_{u,d}$ shortest paths from node d to the exit nodes of the traffic network. We are looking for a path that receives the largest percentage of vehicles, i.e., a path that has the maximum π_u^{veh} defined by

$$\pi_u^{\text{veh}}\left(P_j(d)\right) = \beta^{\text{static}}_{u,d,d_1\left(P_j(d)\right)}(k_\text{c}) \prod_{i=0}^{\bar{n}_j-2} \beta^{\text{static}}_{d_i\left(P_j(d)\right),d_{i+1}\left(P_j(d)\right),d_{i+2}\left(P_j(d)\right)}(k_\text{c}), \quad j \in \left\{1,\dots,N_d^{\text{path}}\right\}, \tag{8.2}$$

where the turning rate of the last link in the path towards the corresponding consecutive virtual link is 1, and the superscript “static” refers to the static situation. Hence, for two connecting links (d_i, d_{i+1}) and (d_{i+1}, d_{i+2}), the turning rate $\beta^{\text{static}}_{d_i,d_{i+1},d_{i+2}}(k_\text{c})$ is given by

$$\beta^{\text{static}}_{d_i,d_{i+1},d_{i+2}}(k_\text{c}) = \beta_{d_i,d_{i+1},d_{i+2}}\left(k_\text{c} + N_\text{p}\right). \tag{8.3}$$

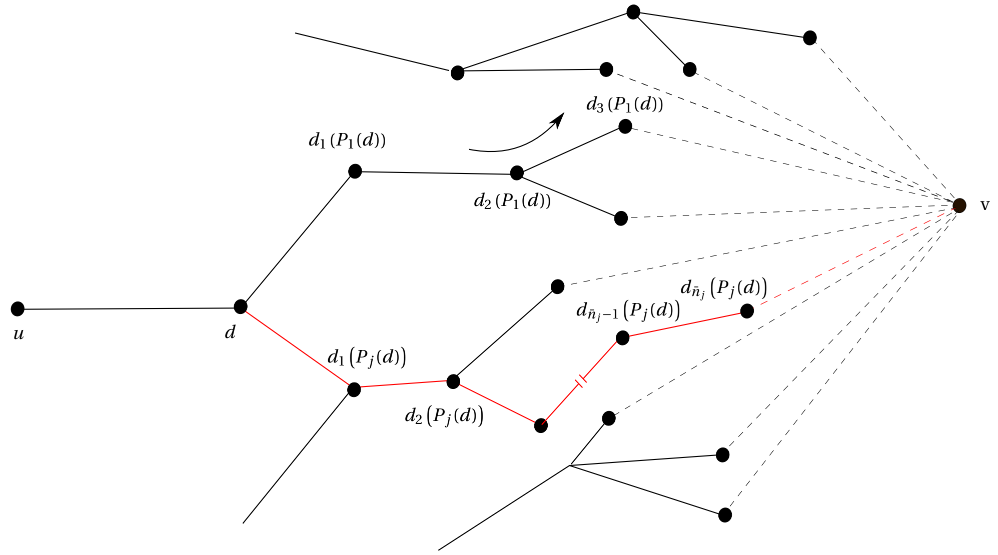


Figure 8.3: Considering a point-to-point problem to find the $K_{u,d}$ most likely used paths for the vehicles on link (u,d) in a traffic network for the endpoint penalties corresponding to the prediction interval $\left[k_\mathrm{c}T_\mathrm{c},\left(k_\mathrm{c}+N_\mathrm{p}\right)T_\mathrm{c}\right)$ (the broken red line indicates a set of intermediate links that have not been shown).

Note that the following equality always holds (assuming that we have already excluded those paths for which π_u^veh is zero):

$$\operatorname*{arg\,max}_{j\in\left\{1,\dots,N_d^\mathrm{path}\right\}}\left(\pi_u^\mathrm{veh}\left(P_j(d)\right)\right)=\operatorname*{arg\,max}_{j\in\left\{1,\dots,N_d^\mathrm{path}\right\}}\left(\log\left(\pi_u^\mathrm{veh}\left(P_j(d)\right)\right)\right). \tag{8.4}$$

Moreover, we have

$$\max_{j\in\left\{1,\dots,N_d^\mathrm{path}\right\}}\left(\log\left(\pi_u^\mathrm{veh}\left(P_j(d)\right)\right)\right)=\min_{j\in\left\{1,\dots,N_d^\mathrm{path}\right\}}\left(-\log\left(\pi_u^\mathrm{veh}\left(P_j(d)\right)\right)\right). \tag{8.5}$$

Therefore, considering (8.2) our problem is equivalent to the problem of finding a path among the j paths starting at node d, with $j\in\left\{1,\dots,N_d^\mathrm{path}\right\}$, for the vehicles that are on link (u,d) that has the minimum value of

$$\chi_u^\mathrm{cum}\left(P_j(d)\right)=-\log\left(\beta^\mathrm{static}_{u,d,d_1\left(P_j(d)\right)}\left(k_\mathrm{c}\right)\prod_{i=0}^{\bar{n}_j-2}\beta^\mathrm{static}_{d_i\left(P_j(d)\right),d_{i+1}\left(P_j(d)\right),d_{i+2}\left(P_j(d)\right)}\left(k_\mathrm{c}\right)\right), \tag{8.6}$$

where from the objective of a shortest-path problem (i.e., finding a path that has the minimum cumulative cost value for the involving links), we see that $\chi_u^\mathrm{cum}\left(P_j(d)\right)$ can indeed represent the cumulative cost of path $P_j(d)$ for our problem. Additionally, for $j\in\left\{1,\dots,N_d^\mathrm{path}\right\}$, we have

$$\begin{aligned}&-\log\left(\beta^{\text{static}}_{u,d,d_1(P_j(d))}(k_{\text{c}})\prod_{i=0}^{\bar{n}_j-2}\beta^{\text{static}}_{d_i(P_j(d)),d_{i+1}(P_j(d)),d_{i+2}(P_j(d))}(k_{\text{c}})\right)=\\&-\log\left(\beta^{\text{static}}_{u,d,d_1(P_j(d))}(k_{\text{c}})\right)+\sum_{i=0}^{\bar{n}_j-2}\left(-\log\left(\beta^{\text{static}}_{d_i(P_j(d)),d_{i+1}(P_j(d)),d_{i+2}(P_j(d))}(k_{\text{c}})\right)\right).\end{aligned}\tag{8.7}$$

Suppose that we denote the cost of link $\left(d_i\left(P_j\left(d\right)\right),d_{i+1}\left(P_j\left(d\right)\right)\right)$ in path $P_j\left(d\right)$ for $i=0$ by $\chi^{0,u}_{d,d_1(P_j(d))}\left(P_j(d)\right)$, and for $i\in\{1,\ldots,\bar{n}_j-1\}$ by $\chi_{d_i(P_j(d)),d_{i+1}(P_j(d))}\left(P_j\left(d\right)\right)$. Then from (8.6) and (8.7), we can write

$$\begin{aligned}&\chi^{0,u}_{d,d_1(P_j(d))}\left(P_j(d)\right)+\sum_{i=0}^{\bar{n}_j-2}\chi_{di+1(P_j(d)),d_{i+2}(P_j(d))}\left(P_j(d)\right)=\\&-\log\left(\beta^{\text{static}}_{u,d,d_1(P_j(d))}(k_{\text{c}})\right)+\sum_{i=0}^{\bar{n}_j-2}\left(-\log\left(\beta^{\text{static}}_{d_i(P_j(d)),d_{i+1}(P_j(d)),d_{i+2}(P_j(d))}(k_{\text{c}})\right)\right),\end{aligned}\tag{8.8}$$

which implies that the cost of a link in the network that belongs to the j^{th} path between the downstream node of that link and the virtual node v, for $j\in\left\{1,\ldots,N^{\text{path}}_d\right\}$, can be defined by

$$\chi^{0,u}_{d,d_1(P_j(d))}\left(P_j(d)\right)=-\log\left(\beta^{\text{static}}_{u,d,d_1(P_j(d))}(k_{\text{c}})\right),\tag{8.9}$$

$$\chi_{d_{i+1}(P_j(d)),d_{i+2}(P_j(d))}\left(P_j(d)\right)=-\log\left(\beta^{\text{static}}_{d_i(P_j(d)),d_{i+1}(P_j(d)),d_{i+2}(P_j(d))}(k_{\text{c}})\right),\quad i\in\{0,\ldots,\bar{n}_j-2\}.\tag{8.10}$$

Note that the cost of the virtual links in the expanded graph is also zero. The definitions given by (8.9) and (8.10) are legitimate definitions for the cost, since we know that the turning rates are less than or equal to one and hence the evaluated functions defined by (8.9) and (8.10) are always greater than or equal to zero. Moreover, as we can see from (8.9) and (8.10), the cost of a link for a given path is defined as a function of the turning rate from the upstream link of that link to the link for the given path. Hence, a link may adopt different cost values in different paths. A shortest-path algorithm, however, accepts only unique cost values per link. Therefore, before we are able to use such an algorithm for a link (u,d), as an intermediate stage we should construct a new graph, called the "expanded graph", that includes all the paths that exist between node d and the virtual node v in the original network for those vehicles that are on link (u,d). In this expanded graph, for each link from the original network, we may consider multiple links (where the number of these links equals the number of the paths in the original network from node d to node v to which the given link belongs to). Therefore, in the expanded, each link can belong to one and only one path. Depending on the path that the link belongs to in the expanded graph, we assign the link the cost value that corresponds to that path and that is computed by either (8.9) or (8.10).

For example, Figure 8.5 illustrates such an expanded graph for the subgraph shown in Figure 8.4. In Figure 8.4, all the paths in between node d and node v for those vehicles

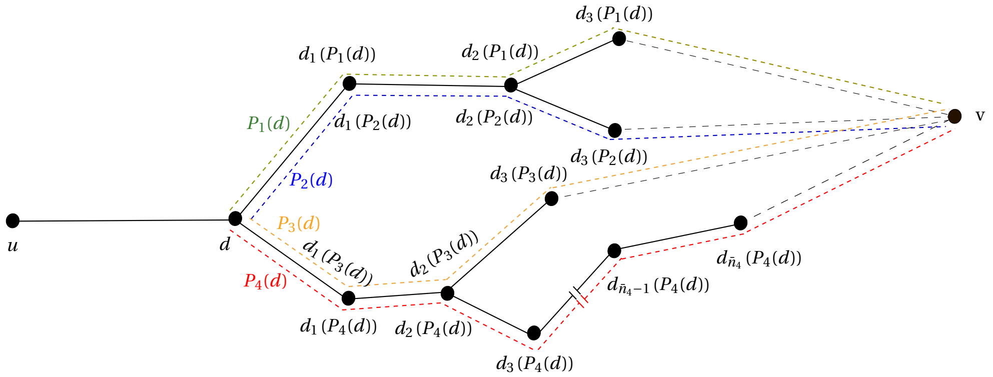


Figure 8.4: The subgraph that includes all the paths in between node d and the virtual node v for the vehicles that are on link (u,d) (there are four paths that are shown by the green, blue, orange, and red dashed curves).

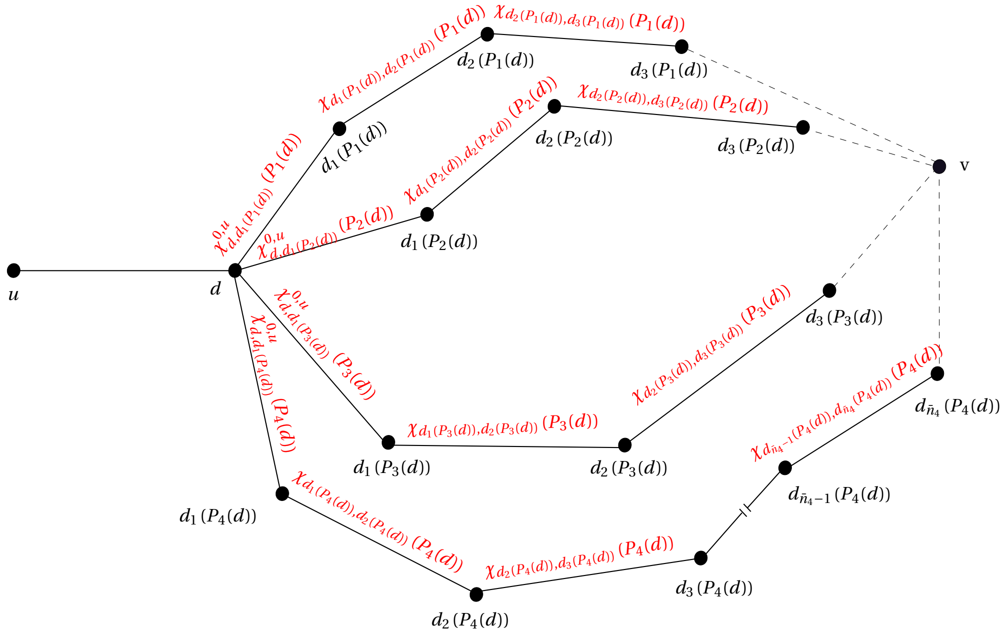

$\chi_{d_1(P_1(d)),d_2(P_1(d))}(P_1(d))$
$\chi_{d_2(P_1(d)),d_3(P_1(d))}(P_1(d))$
$d_2(P_1(d))$
$d_3(P_1(d))$
$\chi_{d_2(P_2(d)),d_3(P_2(d))}(P_2(d))$
$d_1(P_1(d))$
$d_2(P_2(d))$
$d_3(P_2(d))$
v
$\chi_{d_1(P_2(d)),d_2(P_2(d))}(P_2(d))$
$\chi^{0,u}_{d,d_1(P_1(d))}(P_1(d))$
$\chi^{0,u}_{d,d_1(P_2(d))}(P_2(d))$
$d_1(P_2(d))$
u
d
$\chi^{0,u}_{d,d_1(P_3(d))}(P_3(d))$
$\chi^{0,u}_{d,d_1(P_4(d))}(P_4(d))$
$d_3(P_3(d))$
$\chi_{d_2(P_3(d)),d_3(P_3(d))}(P_3(d))$
$d_{\bar{n}_4}(P_4(d))$
$\chi_{d_1(P_3(d)),d_2(P_3(d))}(P_3(d))$
$\chi_{d_{\bar{n}_4-1}(P_4(d)),d_{\bar{n}_4}(P_4(d))}(P_4(d))$
$d_1(P_3(d))$
$d_2(P_3(d))$
$d_{\bar{n}_4-1}(P_4(d))$
$d_1(P_4(d))$
$\chi_{d_1(P_4(d)),d_2(P_4(d))}(P_4(d))$
$\chi_{d_2(P_4(d)),d_3(P_4(d))}(P_4(d))$
$d_3(P_4(d))$
$d_2(P_4(d))$

Figure 8.5: The expanded graph corresponding to all the paths that are in between node d and the virtual node v in the original network for the vehicles that are on link (u,d).

that are on link (u,d) are illustrated. For this specific case, four paths exist (shown by the dashed green, blue, orange, and red curves). The expanded graph of this subgraph is illustrated in Figure 8.5, where each link in the original graph that exists in several paths (e.g., link $(d_1(P_1(d)), d_2(P_1(d)))$, which is the same as link $(d_1(P_2(d)), d_2(P_2(d))))$ will yield multiple links in the expanded graph, and each time adopts the cost value that is computed by either (8.9) or (8.10) for the corresponding path.

At this stage, by using one of the existing shortest-path algorithms, we can determine the $K_{u,d}$ shortest paths from node d to the exit nodes of the traffic network for the vehicles on links (u,d). This way, for each subgraph that corresponds to a link (u,d), another subgraph is obtained, which includes only those paths that are determined via the shortest-path algorithm. For instance, Figure 8.6 illustrates the obtained subgraph corresponding to the subgraph shown in Figure 8.4. The solid black links are those links that belong to the $K_{u,d}$ shortest paths determined by the algorithm, while the eliminated links are shown by gray lines. In this figure, we assume that $K_{u,d} = 2$ and paths $P_1(d)$ and $P_4(d)$ have been returned by the shortest-path algorithm.

8.2.2 Computation of the endpoint penalties for the MPC cost function

Since some links are eliminated by the shortest-path algorithm, the static turning rates defined by (8.3) may not be valid anymore. For example, in Figure 8.6 for the turning rates at node $d_2\left(P_4(d)\right)$, we initially have

$$\beta^{\text{static}}_{d_1(P_4(d)),d_2(P_4(d)),d_3(P_4(d))}\left(k_\text{c}\right) + \beta^{\text{static}}_{d_1(P_4(d)),d_2(P_4(d)),d_3(P_3(d))}\left(k_\text{c}\right) = 1,$$

noting that $d_1\left(P_3(d)\right) = d_1\left(P_4(d)\right)$ and $d_2\left(P_3(d)\right) = d_2\left(P_4(d)\right)$. However, as we are going to eliminate link $\left(d_2\left(P_4(d)\right), d_3\left(P_3(d)\right)\right)$ in the recast problem, if $\beta^{\text{static}}_{d_1(P_4(d)),d_2(P_4(d)),d_3(P_3(d))}\left(k_\text{c}\right) \neq 0$, then the fact that the summation of the turning rates at all nodes in the traffic network should be unity is violated. Therefore, we should redefine and reformulate the turning rates for the recast problem. First, for the sake of simplicity in the notations we assume that the numbering of the $K_{u,d}$ returned paths by the shortest-path algorithm is reset. In our notations, we show the newly numbered shortest paths by $P^{\text{s}}_j(d)$. Since in the recast problem, we will consider the travel times and the emissions on paths instead of on individual links, we define the turning rates for different paths. Hence, the turning rate $\zeta_u\left(P^{\text{s}}_j(d)\right)$ at node d is defined as the percentage of vehicles on link (u,d) that intend to leave the traffic network via path $P^{\text{s}}_j(d)$. Hence, $\zeta_u\left(P^{\text{s}}_j(d)\right)$ is formulated by

$$\zeta_u\left(P^{\text{s}}_j(d)\right) = \frac{\pi^{\text{veh}}_u\left(P^{\text{s}}_j(d)\right)}{\sum\limits_{\ell=1}^{K_{u,d}} \pi^{\text{veh}}_u\left(P^{\text{s}}_\ell(d)\right)}. \tag{8.11}$$

Finally, we can compute the endpoint penalties at control time step k_c, i.e., the total time spent $\text{TTS}^{\text{endpoint}}\left(k_\text{c}\right)$ and the total emissions $\text{TE}^{\text{endpoint}}_p\left(k_\text{c}\right)$ of the pollutant $p \in \mathbb{P}$ (where $\mathbb{P}$ includes different pollutants such as CO, HC, NO_x, etc.) for those vehicles that are already in the traffic network at the end of the prediction interval, from time instant $\left(k_\text{c} + N_\text{p}\right) T_\text{c}$ until they leave the traffic network. We have

$$\text{TTS}^{\text{endpoint}}\left(k_\text{c}\right) = \sum_{d\in\mathcal{N}} \sum_{u\in\mathcal{U}_d} n_{u,d}\left(k_\text{c} + N_\text{p}\right)\left(\text{TTS}^{\text{endpoint}}_{u,d}\left(k_\text{c}\right) + \sum_{j=1}^{K_{u,d}}\left(\zeta_u\left(P^{\text{s}}_j(d)\right)\text{TTS}^{\text{path}}\left(P^{\text{s}}_j(d)\right)\right)\right), \tag{8.12}$$

where $\text{TTS}^{\text{endpoint}}_{u,d}\left(k_\text{c}\right)$ is the total time spent on link (u,d) by the vehicles that are on link (u,d)

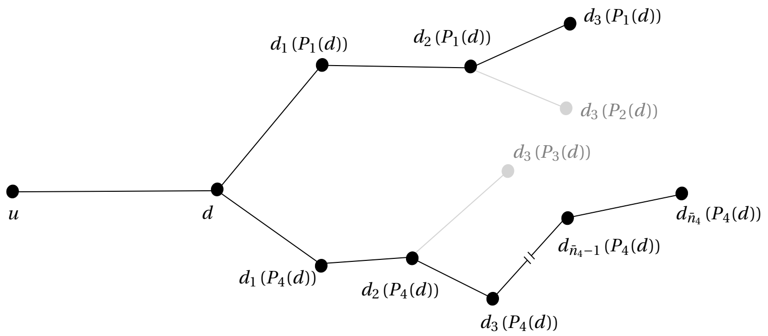


*Figure 8.6: The subgraph for link (u,d) obtained by eliminating those links that are not included in the $K_{u,d}$ shortest paths that connect node d and node*v.

at the end of the prediction interval, $\text{TTS}^{\text{path}}(\cdot)$ is the total time spent by the vehicles that are traveling via a specific path.

Similarly, for $\text{TE}_p^{\text{endpoint}}(k_{\text{c}})$, we have

$$\text{TE}_p^{\text{endpoint}}(k_{\text{c}}) = \sum_{d\in\mathcal{N}}\sum_{u\in\mathcal{U}_d} n_{u,d}\left(k_{\text{c}}+N_{\text{p}}\right)\left(\text{TE}_{p,u,d}^{\text{endpoint}}(k_{\text{c}}) + \sum_{j=1}^{K_{u,d}}\left(\zeta_u\left(P_j^{\text{s}}(d)\right)\text{TE}_p^{\text{path}}\left(P_j^{\text{s}}(d)\right)\right)\right), \tag{8.13}$$

with $\mathcal{N}$ the set of all nodes in the graph that corresponds to the traffic network, $\mathcal{U}_d$ the set of all upstream nodes of node d, $\text{TE}_{p,u,d}^{\text{endpoint}}(k_{\text{c}})$ the total emissions of the pollutant p for the vehicles that are on link (u,d) at the end of the prediction interval while they are traveling on link (u,d), and $\text{TE}_p^{\text{path}}(\cdot)$ the total emissions of the pollutant p for those vehicles that are traveling via a specific path. In order to evaluate (8.12) and (8.13), we assume that all the vehicles that are located on link (d_{i-1},d_i) for $i\in\{0,\dots,\bar{n}_j\}$ and for $j\in\{1,\dots,K_{u,d}\}$ and with $d_{-1}:=u$ at time instant $\left(k_{\text{c}}+N_{\text{p}}\right)T_{\text{c}}$ will keep on traveling in the traffic network with the constant static speed $v_{d_{i-1},d_i}^{\text{static}}(k_{\text{c}})$ (see **Assumption 2**), which can considered to be a weighted average of the individual speeds (based on the flow model that is used) of the vehicles on the link. Then for $j\in\{1,\dots,K_{u,d}\}$ and $d\in\mathcal{N}$, we can write

$$\text{TTS}^{\text{path}}\left(P_j^{\text{s}}(d)\right) = \sum_{i=0}^{\bar{n}_j-1}\frac{C_{d_i\left(P_j^{\text{s}}(d)\right),d_{i+1}\left(P_j^{\text{s}}(d)\right)}\cdot l^{\text{veh}}}{v_{d_i\left(P_j^{\text{s}}(d)\right),d_{i+1}\left(P_j^{\text{s}}(d)\right)}^{\text{static}}(k_{\text{c}})}. \tag{8.14}$$

Moreover, for $u\in\mathcal{U}_d$ we have

$$\text{TTS}_{u,d}^{\text{endpoint}}(k_{\text{c}}) = \frac{C_{u,d}\cdot l^{\text{veh}}}{2v_{u,d}^{\text{static}}(k_{\text{c}})}. \tag{8.15}$$

Note that in order to make the equations simple and to reduce the computational complexity, in (8.12) and (8.13), we have assumed that all the vehicles on link (u,d) are located at the middle of the link at time instant $\left(k_{\text{c}}+N_{\text{p}}\right)T_{\text{c}}$. From Section 6.3, and also considering the

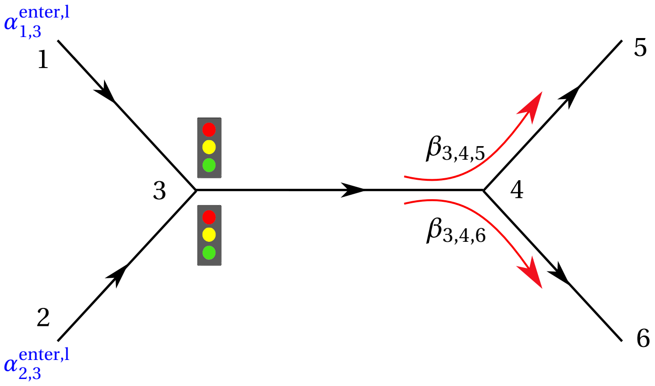

$\alpha_{1,3}^{\text{enter,l}}$
1
2
$\alpha_{2,3}^{\text{enter,l}}$
3
4
5
6
$\beta_{3,4,5}$
$\beta_{3,4,6}$

Figure 8.7: Urban traffic network used for the case study.

assumption of constant speeds, we can write

$$\mathrm{TE}_p^{\text{path}}\left(P_j^{\text{s}}(d)\right) = \sum_{i=1}^{\bar{n}_j-1}\left(\frac{C_{d_i\left(P_j^{\text{s}}(d)\right),d_{i+1}\left(P_j^{\text{s}}(d)\right)}\cdot l^{\text{veh}}}{v^{\text{static}}_{d_i\left(P_j^{\text{s}}(d)\right),d_{i+1}\left(P_j^{\text{s}}(d)\right)}(k_{\text{c}})}\cdot E_p\left(v^{\text{static}}_{d_i\left(P_j^{\text{s}}(d)\right),d_{i+1}\left(P_j^{\text{s}}(d)\right)}(k_{\text{c}}),0\right)\right), \tag{8.16}$$

and

$$\mathrm{TE}_{p,u,d}^{\text{endpoint}}(k_{\text{c}}) = \frac{C_{u,d}\cdot l^{\text{veh}}}{2v_{u,d}^{\text{static}}(k_{\text{c}})}\cdot E_p\left(v_{u,d}^{\text{static}}(k_{\text{c}}),0\right), \tag{8.17}$$

where $E_p(\cdot,\cdot)$ is a microscopic emission model (see, e.g., (5.13)) that computes the instantaneous emissions of the pollutant p based on the instantaneous speed and the instantaneous acceleration of the vehicles.

8.3 Case study

In this section, to examine the effect of considering the endpoint penalties in the formulation of the MPC optimization problem, we consider an urban traffic network (see Figure 8.7) that has 6 nodes and is composed of two entrance links (i.e., links $(1,3)$ and $(2,3)$), two exit links (i.e., links $(4,5)$ and $(4,6)$, and one intermediate link (link $(3,4)$. A traffic light is located at the intersecting node of the two entrance links, i.e., at node 3. The parameters and their corresponding values used for the case study are listed in Table 8.2, where l^{veh} is the average length of the vehicles within the traffic network, $g_{\max}$ and $g_{\min}$ denote the maximum and minimum possible values of the green time for the traffic lights, $C_{u,d}$ with $(u,d)\in\{(1,3),(2,3),(3,4),(4,5),(4,6)\}$ is the capacity of link (u,d), $\mu_{u,d}$ indicates the saturated leaving flow rate of link (u,d), $v_{u,d}^{\text{free}}$ indicates the free-flow speed of the vehicles on link (u,d), c_d is the cycle time of the intersection corresponding to node d, and $\beta_{3,4,5}$ and $\beta_{3,4,6}$ are the turning rates for the vehicles on link $(3,4)$ towards, respectively, nodes 5 and 6. Note that since the network used for the case study is small, network reduction has not been used for this case study.

Table 8.2: Parameters used for the case study.

Parameter	Value
l^{veh} [m]	7
$g_{\max}$ [s]	55
$g_{\min}$ [s]	5
$C_{1,3}$ [veh]	55
$C_{2,3}$ [veh]	50
$C_{3,4}$ [veh]	75
$C_{4,5}$ [veh]	65
$C_{4,6}$ [veh]	50
$\mu_{1,3}$ [veh/s]	1.5
$\mu_{2,3}$ [veh/s]	1.5
$\mu_{3,4}$ [veh/s]	1.5
$\mu_{4,5}$ [veh/s]	1.5
$\mu_{4,6}$ [veh/s]	1.5
$v_{1,3}^{\text{free}}$ [m/s]	12
$v_{2,3}^{\text{free}}$ [m/s]	12
$v_{3,4}^{\text{free}}$ [m/s]	12
$v_{4,5}^{\text{free}}$ [m/s]	12
$v_{4,6}^{\text{free}}$ [m/s]	12
c_3 [s]	60
$\beta_{3,4,5}$ [-]	0.7
$\beta_{3,4,6}$ [-]	0.3

Table 8.3: Initial entering flow rates [veh/s].

	$\alpha_{1,3}^{\text{enter,l}}(0)$	$\alpha_{2,3}^{\text{enter,l}}(0)$
Case 1	1.2	0.5
Case 2	1.0	1.0
Case 3	0.2	1.3

Table 8.4: Effective total time spent for closed-loop simulation [min].

	Without endpoint penalties	With endpoint penalties
TTS [veh.min] for Case 1	1771	1744
TTS [veh.min] for Case 2	2073	2031
TTS [veh.min] for Case 3	1556	1542

We simulate three different traffic scenarios (called case 1, 2, and 3), which include three different pairs of profiles for the entering flow rates from links 1 and 2. Note that we assume that the simulation sampling time and the control sampling time are equal and that the simulation and the control time steps are synchronized. For each of the three cases, the entering flow rates are initially (i.e., at control time step $k_c = 0$) set to those values given in Table 8.3. We consider a sine function with an amplitude equal to $\alpha_i^{\text{enter}}(0)$ and a frequency of $\frac{\pi}{10}$ to model the entering flow rates through time. The aim is to control the two traffic lights using an MPC controller such that the total time spent by the vehicles in the traffic network is minimized. We formulate the MPC optimization problem twice, once without considering the endpoint penalties and once with the endpoint penalties. The total simulation time for each of the three simulations is 75 min. The total time spent by the vehicles in the traffic network during the entire simulation period has been computed for each of the three cases and is presented in Table 8.4. The results show that the total time spent by the vehicles for an MPC controller that considers the effect of the endpoint penalties is smaller for all the three cases compared with an MPC controller that ignores the effect of the endpoint penalties, i.e., by considering the endpoint penalties in the formulation of the MPC optimization problem, the total time spent by the vehicles can be reduced up to 2%.

8.4 Conclusions and future work

In this chapter, we have considered the effect of considering some endpoint penalties in the formulation of the MPC optimization problem of an urban traffic controller on the performance of the control system. Endpoint penalties are implemented in order to approximate an infinite-horizon MPC optimization problem by a finite-horizon MPC optimization problem. We have formulated the endpoint penalties for an arbitrary urban traffic network. The formulas that we have developed in this chapter, are for those vehicles that are in the traffic network at the end of the prediction interval. We estimate the total time spent and the total emissions of these vehicles for the time period from the end of the prediction interval until these vehicles leave the traffic network. A case study has been performed where the total time spent of the vehicles is considered as the cost function that should be minimized by the MPC controller. The results of the case study show that if the effect of the endpoint penalties is taken into account, the total time spent of the vehicles for a 75-min simulation can be reduced by up to 2 % compared with an MPC controller that ignores the effect of the endpoint penalties. Hence, for the given case study in this chapter, the added value of the endpoint penalties is relatively small.

For future work, a general analysis about the endpoint penalties can be done to determine whether they are useful, and in case they are, when and under what conditions their use is recommended. We also propose to perform a case study that considers both the total time spent and the total emissions of the vehicles, and then evaluate more accurately the added value of the endpoint penalties for the proposed formulations. Moreover, alternative approximations and definitions may be used to formulate these endpoint penalties. In order to make a more concrete conclusion about the added value of the endpoint penalties, extensive assessment for different formulations of the endpoint penalties, and for various traffic networks, setups, and demand profiles should be done.

Part III

Predictive model-based fuzzy control: A multi-agent control approach for urban traffic networks

Chapter 9

Background: Type-2 fuzzy sets and agent-based control

9.1 Introduction

In Part III of the thesis, we consider fuzzy logic control (FLC) to design agent-based controllers. We propose to combine the fuzzy control approach with a predictive model-based control approach. The proposed combined control architecture includes two layers, where in the first layer the fuzzy controller performs at every control time step. In the second layer, the predictive model-based controller is located, which is called by the first layer (not necessarily at every control time step) to provide its optimal control decision. This decision is then used to tune the parameters of the fuzzy controller. As the predictive controller can look further in the future, it takes into account the possible effects of the control decision made by the fuzzy controller on the other subsystems in the neighborhood of the fuzzy controller that are not controlled by that fuzzy controller. Additionally, the predictive controller considers the possible effects of the decisions that are made by other controllers within the neighborhood on the future states of the subsystem that is controlled by the fuzzy controller. This way, we can obtain a control system that can be used in a multi-agent coordinative architecture for control of large-scale systems.

The rest of this chapter is organized as follows. In Section 9.2 we discuss multi-agent control systems, which can be used in a decentralized or a distributed framework to control a large-scale system. We briefly indicate the main characteristics of multi-agent control systems. Section 9.3 is about fuzzy logic control (FLC), and in particular, model-based FLC. In Section 9.4 we give the mathematical definitions of type-1 and type-2 fuzzy sets and type-1 and type-2 fuzzy membership functions. Section 9.5 presents an overview of the previous work on using a specific form of type-2 fuzzy membership functions, called interval type-2 fuzzy membership function. Finally, in Section 9.6 an overview of Part III is given. The road map of Chapter 9 is represented in Figure 9.1, and the mathematical notations that are frequently used in this chapter are given in Table 9.1.

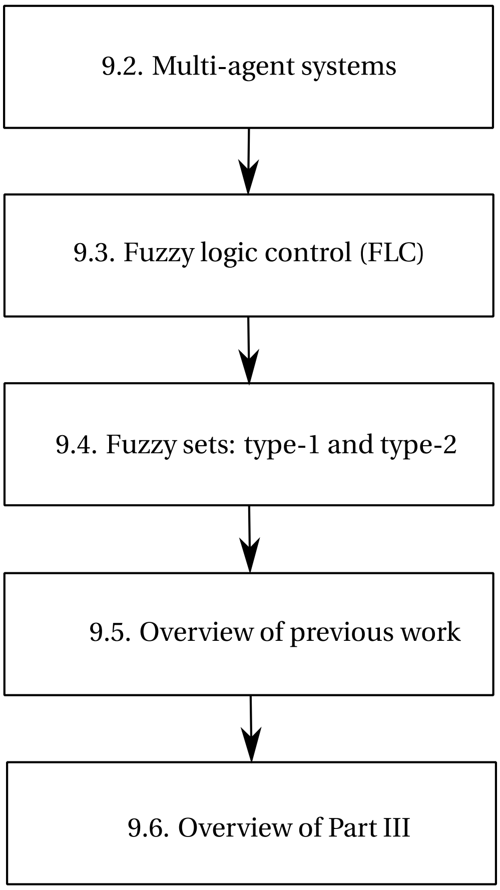

9.2. Multi-agent systems
9.3. Fuzzy logic control (FLC)
9.4. Fuzzy sets: type-1 and type-2
9.5. Overview of previous work
9.6. Overview of Part III

Figure 9.1: Road map of Chapter 9.

Table 9.1: Frequently used mathematical notations for Chapter 9.

$\mathbb{X}$	domain of the independent variable x
F^{t1}	type-1 fuzzy set
$f^{\mathrm{t1}}(\cdot)$	type-1 fuzzy membership function
F^{t2}	type-2 fuzzy set
$f_{\mathrm{p}}^{\mathrm{t1}}(\cdot)$	primary type-1 fuzzy membership function
$f_{\mathrm{s}}^{\mathrm{t1}}(x,\cdot)$	secondary type-1 fuzzy membership function at the point $x \in \mathbb{X}$

9.2 Multi-agent systems

Large-scale control systems can be controlled in several ways, among which are centralized, decentralized, and distributed [107]. In a centralized control architecture, the control problem of the system is formulated and solved as a single problem. In case the control problem is divided into multiple (possibly coupled) smaller problems and each of these smaller problems is solved by one controller, a decentralized or a distributed control architecture arises. In a decentralized control architecture, there is no information exchange among the individual controllers, while in a distributed control architecture the individual controllers may coordinate and/or cooperate before they make their decisions (see [20] for more details).

Although a centralized optimal controller may in theory find the globally optimal control input for the entire controlled system, in practice, when the control problem grows (both in size and in complexity), then solving a single problem may become intractable. Examples of such complicated problems include control of large-scale networks, such as urban and freeway traffic networks, water networks, power networks, etc. Hence, for large-scale systems it is best to divide the controlled system into smaller subsystems and allocate an individual controller to each subsystem. The controllers that are assigned to these subsystems may use different control approaches, including optimization-based control approaches such as optimal control and MPC, intelligent control approaches such as neural network and neuro-fuzzy controllers [155], knowledge-based control such as fuzzy logic control (FLC) [108, 109], etc.

Multi-agent control systems have been developed to tackle complex and/or large-scale problems that are beyond the capabilities of a single agent [76, 110, 125, 144]. Multi-agent control systems have been widely used and proven their efficiency in different fields such as distributed model-predictive control, chilled-water systems, robot teams, industrial problems, data interpretation, power systems, management of business processes, health care management, and transportation systems (see [42, 48, 60, 75, 77, 110, 119–121, 147]).

Division of a centralized control system into multiple distributed interacting controllers, each solving a subproblem extracted from the main control problem, may lead to less computation efforts and a more simple and reliable control system [15, 74, 138]. However, different agents might have incomplete or different views on the controlled system. To deal with this problem, agents can be coordinated and exchange their local information so that the global performance of the system improves. Different coordination policies of agents within a multi-agent architecture include reactive, anticipatory, unilateral, bilateral, selfish, and collaborative coordination (see [23] for more details). Sometimes, individual agents might fail to achieve their predefined control goal or they might face an unexpected situation, for which they have not been prepared [146]. An adaptive scheme for controlling the system may be used to solve such challenges and difficulties [106].

Some of the main advantages of a multi-agent control system according to Sycara [144] include:

- Computational efficiency, which refers to the fact that in a multi-agent system, multiple smaller and simpler problems are solved (in parallel) instead of a complicated central problem (note that when the central problem is divided into multiple coupled smaller problems, some interaction and communication may be needed among the agents, which may bring some computational difficulties itself. Hence, here we assume that the communications are kept minimal and in a high level rather than in a low and detailed level.).

- Reliability, which means that in case an agent fails to achieve its predefined goals, then it can be removed from the multi-agent control architecture causing the least possible disruption for the entire control procedure.

- Extensibility, which means that the number of agents may be increased in response to the dynamic needs of the control system. Additionally, the tasks that are assigned to an agent might alter over time if needed.
- Robustness, which refers to the fact that a multi-agent control system has a higher tolerance towards uncertainties compared with a centralized control system due to the communication and the information exchange among different agents.

Next section briefly discusses fuzzy logic control, including model-based FLC. Note that in Part III, capital letters are used to indicate sets and small letters are used to indicate functions. Moreover, for mathematical notations, regular capital letters are used for sets, bold capital letters are used for matrices, regular small letters are used for scalars, and bold small letters are used for vectors.

9.3 Fuzzy logic control (FLC)

Fuzzy logic and, correspondingly, type-1 fuzzy sets were introduced by Zadeh [157, 159]. Fuzzy logic was later applied to control theory in the context of Fuzzy Logic Control (FLC), which was introduced by Mamdani [108] and Mamdani and Assilian [109]. In FLC, fuzzy logic is implemented to translate human-defined linguistic control rules into automatic strategies for a control system. The main attractions of fuzzy logic include:

- Fuzzy logic converts user-supplied rules formulated in human (sometimes vague) language into their mathematical equivalents.
- Fuzzy logic is capable of handling problems with vague, imprecise, or incomplete information and datasets.
- Fuzzy logic can model various nonlinear functions with different levels of complexity.
- Fuzzy logic-based systems are easily updatable ans maintainable over time.

In fuzzy logic, a truth value within $[0,1]$ is assigned to a logical proposition, while in classical logic, a logical proposition is either “true” or “false”, i.e., it takes a crisp value in $\{0,1\}$. A type-1 fuzzy set F^{t1} is a generalized form of a crisp set. For a type-1 fuzzy set F^{t1}, a type-1 membership function $f^{\mathrm{t1}}: \mathbb{X} \to [0,1]$ is defined, where $\mathbb{X}$ is the domain of the independent variable x. Any realized value of $x \in \mathbb{X}$ belongs to F^{t1} with a certain percentage (i.e., a value in $[0,1]$) called the membership value, $f^{\mathrm{t1}}(x)$. In classical set theory, however, a realized value of x either “belongs to” a crisp set (i.e., it has a membership value of 1 from a fuzzy point-of-view) or “does not belong to” it (i.e., it has a membership value of 0 from a fuzzy point-of-view). A type-2 fuzzy set, compared with a type-1 fuzzy set, can handle the uncertainties that may exist in the membership values (more details regarding type-2 fuzzy sets will be given in Section 9.4).

A fuzzy control system is composed of if–then fuzzy rules of the form "**If** *antecedent statement,* **then** *consequent statement*". The antecedent statement expresses a qualitative value of the system's state, and the consequent statement proposes a qualitative value for the controlled variables [6]. An FLC system is mainly composed of four components [89, 90]:

1. *Fuzzification interface*: Translates the quantitative input into a qualitative one.
2. *Knowledge base*: Includes all the if–then fuzzy rules.
3. *Decision making logic*: Is the core of the FLC system, and finds the best qualitative output using the knowledge base.
4. *Defuzzification interface*: Translates the qualitative output into a quantitative one to be implemented by the actuators.

FLC systems vary in the way they are generated, or in the way the rules in the rule bases are established. In his survey paper, Feng [47] divides the different types of FLC systems into the following six categories:

1. Conventional FLC systems
2. Fuzzy PID control systems
3. Neuro-fuzzy control systems
4. Fuzzy sliding mode control systems
5. Adaptive FLC systems
6. Takagi-Sugeno (TS) model-based FLC systems

Our focus in Part III of the thesis is on designing an adaptive FLC system (item 5) within a multi-agent architecture, with a model-based control approach (item 6). Rules of an FLC system are mostly based on the (possibly heuristic) human-defined (e.g., an expert who is experienced in the field) rules. When the structure of the control system is more complicated, e.g., for a multi-input multi-output control system (see [17, 126]), designing such an FLC system becomes more difficult. In model-based FLC, the fuzzy rules of the controller are designed based on a model of the open-loop system, taking into account the expected characteristics of the closed-loop controlled system. Hence, a more systematic way of training the fuzzy rules can be followed in a model-based FLC system.

In Part III of this thesis, we propose a general representation for a type-2 nonlinear fuzzy model, which is inspired by the main idea of the TS model. Therefore, we first give a brief introduction on the Takagi-Sugeno-Kang (TSK) model, which is an extended form of the TS model.

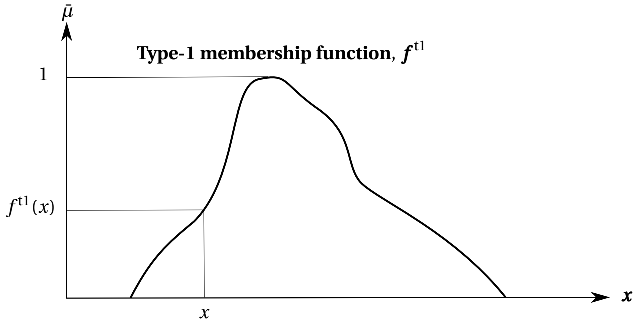


Figure 9.2: Type-1 membership function (continuous domain).

Takagi-Sugeno-Kang (TSK) model

Takagi and Sugeno [145] in 1985 proposed a mathematical tool for constructing fuzzy models of systems, called the Takagi-Sugeno or TS model. It has been proven that TS fuzzy models are universal approximators for smooth nonlinear functions [21, 46, 78], and this was a motivation to apply the TS model for FLC systems. The TS model was later extended by Sugeno and Kang [143], and therefore it is nowadays also known as the Takagi-Sugeno-Kang (TSK) model. The TSK model of a dynamical system includes locally linearized submodels $\mathcal{R}$ of the system represented by

$$\mathcal{R}: \quad \text{If } \boldsymbol{z}_1^{\text{in}} \in F_1 \text{ and } \ldots \text{ and } \boldsymbol{z}_N^{\text{in}} \in F_N, \text{ then } \boldsymbol{z}^{\text{out}} = \boldsymbol{a}_0 + \sum_{i=1}^{N} \boldsymbol{A}_i \cdot \boldsymbol{z}_i^{\text{in}}, \tag{9.1}$$

with $\boldsymbol{z}_i^{\text{in}}$, $i \in \{1, \ldots, N\}$ the input vectors, $\boldsymbol{z}^{\text{out}}$ the output of the dynamical system, $F_1, \ldots, F_N$ fuzzy sets, $\boldsymbol{a}_0$ a vector of the same size as the state vector, and $\boldsymbol{A}_i$ a matrix of appropriate dimension. Therefore, in the TSK model the if-part of a rule activates an affine relationship among input vectors. These submodels are connected smoothly via fuzzy membership functions [47, 145]. For example, suppose that the input set is divided into n crisp subsets, where an affine relationship $\mathcal{R}_r$ for $r \in \{1, \ldots, n\}$ similar to (9.1) describes the submodel that corresponds to subset r. Then these submodels may be connected with a smooth linear relationship such as:

$$\boldsymbol{z}^{\text{out}} = \frac{1}{\sum\limits_{r=1}^{n} \bar{\mu}_r} \sum_{r=1}^{n} \bar{\mu}_r \boldsymbol{z}_r^{\text{out}}, \tag{9.2}$$

where $\boldsymbol{z}_r^{\text{out}}$ indicates the output of the r^{th} affine relationship, and $\bar{\mu}_r$ is the truth value of this output. Note that the truth value is a t-norm of the fuzzy membership values of the input vectors corresponding to the antecedent of $\mathcal{R}_r$.

9.4 Fuzzy sets: Type-1 and type-2

Two types of fuzzy sets, i.e., type-1 and type-2, have been introduced in the literature [157, 158]. A type-1 fuzzy set F^{t1} corresponds to a type-1 membership function $f^{\text{t1}}(\cdot)$ and is defined by

$$F^{\text{t1}} = \left\{ \left(x, f^{\text{t1}}(x)\right), \forall x \in \mathbb{X} \right\}. \tag{9.3}$$

For a type-1 fuzzy set, each point within the domain adopts a single (crisp) membership value (see Figure 9.2). However, sometimes interpretation of the information that is available for a certain dataset may result in more than a single membership value for a specific point. For instance, if we use the opinion or interpretation of the public on a specific topic, we may come up with different conclusions and interpretations that are not even close or similar to each other. Consequently, the membership values that are defined based on this information for each point within the domain are not crisp values, but include uncertainties. In this case, a type-1 fuzzy set cannot be used [112–114], which leads us towards using type-2 fuzzy sets and correspondingly, type-2 membership functions as an alternative.

Type-2 fuzzy sets were introduced by Zadeh [158] to deal with the uncertainties that may exist in datasets. These are fuzzy sets that assign a set of n values (where in the limit $n \to \infty$) instead of a single value to the membership value of a point x within the domain $\mathbb{X}$. In other words, in a type-2 fuzzy set each independent variable x may adopt a fuzzy membership value instead of a crisp one. Each of these values is called a “primary membership value”. Each primary membership value $f^{\text{t1}}_{\text{p},i}(x)$ for point x has a weight $f^{\text{t1}}_{\text{s}}\left(x, f^{\text{t1}}_{\text{p},i}(x)\right)$ (a number between 0 and 1), where this weight is called the “secondary membership value” of point x. Figure 9.3 illustrates a case where the point $x \in \mathbb{X}$ has three different primary membership values, i.e., in this figure we have

$$f^{\text{t1}}_{\text{p},1}(x) = \bar{\mu}_{1,1}(x), \quad f^{\text{t1}}_{\text{p},2}(x) = \bar{\mu}_{1,2}(x), \quad f^{\text{t1}}_{\text{p},3}(x) = \bar{\mu}_{1,3}(x).$$

Moreover, in Figure 9.3, each of these three membership values corresponds to a secondary membership value, i.e.,

$$f^{\text{t1}}_{\text{s}}\left(x, f^{\text{t1}}_{\text{p},1}(x)\right) = \bar{\mu}_2\left(\bar{\mu}_{1,1}(x)\right), \quad f^{\text{t1}}_{\text{s}}\left(x, f^{\text{t1}}_{\text{p},2}(x)\right) = \bar{\mu}_2\left(\bar{\mu}_{1,2}(x)\right), \quad f^{\text{t1}}_{\text{s}}\left(x, f^{\text{t1}}_{\text{p},3}(x)\right) = \bar{\mu}_2\left(\bar{\mu}_{1,3}(x)\right).$$

Correspondingly, we define n primary type-1 membership functions $f^{\text{t1}}_{\text{p},1} < f^{\text{t1}}_{\text{p},2} < \ldots < f^{\text{t1}}_{\text{p},n}$, where $f^{\text{t1}}_{\text{p},i} < f^{\text{t1}}_{\text{p},j}$ indicates that for all $x \in \mathbb{X}$, $f^{\text{t1}}_{\text{p},i}(x) < f^{\text{t1}}_{\text{p},j}(x)$. Then, a type-2 fuzzy set is defined by

$$F^{\text{t2}} = \left\{ \left(x, f^{\text{t1}}_{\text{p},i}(x), f^{\text{t2}}\left(x, f^{\text{t1}}_{\text{p},i}(x)\right)\right), \forall x \in \mathbb{X},\ \forall i \in \mathbb{N}_n \right\}, \tag{9.4}$$

where $\mathbb{N}_n = \{1, \ldots, n\}$. Moreover $f^{\text{t1}}_{\text{s}}(x, \cdot) : f^{\text{t1}}_{\text{p},i}(x) \to f^{\text{t2}}\left(x, f^{\text{t1}}_{\text{p},i}(x)\right)$ for $i \in \mathbb{N}_n$ is called the secondary type-1 membership function of the type-2 membership function f^{t2} at the point $x \in \mathbb{X}$. Membership functions of a type-2 fuzzy set may be illustrated in a 3-dimensional space rather than a 2-dimensional one. Figure 9.4 shows a continuous-domain type-2 membership function with its secondary type-1 membership functions. In this figure, the three dimensions are

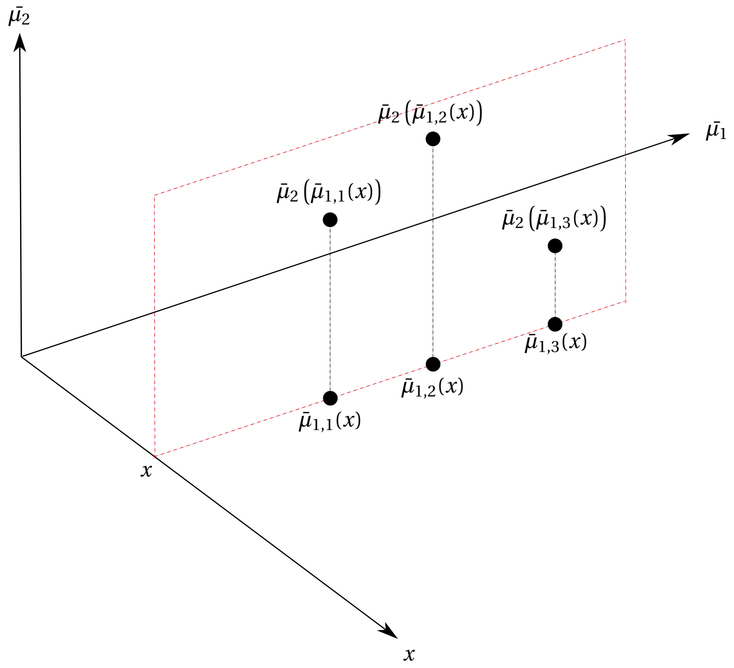

$\bar{\mu}_2$
$\bar{\mu}_2\left(\bar{\mu}_{1,2}(x)\right)$
$\bar{\mu}_1$
$\bar{\mu}_2\left(\bar{\mu}_{1,1}(x)\right)$
$\bar{\mu}_2\left(\bar{\mu}_{1,3}(x)\right)$
$\bar{\mu}_{1,3}(x)$
$\bar{\mu}_{1,2}(x)$
$\bar{\mu}_{1,1}(x)$
x
x

Figure 9.3: Illustration of the primary and secondary membership values for the point $x \in \mathbb{X}$, where it is assumed that x has three different fuzzy membership values.

indicated by x, $\bar{\mu}_1$, and $\bar{\mu}_2$, where x corresponds to the independent variable, $\bar{\mu}_1$ corresponds to the primary membership values, and $\bar{\mu}_2$ corresponds to the secondary membership values. Additionally, $f_{\mathrm{s}}^{\mathrm{t1}}(x,\cdot)$ is the secondary type-1 fuzzy membership function of the type-2 fuzzy membership function at the point $x \in \mathbb{X}$. This function is indeed the cross section of the 3-dimensional type-2 fuzzy membership function with a plane in parallel to the $\bar{\mu}_1 - \bar{\mu}_2$ plane that passes through point x.

In the literature, a special case of the type-2 fuzzy membership functions is particularly taken into account, which is called the "interval type-2 membership function" [112]. The secondary type-1 membership function of an interval type-2 membership function is constant for all $x \in \mathbb{X}$, i.e., it is the unity function. To the best of our knowledge, interval type-2 fuzzy membership functions are the only type-2 fuzzy membership functions that have been used in practice for modeling and for control. In the next section, we briefly introduce the work that has been done on interval type-2 fuzzy membership functions.

9.5 Overview of previous work: Interval type-2 fuzzy membership functions

Hagras [53] develops a hierarchical (2-level) type-2 FLC system for mobile robots that should navigate in a dynamic changing environment with uncertainties involved. For the proposed FLC system, both the input and the output of the system are represented by interval type-2

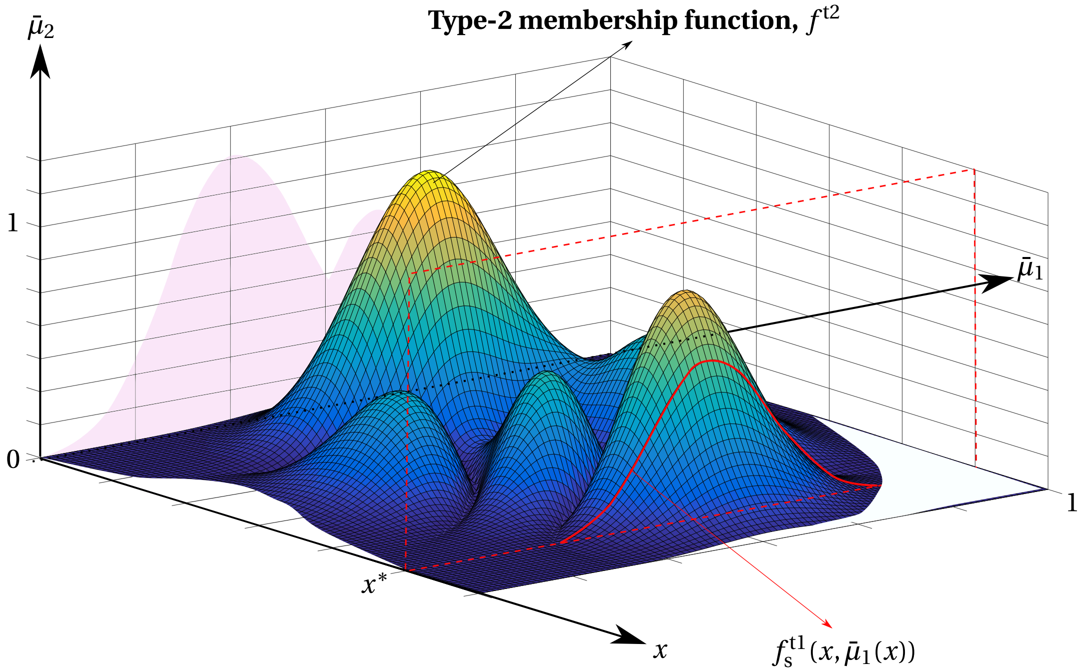


Figure 9.4: Type-2 membership function (continuous domain).

fuzzy sets instead of general type-2 fuzzy sets. A type reduction approach is used, through which the type-2 output fuzzy sets are reduced to type-1 fuzzy sets. After defuzzification, a crisp output is produced.

Wu and Tan [156] introduce a type-2 FLC system for controlling a liquid-level procedure. They use a design approach to find interval type-2 fuzzy sets. This approach is known as the “partially dependent approach”. First, type-1 fuzzy sets are considered, and the design parameters are found via an optimization procedure that applies a genetic algorithm. Then, the input variables of the type-1 fuzzy sets are uniformly blurred to produce the footprint of uncertainties for the corresponding interval type-2 fuzzy sets.

Hagras [54] gives some applications of interval type-2 fuzzy sets for control procedures in industry, such as for the speed control of a marine diesel engine, for control of a buck DC-DC converter, for control of a group of mobile robots, and for control of the ambient intelligent environments.

Hagras et al. [55] propose a model-free approach that is based on designing an adaptive interval type-2 FLC system to control an ambient intelligent environment. Eight phases are considered for the FLC system, where the environment is monitored during phase 1 and input-output data is captured. In phase 2 type-1 fuzzy sets are constructed using data from phase 1; phases 3 and 4 are corresponding to the implementation of the type-1 FLC system and to adding or adapting the rules based on the feedback received from the system. In phase 5 the system is remonitored to collect a new set of input-output data, so that in phase 6 new type-2 fuzzy sets are extracted from this dataset. Phases 7 and 8 are allocated to implementing the type-2 FLC system and to adding or adapting the rules. If needed, the system will go back to phase 5 to repeat the entire sequence of steps of phase 5-phase 8. Note that the work presented by Hagras et al. [55] is a continuation of the work done by Doctor et al. [41], where only the first five phases (i.e., a type-1 FLC system) are discussed.

9.6 Overview of Part III

The main aim of Part III of the thesis is to develop a multi-agent control system that is suited for controlling large-scale systems. To this aim, we first develop a two-layer fuzzy controller. The rules of the fuzzy controller are constructed from type-2 fuzzy sets and are tuned online. A predictive model-based and optimization-based controller is designed next to the fuzzy controller to steer the fuzzy controller. The predictive controller considers the mutual effects of the fuzzy controller's decisions and the decisions made by the neighboring controllers. At every time step that the fuzzy controller tunes its parameters, the computations of the predictive controller can be used to bring the possible effects of the future time steps due to the neighboring controllers into the tuning procedure. The resulting agent-based fuzzy and predictive control system is embedded in a multi-agent architecture to control large-scale systems.

Chapter 10

Multi-agent predictive and adaptive model-based type-2 fuzzy control

10.1 Introduction

In this chapter, we propose to integrate intelligent control approaches, which have a low computation time and fit real-time control applications, with MPC controllers that take care of the optimality requirements of the control system. The main aim is to obtain a fast control system for real-time applications with adaptivity, high performance, and capability of coordinating effectively with other controllers within a multi-agent control architecture. The resulting control system benefits from the characteristics of both control methods (i.e., reasonable computation time and suboptimality), while the intelligent control module takes into account the delayed effect of the previous states, and the MPC controller involves the impact of the current control decision on the expected future states and performance of both the controlled system and the connected systems to it (the dynamics of which may affect the dynamics of the controlled system). A prediction model of the external disturbances can also be linked to the intelligent control module to allow it to make the control decision considering the expected cumulative disturbances in the near future.

For our specific application, i.e., for urban traffic networks, we focus on fuzzy logic control for the intelligent control module. This is because traffic has originally been controlled using human-defined rules, and hence traffic regulations are by nature rule-based, which is in line with the rule-based nature of fuzzy logic control. The discussions are, however, easily extendable to other intelligent control methods such as artificial neural network-based controllers.

We consider the problem of controlling multiple connected subsystems the dynamics of which might be affected by time-delayed states in addition to the current state. We consider subsystems with limited memory and limited storage space for the measured states, with costly or missing measurements, with slow sensors, etc. For such systems the time-delayed states are not always available, and hence using state augmentation [84] is not possible. Therefore, we first introduce type-2 fuzzy models of the subsystems that receive the current and some previous control inputs and estimate the future or the missing previous states. Additionally, we propose two forms of type-2 fuzzy membership functions, i.e., probabilistic-fuzzy and fuzzy-fuzzy. A general adaptive scheme is proposed to produce fuzzy rules that include type-2 fuzzy sets in their antecedent to deal with uncertainties in the dataset, and a generally nonlinear function of the current and/or delayed inputs and outputs, and the disturbances in their consequent. The proposed scheme can be used to provide a prediction model of the controlled system for the MPC controller, to generate a prediction model of the expected external disturbances based on the collected historical data that can be used within a model-based control architecture together with the fuzzy controller, and to design fuzzy controllers that perform within a two-layer adaptive control architecture.

The MPC controller in the second control layer looks ahead in time, and takes into account the interactions among the subsystems. The optimal performance index determined by the MPC controller is used in tuning the adaptive parameters of the fuzzy controller in order to improve the performance of the controlled subsystem and to include the effects of interactions with the other controllers in the multi-agent control architecture. The second layer of control is called by the fuzzy controller at certain tuning time steps, and also when a predefined tuning criterion is triggered.

Contributions and organization of the chapter

Table 10.1: Frequently used mathematical notations for Chapter 10.

e^{r}	a random event
e^{f}	a fuzzy event
k	time step counter
$\mathbb{K}$	set of all time steps
$\mathbb{K}^{\text{identify}}$	set of identification time steps
$\mathbb{K}^{\text{tune}}$	set of tuning time steps
$\boldsymbol{x}(k)$	measured state at time step k
$\boldsymbol{x}^{\mathrm{m}}$	estimated state by the fuzzy model at time state k
$\boldsymbol{u}(k)$	control input vector at time step k
$X_{i,j}^{\mathrm{t2}}(k)$	type-2 fuzzy sets of the antecedent of the model's fuzzy rules at time step k
$\boldsymbol{\theta}^{\mathrm{m}}(k)$	parameter vector of the consequent of the model's fuzzy rules at time step k
$U_{i,j}^{\mathrm{t2}}(k)$	type-2 fuzzy sets of the antecedent of the controller's fuzzy rules
$\boldsymbol{\theta}^{\mathrm{c}}(k)$	parameter vector of the consequent of the controller's fuzzy rules at time step k
$J(\cdot)$	cost function
$\boldsymbol{v}(k)$	vector of disturbances at time step k

The main contributions of the chapter are listed below:

- We present a general and extensive treatment of type-2 fuzzy sets and fuzzy membership functions. Compared with the available literature, we explain type-2 fuzzy sets

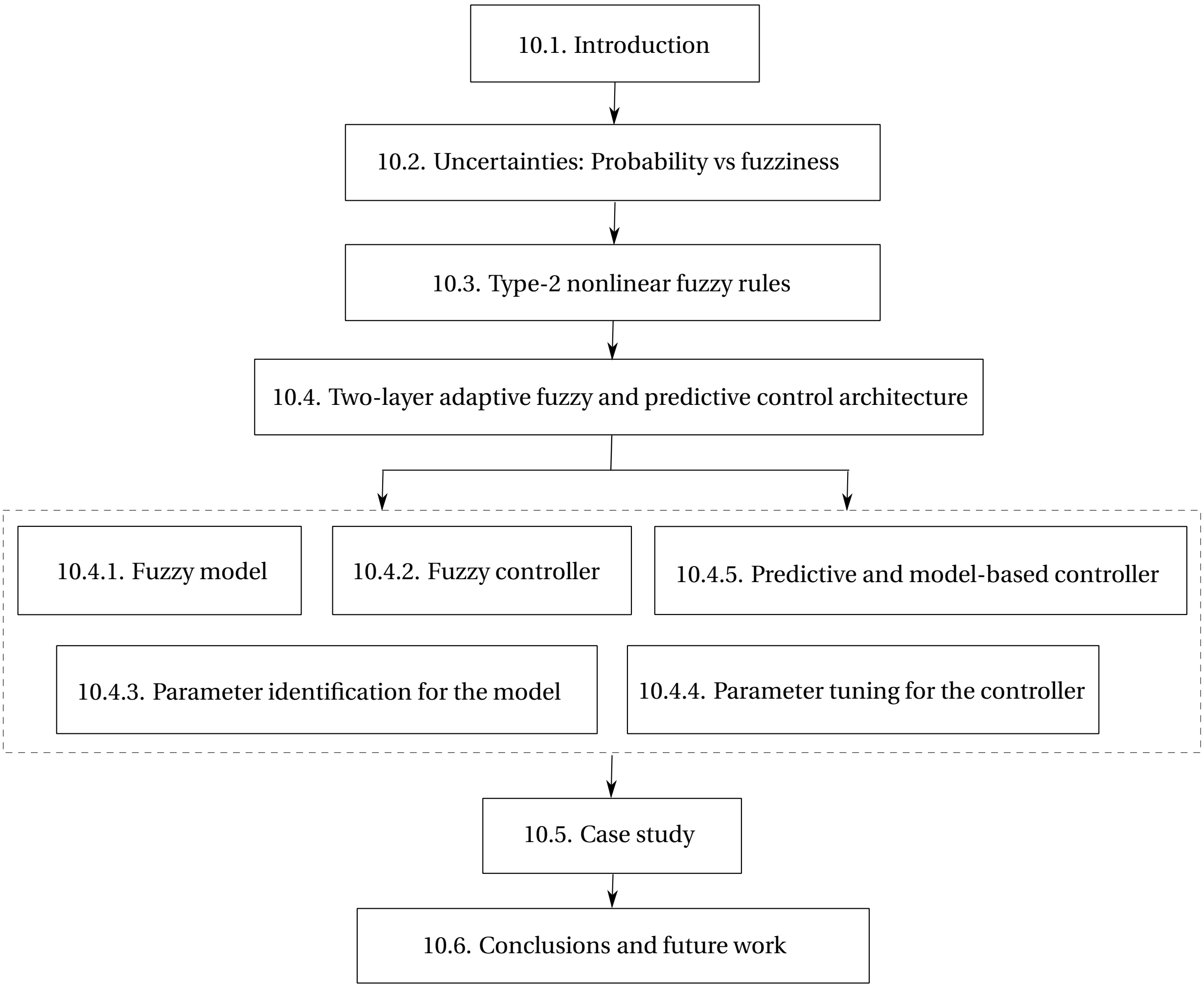

10.1. Introduction
10.2. Uncertainties: Probability vs fuzziness
10.3. Type-2 nonlinear fuzzy rules
10.4. Two-layer adaptive fuzzy and predictive control architecture
10.4.1. Fuzzy model
10.4.2. Fuzzy controller
10.4.5. Predictive and model-based controller
10.4.3. Parameter identification for the model
10.4.4. Parameter tuning for the controller
10.5. Case study
10.6. Conclusions and future work

Figure 10.1: Road map of Chapter 10.

in a 3-dimensional space and with more clear and more structured mathematical formulations. Moreover, we propose a new terminology in our discussions for type-2 fuzzy membership functions, where we divide these functions into two types called "probabilistic-fuzzy" and "fuzzy-fuzzy" membership functions. We believe the new terminology fits well the definitions and the concepts we present.

- We introduce a novel two-layer adaptive control system that integrates an MPC controller in the second layer with an intelligent controller in the first layer. The resulting control system provides adaptivity, efficiency and good performance, and coordination among the intelligent controller and other controllers in a multi-agent architecture.

- We propose a general formulation for producing type-2 fuzzy rules. With this formulation, fuzzy controllers are developed to be used as the intelligent control module within the proposed adaptive control system. Moreover, prediction models of the controlled system and the expected external disturbances can be constructed using these type-2 fuzzy rules, which can be used respectively as the prediction model of the MPC controller in the second control layer and in a model-based structure with the fuzzy controller. The developed control system is implemented and evaluated for reducing the total travel time in an urban traffic network.

In addition, we also discuss two types of uncertainties (probabilistic and fuzzy) that may occur in different datasets, and correspondingly, we propose two forms of type-2 fuzzy membership functions, which we call probabilistic and fuzzy-fuzzy membership functions.

The rest of the chapter is organized as follows. In Section 10.2 we discuss two different forms of uncertainties (probabilistic and fuzzy) that may occur in datasets. Correspondingly, we introduce two forms of type-2 fuzzy membership functions, which we call probabilistic-fuzzy and fuzzy-fuzzy membership functions. Section 10.3 proposes a nonlinear scheme for constructing the rule base of a fuzzy model that corresponds to a time-delayed system with missing or limited data. Additionally, the proposed scheme can also be used to generate a state-feedback fuzzy controller. Section 10.4 presents a two-layer control architecture that can be used in a multi-agent control system. The first layer of the control architecture includes state-feedback fuzzy controllers with rules that follow the general structure proposed in Section 10.3, and the second layer includes a predictive and optimization-based controller. Whenever called by the first layer, the predictive controller steers the fuzzy controller to tune its parameters adaptively. In Section 10.5 a case study is done where an urban traffic network with two adjacent intersections is considered. The traffic network is once controlled by a centralized MPC controller and once by a two-layer adaptive controller as proposed in this chapter. Finally, Section 10.6 concludes the chapter and proposes some topics for future work. The road map of this chapter is shown in Figure 10.1. Moreover, Table 10.1 gives the mathematical notations that are frequently used in Chapter 6.

This chapter of the thesis is based on [72].

10.2 Uncertainties: Probability versus fuzziness

In Chapter 9, we have given the definitions of type-1 and type-2 fuzzy sets, and correspondingly type-1 and type-2 fuzzy membership functions. However, in Section 9.4, we did not expand the basic ideas considered in the literature for type-2 fuzzy sets and fuzzy membership functions. In this section, we look at type-2 fuzzy sets and the motivation for using them in more detail. Since type-2 fuzzy sets have been introduced with the aim of dealing with uncertainties, we first discuss the possible natures of the uncertainties that may occur, i.e., probabilistic and fuzzy uncertainties. Note that in fuzzy formulations, a fuzzy set corresponds to a fuzzy membership function. Correspondingly, we expand the current concept of type-2 fuzzy membership function to two types of fuzzy membership functions, which we call probabilistic-fuzzy and fuzzy-fuzzy membership functions.

Uncertainties may appear in two different ways: probabilistic and fuzzy [44]. The first is observed when a known set $\mathbb{E}^{\mathrm{r}} = \left\{e_1^{\mathrm{r}}, \ldots, e_{n_{\mathbb{E}^{\mathrm{r}}}}^{\mathrm{r}}\right\}$ of random events exists, and based on probability theory [44] each of these events has a certain probability of occurrence less than or equal to 100%. The summation of these probabilities for all the random events in $\mathbb{E}^{\mathrm{r}}$ should exactly be 100%. To elaborate the concept of a random event, we give an example.

Example 1: Random events

Suppose that we have a thermometer with a resolution of 10°C. Now, consider the following three statements, which represent three random events as the given statements are precise and refer to a certain fact. Therefore, these statements describe completely independent (non-overlapping) situations:

e_1^{r} : The temperature in this room is 10°C.

e_2^{r} : The temperature in this room is 20°C.

e_3^{r} : The temperature in this room is 30°C.

Each of these statements refers to a certain value of the state of the room and gives a quantitative expression of this value. Hence, these statements can be interpreted in one way only (i.e., there are no different interpretations for a quantitative expression such as 5°C). At a specific instant of time and in a specific room, only one of these statements can occur. Therefore, if $p_{e_i^{\mathrm{r}}}$ denotes the probability of occurrence of e_i^{r}, then for $i \in \{1,2,3\}$ we have

$$0 \leq p_{e_i^{\mathrm{r}}} \leq 1, \quad \sum_{i=1}^{3} p_{e_i^{\mathrm{r}}} = 1.$$

Hence, for a set of random events, the logical statements corresponding to these events are certain statements, i.e., there is no qualitative information in these statements that may be interpreted differently by various people. For a set of random events, the uncertainty is in the possibility of occurrence of each random event. For instance, in Example 1 somebody who is outside the room and has no information about the temperature of the room except that either of e_1^{r}, e_2^{r}, or e_3^{r} holds, cannot express by certainty that which of the three statements holds in reality.

Alternatively, uncertainty may occur due to the use of qualitative expressions in a logical statement, where these qualitative expressions can have different interpretations by various people. In this case, we have a set $\mathbb{E}^{\mathrm{f}} = \left\{e_1^{\mathrm{f}}, \ldots, e_{n_{\mathbb{E}^{\mathrm{f}}}}^{\mathrm{f}}\right\}$ of fuzzy events, where unlike a random statement e_i^{r}, the qualitative expression of a fuzzy event e_i^{f} may be interpreted as various quantitative expressions that may not be identical at all. For instance, when we say "this student has received a *high* grade in math", one may interpret the qualitative expression of "high grade" to be a grade between 7 and 10, while another person may consider a high grade to be between 8.5 and 10. Each fuzzy event corresponds to a membership value less than or equal to 100%. Due to the different interpretations of fuzzy statements, there may be overlaps in the quantitative interpretation of the fuzzy events in a set (i.e., the occurrence of a fuzzy event may not exclude the probability of occurrence of other fuzzy events). For instance, in the given statement about the student's grade, the range of 8.5 to 10 appears in both given interpretations. Consequently, the summation of the membership values of all the fuzzy events in $\mathbb{E}^{\mathrm{f}}$ may exceed or be lower than 100%.

The following example contains three fuzzy events. The given qualitative statements for the three events include terms (i.e., cold, moderate, and warm) that may be interpreted in different ways by different people.

Example 2: Fuzzy events

e_1^{f} : The climate in this room is *cold.*

e_2^{f} : The climate in this room is *moderate.*

e_3^{f} : The climate in this room is *warm.*

The terms "cold", "moderate", and "warm" are qualitative. Hence, each of them may be interpreted differently. For example, someone may consider the following interpretations for them:

Interpretation 1:

The term "cold" corresponds to a temperature around 2°C or less.

The term "moderate" corresponds to a temperature around 23°C.

The term "warm" corresponds to a temperature around 30°C or more.

From this interpretation, one may conclude that the curves shown in Figure 10.2 illustrate one possible fuzzy description of the climate of the room, where the membership values of 0 and 1 are extracted based on the above interpretations of the terms cold, moderate, and warm, and for the rest of the temperature range, membership functions are defined that gradually transfer from one climate (i.e., cold, moderate, or warm) to another one. Now if the real temperature of the room is 18°C, then from Figure 10.2 the climate of the room may be considered to be cold with a possibility of 2% (i.e., the possibility of the realization of e_1^{f} is 2%). The climate of the room may be considered as moderate with a possibility of 45% (i.e., the possibility of the realization of e_2^{f} is 45%). Moreover, there is no possibility that the climate of the room is considered to be warm (i.e., the possibility of the realization of e_3^{f} is 0%). Therefore, for a set of fuzzy events in contrast to a set of random events, the summation of the chance of occurrence of different fuzzy events at the same time is not necessarily equal to 1 (e.g., in the given example this summation is 47%).

As explained in Section 10.3, type-2 membership functions are mostly considered when there is uncertainty in the realized value of the primary type-1 membership degrees of points within the domain. Considering the two types of uncertainties explained above, we define two types of type-2 membership functions, called the probabilistic-fuzzy and the fuzzy-fuzzy membership functions. We use the term fuzzy-fuzzy membership function, as an alternative term for type-2 fuzzy membership functions defined in Section 9.4. Since the secondary

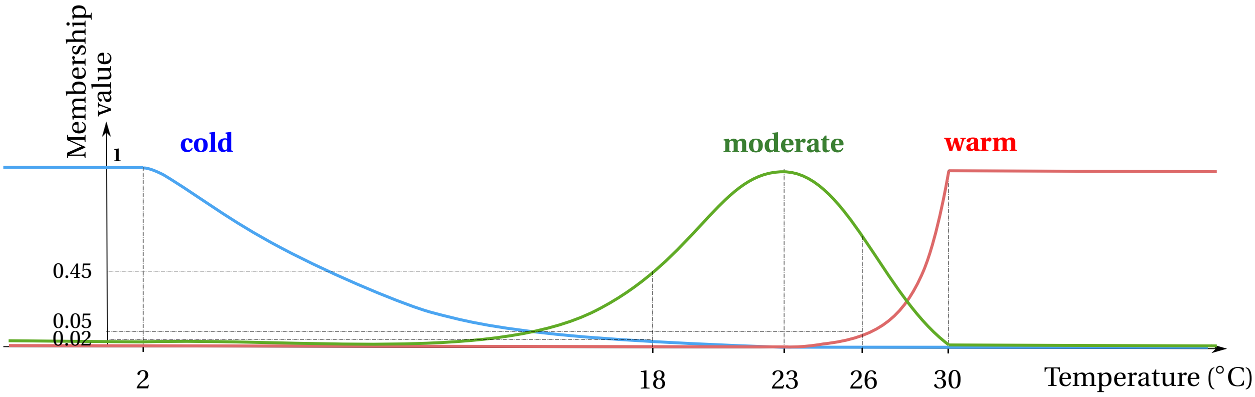


Figure 10.2: A possible option for type-1 fuzzy membership functions of the climates "cold", "moderate", and "warm" based on the statements given in "Interpretation 1".

membership function $f_{\mathrm{s}}^{\mathrm{t1}}(x,\cdot)$ is itself a type-1 fuzzy membership function (see Section 9.4 for more details), i.e., the concept of fuzziness is repeated twice, we find it more suitable to call this form of type-2 fuzzy membership functions fuzzy-fuzzy membership functions.

The probabilistic-fuzzy case is a new concept that we introduce and implement in this chapter. The secondary membership function $f_{\mathrm{s}}^{\mathrm{t1}}(x,\cdot)$ of a probabilistic-fuzzy type-2 membership function can be any probability function that satisfies

$$\int_{M_1(x)} f_{\mathrm{s}}^{\mathrm{t1}}(x,\bar{\mu}_1)\cdot \mathrm{d}\bar{\mu}_1 = 1, \qquad \forall x \in \mathbb{X},$$

where $M_1(x)$ is an interval defined by $M_1(x) = \left[f_{\mathrm{p},1}^{\mathrm{t1}}(x), f_{\mathrm{p},n}^{\mathrm{t1}}(x)\right]$, recalling that n is the total number of the primary type-1 fuzzy membership functions. In order to explain and to clarify the difference between a case that can be formulated by a probabilistic-fuzzy membership function, and a case that can be modeled by a fuzzy-fuzzy membership function, we give two examples.

Example 3: Probabilistic-fuzzy case

Suppose that we ask 10 different people, who are assumed to represent the population perfectly, to give their opinions about the "level of convenience" of a room. The information we provide for these people includes the temperature of the room (i.e., quantitative information) and the following fuzzy statements (i.e., qualitative information):

$e_{1,0}^{\mathrm{f}}$: If the climate in the room is *a little bit warm* or *very warm*, then the room is not convenient.

$e_{2,0}^{\mathrm{f}}$: If the climate in the room is *moderately warm*, then the room is convenient.

These people should first translate the quantitative information about the temperature of the room into a qualitative one about the climate of the room (i.e., "cold", "moderate", and

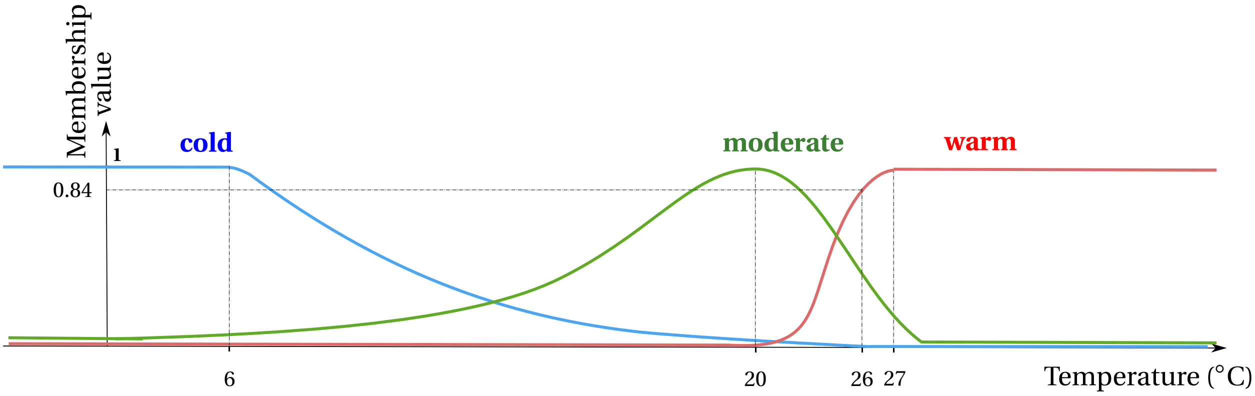


Figure 10.3: A possible option for type-1 fuzzy membership functions of the climates "cold", "moderate", and "warm" based on the statements given in "Interpretation 2".

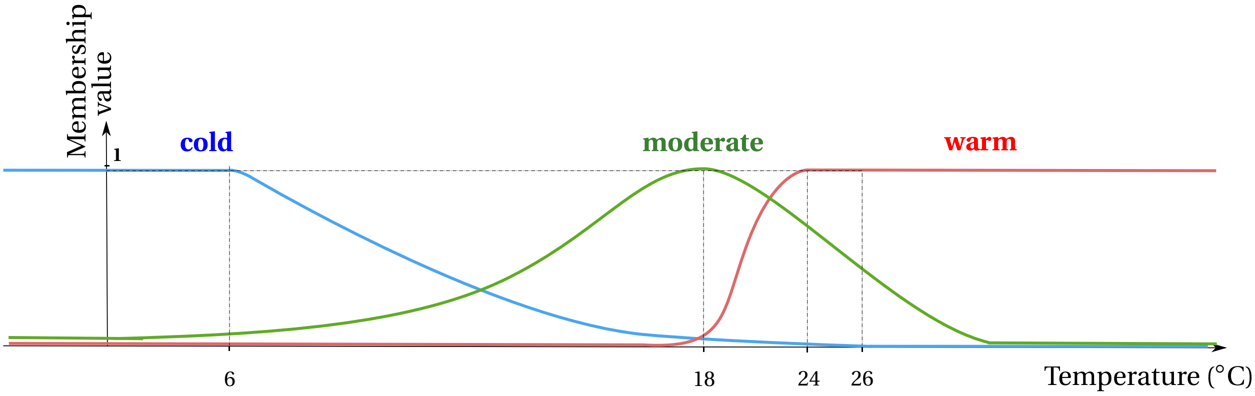


Figure 10.4: A possible option for type-1 fuzzy membership functions of the climates "cold", "moderate", and "warm" based on the statements given in "Interpretation 3".

"warm") and also decide about the qualitative degree of the warmth (i.e., "a little bit", "moderately", or "very"). Based on these translated information, they can decide about the "level of convenience" of the room.

Suppose that 3 out of the 10 people agree with the interpretations given in Example 2 for the terms "cold", "moderate", and "cold", 5 out of the 10 people consider interpretation 2, and the other 2 people consider interpretation 3 for these terms:

Interpretation 2:

The term "cold" corresponds to a temperature around 6°C or less.

The term "moderate" corresponds to a temperature around 20°C.

The term "warm" corresponds to a temperature around 27°C or more.

Interpretation 3:

The term “cold” corresponds to a temperature around 6°C or less.

The term “moderate” corresponds to a temperature around 18°C.

The term “warm” corresponds to a temperature around 24°C or more.

Similar to Figure 10.2, type-1 fuzzy membership functions may be extracted based on the interpretations given by the people in groups 2 and 3 (see Figures 10.3 and 10.4), where compared with Figure 10.2, the midpoints of the fuzzy sets have been shifted to the given values in the second and the third interpretations. Suppose that the temperature of the room is 26°C. The membership value of temperature 26°C for the climate “warm” for the first 3 people is 5% (see Figure 10.2), for the second 5 people it is 84% (see Figure 10.3), and for the last 2 people it is 100% (see Figure 10.4). These membership values should next be translated into the qualitative expressions “a little bit”, “moderately”, and “very”.

Assume that all these 10 people have the following interpretation:

Interpretation 4:

The term “a little bit” refers to a percentage less than 10%.

The term “moderately” refers to a percentage between 10% and 85%.

The term “very” refers to a percentage greater than 85%.

Although the statements that are given in interpretation 4 include qualitative terms, the information that is conveyed about these terms is certain (i.e., these statements express random events as defined earlier on). Therefore, the following fuzzy statements can be extracted based on the interpretation of, respectively, 30%, 50%, and 20% of the people:

$e^{\mathrm{f}}_{1,1}$: The climate in this room is a little bit warm.

$e^{\mathrm{f}}_{2,1}$: The climate in this room is moderately warm.

$e^{\mathrm{f}}_{3,1}$: The climate in this room is very warm.

Now these 10 people can match their translated qualitative interpretations, which are stated by $e^{\mathrm{f}}_{1,1}$, $e^{\mathrm{f}}_{2,1}$, and $e^{\mathrm{f}}_{3,1}$ with the qualitative information they have been given through $e^{\mathrm{f}}_{1,0}$ and $e^{\mathrm{f}}_{2,0}$, which results in the following conclusions:

- Based on the opinion of 50% of the population (i.e., the first 3 people and the last 2 people), the room is not convenient.

- Based on the opinion of 50% of the population (i.e., the second 5 people), the room is convenient.

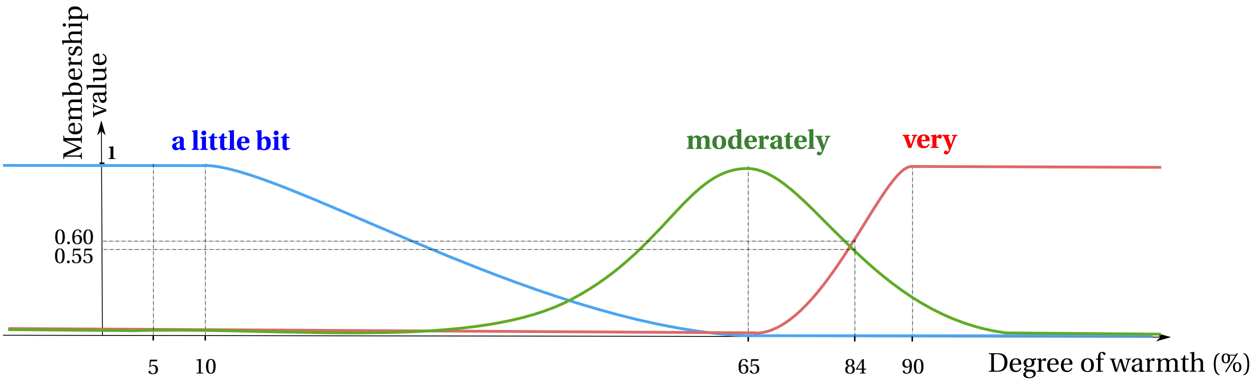


Figure 10.5: A possible option for type-1 fuzzy membership functions of the terms "a little bit", "moderately", and "very" based on the statements given in "Interpretation 5".

Although the procedure of making the above conclusions involves fuzzification (i.e., translation of the quantitative information into qualitative ones) and decision making based on fuzzy statements (given by Interpretations 1–3) in the first stage, for the second stage where the final conclusions are made, probability theory is used. This is because the people involved in making the conclusion have different fuzzy interpretations for the terms "cold", "moderate", and "warm", while they all agree on the same fuzzy interpretation for the terms "a little bit", "moderately, and "very" given by Interpretation 4. Hence, we call such a case a fuzzy-probabilistic case.

Example 4: Fuzzy-fuzzy case

Consider once more the case that is described in Example 3, but this time assume that all these 10 people have the following interpretation for the terms "a little bit", "moderately", and "very":

Interpretation 5:

The term "a little bit" refers to a percentage around 10% or less.

The term "moderately" refers to a percentage around 65%.

The term "very" refers to a percentage around 90% or more.

These statements do not convey a certain opinion about the terms they describe. Hence, they are fuzzy statements and they can be modeled by type-1 fuzzy membership functions. Figure 10.5 shows one example of such fuzzy membership functions. Therefore, from Figure 10.5 for the first group of people (who find the climate to be 5% warm), the membership value of 5% for "a little bit" is 100%, and the membership value for "moderately" and for "very" is zero. For the second group of people (who find the climate to be 84% warm), the membership value of 84% for "a little bit" is zero, for "moderately" it is 55%, and for "very" it is 60%. Finally, for the third group of people (who find the climate to be 100% warm), the membership value of

100% for “a little bit” and for “moderately” it is zero, and for “very” it is 100%. Therefore, the following fuzzy statements can be extracted based on the interpretation of these 10 people:

$e^{\mathrm{f}}_{1,2}$: The climate in this room is a little bit warm.

$e^{\mathrm{f}}_{2,2}$: The climate in this room is 55% moderately warm and 60% very warm.

$e^{\mathrm{f}}_{3,2}$: The climate in this room is very warm.

As we see above, $e^{\mathrm{f}}_{2,2}$ includes both quantitative and qualitative information. However, the information that is given by $e^{\mathrm{f}}_{1,0}$ and $e^{\mathrm{f}}_{2,0}$ involve qualitative information only. Hence, before we can make the final conclusions, $e^{\mathrm{f}}_{2,2}$ should be translated into a completely qualitative statement. This is usually done using a fuzzy inference method. For instance, one may say that since 60% is greater than 55%, the part “55% moderately warm” can be dropped from the statement and only one state, i.e., “very warm”, is considered for the room. This inference results in the following fuzzy statement from $e^{\mathrm{f}}_{2,2}$:

The climate in this room is very warm.

Then, the final conclusion is given by

- Based on the opinion of 100% of the population, the room is not convenient.

In Example 4, the procedure of making the final conclusion involves fuzzification and decision making based on fuzzy statements (given by Interpretations 1–3) in the first stage. Moreover, in the second stage, fuzzy statements (given by Interpretation 5) are again involved. This results in statements that should first be processed using a fuzzy inference approach before a final conclusion can be made. Therefore, the case given by Example 4, is called a fuzzy-fuzzy case. Note that for the specific case described in Example 4, fuzzy inference is not needed for statements $e^{\mathrm{f}}_{1,2}$ and $e^{\mathrm{f}}_{3,2}$, because they include 100% certainty. In different examples, there may be more statements that need a fuzzy inference approach before they can be used to make the final conclusion.

Based on Example 3 and Example 4, we summarize our explanations as follows: when a dataset includes human interpretations of different terms, and inferring a conclusion needs more than one step of interpretation (e.g., in the given examples the quantitative temperature was first translated into a qualitative one, and then the resulting quantitative membership values were again translated into qualitative terms), two cases may occur, i.e., probabilistic-fuzzy (see Example 3) and fuzzy-fuzzy (see Example 4). In particular, if in the second step of interpretation we use probabilistic statements, the probabilistic-fuzzy case occurs. However, if in the second step we use fuzzy statements, the fuzzy-fuzzy case occurs.

10.3 Type-2 nonlinear fuzzy rules for delayed systems

Some systems (including electric networks, pneumatic and hydraulic networks, chemical processes, long transmission lines [38]) involve time-delayed states and/or inputs in their dynamics. This delay can be a result of the time required for transferring the sensed data to the controller, or for transferring the decision made by the controller to the system's actuators. In this section, we consider systems with a time-delayed state. For such systems, the dynamics are affected by the state variables measured at the previous simulation time steps. We discuss a case for which parts of the previous states may not be available. This can happen when due to the limited memory and limited storage space, the measured states cannot be stored, or when parts of the measurements are missing or faulty, or the measurement procedure and tools are costly, or when the state cannot be measured at every simulation time step due to the slow sensors and measurement tools.

For many dynamical systems, the number of input variables is less than the total number of state variables. Hence, it is less costly and needs a smaller memory size to store the input variables instead of the measured state variables for such systems. Moreover, the control inputs are computed by the control systems we design, and hence the corresponding values at every time step are available and do not need to be measured. Correspondingly, we should not be worried about the measurement tools and procedures, and about the missing or faulty measurements. Hence, it is sometimes beneficial to develop a model of dynamical systems with time-delayed state variables that estimates the updated state of the system based on the previous and the current control inputs and the disturbances (as we know that the previous control inputs and disturbances have affected the state variables of the previous time steps).

In this section, we propose a new scheme for a fuzzy model of a dynamical system that considers the current and the delayed input variables in the antecedent of its rules, and that produces the updated state of the dynamical system. The antecedents include type-2 fuzzy sets, where the motivation for using type-2 fuzzy sets in the proposed scheme is to cover the ambiguities that may occur in interpreting the dataset that forms the basis of the fuzzy rule base (for more details see Section 10.2 and Example 4). The antecedent statements are considered to be related by logical "and"s. Inspired by the TSK model explained in Section 9.3, we propose to divide the input set into subsets, and to consider a nonlinear expression of the system's current states, current and time-delayed inputs, and the external inputs (i.e., disturbances) for the dynamics of the system in each resulting subset. Note that these submodels will have a simpler form than a single nonlinear model over the entire input set. Moreover, identifying or tuning the parameters of these submodels independently can be easier and can result in a more accurate model. These submodels can then be connected via a smooth relationship such as (9.2).

For a MIMO system with n^{in} input variables $\boldsymbol{z}_i^{\text{in}}(k),\ i \in \{1,\dots,n^{\text{in}}\}$ and n^{out} output variables $\boldsymbol{z}_j^{\text{out}}(k),\ j \in \{1,\dots,n^{\text{out}}\}$, the proposed type-2 fuzzy model is expressed by rules of the following form:

$$\begin{gathered}
\text{if}\quad \boldsymbol{z}_1^{\text{in}}(k) \in F_{1,0}^{\text{t2}}(k) \quad\text{and}\quad \boldsymbol{z}_1^{\text{in}}\left(k-\delta_{1,1}^{\text{in}}\right) \in F_{1,1}^{\text{t2}}(k) \quad\text{and}\\
\dots \quad\text{and}\quad \boldsymbol{z}_1^{\text{in}}\left(k-\delta_{1,d_1}^{\text{in}}\right) \in F_{1,d_1}^{\text{t2}}(k)\\
\vdots\\
\text{and}\quad \boldsymbol{z}_{n^{\text{in}}}^{\text{in}}(k) \in F_{n^{\text{in}},0}^{\text{t2}}(k) \quad\text{and}\quad \boldsymbol{z}_{n^{\text{in}}}^{\text{in}}\left(k-\delta_{n^{\text{in}},1}^{\text{in}}\right) \in F_{n^{\text{in}},1}^{\text{t2}}(k)
\end{gathered}$$

$$\begin{aligned} &\text{and} \;\; \ldots \;\; \text{and} \;\; \boldsymbol{z}^{\text{in}}_{n^{\text{in}}}\left(k-\delta^{\text{in}}_{1,d_{n^{\text{in}}}}\right) \in F^{\text{t2}}_{1,d_{n^{\text{in}}}}(k) \\ \text{then} \;\; &\boldsymbol{z}^{\text{out}}(k+1) = \boldsymbol{f}\Big(\boldsymbol{\theta}(k), \boldsymbol{z}^{\text{out}}(k), \boldsymbol{z}^{\text{in}}(k), \boldsymbol{z}^{\text{in}}_{\text{d}}(k), \boldsymbol{v}(k)\Big), \end{aligned}$$

where k is the discrete time step counter, $F^{\text{t2}}_{i,j}(k)$ for $i \in \{1,\ldots,n^{\text{in}}\}$, $j \in \{0,\ldots,d_i\}$ indicate type-2 fuzzy sets that may be time-varying and their parameters[1] may be tuned at time steps $k \in \mathbb{K}^{\text{identify}}$ (where the set $\mathbb{K}^{\text{identify}}$ may not necessarily include all the discrete time steps k), $\delta^{\text{in}}_{i,j}$ and d_i are non-negative integers, $\boldsymbol{f}$ is, in general, a nonlinear function, $\boldsymbol{\theta}(k)$ is a vector that contains the parameters of the consequent of the model's fuzzy rules at time step k, and that may be tuned at time steps $k \in \mathbb{K}^{\text{identify}}$, $\boldsymbol{z}^{\text{out}}(k)$ is an n^{out}-dimensional vector that includes all the output vectors of the modeled system at time step k, $\boldsymbol{z}^{\text{in}}(k)$ is an n^{in}-dimensional vector that includes all the input vectors at time step k, and $\boldsymbol{z}^{\text{in}}_{\text{d}}(k)$ is a vector with e^{in} elements $\left(e^{\text{in}} = \sum_{i=1}^{n^{\text{in}}} \sum_{j=0}^{d_i} \delta^{\text{in}}_{i,j}\right)$ that includes all the time-delayed input vectors at time step k, and $\boldsymbol{v}(k)$ is the noise or disturbance vector at time step k.

Note that later on, we will consider two variants for $\boldsymbol{z}^{\text{in}}$ and for $\boldsymbol{z}^{\text{out}}$. The first variant is used for the fuzzy model of the dynamical system (see Section 10.4.1), where $\boldsymbol{z}^{\text{in}}$ is substituted by $\boldsymbol{u}$ (the control input) and $\boldsymbol{z}^{\text{out}}(k)$ is substituted by $\boldsymbol{x}^{\text{m}}$ (the state estimated by the fuzzy model). The second variant is used for expressing the fuzzy controllers (see Section 10.4.2), where $\boldsymbol{z}^{\text{in}}$ is substituted by $\boldsymbol{x}^{\text{m}}$ (the previous states estimated by the system's model) and $\boldsymbol{z}^{\text{out}}$ is substituted by $\boldsymbol{u}$ (the control input produced by the fuzzy controller).

10.4 Two-layer adaptive fuzzy and predictive model-based control architecture

In this section, we propose a two-layer control architecture that aims to minimize an overall cost value for a controlled system with dynamics that involves delayed states. The controlled system is divided into subsystems with local cost functions, where each subsystem is controlled by an agent. The agents make use of controllers with fuzzy rule bases that involve if–then rules with the adaptive parameters and the type-2 nonlinear structure proposed in Section 10.3. These fuzzy controllers form the first layer of control. Suppose that due to limited memory and storage space, missing and/or faulty measurements, and costly or slow measurement tools and sensors, not all the previous measured state variables are available. Hence, each agent uses a fuzzy model of its subsystem that also follows the scheme proposed in Section 10.3 to estimate the missing or unavailable state variables. Moreover, we propose to link a prediction model of the external disturbances to the fuzzy controllers so that they can make the control decisions not only based on the current and the delayed states, but also by taking into account the expected external disturbances in the near future.

Subsystems controlled by the agents may not be completely isolated from each other, i.e., part of the dynamics of these subsystems might be shared. Therefore, the control decision of one agent may affect the dynamics, and consequently the cost value of the connected subsys-

[1] For the sake of compactness in the notations, these parameters are not added as arguments for fuzzy sets.

tems. Hence, the connected agents should coordinate to assure that their decisions will not affect the states of the connecting subsystems in a negative way and vice versa. To that aim, a predictive and optimization-based controller, such as an MPC controller, is proposed in the second layer of control to steer the type-2 fuzzy controllers to adaptively tune their parameters taking into account the effect of the connected subsystems. The optimization problem of the predictive controller is solved by a decomposition method, and the determined optimal solutions are used for tuning the parameters of the type-2 fuzzy controller (see Section 10.4.4 for more details). The second control layer will be activated only at some predefined tuning time steps (i.e., in a periodic way), and possibly at some other control time steps in case a predefined tuning criterion is triggered (i.e., in an event-triggered way). This criterion may be defined in various ways, e.g., in case the realized value of the cost function in the most recent sampling time exceeds a threshold the tuning criterion can be triggered. This avoids solving an optimization problem at every control time step and saves some time for the two-layer control system.

The main advantage of using the proposed two-layer adaptive control architecture compared with a distributed MPC architecture is that fuzzy controllers perform faster than MPC controllers that should solve an optimization problem. In the proposed architecture, the fuzzy controllers consider the delayed state variables of the controlled system and the expected external disturbances in the near future. Since they do not have access to the predicted states of the controlled system in the future, the MPC controller in the second control layer looks in the future and considers these predicted states, in addition to the interactions among different connected subsystems. Hence, for large-scale systems that are affected by some time-delayed variables at every time step (e.g., dynamical systems with time-delayed state variables), and that should deal with limited, missing, or faulty measurements from the previous time steps, the proposed two-layer fuzzy and predictive control architecture can be of more interest than other control architectures, e.g., distributed MPC.

In the following sections, we discuss the formulation of the fuzzy model of the subsystems and the fuzzy controllers used by agents, and we propose efficient approaches for identifying the parameters of the fuzzy submodels and for tuning the parameters of the fuzzy controllers. Note that each subsystem and its controller can be distinguished from other submodels and other controllers using an additional subscript. However, in the following section, we drop this additional subscript for the sake of simplicity of the mathematical notations.

10.4.1 Fuzzy model of subsystems

A subsystem that involves dynamics affected by the delayed states, but does not have access to all the previously measured states, can be modeled by type-2 nonlinear fuzzy rules of the following form, considering the indirect effect of the delayed control inputs on the updated and the previous states:

$$\text{if}\quad \boldsymbol{u}_1(k) \in U^{\text{t2}}_{1,0}(k) \quad \text{and} \quad \boldsymbol{u}_1\left(k-\delta^{\text{c}}_{1,1}\right) \in U^{\text{t2}}_{1,1}(k) \quad \text{and}$$

$$\ldots \quad \text{and} \quad \boldsymbol{u}_1\left(k-\delta^{\text{c}}_{1,d^{\text{c}}_1}\right) \in U^{\text{t2}}_{1,d^{\text{c}}_1}(k)$$

$$
\begin{aligned}
&\vdots\\
&\text{and}\quad \boldsymbol{u}_{n_u}(k) \in U^{\mathrm{t2}}_{n_u,0}(k) \quad \text{and} \quad \boldsymbol{u}_{n_u}\left(k-\delta^{\mathrm{c}}_{n_u,1}\right) \in U^{\mathrm{t2}}_{n_u,1}(k)\\
&\text{and}\quad \ldots \quad \text{and} \quad \boldsymbol{u}_{n_u}\left(k-\delta^{\mathrm{c}}_{n_u,d^{\mathrm{c}}_{nu}}\right) \in U^{\mathrm{t2}}_{n_u,d^{\mathrm{c}}_{nu}}(k)\\
&\text{then}\quad \boldsymbol{x}^{\mathrm{m}}(k+1) = \boldsymbol{f}^{\mathrm{m}}\Big(\boldsymbol{\theta}^{\mathrm{m}}(k), \boldsymbol{x}^{\mathrm{m}}(k), \boldsymbol{u}(k), \boldsymbol{u}_{\mathrm{d}}(k), \boldsymbol{v}(k)\Big),
\end{aligned}
\tag{10.1}
$$

with $k \in \mathbb{K}$ the discrete-time step counter and $\mathbb{K}$ the set of all time steps, $\boldsymbol{u}_i(k), i \in \{1,\ldots,n_u\}$ the control input vectors at time step k, $\boldsymbol{u}(k)$ a vector that includes $\boldsymbol{u}_i(k)$ for all $i \in \{1,\ldots,n_u\}$, $\delta^{\mathrm{c}}_{i,j}$, $j \in \{0,\ldots,d^{\mathrm{c}}_i\}$ and d^{c}_i non-negative integers, $U^{\mathrm{t2}}_{i,j}(k)$ type-2 fuzzy sets that may be time-varying (i.e., their parameters collected in vectors $\boldsymbol{\theta}^{U}_{i,j}$ may be updated at identification time steps $k \in \mathbb{K}^{\mathrm{identify}}$, which are not necessarily every time step $k \in \mathbb{K}$), $\boldsymbol{x}^{\mathrm{m}}(k)$ the state vector of the system estimated by the fuzzy model at time step k, $\boldsymbol{f}^{\mathrm{m}}$ a nonlinear function, $\boldsymbol{u}_{\mathrm{d}}(k)$ a vector that includes all the delayed control inputs with e_u elements ($e_u = \sum_{i=1}^{n_u}\sum_{j=0}^{d^{\mathrm{c}}_i}\delta^{\mathrm{c}}_{i,j}$) at time step k, $\boldsymbol{v}(k)$ the disturbance vector at time step k (note that the disturbance of each subsystem may include the state vectors of the connecting subsystems), and $\boldsymbol{\theta}^{\mathrm{m}}(k)$ a vector that includes all parameters of the consequent of the fuzzy rule at time step k (similar to the type-2 fuzzy sets of the fuzzy models, the elements of $\boldsymbol{\theta}^{\mathrm{m}}$ may be updated at identification time steps $k \in \mathbb{K}^{\mathrm{identify}}$).

The proposed model can be used to provide estimations for the previous states in cooperation with a controller that is affected by the delayed states of the subsystem (see Figure 10.6). At time step k, the controller receives the estimated states (including the delayed states) from the fuzzy model, and produces the control input $\boldsymbol{u}(k)$, which is sent to the subsystem and to the fuzzy model together with the required delayed control inputs that may be stored in a memory unit. At time step $k \in \mathbb{K}^{\mathrm{identify}}$, when the state of the subsystem is measured, the parameters of the fuzzy model are re-tuned (see Section 10.4.3 for more details).

Note that a prediction model of the subsystem for the MPC controller in the second layer can also be developed in a similar way. Moreover, using some historical data from the external disturbances, a fuzzy model can be trained that predicts the future external disturbances and that can be linked to the fuzzy controllers in the first layer.

10.4.2 Fuzzy controller of subsystems

Each agent uses a two-layer adaptive fuzzy and predictive control architecture for its subsystem. The first layer includes a state-feedback controller that is defined by the following fuzzy rules:

$$
\begin{aligned}
&\text{if}\quad \boldsymbol{x}^{\mathrm{m}}_1(k) \in X^{\mathrm{t2}}_{1,0}(k) \quad \text{and} \quad \boldsymbol{x}^{\mathrm{m}}_1\left(k-\delta^{\mathrm{m}}_{1,1}\right) \in X^{\mathrm{t2}}_{1,1}(k) \quad \text{and}\\
&\ldots \quad \text{and} \quad \boldsymbol{x}^{\mathrm{m}}_1\left(k-\delta^{\mathrm{m}}_{1,d^{\mathrm{m}}_1}\right) \in X^{\mathrm{t2}}_{1,d^{\mathrm{m}}_1}(k)\\
&\vdots\\
&\text{and}\quad \boldsymbol{x}^{\mathrm{m}}_{n_x}(k) \in X^{\mathrm{t2}}_{n_x,0}(k) \quad \text{and} \quad \boldsymbol{x}^{\mathrm{m}}_{n_x}\left(k-\delta^{\mathrm{m}}_{n_x,1}\right) \in X^{\mathrm{t2}}_{n_x,1}(k)
\end{aligned}
$$

$$\text{and} \quad \ldots \quad \text{and} \quad \boldsymbol{x}^{\mathrm{m}}_{n_x}\left(k-\delta^{\mathrm{m}}_{n_x,d^{\mathrm{m}}_{n_x}}\right) \in X^{\mathrm{t2}}_{1,d^{\mathrm{m}}_{n_x}}(k)$$

$$\text{then} \quad \boldsymbol{u}(k) = \boldsymbol{f}^{\mathrm{c}}\Big(\boldsymbol{\theta}^{\mathrm{c}}(k), \boldsymbol{x}^{\mathrm{m}}(k), \boldsymbol{x}^{\mathrm{m}}_{\mathrm{d}}(k), J(k-1)\Big), \tag{10.2}$$

with $\boldsymbol{x}^{\mathrm{m}}_i(k), i \in \{1,\ldots,n_x\}$ state vectors, which together construct a vector $\boldsymbol{x}^{\mathrm{m}}(k)$ at time step k, $\delta^{\mathrm{m}}_{i,j}$, $j \in \{0,\ldots,d^{\mathrm{m}}_i\}$ and d^{m}_i non-negative integers, $X^{\mathrm{t2}}_{i,j}(k)$ type-2 fuzzy sets that may be time-varying (i.e., their parameters collected in vectors $\boldsymbol{\theta}^X_{i,j}$ may be updated at tuning time steps $k \in \mathbb{K}^{\mathrm{tune}}$, which are not necessarily every time step $k \in \mathbb{K}$), $\boldsymbol{u}(k)$ the control input vector at time step k, $\boldsymbol{f}^{\mathrm{c}}$ a nonlinear function, $\boldsymbol{\theta}^{\mathrm{c}}(k)$ a vector including the adjustment parameters for the adaptive fuzzy controller at time step k (similar to the type-2 fuzzy sets of the fuzzy controller, the elements of $\boldsymbol{\theta}^{\mathrm{c}}$ may be updated at tuning time steps $k \in \mathbb{K}^{\mathrm{tune}}$), $\boldsymbol{x}^{\mathrm{m}}_{\mathrm{d}}(k)$ a vector that includes all the delayed states with e_x elements ($e_x = \sum_{i=1}^{n_u}\sum_{j=0}^{d^{\mathrm{m}}_i}\delta^{\mathrm{m}}_{i,j}$) at time step k, and $J(\cdot)$ the cost function of the subsystem, given by

$$J(k) = \mathcal{J}(\boldsymbol{x}^{\mathrm{m}}(k), \Delta\boldsymbol{u}(k)), \tag{10.3}$$

where $\mathcal{J} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ is, in general, a nonlinear function that computes the cost function, and $\Delta\boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1)$.

Note that due to the limitations in the memory and storage space, we do not save the estimated values of the delayed state variables. Whenever we need to know these values, we estimate them via (10.1) using the updated parameter values.

10.4.3 Parameter identification for the fuzzy model

The parameters $\boldsymbol{\theta}^U_{i,j}(k)$ of the type-2 fuzzy sets in the antecedent of the fuzzy rules and the parameter vector $\boldsymbol{\theta}^{\mathrm{m}}(k)$ in the consequent of the fuzzy rules for the fuzzy model of a subsystem are re-identified at time steps $k \in \mathbb{K}^{\mathrm{identify}}$. The set $\mathbb{K}^{\mathrm{identify}}$ includes those time steps at which a measurement of the states of the subsystem is available. Re-identification of $\boldsymbol{\theta}^U_{i,j}(k)$ is based on the most recent value determined for these parameters, i.e., $\boldsymbol{\theta}^U_{i,j}\left(k - k^{-,\mathrm{identify}}\right)$ and the measured state $\boldsymbol{x}(k)$ of the subsystem at time step $k \in \mathbb{K}^{\mathrm{identify}}$. Note that $k^{-,\mathrm{identify}}$ is the most recent element of $\mathbb{K}^{\mathrm{identify}}$ before time step k. To re-identify $\boldsymbol{\theta}^U_{i,j}(k)$, we use dataset $\mathbb{D}^{\mathrm{m}}_k$, which includes the data that has been collected in the last N sampling time steps. Then the antecedent's parameters are updated as follows (also see [26]):

$$\boldsymbol{\theta}^U_{i,j}(k) = \Theta^U\left(\boldsymbol{\theta}^U_{i,j}(k-1), \mathbb{D}^{\mathrm{m}}_k\right), \qquad i \in \{1,\ldots,n_u\}, \quad j \in \{0,\ldots,d^{\mathrm{c}}_i\}, \tag{10.4}$$

and the updated type-2 fuzzy sets for the antecedents of the fuzzy rules of the submodels are obtained by

$$U^{\mathrm{t2}}_{i,j}(k) = \Pi_U\left(\boldsymbol{\theta}^U_{i,j}(k)\right), \qquad i \in \{1,\ldots,n_u\}, \quad j \in \{0,\ldots,d^{\mathrm{c}}_i\}, \tag{10.5}$$

where $\Theta^U(\cdot)$ is a nonlinear operator used for updating the parameters of the type-2 fuzzy sets, and $\Pi_U(\cdot)$ is an operator that receives the corresponding parameters and gives the fuzzy sets

$U^{\mathrm{t2}}_{i,j}$.

The parameter vector $\boldsymbol{\theta}^{\mathrm{m}}$ of the consequent of the rules for the fuzzy model can be updated at time step $k \in \mathbb{K}^{\mathrm{identify}}$ by minimizing the cumulative error of the states estimated by the fuzzy model within a time window $\mathbb{L}^{\mathrm{identify}}(k)$ that includes the p^{identify} most recent elements of $\mathbb{K}^{\mathrm{identify}}$ at time step k, w.r.t. the measured states at these time steps. Therefore, we can write

$$\begin{aligned}&\min_{\boldsymbol{\theta}^{\mathrm{m}}(k)} \left(\sum_{l \in \mathbb{L}^{\mathrm{identify}}(k)} \left\| \boldsymbol{x}(l) - \boldsymbol{x}^{\mathrm{m}}\left(l \middle| \boldsymbol{\theta}^{\mathrm{m}}(l) = \boldsymbol{\theta}^{\mathrm{m}}(k)\right) \right\| \right),\\ &\text{s.t.}\\ &\qquad \text{(10.1) and (10.2) hold for } l \in \mathbb{L}^{\mathrm{identify}}(k),\end{aligned} \tag{10.6}$$

which is a minimization problem with $\boldsymbol{\theta}^{\mathrm{m}}(k)$ the optimization variable, and with $\boldsymbol{x}(l)$ the measured value of the state vector of the controlled system at time step l. In (10.6), the fuzzy model given by (10.1) is re-run within the time window $\mathbb{L}^{\mathrm{identify}}(k)$, considering the updated $\boldsymbol{\theta}^{\mathrm{m}}(k)$ at time step k.

Note that the optimization problem given by (10.6) is in general a nonlinear, nonsmooth and nonconvex optimization problem, hence it can be solved by standard optimization algorithms such as pattern search, genetic algorithm, or gradient-based optimization approaches, using multiple starting points (see Chapter 7 for detailed information about these optimization methods).

Figure 10.6 shows the identification procedure for the fuzzy model. As we see in this figure, the current and the delayed input vectors are injected into the fuzzy model. At the identification time steps $k \in \mathbb{K}^{\mathrm{identify}}$, the state $\boldsymbol{x}(k)$ of the system is measured, and the corresponding error of the estimated state by the fuzzy model w.r.t. the measured state, which is indicated by $\boldsymbol{e}^{\mathrm{m}}(k)$ in Figure 10.6, is sent to the identification module, where $\boldsymbol{\theta}^{\mathrm{m}}(k)$ is re-identified via (10.6). Moreover, the parameters $\boldsymbol{\theta}^{\mathrm{U}}_{i,j}(k)$ of the type-2 fuzzy sets are re-identified using (10.4).

10.4.4 Parameter tuning for the fuzzy controller

The second layer of each fuzzy controller is responsible for tuning the parameters of the fuzzy rules in the first layer of the controller at time steps $k \in \mathbb{K}^{\mathrm{tune}}$. The set $\mathbb{K}^{\mathrm{tune}}$ is constructed using a combination of a repetitive approach and an event-triggered one. Therefore, at every N^{tune} time steps tuning can occur, in addition to some intermediate time steps at which a tuning criterion is triggered (note that $\mathbb{K}^{\mathrm{tune}}$ may be different from $\mathbb{K}^{\mathrm{identify}}$). The second layer of a fuzzy controller updates the controller's parameters $\boldsymbol{\theta}^{X}_{i,j}$ in a similar way as for $\boldsymbol{\theta}^{U}_{i,j}$, i.e.,

$$\boldsymbol{\theta}^{X}_{i,j}(k) = \Theta^{X}\left(\boldsymbol{\theta}^{X}_{i,j}(k-1), \mathbb{D}^{\mathrm{c}}_{k}\right). \tag{10.7}$$

Additionally, we have

$$X^{\mathrm{t2}}_{i,j}(k) = \Pi_{X}\left(\boldsymbol{\theta}^{X}_{i,j}(k)\right), \qquad i \in \{1, \ldots, n_x\}, \quad j \in \{0, \ldots, d^{\mathrm{m}}_{i}\}, \tag{10.8}$$

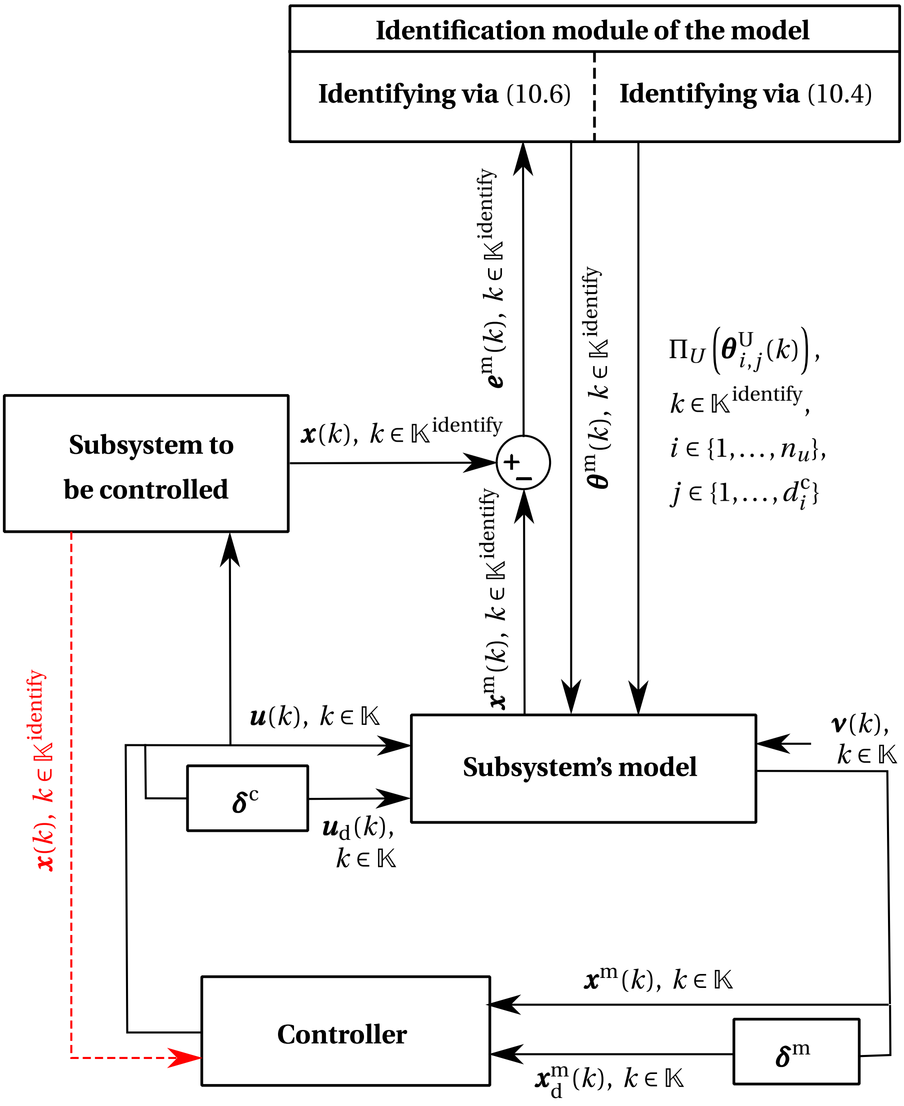

Identification module of the model
Identifying via (10.6)
Identifying via (10.4)
$e^{m}(k),\ k \in \mathbb{K}^{\text{identify}}$
$\theta^{m}(k),\ k \in \mathbb{K}^{\text{identify}}$
$\Pi_U\left(\theta^{U}_{i,j}(k)\right)$, $k \in \mathbb{K}^{\text{identify}}$, $i \in \{1,\dots,n_u\}$, $j \in \{1,\dots,d^{c}_{i}\}$
Subsystem to be controlled
$x(k),\ k \in \mathbb{K}^{\text{identify}}$
+
−
$x^{m}(k),\ k \in \mathbb{K}^{\text{identify}}$
$x(k),\ k \in \mathbb{K}^{\text{identify}}$
$u(k),\ k \in \mathbb{K}$
δ^{c}
$u_{d}(k),\ k \in \mathbb{K}$
Subsystem's model
$v(k),\ k \in \mathbb{K}$
Controller
$x^{m}(k),\ k \in \mathbb{K}$
δ^{m}
$x^{m}_{d}(k),\ k \in \mathbb{K}$

Figure 10.6: Closed-loop model-based control system with state delays (model parameters are re-identified at $k \in \mathbb{K}^{\text{identify}}$, when measurements from the state of the subsystem are available).

where $\Theta^{X}(\cdot)$ is a nonlinear operator used for updating the parameters of the type-2 fuzzy sets, $\mathbb{D}^{c}_{k}$ for $k \in \mathbb{K}^{\text{tune}}$ is a dataset that includes the data that has been collected in the last N sampling time steps, and $\Pi_{X}(\cdot)$ is an operator that receives the corresponding parameters and produces the fuzzy sets $X^{t2}_{i,j}$.

The aim of the proposed two-layer controller is to reduce the realized value of a predefined cost function (see (10.3)). Therefore, the parameters θ^{c} of the fuzzy controller should be updated taking into account this aim of the controller. We define the following minimization

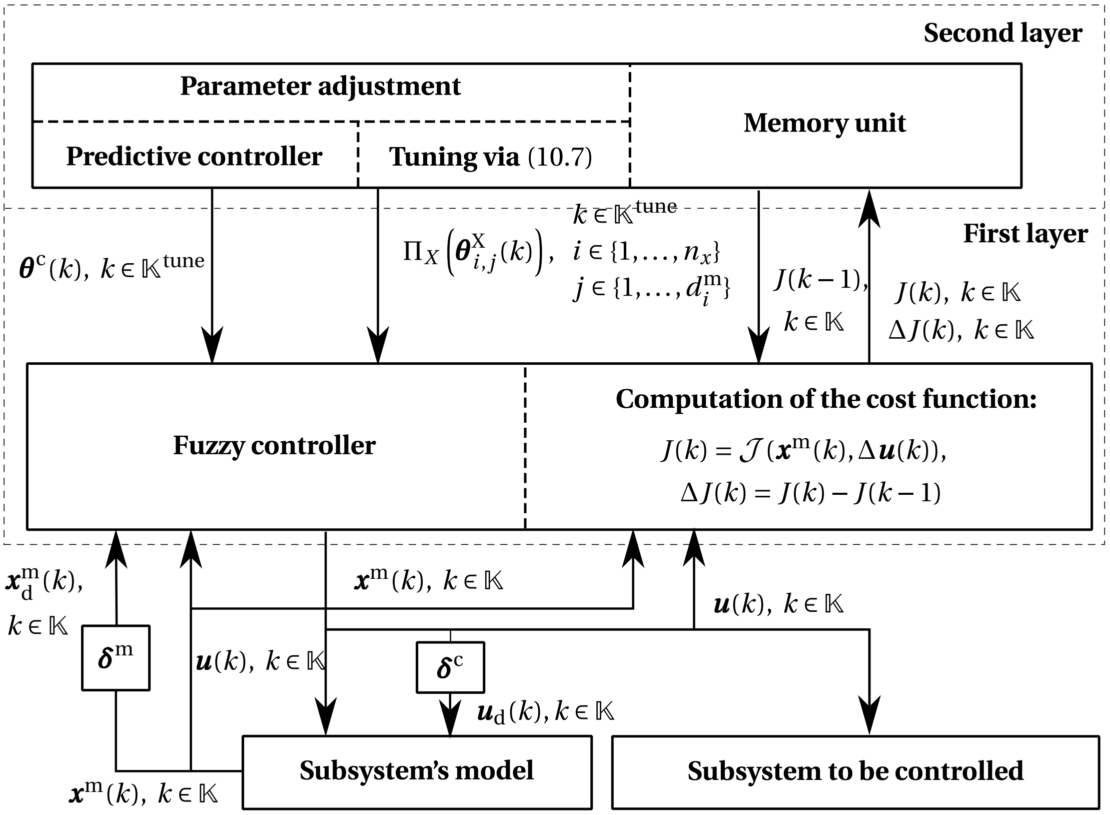

Second layer
Parameter adjustment
Memory unit
Predictive controller
Tuning via (10.7)
$k \in \mathbb{K}^{\text{tune}}$
$\boldsymbol{\theta}^{\text{c}}(k),\ k \in \mathbb{K}^{\text{tune}}$
$\Pi_X\left(\boldsymbol{\theta}^{\text{X}}_{i,j}(k)\right),\ i \in \{1,\ldots,n_x\}$
$j \in \{1,\ldots,d^{\text{m}}_i\}$
$J(k-1),$
$k \in \mathbb{K}$
First layer
$J(k),\ k \in \mathbb{K}$
$\Delta J(k),\ k \in \mathbb{K}$
Fuzzy controller
Computation of the cost function:
$J(k) = \mathcal{J}(\boldsymbol{x}^{\text{m}}(k), \Delta \boldsymbol{u}(k)),$
$\Delta J(k) = J(k) - J(k-1)$
$\boldsymbol{x}^{\text{m}}_{\text{d}}(k),$
$k \in \mathbb{K}$
$\boldsymbol{\delta}^{\text{m}}$
$\boldsymbol{x}^{\text{m}}(k),\ k \in \mathbb{K}$
$\boldsymbol{u}(k),\ k \in \mathbb{K}$
$\boldsymbol{u}(k),\ k \in \mathbb{K}$
$\boldsymbol{\delta}^{\text{c}}$
$\boldsymbol{u}_{\text{d}}(k), k \in \mathbb{K}$
Subsystem's model
Subsystem to be controlled
$\boldsymbol{x}^{\text{m}}(k),\ k \in \mathbb{K}$

Figure 10.7: A two-layer architecture for online adaptive tuning of the parameters of a fuzzy controller.

problem, where $\boldsymbol{\theta}^{\text{c}}(k)$ is the optimization variable:

$$\min_{\boldsymbol{\theta}^{\text{c}}(k)} \left(w^{\text{p}} \sum_{l \in \mathbb{L}^{\text{tune}}(k)} \mathcal{J}\left(\boldsymbol{x}^{\text{m}}(l|\boldsymbol{\theta}^{\text{c}}(l) = \boldsymbol{\theta}^{\text{c}}(k)), \Delta \boldsymbol{u}(l|\boldsymbol{\theta}^{\text{c}}(l) = \boldsymbol{\theta}^{\text{c}}(k))\right) + \right.$$
$$\left. w^{\text{f}} \left| \Theta(k) - \sum_{l=k}^{k+2} \mathcal{J}\left(\boldsymbol{x}^{\text{m}}(l|\boldsymbol{\theta}^{\text{c}}(l) = \boldsymbol{\theta}^{\text{c}}(k)), \Delta \boldsymbol{u}(l|\boldsymbol{\theta}^{\text{c}}(l) = \boldsymbol{\theta}^{\text{c}}(k))\right) \right| \right), \tag{10.9}$$

s.t.

(10.1) and (10.2) hold for $l \in \mathbb{L}^{\text{tune}}(k)$,

where w^{p} and w^{f} are weighting factors, $\mathbb{L}^{\text{tune}}(k)$ is a time window that includes the most recent p elements of $\mathbb{K}$ before time step k, and $\Theta(k)$ is the optimal cumulative future cost function from now until two time steps further in the future, which is computed by the predictive and model-based controller in the second layer of the proposed control architecture. Details regarding computation of $\Theta(k)$ will be presented in Section 10.4.5. The first term in the minimization problem of (10.9), i.e.,

$$\sum_{l \in \mathbb{L}^{\text{tune}}(k)} \mathcal{J}\left(\boldsymbol{x}^{\text{m}}(l|\boldsymbol{\theta}^{\text{c}}(l)=\boldsymbol{\theta}^{\text{c}}(k)), \Delta\boldsymbol{u}(l|\boldsymbol{\theta}^{\text{c}}(l)=\boldsymbol{\theta}^{\text{c}}(k))\right),$$

formulates the cumulative cost of the last p time steps. We should re-run the fuzzy model of (10.1) to re-estimate the previous states using $\boldsymbol{\theta}^{\text{c}}(k)$ for the parameters of the antecedent. The second term in the minimization problem of (10.9), i.e.,

$$\left|\Theta(k) - \sum_{l=k}^{k+2} \mathcal{J}\left(\boldsymbol{x}^{\text{m}}(l|\boldsymbol{\theta}^{\text{c}}(l)=\boldsymbol{\theta}^{\text{c}}(k)), \Delta\boldsymbol{u}(l|\boldsymbol{\theta}^{\text{c}}(l)=\boldsymbol{\theta}^{\text{c}}(k))\right)\right|,$$

formulates the difference between an optimal value for the cumulative cost function for two time steps in future and the realized value of the cumulative future cost function, if the fuzzy controller is used to control the subsystem in the coming two time steps (see the next section for more details regarding using two future time steps). These two terms are weighted by w^{p} and w^{f} in (10.9). The optimization problem given by (10.9) is in general a nonlinear, nonsmooth and nonconvex optimization problem, and can be solved by standard optimization algorithms such as pattern search, genetic algorithm, or gradient-based optimization approaches, using multiple starting optimization points (see Chapter 7 for more details).

Figure 10.7 shows the main structure of the two-layer adaptive fuzzy and predictive model-based controller proposed in this chapter. Note that the illustrated two-layer controller enclosed by the dashed box in the figure, elaborates the controller block that is shown in Figure 10.6. From Figure 10.7, the type-2 fuzzy controller, which is located in the first layer of the control architecture, computes the control input $\boldsymbol{u}(k)$ at every time step $k \in \mathbb{K}$. At tuning time steps $k \in \mathbb{K}^{\text{tune}}$, the parameters $\boldsymbol{\theta}^{\text{X}}_{i,j}(k)$ of the type-2 fuzzy sets are re-tuned using (10.7). Additionally, the parameter vector $\boldsymbol{\theta}^{\text{c}}(k)$ of the fuzzy controller is also re-tuned using the predictive and model-based controller, which is located in the second layer of the control architecture, and via the minimization problem of (10.9).

10.4.5 Predictive and optimization-based controller

As indicated before, the two-layer control architecture proposed in this chapter is considered for use in a multi-agent control system. In a multi-agent control system, in general, all or some agents are connected, i.e., their local dynamics affect each other's dynamics. Hence, each of these agents should take into account the connected agents whenever it manipulates its local dynamics by implementing a control input that is produced by the two-layer controller. For the sake of simplicity, we consider two connected agents and indicate them by the indices 1 and 2 (see Figure 10.8). Hence, from now on, whenever we refer to a variable that has been used in the previous sections, we may add an additional subscript that distinguishes the corresponding agent in the multi-agent architecture. Note that extension of the discussions given in this section to a multi-agent system with more than two agents is straightforward.

Each agent uses a fuzzy model of its subsystem given by (10.1) and a two-layer controller, where the fuzzy controller in the first layer is given by (10.2). Some elements of the state vector

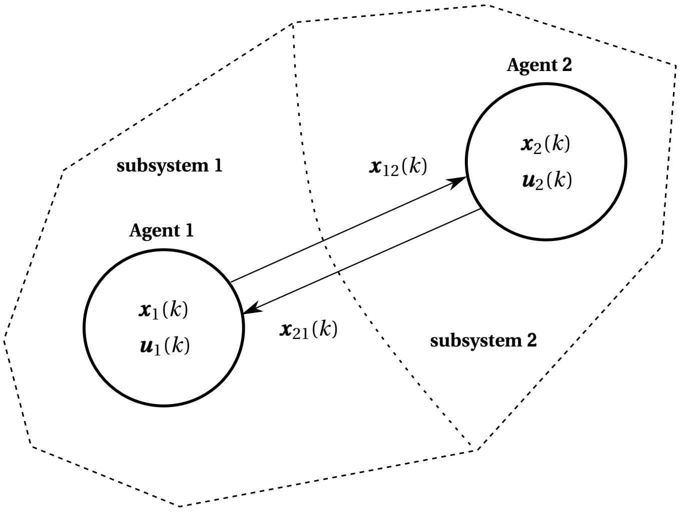


Figure 10.8: Two connected agents with fuzzy models of subsystems and two-layer fuzzy and predictive model-based controllers.

of one subsystem may directly affect the dynamics of the other subsystem. Suppose that $\boldsymbol{x}_{12}$ denotes the subvector of $\boldsymbol{x}_1$ that directly affects the dynamics of subsystem 2, and $\boldsymbol{x}_{21}$ denotes the subvector of $\boldsymbol{x}_2$ that directly affects the dynamics of subsystem 1 (see Figure 10.8). Hence, we have

$$\boldsymbol{v}_1(k) = \boldsymbol{x}_{21}(k), \tag{10.10}$$

$$\boldsymbol{v}_2(k) = \boldsymbol{x}_{12}(k). \tag{10.11}$$

Note that we assume that the two agents have synchronized time steps. The effect of the interactions among connected agents is represented in the second term of the minimization problem (10.9), where we consider the error between the cumulative future cost function resulting from using the fuzzy controller w.r.t. Θ, which is the cumulative future cost function computed by the predictive controller.

In this section, we show how $\Theta(k)$ can be determined[2] using a decomposition method [12], such as the primal decomposition approach. Consider the following minimization problem:

$$\min_{\phi} \Phi(\phi) = \min_{\phi_1,\phi_2,\theta} \big(\Phi_1(\phi_1,\theta) + \Phi_2(\phi_2,\theta)\big), \tag{10.12}$$

with private variables ϕ_1 and ϕ_2, and complicating variable θ, i.e., $\phi = [\phi_1, \phi_2, \theta]^\top$. The minimization problem (10.12) yields the following equivalent optimization problem, called the "master problem", which involves only the complicating variable as the optimization variable:

$$\min_{\theta} (\Theta_1(\theta) + \Theta_2(\theta)). \tag{10.13}$$

[2] For the sake of simplicity in the mathematical notations from now on we skip the argument k.

In (10.13), $\Theta_1(\theta)$ and $\Theta_2(\theta)$ are optimal values for the following two optimization problems:

- $\min\limits_{\phi_1} \Phi_1(\phi_1, \theta)$,
- $\min\limits_{\phi_2} \Phi_1(\phi_2, \theta)$.

Using a standard optimization method, the master problem (10.13) can be solved to find an optimal θ^{opt}, and to determine optimal values of $\Theta_1(\theta^{\text{opt}})$ and $\Theta_2(\theta^{\text{opt}})$ correspondingly.

Next, we formulate the optimization problem of the multi-agent control system shown in Figure 10.8 using the primal decomposition approach. The overall cost function of the system at time control step k is indicated by $J^{\text{sys}}(k)$. Note that from (10.3), $\boldsymbol{u}_1(k)$ affects $J^{\text{sys}}(k)$ and $J^{\text{sys}}(k+1)$ by affecting the cost function of subsystem 1. The same holds for agent 2. Moreover, from (10.1), the decision $\boldsymbol{u}_1(k)$ of agent 1 at time step k affects the state of subsystem 1 at the next time step, i.e., $\boldsymbol{x}_1^{\text{m}}(k+1)$, whereas the decision $\boldsymbol{u}_2(k)$ of agent 2 at time step k affects $\boldsymbol{x}_2^{\text{m}}(k+1)$. The state $\boldsymbol{x}_1^{\text{m}}(k+1)$ acts as disturbance for the connected subsystem 2 at time step $k+1$, while $\boldsymbol{x}_2^{\text{m}}(k+1)$ acts as disturbance for subsystem 1 at time step $k+1$. Therefore, based on from (10.1), the effect of $\boldsymbol{u}_1(k)$ on the state of subsystem 2 and the effect of $\boldsymbol{u}_2(k)$ on the state of subsystem 1 are observed at time step $k+2$ only. Hence, in addition to $J^{\text{sys}}(k)$ and $J^{\text{sys}}(k+1)$, $J^{\text{sys}}(k+2)$ will also be affected by $\boldsymbol{u}_1(k)$ and $\boldsymbol{u}_2(k)$.

Therefore, in order to find an optimal value of the control input $\boldsymbol{u}(k) = [\boldsymbol{u}_1(k) \quad \boldsymbol{u}_2(k)]^\top$ at time step k, in addition to the current cost functions of subsystems 1 and 2, we also look at the cost functions of these subsystems within two time steps from k considering all the effective control inputs as the optimization variable. Hence, at time step k, we minimize the summation of the cost functions of subsystems 1 and 2 at time steps k, $k+1$, and $k+2$, finding optimal profiles for $\boldsymbol{u}(k), \boldsymbol{u}(k+1), \boldsymbol{u}(k+2)$. So we obtain

$$\min_{\substack{\boldsymbol{u}_1(k),\boldsymbol{u}_1(k+1),\boldsymbol{u}_1(k+2),\\ \boldsymbol{u}_2(k),\boldsymbol{u}_2(k+1),\boldsymbol{u}_2(k+2)}} \sum_{l=k}^{k+2} \sum_{\ell\in\{1,2\}} \mathcal{J}_\ell\left(\boldsymbol{x}_\ell^{\text{m}}(l), \Delta\boldsymbol{u}_\ell(l)\right). \tag{10.14}$$

Next, from (10.1), we can write:

$$\begin{aligned}&\mathcal{J}_\ell\left(\boldsymbol{x}_\ell^{\text{m}}(k+1), \|\boldsymbol{u}_\ell(k+1) - \boldsymbol{u}_\ell(k)\|\right) = \\ &\mathcal{J}_\ell\Big(\boldsymbol{f}_\ell^{\text{m}}\Big(\boldsymbol{\theta}_\ell^{\text{m}}(k+1), \boldsymbol{x}_\ell^{\text{m}}(k), \boldsymbol{u}_\ell(k), \boldsymbol{u}_{\text{d},\ell}(k), \boldsymbol{x}_{\bar{\ell}}^{\text{m}}(k)\Big), \|\boldsymbol{u}_\ell(k+1) - \boldsymbol{u}_\ell(k)\|\Big).\end{aligned} \tag{10.15}$$

Using (10.1) once more, we obtain

$$\begin{aligned}&\mathcal{J}_\ell\left(\boldsymbol{x}_\ell^{\text{m}}(k+2), \Delta\boldsymbol{u}_\ell(k+2)\right) = \\ &\mathcal{J}_\ell\Big(\boldsymbol{f}_\ell^{\text{m}}\Big(\boldsymbol{\theta}_\ell^{\text{m}}(k+2), \boldsymbol{x}_\ell^{\text{m}}(k+1), \boldsymbol{u}_\ell(k+1), \boldsymbol{u}_{\text{d},\ell}(k+1), \boldsymbol{x}_{\bar{\ell}}^{\text{m}}(k+1)\Big), \|\boldsymbol{u}_\ell(k+2) - \boldsymbol{u}_\ell(k+1)\|\Big),\end{aligned} \tag{10.16}$$

with

$$\begin{aligned}
&\boldsymbol{x}_{\ell}^{\mathrm{m}}(k+1) = \boldsymbol{f}_{\ell}^{\mathrm{m}}\Big(\boldsymbol{\theta}_{\ell}^{\mathrm{m}}(k+1), \boldsymbol{x}_{\ell}^{\mathrm{m}}(k), \boldsymbol{u}_{\ell}(k), \boldsymbol{u}_{\mathrm{d},\ell}(k), \boldsymbol{x}_{\bar{\ell}}^{\mathrm{m}}(k)\Big),\\
&\boldsymbol{x}_{\bar{\ell}}^{\mathrm{m}}(k+1) = \boldsymbol{f}_{\bar{\ell}}^{\mathrm{m}}\Big(\boldsymbol{\theta}_{\bar{\ell}}^{\mathrm{m}}(k+1), \boldsymbol{x}_{\bar{\ell}}^{\mathrm{m}}(k), \boldsymbol{u}_{\bar{\ell}}(k), \boldsymbol{u}_{\mathrm{d},\bar{\ell}}(k), \boldsymbol{x}_{\ell}^{\mathrm{m}}(k)\Big),\\
&\boldsymbol{u}_{\mathrm{d},\ell}(k+1) = [\boldsymbol{u}_{\ell}^{\top}(k), \boldsymbol{u}_{\mathrm{d},\ell}^{\top}(k)]^{\top}.
\end{aligned} \tag{10.17}$$

Then for $\ell \in \{1,2\}$, we define

$$Q_{\ell}(\boldsymbol{u}_{\ell}(k), \boldsymbol{u}_{\ell}(k+1), \boldsymbol{u}_{\ell}(k+2), \boldsymbol{u}_{\bar{\ell}}(k)) := \sum_{l=k}^{k+2} \mathcal{J}_{\ell}\left(\boldsymbol{x}_{\ell}^{\mathrm{m}}(l), \Delta\boldsymbol{u}_{\ell}(l)\right), \tag{10.18}$$

with $\bar{\ell} = 2$ for $\ell = 1$ and $\bar{\ell} = 1$ for $\ell = 2$. Equation (10.18) simplifies the given optimization problem to a formulation similar to (10.12) with

- Complicating variables:

$$\boldsymbol{\theta}(k) := \left[\boldsymbol{u}_{\ell}^{\top}(k), \boldsymbol{u}_{\bar{\ell}}^{\top}(k)\right]^{\top}. \tag{10.19}$$

- Private variables:

$$\begin{aligned}
\boldsymbol{\phi}_{\ell}(k) &:= \left[\boldsymbol{u}_{\ell}^{\top}(k+1), \boldsymbol{u}_{\ell}^{\top}(k+2)\right]^{\top}\\
\boldsymbol{\phi}_{\bar{\ell}}(k) &:= \left[\boldsymbol{u}_{\bar{\ell}}^{\top}(k+1), \boldsymbol{u}_{\bar{\ell}}^{\top}(k+2)\right]^{\top}
\end{aligned}. \tag{10.20}$$

- Decomposed (local) cost functions:

$$\begin{aligned}
\Phi_{\ell}(k, \boldsymbol{\phi}_{\ell}, \boldsymbol{\theta}) &= Q_{\ell}(\boldsymbol{\phi}_{\ell}, \boldsymbol{\theta})\\
\Phi_{\bar{\ell}}(k, \boldsymbol{\phi}_{\bar{\ell}}, \boldsymbol{\theta}) &= Q_{\bar{\ell}}(\boldsymbol{\phi}_{\bar{\ell}}, \boldsymbol{\theta})
\end{aligned}. \tag{10.21}$$

By solving the resulting optimization problem, we obtain an optimal value for $\boldsymbol{\theta}$, $\boldsymbol{\phi}_{\ell}$, and $\boldsymbol{\phi}_{\bar{\ell}}$ denoted by $\boldsymbol{\theta}^{\mathrm{opt}}$, $\boldsymbol{\phi}_{\ell}^{\mathrm{opt}}$, and $\boldsymbol{\phi}_{\bar{\ell}}^{\mathrm{opt}}$, and also for $\Phi_{\ell}(k, \boldsymbol{\phi}_{\ell}, \boldsymbol{\theta})$ and $\Phi_{\bar{\ell}}(k, \boldsymbol{\phi}_{\bar{\ell}}, \boldsymbol{\theta})$ denoted by, respectively, $\Theta_{\ell}(k, \theta^{\mathrm{opt}})$ and $\Theta_{\bar{\ell}}(k, \theta^{\mathrm{opt}})$. In the second term of (10.9), we can then use $\Theta_{\bar{\ell}}(k, \theta^{\mathrm{opt}})$ and $\Theta_{\bar{\ell}}(k, \theta^{\mathrm{opt}})$ to tune the parameters of the fuzzy controller. Since these two optimal values have been computed taking into account the complicating variable, they include the effects of interactions among the connected agents. Note that when (10.9) is solved for a fuzzy controller in a multi-agent control architecture, we assume that the other fuzzy controllers in the system are using the most recent updated values for their tuning parameters.

Also note that using a prediction window including 3 time steps provides the minimum horizon needed for providing the coordination of the connected agents (see the explanations given above at the beginning of this section). However, one may choose a horizon of any multiple of 3 time steps for the predictive controller.

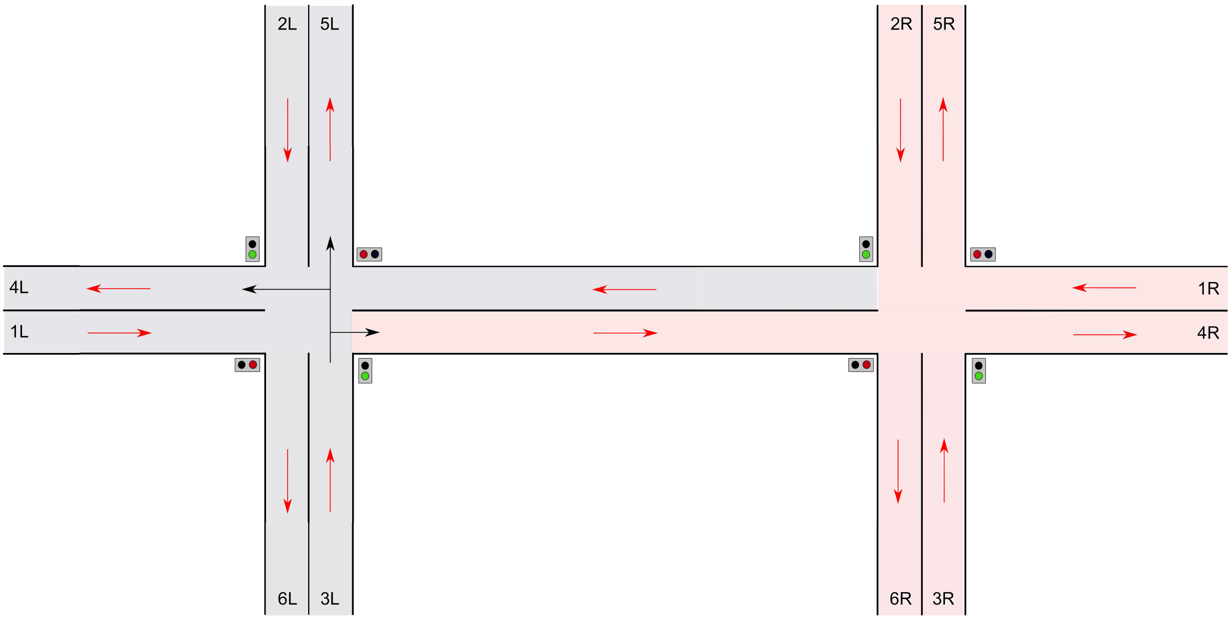


Figure 10.9: Urban traffic network with two adjacent intersections used for the case study.

10.5 Case study

In this section, the two-layer adaptive control system proposed in Section 10.4 is implemented to an urban traffic network consisting of two adjacent intersections (see Figure 10.9). All links that are depicted in Figure 10.9 include two lanes, where the right-hand side lane of each link at an intersection acts as an entrance lane and the left-hand side lane acts as en exit lane (see the red arrows in Figure 10.9). The vehicles entering an intersection via a specific link are allowed to move to any of the exit lanes of the intersection, except for the exit lane on the same link (i.e., U-turns are forbidden at the intersections). The black arrows shown in Figure 10.9, illustrate the three possible routes that can be taken by a vehicle that enters the left-hand side intersection via the southern link. Each of the two intersections is controlled by four traffic signals, where each traffic signal corresponds to all the rights-of-way of the corresponding lane of the intersection. The traffic signals that are located at opposite entrance lanes of an intersection will follow the same schedule (i.e., their green and red phases coincide).

A simulation of half an hour is considered and is repeated twice, once with the two-layer adaptive control system proposed in Section 10.4, where the urban traffic network is divided into two subnetworks colored in grey and in pink in Figure 10.9, and once with a centralized MPC controller for the entire urban traffic network. The traffic on the links is simulated using the extended version of the S-model (see Chapter 6). The parameters used for the case study are given in Table 10.2, where C and C_{con} refer to the capacity of the boundary and the connecting lanes respectively, β is the turning rate for all the lanes in the traffic network (note that there are three possible turning directions for each lane as illustrated in Figure 10.9, and we assume that all the corresponding turning rates are equal), the saturation leaving flow rates for all the lanes are the same and are denoted by μ, c is the cycle time of the two intersections,

Table 10.2: Parameters used for the case study.

C [veh]	C_{con} [veh]	β	μ [veh/s]	c [s]	l^{veh} [m]	v^{free} [m/s]
70	120	1/3	0.8	60	4.5	13.5

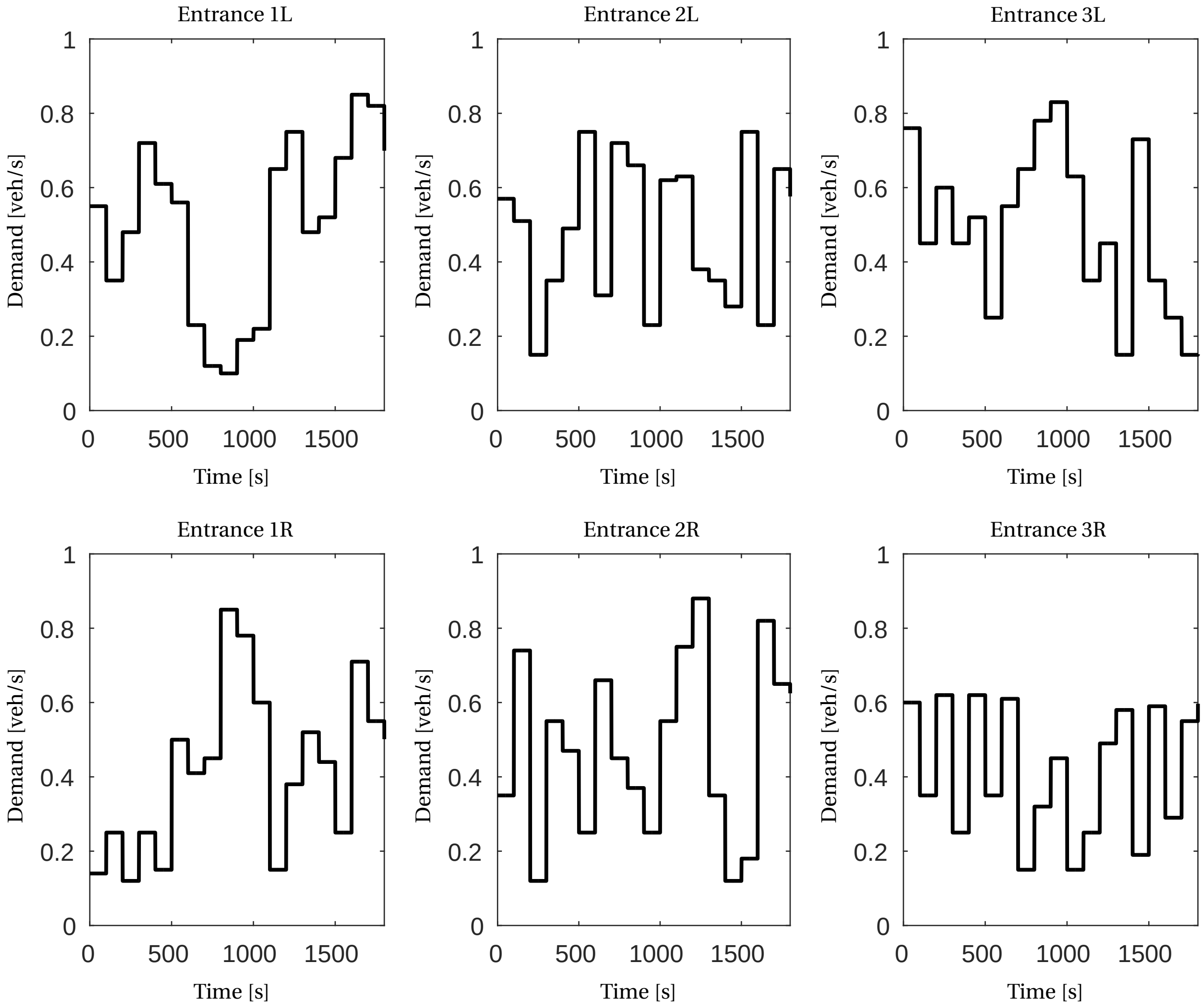


Figure 10.10: Demand in vehicles per second at the entrances of the urban traffic network.

l^{veh} is the average vehicle length, and v^{free} is the free-flow speed for the vehicles on all lanes of the urban traffic network. For both simulations, the minimum and maximum allowed values of the green time are, respectively, 6 s and 55 s. The demand profiles at the entrances of the urban traffic network (i.e., sources 1L, 2L, 3L, 1R, 2R, and 3R in Figure 10.9) for the simulation period are illustrated in Figure 10.10. We assume the simulations to start from an empty traffic network.

The two-layer adaptive control system includes two fuzzy controllers for the two subnetworks in its first layer. These fuzzy controllers include fuzzy rules that have the following structure for each of the entrance lanes of the intersection:

"If *queue length* is *short/medium/long* and *expected cumulative demand* is *low/high*, then green time of the lane = $a_{0,l}+a_{1,l}\cdot$(current queue length)$+a_{2,l}\cdot$(expected cumulative demand)".

The parameters $a_{0,l}$, $a_{1,l}$, and $a_{2,l}$ are the tuning parameters for the entrance lane l. We assume to have a perfect model of the external disturbances. Moreover, the expected cumulative demand is computed from the current time step until the end of the intersection's cycle time. Since each fuzzy controller takes care of four entrance lanes of its intersection, and six fuzzy rules (resulting from the 3 fuzzy state values for the queue length and the 2 fuzzy state values for the expected cumulative demand) govern each of these lanes, the total number of the tuning parameters for each fuzzy controller is 72 (i.e., 4 lanes · 6 rules · 3 parameters). The terms "short", "medium", and "long" for the boundary and the connecting lanes are formulated via triangular fuzzy membership functions that are shown in Figure 10.11. Note that the range of the queue lengths for the boundary lanes is $[0, C]$ and for the connecting lanes is $[0, C_{\text{con}}]$. The triangular fuzzy membership functions corresponding to the terms "low" and "high" for the expected cumulative demand for a lane in the upcoming cycle are shown in Figure 10.12. Note that we assume the demand varies from 0 veh/s to 1 veh/s. Therefore, the range of the expected cumulative demand of a lane per cycle is between 0 [veh] and 1 [veh/s]$\cdot c$ [s], i.e., in $[0, c]$.

In the second control layer, an MPC controller is considered that has a prediction horizon of three, and that uses the S-model as the prediction model. To solve the nonsmooth optimization problem, the MPC controller uses pattern search with a single starting point (note that by repeating the simulation for different numbers of starting points, we noticed that a single starting point works well). Moreover, for tuning the parameters of the fuzzy controller using (10.9), the cost function is considered to be the total time spent by the vehicles, and for estimation of the past states within the time window $\mathbb{L}^{\text{tune}}$ of length 3, we have again used the S-model. The tuning is redone in a periodic way at control time steps 5, 10, 15, 20, and 25.

For the second simulation with a centralized MPC controller, the cost function of the optimization problem is the total time spent by the vehicles in the urban traffic network. We consider a prediction horizon of seven, the S-model as the prediction model, pattern search as the optimization solver, and 5 random starting points per optimization problem for dealing with the nonconvexity of the optimization problem (the number 5 was determined by repeating the simulation for different numbers of starting points each time and by selecting a number that works well for all the control time steps).

For both simulations with the centralized MPC controller and with the two-layer adaptive controller, the total time spent of the vehicles in the urban traffic network has been computed and the computation times for solving the optimization problems have been measured. The results are shown in Tables 10.3 and 10.4. From these results, we see that the total time spent of the vehicles for the two-layer adaptive controller compared with the total time spent of the vehicles for the centralized MPC controller is very satisfactory. Indeed, the reduction of the total time spent using the centralized MPC controller w.r.t. the two-layer adaptive controller is only 2%, while the average computation time for the centralized MPC controller per control step is almost 10.5 s (i.e., 315 [s]/30, with 30 the number of control time steps within the 1/2

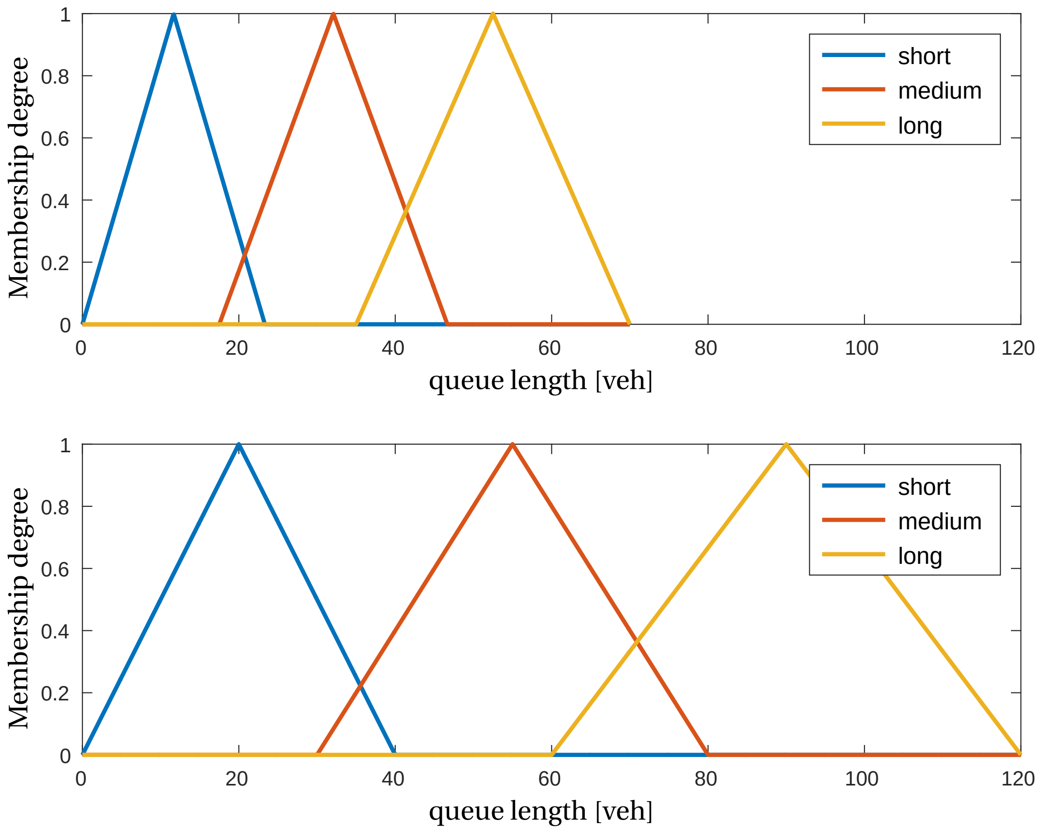


Figure 10.11: Membership functions corresponding to the terms "short", "medium", and "long" for the boundary links (top plot) and the connecting links (bottom plot).

hour simulation run), and the online computation time for the fuzzy controllers is almost zero (fuzzy controllers do not need to solve time-consuming optimization problems). Moreover, the computation time needed for tuning the parameters of the fuzzy controllers (see the last row of Table 10.4 at control time steps 5, 10, 15, 20, and 25) are all less than one sampling cycle of the intersections, i.e., 60 s. This indicates that the updated values of the tuning parameters are always available at the first control time step after each tuning operation.

For this case study, we can conclude that while the CPU time of the online computations for the two-layer adaptive control system is almost zero, its performance is very close to that of a centralized MPC controller with a computation time of 315 s. Hence, the approach proposed in this chapter can result in a promising control system that provides a high performance (i.e., comparable to that of a centralized MPC controller) with a very low computation time.

10.6 Conclusions and future work

In this chapter, we have combined fuzzy logic control, model-based control, and predictive control approaches to obtain an efficient controller with a satisfactory performance for real-time applications. We have considered the problem of controlling multiple connected sub-

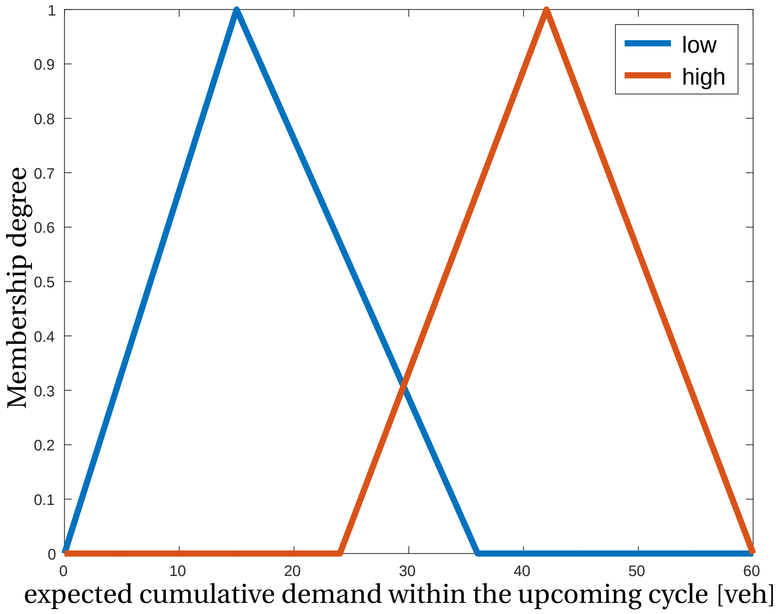


Figure 10.12: Membership functions corresponding to the terms "low" and "high" for the expected cumulative demand for the lanes in the urban traffic network within the upcoming cycle.

Table 10.3: Total time spent of the vehicles [veh.s] for the simulations with the centralized MPC controller and with the two-layer adaptive controller.

Centralized MPC controller	$8.15 \cdot 10^7$
Two-layer adaptive controller	$8.38 \cdot 10^7$

Table 10.4: Computation times [s] for the simulations with the centralized MPC controller (total computation time) and with the two-layer adaptive controller (computation times at periodic tuning time steps).

Centralized MPC controller	Total computation time				
	315				
Two-layer adaptive controller	$k = 5$	$k = 10$	$k = 15$	$k = 20$	$k = 25$
	4.8	10.2	6.3	5.5	3.8

systems with time-delayed states. We have taken into account those cases where due to the costly or missing measurements, slow sensors, limited memory and limited storage space for the measured states, etc. the previous states might not always be available, and hence state augmentation cannot be used. We have proposed type-2 fuzzy rules, which involve type-2 fuzzy sets in their antecedents and a generally nonlinear function of the inputs, outputs, and disturbances in their consequents. Using the proposed fuzzy rules, we have generated type-2 fuzzy models and type-2 fuzzy controllers for the subsystems. The fuzzy controllers, which are fast and hence suit real-time computations, are embedded in a two-layer control architecture. The fuzzy controllers are located in the first layer and perform at all control time steps, and they take care of the uncertainties in the state or in the system's model, and take into account the effect of the delayed states. These fuzzy controllers are also linked to a prediction model that estimates the expected external disturbances of the controlled system in the near future. The second layer of the proposed control system includes an MPC controller that is called at certain tuning time steps, or whenever a predefined criterion is triggered. The MPC controller steers the fuzzy controller in the first layer to tune its parameters adaptively. Additionally, in this chapter, we have discussed different ways that uncertainties may appear (i.e., probabilistic and fuzzy). Accordingly, we have proposed two forms of type-2 fuzzy membership functions, which we call probabilistic-fuzzy and fuzzy-fuzzy membership functions. Finally, we have performed a case study for an urban traffic network to evaluate the performance of the integrated fuzzy, model-based, and predictive controller compared with a centralized MPC controller. The results of the simulation show that the adaptive two-layer fuzzy controller performs almost as good as the centralized MPC controller, while the computation time of the fuzzy controllers are almost zero.

As a topic for future work, we propose for the case study to use trained adaptive fuzzy models for the urban traffic subnetworks and for the external disturbances (i.e., for the demand profiles) based on some historical data, together with fuzzy-fuzzy and probabilistic-fuzzy membership functions for defining the fuzzy rule bases. We also propose to use a traffic microsimulator both for evaluation of the controller, and for identifying the parameters of the fuzzy models. Finally, we propose to extend the case study to a larger traffic network with more than two subnetworks.

Chapter 11

Overall conclusions and topics for future research

In this thesis, we have presented efficient microscopic formulas and algorithms for estimation of the fundamental traffic variables. Additionally, we have developed and investigated fast, accurate, and efficient modeling and control methods for green urban mobility. In that context, we have developed a general mesoscopic modeling framework for integrating traffic flow and emission models, and we have discussed efficient control approaches with a focus on comparing smooth and nonsmooth optimization solvers. Finally, we have included and expanded fuzzy control approaches within a predictive and coordinative multi-agent architecture to develop an efficient control architecture that can also be used for urban traffic control.

In this last chapter of the thesis, we first briefly present the main contributions and results of the thesis. Moreover, we suggest some topics, which we believe to be very promising for further research in the future.

11.1 Main contributions of the thesis

The major contributions of the thesis are as follows:

- **Efficient microscopic formulas for temporal-spatial traffic variables**
  In Part I of the thesis, we have introduced new formulas for accurate estimation of the time-space fundamental traffic variables (in particular the time-space mean speed or the TSMS) using point measurements such as inductive loop detectors. According to the literature (see Section 2.3 of the thesis), loop detectors will still be widely in use during the upcoming 10-15 years. The formulas developed in Part I of the thesis have proven via simulations of Chapter 3 and 4 to be much more accurate than other available formulas for estimation of the average traffic speed. Hence, these formulas are of high importance for those traffic management and analysis centers worldwide that still collect their traffic data from point measurements.

- **A general mesoscopic framework for integration of the traffic flow and emission models**
  In Chapter 6 of the thesis, we have introduced a general framework to interface and integrate macroscopic traffic flow models and microscopic emission models. As a result, a new mesoscopic integrated flow-emission model is obtained. Due to its high accuracy and high speed in computations, the proposed modeling approach is very suitable for real-time model-based control approaches.

- **Efficient smoothening methods for nonsmooth models and optimization problems**
  In Chapter 7, we have proposed general smoothening methods that can be used to make the nonsmooth mathematical models of different physical systems, and the nonsmooth constrains and cost functions of optimization problems smooth. Consequently, available efficient gradient-based optimization approaches can be implemented to solve the resulting smooth optimization problem.

- **Coordinative and predictive multi-agent type-2 fuzzy control**
  In Part III of the thesis, we have proposed a general adaptive scheme for producing fuzzy rules that include type-2 fuzzy sets in order to deal with uncertainties in the datasets. We have also introduced two forms of type-2 fuzzy membership functions that we have called probabilistic-fuzzy and fuzzy-fuzzy membership functions. Using the proposed scheme, we have developed a two-layer adaptive fuzzy control architecture that is capable of tuning its parameters adaptively. We have also used a predictive and optimization-based controller next to the fuzzy control system to provide coordination and interaction among different fuzzy controllers within a multi-agent control architecture.

In addition to the major contributions given above, in this thesis we have also considered the following additional contributions:

- Considering the initial conditions of the traffic estimations that are based on point measurements.

- Developing an extension to the urban traffic flow S-model for increased accuracy.

- Formulating the endpoint penalties for urban traffic MPC.

11.2 Suggestions for further research

In this section, we introduce challenging open issues and topics for future research. These topics are summarized in the following list.

Further theoretical research

In continuation of the topics that have been discussed and developed in this thesis, we propose the following fundamental and theoretical topics for further research.

- **Consideration of lane changing in approximating the trajectories of the vehicles**
  In Chapter 4 of the thesis, we have proposed some approaches for approximating the trajectories of vehicles in between two consecutive loop detectors. In these formulations, we have not considered the effect of lane changing by the vehicles. In order to have a possibly more realistic trajectory approximation method, it will be of interest to consider well-known lane-changing models in the formulations.

- **Nonsmooth analysis of nonsmooth optimization problems**
  The term “nonsmooth analysis” [29] refers to the theory of developing differential calculus for those functions that are not differentiable. The main elements of nonsmooth analysis include: generalized gradients and Jacobians, proximal subgradients, sub-differentials, and generalized directional derivatives. Nonsmooth analysis is normally used when linearization of nonlinear (nonsmooth) systems is not applicable. Further theoretical investigation and analysis is of interest, in particular, in the fields of optimal control, stability analysis, distributed control, hybrid systems, and LPV (linear parameter-varying) control systems.

Further application-based research

In addition to the applications we have considered in this thesis, we propose the following application-based topics for further research.

- **Extensive validation of the formulas introduced in Chapter 3 for estimation of the TSMS**
  The formulas developed in Chapter 3 for estimation of the time-space-mean speed and tight upper and lower bounds for this mean speed can more extensively be validated using real-life datasets. We also propose an extensive analysis of different cases for detecting the traffic scenarios and conditions for which each of the formulas discussed in Chapter 3 might give a more reliable estimate of the average traffic speed.

- **Validation of the approach of Chapter 4 for approximating the trajectories of the vehicles**
  The trajectory approximating approach proposed in Chapter 4, has not been tested and evaluated in this thesis. Using real-life data for this aim can help to evaluate and, if needed, to improve the proposed formulations. Note that improvement of the proposed trajectory approximator might need a detailed microscopic study of different traffic behaviors and psychological aspects of different drivers in making their decisions.

- **Calibration and validation of the integrated mesoscopic model for different flow and emission models**
  The proposed general integrating and interfacing framework for urban traffic flow and emission models has been used in Chapter 6 to integrate the flow S-model [98] and VT-micro [3]. The resulting model has proven to provide a balanced trade-off between a

high computation speed and a high accuracy of computations, which makes it a perfect candidate for the prediction model of an urban traffic MPC controller. Hence, it is of value to integrate other macroscopic flow models (e.g., the cell transmission model [33]) and microscopic emission model (e.g., VERSIT+ [95]), and to calibrate the resulting mesoscopic model with real-life data and various demand scenarios.

- **Extensive practical comparison between nonsmooth and smooth approaches for nonsmooth optimization problems**
  Further practical investigations are recommended for determining the advantages and disadvantages of using smooth analysis, which has been proposed in this thesis in Chapter 7 compared with the advantages and disadvantages of using the known approaches of nonsmooth analysis (see, e.g., [62]). Some aspects that might be of interest within this context are further analysis and reasoning regarding the relatively poor performance of the known nonsmooth optimization solvers such as the genetic algorithm and the pattern search compared with the smooth approaches, the rate of increase of the computational complexity and the computation time for each of the smooth and nonsmooth optimization approaches w.r.t. the size of the traffic network (or the other controlled systems that are considered in the further case studies).

- **Further investigation of the added value of endpoint penalties in finite-horizon MPC**
  In Chapter 8 of this thesis, we have proposed some general formulations for approximating the endpoint penalties for urban traffic networks, where the cost function of the MPC controller includes a weighted summation of the total time spent and the total emissions of the vehicles. The added value of endpoint penalties has not become clear yet. Hence, further research in this field is of interest in order to compare the truncation error of a finite-horizon MPC controller w.r.t. an infinite MPC controller and the error of approximating the finite-horizon MPC controller with an infinite-horizon MPC controller considering endpoint penalties in formulating the cost function of the MPC optimization problem.

Additional research beyond the topics of the thesis

Going further than the topics within the field of this thesis, we also propose the following additional topics for future research.

- **Fast MPC methods for urban traffic control**
  One important aspect of many MPC controllers that are used to control a dynamical system is that they should be capable of operating in real time. Hence, taking further steps with the aim of providing an MPC controller that performs fast enough for real-time control purposes is of a high importance. Some techniques that we propose to consider for this aim include parameterized MPC approaches [22, 162], developing methods for providing good starting points for the optimization solver of the MPC controller, and fast MPC approaches (see, e.g., [4, 101]).

- **Stability analysis of the proposed smooth MPC controller**
  Existence of conditions that can guarantee the stability of the smooth MPC controller developed in Chapter 7 should be investigated. Hence, as a topic for future research we propose the stability analysis of the smooth MPC controller.

- **Robust control of time-delayed LPV systems**
  Time delays exist in both modeling (see Section 6.2.2 of the thesis) and control of traffic networks (see, e.g., [45]). Moreover, recently more researches consider LPV models for traffic networks. For instance, Luspay et al. [102] use an LPV formalism to model the traffic on freeway networks, and [103], which develops a control system for such a freeway network that is modeled by an LPV system. Hence, theoretical research in the field of LPV systems with time-delayed state and/or control variables aimed at developing robust MPC controllers is suggested.

- **Embedding several smooth MPC controllers within a distributed control architecture for large-scale traffic networks**
  In Part II of the thesis, we have proposed efficient methods for developing a smooth MPC controller for green urban mobility. For a large-scale urban traffic network, which includes a relatively large number of state and control variables, however, using a centralized smooth MPC controller might not be computationally efficient. Moreover, any failure of the centralized MPC controller will disturb the entire traffic network. Hence, for future work we propose to consider a distributed control architecture, where multiple smooth MPC controllers can exchange information and can coordinate to achieve the overall objective of the urban traffic network.

- **Constructing a multi-level control architecture with multiple control levels**
  Traffic networks are, in general, large-scale systems. Hence, it would be of benefit, both computationally and from a performance point-of-view, if such systems are controlled by a multi-level control system, where the levels will operate at different spatial and temporal scales. Hence, we propose to design such a multi-level control system that can use available macroscopic control approaches (e.g., control methods based on networked fundamental diagrams (see [50] for more details)) in the higher levels that include larger traffic subnetworks (e.g., an entire city). For the middle levels, which correspond to the average-sized traffic subnetworks (e.g., a relatively big collection of several intersections and main roads of a city), MPC controllers that are based on the approaches proposed in Part II of the thesis can be used. For lower control levels that will include small urban traffic subnetworks (e.g., a collection of a few intersections and urban links), the coordinative and predictive fuzzy controller proposed in Chapter 10 can be a suitable choice. The overall objective of the control system (i.e., finding a balanced trade-off between reduction of the total time spent and reduction of the total emissions of the vehicles) will then be distributed among these different control levels. One challenge for this topic that should be tackled will be to provide coordination across and within these control levels.

- **Efficient smooth MPC for green mobility in integrated urban-freeway traffic networks**
  The control approaches proposed in this thesis have been applied to urban traffic networks. However, in reality, urban traffic networks might be connected to some freeways. This results in a mutual effect of the traffic situations of the two traffic networks on each other. If this effect is ignored in designing the control system for either of these traffic networks, it may results in failures and/or significant performance reduction of either of these control systems in practice. The efficient MPC controller introduced in Chapters 6 and 7 together with the endpoint penalties developed and formulated in Chapter 8 should further be extended to a freeway traffic network. Moreover, one aspect that should in particular be considered for designing the integrated MPC controller is efficient formulation of the shared states and shared control variables in between the two traffic networks (e.g., these mutual effects might be formulated as constraints for the optimization problems of the MPC controllers). If we consider a centralized MPC controller, then we need an integrated flow and emission model that covers the integrated urban-freeway traffic network. However, we can simplify the structure of the model and the controller by considering a distributed control architecture, where a few of these MPC controllers take care of the connecting boundaries of the two traffic networks and hence can use a simpler integrated model.

- **Applying predictive model-based fuzzy controller for multi-agent control systems in unknown environments**
  The proposed control architecture of Chapter 10 considers a type-2 fuzzy model of the controlled system that is developed based on the historical data of each agent and that is modified adaptively based on the information collected by each agent during the online control procedure. Hence, a promising application of the proposed control architecture is for agents that should operated in an environment about which the agent has only limited or no information at all. Examples of such situations include search-and-rescue missions in buildings after an incident or in cities struck by earthquakes, deep space exploration, underwater operations, and autonomous aircraft.

Bibliography

[1] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos. Store-and-forward based methods for the signal control problem in large-scale congested urban road networks. *Transportation Research Part C*, 17(2):163–174, 2009.

[2] K. Aboudolas, M. Papageorgiou, A. Kouvelas, and E. Kosmatopoulos. A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks. *Transportation Research Part C*, 18(5):680–694, 2010.

[3] K. Ahn, A. Trani, H. Rakha, and M. van Aerde. Microscopic fuel consumption and emission models. In *Proceedings of the* 78th *Annual Meeting of the Transportation Research Board*, USA, June–July 1999.

[4] M. Alamir. Fast NMPC: A reality-steered paradigm: Key properties of fast NMPC algorithms. In *Proceedings of the 2014 European Control Conference (ECC)*, pages 2472–2477, France, June 2014.

[5] W. P. Anderson, S. K. Pavlos, E. J. Miller, and R. Buliung. Simulating automobile emissions in an integrated urban model. *Transportation Research Record*, 1520:71–80, 1996.

[6] R. Babuška. *Fuzzy Modeling for Control*. Kluwer Academic Publishers, USA, 1998.

[7] T. M. Bailey. Convergence of Rprop and variants. *Neurocomputing*, 159:90–95, 2015.

[8] A. Barisone, D. Giglio, R. Minciardi, and R. Poggi. A macroscopic traffic model for real-time optimization of signalized urban areas. In *Proceedings of the* 41st *IEEE Conference on Decision and Control*, pages 900–903, USA, 2002.

[9] M. Barth and K. Boriboonsomsin. Real-world carbon dioxide impacts of traffic congestion. *Transportation Research Record*, 2058:163–171, 2008.

[10] M. Barth, F. An, J. Norbeck, and M. Ross. Modal emissions modeling: A physical approach. *Transportation Research Record*, 1520:81–88, 1996.

[11] 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, 2006.

[12] A. Bemporad, M. Heemels, and M. Johansson. *Networked Control Systems*. Springer-Verlag, Germany, 2010.

[13] P. J. Bickel, C. Chen, J. Kwon, E. van Zwet, and P. Varaiya. Measuring traffic. *Statistical Science*, 22(4):581–597, 2007.

[14] H. H. J. Bloemen, T. J. J. van den Boom, and H. B. Verbruggen. Optimizing the end-point state-weighting matrix in model-based predictive control. *Automatica*, 38(6): 1061–1068, 2002.

[15] A. H. Bond and L. Gasser. *Readings in Distributed Artificial Intelligence*. Morgan Kaufmann, USA, 1988.

[16] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. Elsevier Science Publishing Co., USA, 1976.

[17] S. Boverie, B. Demaya, J. M. Lequellec, and A. Titli. Fuzzy logic control for high order systems. In *Proceedings of the* 2$^{\text{nd}}$ *IEEE International Conference on Fuzzy Systems*, pages 117–121, USA, March–April 1993.

[18] D. Boyce and C. Xiong. Forecasting travel for very large cities: Challenges and opportunities for china. *Transportmetrica*, 3(1):1–19, 2007.

[19] M. Brackstone, B. Waterson, and M. McDonald. Determinants of following headway in congested traffic. *Transportation Research Part F: Traffic Psychology and Behaviour*, 12 (2):131–142, 2009.

[20] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems Magazine*, 22(1):44–52, 2002.

[21] S. G. Cao, N. W. Rees, and G. Feng. Analysis and design for a class of complex control systems part I: Fuzzy modelling and identification. *Automatica*, 33(6):1017–1028, 1997.

[22] A. Casavola, D. Famularo, and G. Franze. A predictive control strategy for norm-bounded LPV discrete-time systems with bounded rates of parameter change. *International Journal of Robust and Nonlinear Control*, 18(7):714–740, 2007.

[23] C. Castelfranchi. Modelling social action for AI agents. *Artificial Intelligence*, 103(1–2): 157–182, 1998.

[24] Center for Economics and Business Research. The future economic and environmental costs of gridlock in 2030. Technical report, INRIX, USA, July 2014.

[25] C. Chen and O. L. Mangasarian. Smoothing methods for convex inequalities and linear complementarity problems. *Mathematical Programming*, 71(1):51–69, 1995.

[26] G. Chen and T. T. Pham. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*. CRC Press, USA, 2000.

[27] K. Chen and L. Yu. Microscopic traffic-emission simulation and case study for evaluation of traffic control strategies. *Journal of Transportation Systems Engineering and Information Technology*, 7(1):93–99, 2007.

[28] D. W. Clarke and R. Scattolini. Constrained receding horizon predictive control. *IEE Proceedings – Control Theory and Applications*, 138(4):347–354, 1991.

[29] F. H. Clarke. *Optimization and Nonsmooth Analysis.* John Wiley & Sons, USA, 1983.

[30] F. H. Clarke. *Optimization and Nonsmooth Analysis.* Classics in Applied Mathematics. Society for Industrial and Applied Mathematics (SIAM), USA, 1990.

[31] B. Coifman. Estimating travel times and vehicle trajectories on freeways using dual loop detectors. *Transportation Research Part A*, 36(4):351–364, 2002.

[32] A. Csikós, T. Tettamanti, and I. Varga. Macroscopic modeling and control of emission in urban road traffic networks. *Transport*, 30(2):152–161, 2015.

[33] C. F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994.

[34] C. F. Daganzo. The cell transmission model, part II: Network traffic. *Transportation Research Part B*, 29B(2):79–93, 1995.

[35] C. F. Daganzo. *Fundamentals of Transportation and Traffic Operations.* Pergamon-Elsevier, UK, 1997.

[36] D. J. Dailey. A statistical algorithm for estimating speed from single loop volume and occupancy measurements. *Transportation Research Part B: Methodological*, 33(5):313–322, 1999.

[37] B. De Schutter. Model predictive traffic control for green mobility. In *Proceedings of the 2014 European Control Conference*, pages 2260–2263, France, June 2014.

[38] D. Debeljković. *Time-Delay Systems.* InTech, Croatia, 2011.

[39] C. Diakaki, M. Papageorgiou, and K. Aboudolas. A multi-variable regulator approach to traffic-responsive network-wide signal control. *Control Engineering Practice*, 10(2):183–195, 2002.

[40] C. Diakaki, V. Dinopoulou, K. Aboudolas, M. Papageorgiou, E. Ben-Shabat, E. Seider, and A. Leibov. Extensions and new applications of the traffic-responsive urban control strategy: Coordinated signal control for urban networks. *Transportation Research Record*, 1856:202–211, 2003.

[41] F. Doctor, H. Hagras, and V. Callaghan. A fuzzy embedded agent-based approach for realizing ambient intelligence in intelligent inhabited environments. *IEEE Transactions on Systems, Man, and Cybernetics*, 35(1):55–65, 2005.

[42] G. Droge and M. Egerstedt. Distributed parameterized model predictive control of networked multi-agent systems. In *Proceedings of the 2013 American Control Conference (ACC)*, pages 1332–1337, USA, 2013.

[43] L. C. Edie. Discussion of traffic stream measurements and definitions. In *Proceedings of the* 2nd *International Symposium on the Theory of Traffic Flow*, pages 139–154, France, 1963.

[44] M. Eisen. *Introduction to Mathematical Probability Theory*. Prentice Hall, USA, 1969.

[45] M. Keyvan Ekbatani, M. Papageorgiou, and V. L. Knoop. Comparison of online time-delayed and non-time-delayed urban traffic control via remote gating. In *Proceedings of the* 94th *Annual Meeting of the Transportation Research Board*, USA, January 2015.

[46] C. Fantuzzi and R. Rovatti. On the approximation capabilities of the homogeneous Takagi-Sugeno model. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, pages 1067–1072, September 1996.

[47] G. Feng. A survey on analysis and design of model-based fuzzy control systems. *IEEE Transactions on Fuzzy Systems*, 14(5):676–697, 2006.

[48] R. J. Gallimore, N. R. Jennings, H. S. Lamba, C. L. Mason, and B. J. Orenstein. Cooperating agents for 3-D scientific data interpretation. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, 29(1):110–126, 1999.

[49] D. Gazis, R. Herman, and R. Rothery. Nonlinear follow the leader models of traffic flow. *Operations Research*, 9(4):545–567, 1961.

[50] N. Geroliminis and C. F. Daganzo. Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. *Transportation Research Part B: Methodological*, 42(9):759–770, 2008.

[51] S. Gori, S. La Spada, L. Mannini, and M. Nigro. A dynamic mesoscopic emission model for signalized intersections. In *Proceedings of the* 16th *International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 2212–2217, The Netherlands, October 2013.

[52] R. M. P. Goverde. A delay propagation algorithm for large-scale railway traffic networks. *Transportation Research Part C: Emerging Technologies*, 18(3):269–287, 2010.

[53] H. Hagras. A hierarchical type-2 fuzzy logic control architecture for autonomous mobile robots. *IEEE Transactions on Fuzzy Systems*, 12(4):524–539, 2004.

[54] H. Hagras. Type-2 FLCs: A new generation of fuzzy controllers. *IEEE Computational Intelligence Magazine*, 2(1):30–43, 2007.

[55] H. Hagras, F. Doctor, and A. Lopez. An incremental adaptive life long learning approach for type-2 fuzzy embedded agents in ambient intelligent environments. *IEEE Transactions on Fuzzy Systems*, 15(1):41–55, 2007.

[56] J. Han, J. W. Polak, J. Barria, and R. Krishnan. On the estimation of space-mean speed from inductive loop detector data. *Transportation Planning and Technology*, 33(1):91–104, 2010.

[57] HCM. *Highway Capacity Manual.* TRB, Pub No: LP-608, USA, 2000.

[58] A. Hegyi, B. De Schutter, and H. Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C*, 13(3):185–209, June 2005.

[59] S. P. Hoogendoorn and P. H. L. Bovy. State-of-the-art of vehicular traffic flow modelling. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 215(4):283–303, 2001.

[60] J. Huang, N. R. Jennings, and J. Fox. Agent-based approach to health care management. *Applied Artificial Intelligence*, 9(4):401–420, 1995.

[61] R. Huey, D. De Leonardis, and M. Freedman. *National Traffic Speeds Survey II: 2009.* National Technical Information Service, USA, 2012.

[62] K. Ito and K. Kunisch. Karush-Kuhn-Tucker conditions for nonsmooth mathematical programming problems in function spaces. *SIAM Journal on Control and Optimization*, 49(5):2133–2154, 2011.

[63] A. Jamshidnejad and B. De Schutter. Estimation of the generalized average traffic speed based on microscopic measurements. *Transportmetrica A: Transport Science*, 11(6): 525–546, 2015.

[64] A. Jamshidnejad and B. De Schutter. An iterative procedure for estimating the generalized average speed using microscopic point measurements. In *Proceedings of the 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, pages 38–44, Hungary, June 2015.

[65] A. Jamshidnejad and B. De Schutter. An algorithm for estimating the generalized fundamental traffic variables from point measurements using initial conditions. *Transportmetrica B: Transport Dynamics*, 2017. doi: 10.1080/21680566.2017.1279991. To appear.

[66] A. Jamshidnejad, B. De Schutter, and M.J. Mahjoob. Urban traffic control using a fuzzy multi-agent system. In *Proceedings of the 2015 European Control Conference*, pages 3046–3051, Austria, July 2015.

[67] A. Jamshidnejad, H. Hellendoorn, S. Lin, and B. De Schutter. Smoothening for efficient solution of model predictive control for urban traffic networks considering endpoint penalties. In *Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems*, pages 2837–2842, Spain, September 2015.

[68] A. Jamshidnejad, M. Papageorgiou, I. Papamichail, and B. De Schutter. A multi-objective model-predictive control approach dealing with congestion and emissions in urban traffic networks. In *Proceedings of the* 1st *ARTS ECR Conference*, Malta, May 2015.

[69] A. Jamshidnejad, I. Papamichail, H. Hellendoorn, M. Papageorgiou, and B. De Schutter. Gradient-based model-predictive control for green urban mobility in traffic networks. In *Proceedings of the* 19th *IEEE Intelligent Transportation Systems Conference (ITSC 2016)*, pages 1077–1082, Brazil, November 2016.

[70] A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter. A model-predictive urban traffic control approach with a modified flow model and endpoint penalties. In *Proceedings of the* 14th *IFAC Symposium on Control in Transportation Systems (CTS 2016)*, pages 147–152, Turkey, May 2016.

[71] A. Jamshidnejad, H. Hellendoorn, M. J. Mahjoob, and B. De Schutter. Multi-agent predictive and adaptive model-based type-2 fuzzy control for state-delayed systems – addendum. Technical Report 15-033a, Delft Center for Systems and Control, Delft University of Technology, The Netherlands, February 2017.

[72] A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter. Sustainable model-predictive control in urban traffic networks: Efficient solution based on general smoothening methods. *IEEE Transactions on Control Systems Technology*, 2017. doi: 10.1109/TCST.2017.2699160. To appear.

[73] A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter. A mesoscopic integrated urban flow-emission model. *Transportation Research Part C: Emerging Technologies*, 75:45–83, 2017.

[74] N. R. Jennings. Commitments and conventions: The foundation of coordination in multi-agent systems. *The Knowledge Engineering Review*, 8(3):223–250, 1993.

[75] N. R. Jennings. Controlling cooperative problem solving in industrial multi-agent systems using joint intentions. *Artificial Intelligence*, 75(2):195–240, 1995.

[76] N. R. Jennings. On agent-based software engineering. *Artificial Intelligence*, 117(2): 277–296, 2000.

[77] N. R. Jennings, P. Faratini, M. J. Johnson, T. J. Norman, P. O'Brien, and M. E. Wiegand. Agent-based business process management. *International Journal of Cooperative Information Systems*, 5(2–3):105–130, 1996.

[78] T. A. Johansen, R. Shorten, and R. Murray-Smith. On the interpretation and identification of dynamic Takagi-Sugeno fuzzy models. *IEEE Transactions on Fuzzy Systems*, 8(3): 297–313, 2000.

[79] H. Kagiwada, R. Kalaba, and Y. Thomas. Exact solution of Pontryagin's equations of optimal control - part 1. *Journal of Optimization Theory and Applications*, 5(1):12–22, 1970.

[80] H. Kashani and G. Saridis. Intelligent control for urban traffic systems. *Automatica*, 2 (19):191–197, 1983.

[81] S. S. Keerthi and E. G. Gilbert. Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations. *Journal of Optimization Theory and Applications*, 57(2):265–293, 1988.

[82] C. J. Khisty and B. K. Lall. *Transportation Engineering: An Introduction*. Prentice Hall, USA, 2003.

[83] L. A. Klein, M. K. Mills, and D. R. Gibson. *Traffic Detector Handbook*. U.S. Department of Transportation, Federal Highway Administration, third edition, 2006.

[84] R. W. Koepcke. On the control of linear systems with pure time delay. *Journal of Basic Engineering*, 87(1):74–80, 1965.

[85] D. Krajzewicz, G. Hertkorn, C. Rösel, and P. Wagner. SUMO (Simulation of Urban MObility); An open-source traffic simulation. In *Proceedings of the* 4th *Middle East Symposium on Simulation and Modelling (MESM'02)*, pages 183–187, United Arab Emirates, September 2002.

[86] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker. Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal on Advances in Systems and Measurements*, 5(3–4):128–138, 2012.

[87] W. H. Kwon, A. M. Bruckstein, and T. Kailath. Stabilizing state-feedback design via the moving horizon method. *International Journal of Control*, 37(3):631–643, 1983.

[88] C. Lee, B. Hellinga, and F. Saccomanno. Evaluation of variable speed limits to improve traffic safety. *Transportation Research Part C*, 14(3):213–228, 2006.

[89] C. C. Lee. Fuzzy logic in control systems: Fuzzy logic controller - Part I. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2):404–418, 1990.

[90] C. C. Lee. Fuzzy logic in control systems: Fuzzy logic controller - Part II. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2):419–435, 1990.

[91] I. J. Lee and A. S. Oh. An accident detection system on highway through CCTV with Calogero-Moser system. *Information Engineering*, 2(1):1–5, 2013.

[92] F. L. J. Lewis. *Optimal Control.* John Wiley & Sons, USA, 1986.

[93] L. Li, R.R. Negenborn, and B. De Schutter. A sequential linear programming approach for flow assignment in intermodal freight transport. In *Proceedings of the* 16th *International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 1224–1230, The Netherlands, October 2013.

[94] M. J. Lighthill and G. B. Whitham. On kinematic waves. II. a theory of traffic flow on long crowded roads. *Proceedings of the Royal Society of London*, 229(1178):317–345, 1955.

[95] N. E. Ligterink, R. De Lange, and E. Schoen. Refined vehicle and driving-behaviour dependencies in the VERSIT+ emission model. In *Proceedings of the ETAPP Symposium*, pages 177–186, France, June 2009.

[96] S. Lin. *Efficient Model Predictive Control for Large-Scale Urban Traffic Networks.* PhD thesis, Delft University of Technology, The Netherlands, April 2011.

[97] S. Lin and Y. Xi. An efficient model for urban traffic network control. In *Proceedings of the* 17th *World Congress the International Federation of Automatic Control*, pages 14066–14071, Korea, 2008.

[98] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn. Efficient network-wide model-based predictive control for urban traffic networks. *Transportation Research Part C*, 24:122–140, 2012.

[99] S. Lin, B. De Schutter, Y. Xi, and H. Hellendoorn. Integrated urban traffic control for the reduction of travel delays and emissions. *IEEE Transactions on Intelligent Transportation Systems*, 1(4):1609–1619, 2013.

[100] S. Liu, H. Hellendoorn, and B. De Schutter. Model predictive control for freeway networks based on multi-class traffic flow and emission models. *IEEE Transactions on Intelligent Transportation Systems*, 18(2):306–320, 2016.

[101] R. Lopez-Negrete, F. J. D'Amato, L. T. Biegler, and A. Kumar. Fast nonlinear model predictive control: Formulation and industrial process applications. *Computers and Chemical Engineering*, 51:55–64, 2013.

[102] T. Luspay, B. Kulcsár, J. W. van Wingerden, M. Verhaegen, and J. Bokor. Linear parameter varying identification of freeway traffic models. *IEEE Transactions on Control Systems Technology*, 19(1):31–45, 2011.

[103] T. Luspay, T. Péni, and B. Kulcsár. Constrained freeway traffic control via linear parameter varying paradigm. In J. Mohammadpour and C. W. Scherer, editors, *Control of Linear Parameter Varying Systems with Applications*, chapter 18, pages 461–482. Springer, Germany, 2012.

[104] Y. Ma, J. van Dalen, C. de Blois, and J. van Nuen. Estimating dynamic transport population for official statistics based on GPS/GSM. In *Proceedings of the* 7th *International Conference on Traffic and Transportation Studies (ICTTS)*, pages 1076–1088, China, August 2010.

[105] J. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, UK, 2002.

[106] P. Maes. Modeling adaptive autonomous agents. *Artificial Life*, 1(1–2):135–162, 1993.

[107] J. M. Maestre and R. R. Negenborn. *Distributed Model-Predictive Control Made Easy*. Springer-Verlag, Germany, 2014.

[108] E. H. Mamdani. Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institution of Electrical Engineers*, 121(12):1585–1588, 1974.

[109] E. H. Mamdani and S. Assilian. An experiment in linguistic synthesis with a fuzzy logic controller. *International Journal of Man-Machine Studies*, 7(1):1–13, 1975.

[110] F. P. Maturana, R. J. Staron, and K. H. Hall. Methodologies and tools for intelligent agents in distributed control. *IEEE Intelligent Systems*, 20(1):42–49, 2005.

[111] A. D. May. *Traffic Flow Fundamentals*. Prentice Hall, USA, 1990.

[112] J. M. Mendel. *Uncertain Rule-Based Fuzzy Logic Systems*. Prentice Hall, USA, 2000.

[113] J. M. Mendel. Type-2 fuzzy sets and systems: An overview. *IEEE Computational Intelligence Magazine*, 2(2):20–29, 2007.

[114] J. M. Mendel and R. I. B. John. Type-2 fuzzy sets made simple. *IEEE Transactions on Fuzzy Systems*, 10(2):117–127, 2002.

[115] A. Messmer and M. Papageorgiou. METANET: A macroscopic simulation program for motorway networks. *Traffic Engineering and Control*, 31(9):466–470, 1990.

[116] M. Morari and J. H. Lee. Model predictive control: Past, present, and future. *Computers and Chemical Engineering*, 23(4–5):667–682, 1999.

[117] U. Mori, A. Mendiburu, M. Álvarez, and J. A. Lozano. A review of travel time estimation and forecasting for advanced traveller information systems. *Transportmetrica A: Transport Science*, 11(2):119–157, 2015.

[118] E. Mosca and J. Zhang. Stable redesign of predictive control. *Automatica*, 28(6):1229–1233, 1992.

[119] M. A. Müller, M. Reble, and F. Allgöwer. Cooperative control of dynamically decoupled systems via distributed model predictive control. *International Journal of Robust and Nonlinear Control*, 22(12):1376–1397, 2012.

[120] R. R. Negenborn, B. De Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence*, 21(3):353–366, 2008.

[121] R. R. Negenborn, G. Hug-Glanzmann, B. De Schutter, and G. Andersson. A novel coordination strategy for multi-agent control using overlapping subnetworks with application to power systems. In J. Mohammadpour and K.M. Grigoriadis, editors, *Efficient Modeling and Control of Large-Scale Systems*, pages 251–278. Springer, USA, 2010.

[122] G. F. Newell. *Theory of Highway Traffic Flow 1945-1965*. Transportation Research Board, University of California at Berkeley, 1995.

[123] G. F. Newell. A simplified car-following theory: A lower order model. *Transportation Research Part B*, 36(3):195–205, 2002.

[124] L. Ntziachristos, D. Gkatzoflias, K. Chariton, and Z. Samaras. *COPERT: A European Road Transport Emission Inventory Model Information Technologies in Environmental Engineering*. Springer, Germany, 2009.

[125] E. Oliviera, K. Fischer, and O. Stepankova. Multi-agent systems: Which research for which applications. *Robotics and Autonomous Systems*, 27(1–2):91–106, 1999.

[126] R. Palm, H. Hellendoorn, and D. Driankov. *Model-Based Fuzzy Control*. Springer, Germany, 1997.

[127] M. Papageorgiou and A. Kotsialos. Nonlinear optimal control applied to coordinated ramp metering. *IEEE Transactions on Intelligent Transportation Systems*, 12:920–933, 2004.

[128] M. Papageorgiou and M. Marinaki. A feasible direction algorithm for the numerical solution of optimal control problems. Technical report, Dynamic Systems & Simulation Laboratory, Technical University of Crete, Greece, 1995.

[129] P. M. Pardalos and J. B. Rosen. *Constrained Global Optimization: Algorithms and Applications*. Lecture Notes in Computer Science. Springer-Verlag, Germany, 1987.

[130] C. Pasquale, I. Papamichail, C. Roncoli, S. Sacone, S. Siri, and M. Papageorgiou. Two-class freeway traffic regulation to reduce congestion and emissions via nonlinear optimal control. *Transportation Research Part C: Emerging Technologies*, 55:85–99, 2015.

[131] A. Paz, V. Molano, E. Martinez, C. Gaviria, and C. Arteaga. Calibration of traffic flow models using a memetic algorithm. *Transportation Research Part C*, 55:432–443, 2015.

[132] L. A. Pipes. An operational analysis of traffic dynamics. *Journal of Applied Physics*, 24 (3):274–281, 1953.

[133] Y. G. Qi, H. H. Teng, and L. Yu. Microscale emission models incorporating acceleration and deceleration. *Journal of Transport Engineering*, 130(3):348–359, 2004.

[134] H. Rakha and W. Zhang. Estimating traffic stream space-mean speed and reliability from dual and single loop detectors. *Transportation Research Record*, (1925):38–47, 2005.

[135] H. Rakha, H. Yue, and F. Dion. VT-meso model framework for estimating hot-stabilized light-duty vehicle fuel consumption and emission rates. *Canadian Journal of Civil Engineering*, 38(11):1274–1286, 2011.

[136] J. B. Rawlings and K. R. Muske. The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, 38(10):1512–1516, 1993.

[137] M. Riedmiller and H. Braun. A directive adaptive method for faster backpropagation learning: the RPROP algorithm. In *Proceedings of the IEEE International Conference on Neural Networks*, pages 586–591, USA, 1993.

[138] N. R. Sandell, P. Varaiya, M. Athans, and M. G. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Transactions on Automatic Control*, 23(2):108–128, 1978.

[139] J. B. Sheu. A stochastic optimal control approach to real-time incident-responsive traffic signal control at isolated intersections. *Transportation Science*, 36(4):418–434, 2002.

[140] A. Sjodin, K. Persson, K. Andreasson, B. Arlander, and B. Galle. On-road emission factors derived from measurements in a traffic tunnel. *International Journal of Vehicle Design*, 20(1–4):147–158, 1998.

[141] F. Soriguera and F. Robusté. Estimation of traffic stream space mean speed from time aggregations of double loop detector data. *Transportation Research Part C*, 19(1):115–129, 2011.

[142] F. Soriguera and F. Robusté. Highway travel time accurate measurement and short-term prediction using multiple data sources. *Transportmetrica*, 7(1):85–109, January 2011.

[143] M. Sugeno and G. T. Kang. Structure identification of fuzzy model. *Fuzzy Sets and Systems*, 28(1):15–33, 1988.

[144] K. P. Sycara. Multi-agent systems. *AI Magazine*, 19(2):79–92, 1998.

[145] T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man, and Cybernetics*, 15(1):116–132, 1985.

[146] M. Tambe. Towards flexible teamwork. *Journal of Artificial Intelligence Research*, 7(1): 83–124, 1997.

[147] M. Tambe. Implementing agent teams in dynamic multi-agent environments. *Applied Artificial Intelligence*, 12(2–3):189–210, 1998.

[148] J. M. Thomson. Reflections on the economics of traffic congestion. *Journal of Transport Economics and Policy*, 32(1):93–112, 1997.

[149] M. Treiber and A. Kesting. *Traffic Flow Dynamics*. Springer, Germany, 2013.

[150] M. van den Berg, A. Hegyi, B. De Schutter, and J. Hellendoorn. Integrated traffic control for mixed urban and freeway networks: A model predictive control approach. *European Journal of Transport and Infrastructure Research*, 7(3):223–250, 2007.

[151] Y. Wang and N. L. Nihan. Can single-loop detectors do the work of double-loop detectors. *Journal of Transportation Engineering*, 129(2):169–176, 2003.

[152] Y. Wang, M. Papageorgiou, and A. Messmer. RENAISSANCE – A unified macroscopic model-based approach to real-time freeway network traffic surveillance. *Transportation Research Part C*, 14(3):190–212, 2006.

[153] J. G. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers*, 1:325–378, 1952.

[154] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J. P. Hubaux. TraCI: An interface for coupling road traffic and network simulators. In *Proceedings of the* 11th *Communications and Networking Simulation Symposium (CNS'08)*, pages 155–163, USA, 2008.

[155] D. White and D. A. Sofge. *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*. Van Nostrand Reinhold, USA, 1992.

[156] D. Wu and W. Tan. A type-2 fuzzy logic controller for the liquid-level process. In *Proceedings of the IEEE International Conference on Fuzzy Systems*, volume 2, pages 953–958, Hungary, 2004.

[157] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[158] L. A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning. *Information Sciences*, 8(3):199–249, 1975.

[159] L. A. Zadeh. From computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions. *IEEE Transactions on Circuits and Systems*, 45(1):105–119, 1999.

[160] S. K. Zegeye. *Model-Based Traffic Control for Sustainable Mobility*. PhD thesis, Delft University of Technology, The Netherlands, Oct. 2011.

[161] S. K. Zegeye, B. De Schutter, J. Hellendoorn, and E. A. Breunesse. Reduction of area-wide emissions using an efficient model-based traffic control strategy. In *Proceedings of the 2011 IEEE Forum on Integrated and Sustainable Transportation Systems (FISTS 2011)*, pages 307–312, Austria, June–July 2011.

[162] S. K. Zegeye, B. De Schutter, J. Hellendoorn, E. A. Breunesse, and A. Hegyi. A predictive traffic controller for sustainable mobility using parameterized control policies. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1420–1429, 2012.

[163] Y. Zou, Y. Zhang, and X. Zhu. Constructing a bivariate distribution for freeway speed and headway data. *Transportmetrica A: Transport Science*, 10(3):255–272, 2014.

TRAIL Thesis Series

The following list contains the most recent dissertations in the TRAIL Thesis Series. For a complete overview of more than 150 titles see the TRAIL website: www.rsTRAIL.nl.

The TRAIL Thesis Series is a series of the Netherlands TRAIL Research School on transport, infrastructure and logistics.

Jamshidnejad, A., *Efficient Predictive Model-Based and Fuzzy Control for Green Urban Mobility*, T2017/6, June 2017, TRAIL Thesis Series, the Netherlands

Araghi, Y., *Consumer Heterogeneity, Transport and the Environment*, T2017/5, May 2017, TRAIL Thesis Series, the Netherlands

Kasraian Moghaddam, D., *Transport Networks, Land Use and Travel Behaviour: A long term investigation*, T2017/4, May 2017, TRAIL Thesis Series, the Netherlands

Smits, E.-S., *Strategic Network Modelling for Passenger Transport Pricing*, T2017/3, May 2017, TRAIL Thesis Series, the Netherlands

Tasseron, G., *Bottom-Up Information Provision in Urban Parking: An in-depth analysis of impacts on parking dynamics*, T2017/2, March 2017, TRAIL Thesis Series, the Netherlands

Halim, R.A., *Strategic Modeling of Global Container Transport Networks: Exploring the future of port-hinterland and maritime container transport networks*, T2017/1, March 2017, TRAIL Thesis Series, the Netherlands

Olde Keizer, M.C.A., *Condition-Based Maintenance for Complex Systems: Coordinating maintenance and logistics planning for the process industries*, T2016/26, December 2016, TRAIL Thesis Series, the Netherlands

Zheng, H., *Coordination of Waterborn AGVs*, T2016/25, December 2016, TRAIL Thesis Series, the Netherlands

Yuan, K., *Capacity Drop on Freeways: Traffic dynamics, theory and Modeling*, T2016/24, December 2016, TRAIL Thesis Series, the Netherlands

Li, S., *Coordinated Planning of Inland Vessels for Large Seaports*, T2016/23, December 2016, TRAIL Thesis Series, the Netherlands

Berg, M. van den, *The Influence of Herding on Departure Choice in Case of Evacuation: Design and analysis of a serious gaming experimental set-up*, T2016/22, December 2016, TRAIL Thesis

Series, the Netherlands

Luo, R., *Multi-Agent Control of urban Transportation Networks and of Hybrid Systems with Limited Information Sharing*, T2016/21, November 2016, TRAIL Thesis Series, the Netherlands

Campanella, M., *Microscopic Modelling of Walking Behavior*, T2016/20, November 2016, TRAIL Thesis Series, the Netherlands

Horst, M. van der, *Coordination in Hinterland Chains: An institutional analysis of port-related transport*, T2016/19, November 2016, TRAIL Thesis Series, the Netherlands

Beukenkamp, W., *Securing Safety: Resilience time as a hidden critical factor*, T2016/18, October 2016, TRAIL Thesis Series, the Netherlands

Mingardo, G., *Articles on Parking Policy*, T2016/17, October 2016, TRAIL Thesis Series, the Netherlands

Duives, D.C., *Analysis and Modelling of Pedestrian Movement Dynamics at Large-scale Events*, T2016/16, October 2016, TRAIL Thesis Series, the Netherlands

Wan Ahmad, W.N.K., *Contextual Factors of Sustainable Supply Chain Management Practices in the Oil and Gas Industry*, T2016/15, September 2016, TRAIL Thesis Series, the Netherlands

Liu, X., *Prediction of Belt Conveyor Idler Performance*, T2016/14, September 2016, TRAIL Thesis Series, the Netherlands

Gaast, J.P. van der, *Stochastic Models for Order Picking Systems*, T2016/13, September 2016, TRAIL Thesis Series, the Netherlands

Wagenaar, J.C., *Practice Oriented Algorithmic Disruption Management in Passenger Railways*, T2016/12, September 2016, TRAIL Thesis Series, the Netherlands

Psarra, I., *A Bounded Rationality Model of Short and Long-Term Dynamics of Activity-Travel Behavior*, T2016/11, June 2016, TRAIL Thesis Series, the Netherlands

Ma, Y., *The Use of Advanced Transportation Monitoring Data for Official Statistics*, T2016/10, June 2016, TRAIL Thesis Series, the Netherlands

Li, L., *Coordinated Model Predictive Control of Synchromodal Freight Transport Systems*, T2016/9, June 2016, TRAIL Thesis Series, the Netherlands

Vonk Noordegraaf, D.M., *Road Pricing Policy Implementation*, T2016/8, June 2016, TRAIL Thesis Series, the Netherlands

Liu, S., Modeling, R*obust and Distributed Model Predictive Control for Freeway Networks*, T2016/7, May 2016, TRAIL Thesis Series, the Netherlands

Calvert, S.C., *Stochastic Macroscopic Analysis and Modelling for Traffic Management*, T2016/6, May 2016, TRAIL Thesis Series, the Netherlands

Sparing, D., *Reliable Timetable Design for Railways and Connecting Public Transport Services*, T2016/5, May 2016, TRAIL Thesis Series, the Netherlands

Summary

Efficient Predictive Model-Based and Fuzzy Control for Green Urban Mobility

In this thesis, we develop efficient predictive model-based control approaches, including model-predictive control (MPC) and model-based fuzzy control, for application in urban traffic networks with the aim of reducing a combination of the total time spent by the vehicles within the network and the total emissions. The thesis includes three main parts, where in the first part the main focus is on accurate approaches for estimating the macroscopic traffic variables, such as the temporal-spatial averages, from a microscopic point-of-view. The second part includes efficient approaches for solving the optimization problem of the nonlinear MPC controller. The third and last part of the thesis proposes an adaptive and predictive model-based type-2 fuzzy control scheme that can be implemented within a multi-agent control architecture.

Flow and density are mainly used to characterize partly the state of physical systems with moving particles. Flow and density are macroscopic concepts, i.e., they involve average values. For dynamic systems that include moving particles, such as for traffic networks, these averages may be defined in three different ways: temporal, spatial, and temporal-spatial. Computation of the first two averages is straightforward, but for the third average only a general formulation is suggested by Edie (1963), while details regarding computation of this general formula are missing in literature. Since flow and density play a prominent role in model-based analysis and control of long traffic roads, in the first part of the thesis, we focus on microscopic approaches for accurate estimation of these variables in the temporal-spatial sense. The proposed approaches can be applied to any dynamic physical system with moving particles, in particular traffic networks, which are the main concern of this thesis.

The second part of the thesis is focused on developing efficient and fast model-predictive control approaches for systems with a highly nonlinear behavior (such as traffic networks), gradient-based optimization is an efficient and fast method for finding local optima of nonlinear functions. For many nonconvex functions, this approach can still be efficiently applied considering multiple initial starting points for searching. We apply this approach to solve the optimization problem of the MPC controller. In this context, we discuss two cases that may occur in solving a nonlinear MPC optimization problem: smooth case and nonsmooth case, where the first one can be dealt with via gradient-based methods. We then develop general

smoothening approaches to readjust nonsmooth optimization problems into smooth ones, which can be solved by a gradient-based optimization method. The resulting control system is implemented in an urban traffic network with the aim of finding a balanced trade-off between prevention/reduction of traffic congestion and decreasing the level of emitted pollutants. Additionally, to predict the future evolution of the states of the traffic network in a reliable way and within a reasonable time span, we develop a general framework to integrate and interface macroscopic traffic flow models and microscopic emission models, which results in a computationally efficient and accurate mesoscopic traffic flow and emission model that can be used as prediction model for MPC.

In the third and last part of the thesis, we combine predictive and model-based control methods with intelligent control approaches with a low computation time. Consequently, a two-layer adaptive control system with an MPC controller in the top layer and a fuzzy controller in the bottom layer is developed that can be used in a coordinative multi-agent control architecture, in particular for processes with time-delayed states. The fuzzy controller uses a prediction model to estimate the expected inputs of the controlled system within a near future. To this aim, we first extend the affine formulation of the Takagi-Sugeno model-based fuzzy approach to a nonlinear one, and propose a general nonlinear type-2 fuzzy formulation for modeling time-delayed systems. The proposed fuzzy rules are constructed from type-2 fuzzy sets, which can deal with uncertainties and ambiguities regarding the available information for the modeled system. This new formulation can be applied to both the fuzzy controller and the prediction models of both controllers. The fuzzy controller includes a set of adaptive parameters that will be tuned during the online control procedure in both a periodic and an event-triggered way. At the tuning time steps, an optimizer determines an optimal value for the tuning parameters minimizing a cost function that takes into account a given number of past inputs and states, as well as the expected inputs and states in the near future.

In summary, the main contributions of this thesis are

- We introduce accurate microscopic approaches for estimating the temporal-spatial macroscopic traffic variables.

- We develop efficient methods for solving the nonlinear optimization problem of the MPC controller by smoothening the nonsmooth optimization problem and via implementing a gradient-based optimization approach.

- We introduce a general mesoscopic framework for integrating macroscopic traffic flow models with microscopic emission models. The resulting model can provide fast and accurate estimates of the future emissions and the future total time spent by the vehicles.

- We propose an adaptive nonlinear control scheme that integrates fuzzy, model-based, and predictive control approaches and that can be used within a multi-agent control architecture, in particular, for processes that involve time-delayed states.

Anahita Jamshidnejad

Samenvatting

Efficiënte voorspellende modelgebaseerde en vage regelingen voor groene stedelijke mobiliteit

In dit proefschrift ontwikkelen we efficiënte modelgebaseerde regelmethoden, in het bijzonder modelgebaseerde voorspellende regelingen (in het Engels: *model-predictive control* (MPC)) en modelgebaseerde vage regelingen (in het Engels: *fuzzy control*) voor gebruik in stedelijke verkeersnetwerken met het oog op het verminderen van zowel de totale reistijd van de voertuigen in het netwerk als de totale uitstoot. Het proefschrift omvat drie delen, waarbij in het eerste deel de focus ligt op nauwkeurige methoden voor het schatten van macroscopische verkeersgrootheden zoals tijds-ruimte-gemiddelden vanuit een microscopisch oogpunt. Het tweede deel omvat het ontwikkelen van efficiënte methoden voor het oplossen van niet-lineaire MPC-optimalisatieproblemen. In het derde en laatste deel van het proefschrift stellen we een adaptief en voorspellend modelgebaseerd type-2 vage-regelingschema voor dat geïmplementeerd kan worden in een multi-agent regelarchitectuur.

Intensiteit en dichtheid worden in hoofdzaak gebruikt om gedeeltelijk de toestand te karakteriseren van fysische systemen met bewegende deeltjes. Intensiteit en dichtheid zijn macroscopische concepten, d.w.z. dat ze gemiddelde waarden zijn. Voor dynamische systemen met bewegende deeltjes, zoals verkeersnetwerken, kunnen deze gemiddelden op drie verschillende manieren gedefinieerd worden: als tijdsgemiddelde, ruimtegemiddelde, of tijds-ruimte-gemiddelde. De berekening van de eerste twee gemiddelden is eenvoudig, maar voor het derde gemiddelde heeft Edie (1963) enkel een algemene formulering voorgesteld, terwijl details i.v.m. de berekeningen ontbreken in de literatuur. Omdat intensiteit en dichtheid een prominente rol spelen in de modelgebaseerde analyse en regeling van verkeersnetwerken, focussen we in het eerste deel van het proefschift op microscopische methoden voor accurate schatting van deze variabelen in een tijds-ruimte-context. De voorgestelde methoden kunnen gebruikt worden voor elk dynamisch fysisch systeem met bewegende deeltjes, en in het bijzonder voor verkeersnetwerken, die de hoofdonderwerp vormen van dit proefschift.

Het tweede deel van het proefschift richt zich op het ontwikkelen van efficiënte en snelle modelgebaseerde regelmethoden voor systemen met een sterk niet-lineair gedrag (zoals verkeersnetwerken). Gradiëntgebaseerde optimalisatie is een efficiënte en snelle methode voor het vinden van lokale optima van niet-lineaire functies. Voor vele niet-convexe functies kan deze werkwijze nog altijd efficiënt toepast worden door verscheidene startpunten te beschouwen voor de optimalisatie. Wij passen deze werkwijze toe om MPC-optimalisatieproblemen op te lossen. In dit verband beschouwen we twee gevallen die kunnen voorkomen bij het oplossen van niet-lineaire MPC-optimalisatieproblemen: het gladde (in het Engels: *smooth*) geval en het niet-gladde geval, waarbij het eerste geval opgelost kan worden met behulp van gradiëntgebaseerde optimalisatiemethoden. Vervolgens ontwikkelen we algemene methoden om niet-gladde optimalisatieproblemen om te vormen in gladde optimalisatieproblemen, die opgelost kunnen worden met behulp van gradiëntgebaseerde optimalisatiemethoden. Het resulterende regelsysteem wordt vervolgens geïmplementeerd voor een stedelijk verkeersnetwerk met het doel een gebalanceerde afweging te vinden tussen het voorkomen en verminderen van opstoppingen en het verminderen van de uitstoot. Daarnaast ontwikkelen we — met het oog op het betrouwbaar voorspellen van de toekomstige evolutie van de toestanden van een verkeersnetwerk binnen een aanvaardbare tijdspanne — een algemeen raamwerk om macroscopische verkeersstroommodellen en microscopische uitstootmodellen te integreren en met elkaar te koppelen. Dit resulteert in een vanuit rekenoogpunt efficiënt en accuraat mesoscopisch model voor verkeersstromen en uitstoot dat gebruikt kan worden als voorspellingsmodel voor MPC-regeling.

In het derde en laatste deel van het proefschift combineren we voorspellende en modelgebaseerde regelmethoden met werkwijzen uit de intelligente regeling met een lage rekentijd. Op deze manier wordt een twee-laags adaptief vage-regeling-systeem ontwikkeld met een MPC-regelaar in de bovenste laag en een vage regelaar in de onderstaande laag dat gebruikt kan worden in een coördinatieve multi-agent regelarchitectuur, in het bijzonder voor processen met tijdsvertraagde ingangen en toestanden. De vage regelaar gebruikt een voorspellingsmodel om de verwachte ingangen van het geregelde systeem in de nabije toekomst te schatten. Daartoe breiden we eerst de affiene formulering van de Takagi-Sugeno modelgebaseerde vage aanpak uit tot een niet-lineaire formulering en stellen we een algemene niet-lineaire type-2 vage formulering voor voor het modelleren van systemen met tijdsvertragingen. De voorgestelde vage regels worden geconstrueerd op basis van type-2 vage verzamelingen, die kunnen omgaan met onzekerheden en dubbelzinnigheden met betrekking tot de beschikbare informatie over het gemodelleerde systeem. Deze nieuwe formulering kan toegepast worden op zowel de vage regelaar als de voorspellingsmodellen van beide regelaars. De vage regelaar gebruikt een aantal adaptieve parameters die ingesteld worden tijdens de online regelprocedure op zowel een periodieke als gebeurtenis-geïnduceerde manier. Op de insteltijdstippen bepaalt een optimalisatieprocedure de optimale waarden van de instelparameters zodat een kostfunctie geminimaliseerd wordt die rekening houdt met een gegeven aantal ingangen en toestanden uit het verleden alsmede de verwachte ingangen en toestanden in de nabije toekomst.

De hoofdbijdragen van dit proefschift kunnen als volgt samengevat worden:

- We stellen nauwkeurige microscopische methoden voor voor het schatten van tijdsruimte macroscopische verkeersgrootheden.

- We ontwikkelen efficiënte methoden voor het oplossen van niet-lineaire MPC-optimalisatieprobleem door het niet-gladde optimalisatieprobleem glad te maken en door een gradiëntgebaseerde optimalisatiemethode te gebruiken.

- We stellen een algemeen mesoscopisch raamwerk voor voor het integreren van macroscopische verkeersstroommodellen met microscopische uitstootmodellen. Het resulterende model kan snel nauwkeurige schattingen genereren van de toekomstige uitstoot en van de toekomstige reistijd van de voertuigen in een verkeersnetwerk.

- We ontwikkelen een adaptief niet-lineair regelschema dat vage, modelgebaseerde en voorspellende regelaanpakken integreert en dat kan gebruikt worden in een multi-agent regelarchitectuur, in het bijzonder voor processen met tijdsvertraagde toestanden.

Anahita Jamshidnejad

List of Publications

Published International Peer-Reviewed Journal Papers

8. A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter, “Sustainable Model-Predictive Control in Urban Traffic Networks: Efficient Solution Based on General Smoothening Methods", *IEEE Transactions on Control Systems Technology*, 2017. doi: 10.1109/TCST.2017.2699160. To appear.

7. A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter, “A Mesoscopic Integrated Urban Traffic Flow-Emission Model", *Transportation Research part C: Emerging Technologies*, vol. 75, pp. 45–83 , 2017. doi: 10.1016/j.trc.2016.11.024.

6. A. Jamshidnejad, B. De Schutter,“An Algorithm for Estimating the Generalized Fundamental Traffic Variables From Point Measurements Using Initial Conditions", *Transportmetrica B: Transport Dynamics*, 2017. doi: 10.1080/21680566.2017.1279991. To appear.

5. A. Jamshidnejad, B. De Schutter,“Estimation of the Generalized Traffic Average Speed Based on Microscopic Measurements", *Transportmetrica A: Transport Science*, vol. 11, no. 6, pp. 525–546, 2015. doi: 10.1080/23249935.2015.1026957.

Journal Papers Under Review

4. A. Jamshidnejad, H. Hellendoorn, M. J. Mahjoob, and B. De Schutter, “Multi-Agent Adaptive Model-Based Predictive Type-2 Fuzzy Control System", submitted to *IEEE Transactions on Cybernetics*.

3. A. Jamshidnejad, G. Gomes, D. Haziza, F. Belletti, A. M. Bayen, and B. De Schutter, “Integrated Offline and Online Predictive Control System within a Base-Parallel Architecture”, submitted to *IEEE Transactions on Control Systems Technology*.

Journal Papers Under Preparation

2. A. Jamshidnejad, A. Ferrara, E. Boerci, and B. De Schutter, “Hierarchical Model-Predictive Urban Traffic Control with Spatial and Temporal Distribution for Green Mobility", to be submitted to *IEEE Transactions on Control Systems Technology*.

1. A. Jamshidnejad, R. van Kooten, P. Imhof, K. Brummelhuis, M. van Pampus, and B. De Schutter, “ART-UTC: A Novel Adaptive Real-Time Urban Traffic Control Strategy”, to be submitted to *IEEE Transactions on Intelligent Transportation Systems*.

International Peer-Reviewed Conference Papers

8. A. Jamshidnejad, I. Papamichail, H. Hellendoorn, M. Papageorgiou, and B. De Schutter, “Gradient-Based Model-Predictive Control for Green Urban Mobility in Traffic Networks”, Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems, Rio de Janeiro, Brazil, pp. 1077–1082 Nov. 2016.

7. A. Jamshidnejad, I. Papamichail, M. Papageorgiou, and B. De Schutter, “A Model-Predictive Urban Traffic Control Approach with a Modified Flow Model and Endpoint Penalties”, Proceedings of the IFAC Control in Transportation Systems 2016, Istanbul, Turkey, pp. 147–152, May 2016.

6. A. Jamshidnejad, H. Hellendoorn, S. Lin, and B. De Schutter, “Smoothening for Efficient Solution of Model-Predictive Control for Urban Traffic Networks Considering Endpoint Penalties”, Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Las Palmas de Gran Canaria, Spain, pp. 2837-2842, Sept. 2015.

5. A. Jamshidnejad, B. De Schutter, and M. Mahjoob, “Urban Traffic Control Using a Fuzzy Multi-Agent System”, Proceedings of the 2015 European Control Conference (ECC), Linz, Austria, pp. 3046-3051, July 2015.

4. A. Jamshidnejad, B. De Schutter,“An Iterative Procedure for Estimating the Generalized Average Speed Using Microscopic Point Measurements”, Proceedings of the 2015 Models and Technologies for Intelligent Transportation Systems (MT ITS), Budapest, Hungary, pp. 38-44, June 2015.

3. A. Jamshidnejad, B. De Schutter,“A Multi-Objective Model-Predictive Control Approach Dealing with Congestion and Emissions in Urban Traffic Networks”, Proceedings of the 2015 ARTS ECR, La Valletta, Malta, 6 pp., May 2015.

2. A. Jamshidnejad, M. J. Mahjoob, “Urban Traffic Fuzzy Prototypes Using a Graph-Based Two-Stage Clustering Algorithm”, Proceedings of the 2nd International Conference on Control, Instrumentation, and Automation (ICCIA 2011), Shiraz, Iran, pp. 102-105, Dec. 2011.

1. A. Jamshidnejad, M. J. Mahjoob, “Traffic Simulation of an Urban Network System Using Agent-Based Modeling”, Proceedings of the IEEE Colloquium on Humanities, Science and Engineering Research (CHUSSER 2011), Penang, Malaysia, pp. 300-304, Dec. 2011.

Curriculum Vitae

Anahita Jamshidnejad finished her pre-university educations at the National Organization for Development of Exceptional Talents (NODET/SAMPAD) in Iran, and was accepted to the AmirKabir University of Technology (Tehran Polytechnic), in Tehran, Iran as an "Elite student". She obtained the B.Sc. degree in Mechanical Engineering from the School of Mechanical Engineering of Tehran Polytechnic in 2009, and got admitted to the University of Tehran, in Tehran, Iran for continuing her M.Sc. education. She received an M.Sc. degree as top 3 students in Mechanical Engineering (Systems dynamics and control) in January 2012.

From May 2013 she started her PhD at the Delft Center for Systems and Control (DCSC), Delft University of Technology. She succeeded to obtain the DISC certificate of the Dutch Institute of Systems and Control (DISC) in 2013. As the main topics for her PhD research, she worked on efficient modeling and control approaches (including model-predictive control) for freeway and urban traffic networks, and on coordinative fuzzy multi-agent systems under the supervision of Prof. dr. ir. Bart De Schutter, Prof. dr. .ir. Hans Hellendoorn, and Prof. dr. Markos Papageorgiou. She had a 2-month research visit at the Dynamic Systems & Simulations Laboratory, University of Crete, in Chania, Greece in 2015, where she collaborated with the group of Prof. Markos Papageorgiou and Prof. Ioannis Papamichail. She also visited the Institute of Transportation Studies, University of California, Berkeley, in US during October-November 2016, and collaborated with Prof. Alexandre Bayen and his research group.

Anahita Jamshidnejad's main research interests include fast model-predictive control, cooperative and coordinative multi-agent systems, intelligent and distributed control of large-scale systems, and event-triggered and nonlinear systems.