

Technische Universität  
München

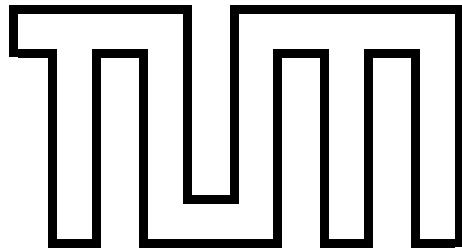
Department of Mathematics

Master's Thesis in Mathematics

Modelling and control of traffic flow on networks

Markus Stachl





Technische Universität  
München

Department of Mathematics

Master's Thesis in Mathematics

Modelling and control of traffic flow on networks

Modellierung und Kontrolle von Fahrzeugverkehr auf  
Netzwerken

**Author:** Markus Stachl

**Supervisor:** Prof. Dr. Massimo Fornasier

**Advisor:** Dr. Giacomo Albi

**Submission:** 18.04.2017

I assure the single handed composition of this master's thesis only supported by declared resources.

München, 18.04.2017

*(Markus Stachl)*



## Abstract

The flow of motorised vehicles through urban road networks is known to be one of the main reasons for high pollution in metropolitan areas. Continuous growth of cities and therefore an increased traffic volume on roads, mainly caused by commuter traffic, triggers the need for improved planning and optimal control of vehicular traffic. Although an improved traffic flow is assumed to result in lower vehicle emissions, little scientific research has been spent on the effects of coordinated traffic lights on emission. In this thesis we try to close this gap. We resort to a macroscopic model to simulate traffic flow on networks, which was derived by Bressan et al and which models junctions using buffers with limited capacity. This guarantees well-posedness and unique solvability of the underlying conservation law. The management of traffic flow at junctions is handled by traffic lights, whose control influences the traffic volume on the entire road network. We show that by controlling traffic lights using responsive techniques like *Model Predictive Control* (MPC) the cumulated traffic flux can be improved compared to traffic lights with fixed cycle length (the time difference between red and green phases). Coupling the model of vehicular dynamics with a one-dimensional pollution model on a road network in Munich shows that a decrease of  $CO$ -emissions by 5-10 % can be achieved by the optimal control of traffic lights.

## Zusammenfassung

Der Strom motorisierter Fahrzeuge durch innerstädtische Straßennetzwerke ist bekanntermaßen eine der Hauptursachen für die hohe  $CO$ -Belastung in Metropolregionen. Durch das anhaltende Städtewachstum und das damit einhergehende erhöhte Verkehrsaufkommen auf den Straßen, verursacht vor allem durch intensiven Berufs- und Pendelverkehr, bildet sich die Notwendigkeit der verbesserten Planung und optimalen Steuerung des Fahrzeugverkehrs. Obwohl angenommen wird, dass ein verbesserter Verkehrsfluss auch die Menge ausgestoßener Schadstoffe reduziert, wurde dies in nur geringem Maße wissenschaftlich untersucht. In dieser Arbeit versuchen wir diese Lücke zu schließen. Dazu greifen wir auf ein makroskopisches Modell zur Simulation des Verkehrsflusses auf Netzwerken zurück, welches von Bressan et. al entwickelt wurde und Kreuzungen durch Zwischenspeicher mit beschränkter Kapazität modelliert. Dies garantiert eindeutige Lösbarkeit der zugrundeliegenden Erhaltungsgleichungen. Die Regelung des Verkehrsflusses an Kreuzungen wird von Ampeln übernommen, deren Steuerung die Verkehrsbelastung auf dem gesamten Straßennetz beeinflusst. Wir zeigen, dass durch die Steuerung der Ampeln mithilfe eines responsiven Verfahrens wie *Model Predictive Control* (MPC) der Verkehrsfluss gegenüber Ampeln mit fixierten Schaltzyklen verbessert werden kann. Eine Kopplung des Modells für die Verkehrsdynamik auf einem Straßennetzwerk in München mit einem eindimensionalen Emissionsmodell zeigt anschließend, dass durch optimale Steuerung der Verkehrampeln eine Verringerung des  $CO$ -Ausstoßes um 5-10% erzielt werden kann.

## Acknowledgements

I would like to express my sincere thanks to those who offered feedback and support throughout this thesis. Starting with *Prof. Dr. Massimo Fornasier* for offering me this applied topic and his valuable input. Next, my supervisor *Dr. Giacomo Albi* for the introduction to an interesting research field, his continuous support and our fruitful discussions in Munich and Verona.

Furthermore, I want to thank my family for their faith and their constant support during the past years and particularly months. And last but not least, I want to thank Stephan Roder, Matthias Kornherr and Martin Schlecker for proofreading this thesis.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Traffic flow modelling . . . . .	2
1.3	Traffic flow optimization . . . . .	5
1.4	Contributions . . . . .	6
1.5	Organization . . . . .	6
<b>2</b>	<b>The LWR model on networks</b>	<b>8</b>
2.1	Related literature . . . . .	8
2.2	Basic principles of networks . . . . .	9
2.3	Modelling of junctions - the buffer model . . . . .	12
<b>3</b>	<b>Numerical approximations</b>	<b>19</b>
3.1	The Godunov scheme for nonlinear conservation laws . . . . .	19
3.1.1	Theory of conservative numerical schemes . . . . .	20
3.1.2	The Godunov scheme for the buffer model . . . . .	23
3.2	Numerical example . . . . .	24
<b>4</b>	<b>Flux optimization via traffic light control</b>	<b>26</b>
4.1	Green wave strategy: Traffic light coordination . . . . .	29
4.1.1	The model . . . . .	29
4.1.2	Experiments on coordinated light signals . . . . .	30
4.2	Optimization via Model Predictive Control . . . . .	33
4.2.1	Problem formulation . . . . .	36
4.2.2	Parameter reduction: minimum green phase time . . . . .	40
4.2.3	Numerical experiments . . . . .	43
<b>5</b>	<b>Pollution modelling</b>	<b>50</b>
5.1	CO emission in urban traffic: an instantaneous model . . . . .	50
5.2	The 2D diffusion model . . . . .	54
5.3	A case study: Munich - Schleißheimer Straße . . . . .	58
<b>6</b>	<b>Discussion and further research</b>	<b>71</b>

<i>CONTENTS</i>	vi
-----------------	----

<b>Appendices</b>	<b>73</b>
<b>A Solutions to scalar hyperbolic equations</b>	<b>74</b>
<b>B Limitations of the junction model by Garavello</b>	<b>76</b>
<b>C Finite difference schemes</b>	<b>79</b>
<b>D Traffic modelling toolbox</b>	<b>81</b>

# Chapter 1

## Introduction

### 1.1 Motivation

A green wave for a driver occurs when sequential traffic lights are coordinated in a way such that the driver is allowed to cross every intersection without having to wait in front of a red light.

First and foremost this is convenient for the driver who can reach his destination in the shortest time possible. Additional studies have shown that the establishment of green waves can have benefits not only for the individual driver, but also for the general public, in particular on an environmental level [16, 35, 40].



Figure 1.1: Traffic congestion in Minnesota (2006)<sup>1</sup>



Figure 1.2: Circular traffic congestion in Sabah, Malaysia (2014)<sup>2</sup>

Emissions from automobiles and trucks operating on public roads represent a major portion of the pollutants in urban areas (see for instance [23, 59]). Besides other substances the highly poisonous gas carbon monoxide ( $CO$ ) is emitted mainly by urban traffic.  $CO$  is the most abundant air pollutant in the lower atmosphere with exception of  $CO_2$ . It is a byproduct caused by incomplete combustion of fuel used to operate cars.

The study by De Coensel [16] in 2012 has shown that green wave policies have the potential to

- reduce fuel consumption of vehicles,
- reduce  $CO, CO_2$  and particulate matter (PM) caused by urban traffic by 10-40%,
- reduce waiting times for the driving public
- and reduce the noise caused by rapid acceleration and deceleration of cars by up to 1.5 dB(A).

In this thesis we will discuss different mathematical tools to generate green waves over a network of several intersections in one main direction. The general approach here is to maximize the total traffic flux on the network. The goal is to create a generic simulation tool that compares the different mathematical methods in terms of computational effort and optimal flux. The obtained results finally are post-processed in order to compare the performance of the different methods with respect to generated pollution.

## 1.2 Traffic flow modelling

Analyzing traffic flow has been an interdisciplinary research field of both mathematicians and civil engineers since the early 1950s. Over time different approaches for modelling traffic have been introduced and applied in extensive studies.

- On the **microscopic** scale every vehicle is considered as an individual agent whose dynamics are determined by the solution of an ordinary differential equation (ODE). Interaction between neighboring vehicles are usually based on simple equations and determine the behaviour of the collective. The most famous models of this approach are those of the *follow-the-leader* kind [48]. In those models the entire behaviour of the collection of cars is fully determined by the dynamics of the first car, namely the leader. Microscopic models can not only illustrate collective phenomena like traffic jams, but it can also be shown that under certain conditions the microscopic solution converges to the macroscopic solution as the number of vehicles approaches infinity [21].
- **Mesoscopic** - or kinetic - models analyze transportation elements in small homogeneous groups, where the probability of each group to be at time  $t$  at location  $x$  with velocity  $v$  can be described by a function  $f(t, x, v)$ . The family of mesoscopic models includes headway distribution models [13] and cluster models [41]. This modelling approach often requires the use of methods from statistical mechanics.

---

<sup>1</sup><https://www.flickr.com/photos/146264147@N03/33417084342/>. Attribution: non-commercial (<https://creativecommons.org/licenses/by-nc/2.0/>), Photo attribution by <http://www.photosforclass.com>

<sup>2</sup><https://www.flickr.com/photos/beggs/34117133>. Attribution: non-commercial (<https://creativecommons.org/licenses/by-nc/2.0/>)

- **Macroscopic** models utilize the similarities of traffic flow and fluid dynamics. In those models the change of averaged quantities, like density, velocity etc., is described by means of partial differential equations (PDE). The oldest representatives of macroscopic models are the LWR-models [51], named after their inventors M. J. Lighthill, G. B. Whitham and P. I. Richards. Others include the Aw-Rascle model and the Payne-Whitham approach [2, 47]. For the rest of this thesis this is the modelling approach of choice.

In macroscopic models the target is to analyze the change of measurable quantities. A typical quantity used in the analysis of traffic flow, analogous to fluid dynamics, is the density of mass. Let us denote the traffic density on a space interval  $[x_1, x_2]$  at some time  $t \in \mathbb{R}$  as  $\rho(x, t)$ , then we can write the total mass on this interval as

$$\text{amount of cars} = \int_{x_1}^{x_2} \rho(x, t) dx. \quad (1.2.1)$$

For conserved quantities no mass is neither destroyed nor created. Therefore the mass on an interval can only change by inflowing and outflowing quantities. The quantity mass crossing the point  $x$  at a time  $t$  is given by the **traffic flux**  $f : [0, \rho_{max}] \rightarrow [0, f^{max}]$ , where  $\rho_{max}$  denotes the maximum car density on the network.

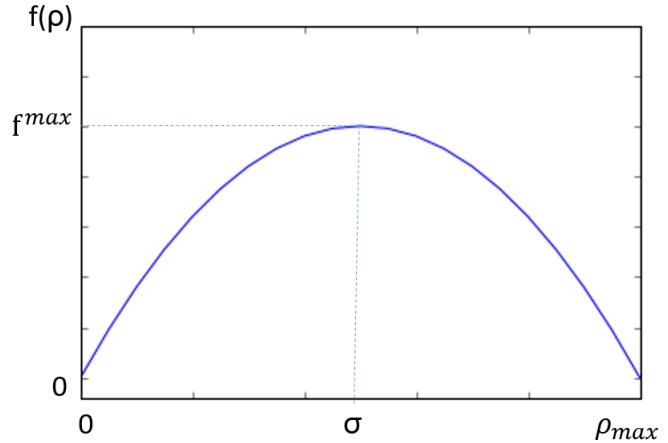


Figure 1.3: Typical shape of the fundamental diagram of traffic flow.

The flux  $f$ , by assumption, is a Lipschitz continuous function satisfying  $f(0) = f(\rho_{max}) = 0$ , meaning there is no movement of cars if the road is empty or fully congested. We also assume that the traffic flux, or also referred to as *fundamental diagram*, follows the rule

$$f'(\rho)(\sigma - \rho) > 0,$$

where

$$\sigma := \arg \max_{\rho} f(\rho) \quad (1.2.2)$$

denotes the density where the maximum flux

$$f^{max} := f(\sigma)$$

is attained. This value is usually referred to as *road capacity*.

Since the total mass on the global space interval is preserved, the rate of change of the mass on the interval  $[x_1, x_2]$  is defined by the difference of the in- and outfluxes of the interval (assuming that  $\rho(x, \cdot) \in L^1_{loc}$ ), e.g.

$$\partial_t \int_{x_1}^{x_2} \rho(x, t) dx = f(\rho(x_1, t)) - f(\rho(x_2, t)).$$

Rewriting this under the assumption of differentiability of both functions  $\rho$  and  $f$  and integration over the time interval  $[t_1, t_2]$  leads to:

$$\int_{t_1}^{t_2} \int_{x_1}^{x_2} \partial_t \rho(x, t) dx dt = \int_{t_1}^{t_2} \int_{x_1}^{x_2} \partial_x \{-f(\rho)\} dx dt. \quad (1.2.3)$$

Assuming that  $f$  is smooth, this leads to the differential form of the **conservation law**:

$$\partial_t \rho + \partial_x (f(\rho)) = 0. \quad (1.2.4)$$

The LWR-model, as it will be the basis for the remaining thesis, assumes that the flux  $f(\rho) = \rho(x, t)v(x, t)$  is linearly dependent on the velocity  $v(x, t)$  of the wave at point  $x$ . In particular the flux used for the LWR-model can be written as

$$f(\rho(x, t)) = \rho(x, t)v_{max} \left( 1 - \frac{\rho(x, t)}{\rho_{max}} \right), \quad (1.2.5)$$

where  $u_{max}$  is the maximum density permitted by the road, and  $v_{max}$  the maximum velocity obtainable by the wave.

Other approaches like the Greenberg and Payne-Whitham model use highly nonlinear functions for their wave velocities, see [43, 47].

Extending the conservation law 1.2.4 with suitable initial conditions, we can formulate the full **Lighthill-Whitham-Richards model** as

$$\begin{aligned} \partial_t \rho + \partial_x \left[ v_{max} \rho \left( 1 - \frac{\rho}{\rho_{max}} \right) \right] &= 0 & t > 0 \\ \rho(x, 0) &= \rho_0(x). \end{aligned} \quad (1.2.6)$$

The solution to this type of problems can be found via the method characteristics or via solving the corresponding Riemann problem (see Appendix A).

For an exhaustive and comprehensive genealogy of the history of traffic flow modelling we refer to [60].

## 1.3 Traffic flow optimization

In recent years the focus of academic research has shifted towards the optimization of traffic flow on complex networks. The optimal control of traffic flow is an active field of mathematical research, occupying scientists from different backgrounds. As urbanization continues, regulating and guiding vehicular traffic becomes an increasingly difficult but important task. Typically the studied models, ranging from cellular automata to macroscopic models, attempt to minimize the average travel time or maximize the total traffic flux by controlling the cycle length (the time difference between red and green phases) of traffic lights.

Depending on the target online or offline optimization tools can be used to regulate the traffic, depending on the availability and accessibility of current traffic flow data. Two widely used adaptive optimization packages are the 'spit cycle offset optimization technique' (SCOOT) [34] and 'Sydney coordinated adaptive traffic systems' (SCATS) [39]. These kinds of systems use traffic detectors often embedded in roads in order to measure the current rate of road utilization.

Google Maps offers a feature called Google Traffic, displaying traffic conditions in real time. The data obtained from GPS-determined locations by a large number of mobile phone users can be useful to gather information about current traffic conditions and adjust traffic lights accordingly.

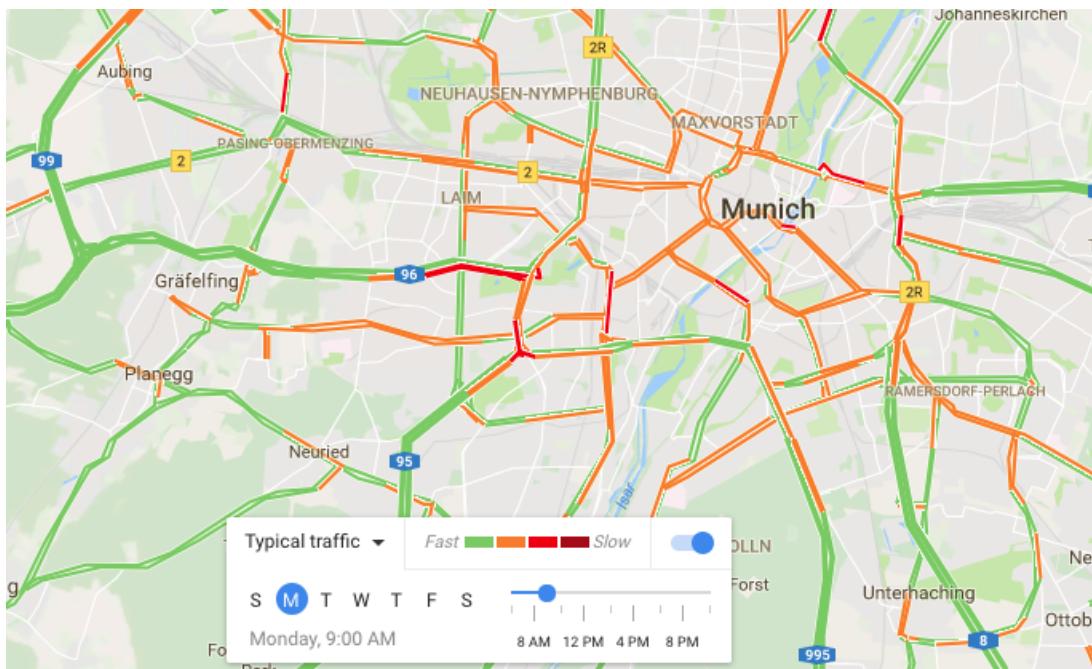


Figure 1.4: Current traffic displayed on Google Maps.<sup>3</sup>

---

<sup>3</sup>Map data ©2017 GeoBasis-DE/BKG (©2009), Google

Mathematical methods like mixed-integer programming, Model Predictive Control and gradient descent are used in this context to regulate traffic via variable speed limits [46] or by regulating junction flows with traffic lights [30, 29].

## 1.4 Contributions

This thesis offers several contributions. Firstly, to the best of our knowledge, we offer the first numerical implementation of the buffer model proposed by Bressan et al. We provide an extensive discussion of the benefits and limitations of this modelling approach and the significance of buffers in real traffic networks. Secondly, the switching patterns of traffic light signals are optimized using Model Predictive Control. We can show that the total flux can be improved significantly compared to traffic lights operating at a fixed control strategy due to higher responsivity to changing traffic situations. Furthermore, we couple the traffic dynamics with a non-linear pollution model, where we show that the amount of pollution generated by vehicular traffic on road networks can also be reduced. Finally, a Python framework is provided enabling the analysis of arbitrary networks in terms of traffic flux and generated pollution.

## 1.5 Organization

The subsequent chapters of this thesis have the following structure:

**Chapter 2** extends the given LWR-model on a network of roads in order to consider urban traffic situations. Moreover, we introduce the concept of buffers to guarantee well-posedness of the model. We also give an overview of different modelling techniques for traffic flow as well as current research areas.

In **Chapter 3** we provide a numerical formulation for general conservation laws and the buffer model in particular, permitting us to simulate traffic flow computationally. We introduce the concept based on the *Godunov scheme* and discuss its benefits compared to general finite difference schemes.

**Chapter 4** is devoted to the optimization of cars flowing on the network through consideration of traffic light signals. We propose two strategies, the first one consisting on direct synchronization of traffic lights, the second one based on *Model Predictive Control* (MPC). The control schemes are implemented and discussed through experiments on simple road networks.

In **Chapter 5** the LWR-model is extended by the concept of *pollution* where the LWR model is coupled with a non-linear pollution term. The acquired knowledge is subsequently applied to a road network part of the inner city of Munich, where the optimization schemes of traffic flow are discussed intensively.

The thesis concludes in **Chapter 6** with a discussion and an overview on future research.

# Chapter 2

## The LWR model on networks

The introductory part in Chapter 1 describes traffic flow problems on a single, infinitely long road. The analysis of more sophisticated traffic scenarios, in particular in urban environments, requires the introduction of junctions. Urban road systems can be represented by networks of roads and junctions where the mathematical knowledge on graph theory can be harnessed for deeper analysis. In the following traffic networks are modelled via directed graphs, consisting of a set of unidirectional edges, representing the roads, and a set of nodes for the junctions. Every junction is thereby characterized by a finite number of incoming and outgoing roads. Junctions represent points of discontinuity, where cars transit from incoming to outgoing roads.

The Lighthill-Whitham-Richards equation is assumed to hold on the edges of the network, away from intersections, i.e.

$$\begin{aligned}\partial_t \rho_i(x, t) + \partial_x f(\rho_i(x, t)) &= 0 \quad \forall e_i \in \mathcal{E}, x \in (a_i, b_i), t \geq 0, \\ \rho_i(x, 0) &= \rho_{i,0}(x) \quad \forall x \in (a_i, b_i).\end{aligned}\tag{2.0.1}$$

In the following section we first give a short overview over the field of traffic flow analysis on networks.

### 2.1 Related literature

Traffic flow on road networks has been a widely discussed topic over the past 70 years with an increasing interest in the past century due to the continuous growth of cities and as a consequence of the growing complexity of road networks. Beginning with the application of the wavefront tracking method introduced by Dafermos [20] and applied to road networks (e.g. by Bressan [8, 10]), a wide variety of models to describe traffic flow on networks has been developed in the past years.

Garavello introduced his junction model in [26] which distributes fluxes at a junction based on a convex optimization approach. Unfortunately it can be shown that this approach does not provide uniqueness for resulting solutions under certain conditions (see [9, 11] and Appendix B).

Briani et al [12] introduced another approach in 2014, the so-called *multi-path model*. Their model considers different populations of cars depending on the path the vehicle is taking through the network and the LWR PDE is then individually solved for every single population. The main advantage of this approach is that junctions are not defined explicitly but only appear as discontinuities in the traffic density on the path. As this approach assumes the knowledge of every path of every single vehicle on the network and as the number of different paths increases rapidly for more complex networks this approach might not be as useful for bigger networks.

The approach by Claudel et al. [44] uses the cumulative number of vehicles (CVN function) as an intermediate computational abstraction. This CVN function is the integral form of the density function and solves a Hamilton-Jacobi PDE, while the density function solves the LWR PDE.

## 2.2 Basic principles of networks

Rigorously we can define a network as followed, cf. [26]:

**Definition 2.2.1** A network is a tuple  $(\mathcal{N}, \mathcal{E})$ , where  $\mathcal{N}$  is a finite collection of vertices, and  $\mathcal{E}$  a finite collection of  $n_R$  edges, where every node represents a junction and every edge represents a unidirectional road. Each road  $e_i = [a_i, b_i] \in \mathbb{R}$ ,  $i = 1 \dots n_R$  is defined over a real interval.

We further require the following properties:

- 1) Every junction  $J \in \mathcal{N}$  defines a union of two non-empty sets  $\delta^{in}(J)$  and  $\delta^{out}(J)$  where  $\delta^{in}(J), \delta^{out}(J) \subset \mathcal{E}$ , representing incoming and outgoing roads of the junction  $J$ , respectively.
- 2) For junctions  $I, J \in \mathcal{N}$ ,  $I \neq J$  we require  $\delta^{in}(I) \cap \delta^{in}(J) = \emptyset$  and  $\delta^{out}(I) \cap \delta^{out}(J) = \emptyset$ .
- 3) If  $e_i \notin \bigcup_{J \in \mathcal{N}} \delta^{in}(J)$  then  $b_i = \infty$ . In this case  $e_i \in \mathcal{E}^{out}$  where  $\mathcal{E}^{out}$  denotes the set of all outgoing roads for the network. Furthermore, if  $e_i \notin \bigcup_{J \in \mathcal{N}} \delta^{out}(J)$  then  $a_i = -\infty$ . In this case  $e_i \in \mathcal{E}^{in}$  where  $\mathcal{E}^{in}$  denotes the set of all incoming roads for the network.

These assumptions guarantee that the resulting network is indeed a valid graph. Requirement 2) implies that every road starts and ends in at most one junction. Requirement 3) then says that if a road does not start (end) at a junction then it is considered as an inflow (outflow) of the network.

Another method to represent a road network is via so-called **adjacency matrices** [4].

**Definition 2.2.2** Let  $\mathcal{N}$  be a set of edges with  $|\mathcal{N}| = n_R$ . Then the adjacency matrix  $A$  is a square  $n_R \times n_R$ -matrix with

$$A_{ij} = \begin{cases} 1 & \text{if there exists a junction between road } e_j \text{ and } e_i \\ 0 & \text{else.} \end{cases}$$

The adjacency matrix stores information if there is a direct connection between road  $e_j$  and  $e_i$ .

### Remark 1

The adjacency matrix only provides information about the general infrastructure of the network. Later on we will incorporate information about the drivers' turning preferences  $\Theta_{i,j}$ . We then speak of the transition matrix  $\Theta$  with

$$\Theta_{ji} = \begin{cases} \Theta_{i,j} & \text{if there exists a junction between road } e_j \text{ and } e_i \\ 0 & \text{else,} \end{cases}$$

where  $\Theta_{i,j} \in [0, 1]$  reflects the preference of a driver coming from road  $e_i$  and leaving the junction in direction of road  $e_j$ .

**Example 1** Let us consider the network displayed in Figure 2.1. Then the corresponding adjacency matrix can be written as

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}.$$

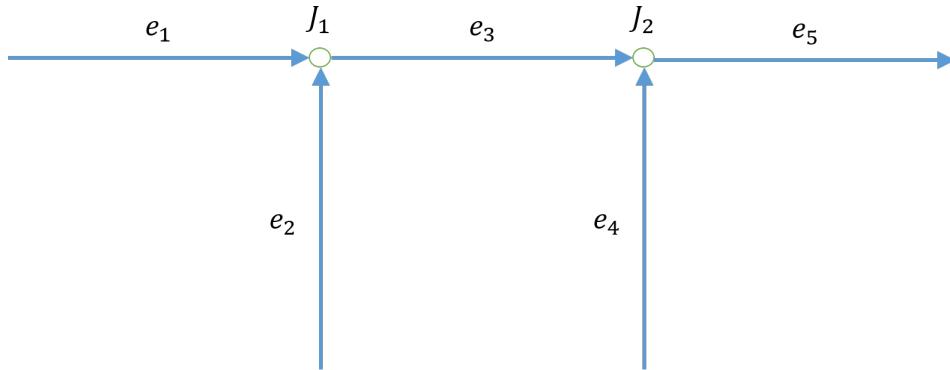


Figure 2.1: Example of a network.

Generally, for every road  $e_i$  we write  $\rho_i : [a_i, b_i] \times [0, \infty) \rightarrow [0, 1]$  for the density of cars at any point  $x \in [a_i, b_i]$  at any time  $t \geq 0$ . We then want  $\rho_i$  to be a weak entropy solution on  $[a_i, b_i]$ , i.e. for every smooth test function  $\phi : [a_i, b_i] \times [0, \infty) \rightarrow \mathbb{R}$  with compact support on  $(a_i, b_i) \times (0, \infty)$

$$\int_0^\infty \int_{a_i}^{b_i} (\rho_i \partial_t \phi + f(\rho_i) \partial_x \phi) dx dt = 0. \quad (2.2.1)$$

**Definition 2.2.3** Let  $J$  be a junction with  $n$  incoming roads and  $m$  outgoing roads. A weak solution to equation 2.0.1 at  $J$  is a collection of functions  $\rho_i : [a_i, b_i] \times [0, \infty) \rightarrow [0, 1]$ ,  $i = 1, \dots, n+m$ , such that

$$\sum_{i=1}^{n+m} \left( \int_0^\infty \int_{a_i}^{b_i} (\rho_i \partial_t \phi_i + f(\rho_i) \partial_x \phi_i) dx dt + \int_{a_i}^{b_i} \rho_i(x, 0) \phi_i(x, 0) dx \right) = 0 \quad (2.2.2)$$

for smooth functions  $\phi_i$ ,  $i = 1, \dots, n+m$ . In particular  $\phi$  is also smooth across junctions, namely

$$\phi_i(b_i, \cdot) = \phi_j(a_j, \cdot), \quad \partial_x \phi_i(b_i, \cdot) = \partial \phi_j(a_j, \cdot).$$

One is now looking for a well-defined problem on the whole network. To do so one has to give a formulation of a well-defined problem at junction. From Definition 2.2.3 we can also follow the so-called Rankine-Hugoniot condition.

**Lemma 2.2.4** Rankine-Hugoniot condition

Let  $\rho = (\rho_1, \dots, \rho_{n+m})$  be a weak solution at junction  $J$ . Then  $\rho$  satisfies the Rankine-Hugoniot condition

$$\sum_{i=1}^n f(\rho_i(b_i, t)) = \sum_{j=1}^m f(\rho_j(a_j, t)) \quad \forall t. \quad (2.2.3)$$

Following [26] we can define the Riemann problem at junctions as in Definition 2.2.5. A Riemann solver is a map assigning a solution to each Riemann initial data. Since we consider only centered solutions (at the junction), it is sufficient to study constant initial data  $\tilde{\rho} : i$  on each road.

**Definition 2.2.5** Riemann solver at a junction

A Riemann solver for the vertex  $J$  with  $n$  incoming and  $m$  outgoing roads is a function

$$RS : \mathbb{R}^{n+m} \rightarrow \mathbb{R}^{n+m}$$

that associates to every Riemann data  $\tilde{\rho} = (\tilde{\rho}_0, \tilde{\rho}_1, \dots, \tilde{\rho}_{n+m-1})$  at  $J$  a vector  $\hat{\rho} = (\hat{\rho}_0, \hat{\rho}_1, \dots, \hat{\rho}_{n+m-1})$  so that the following holds:

On each edge  $e_i$ ,  $i = 0, \dots, n+m-1$ , the solution is given by the solution to the initial value boundary problem (IVBP) with initial data  $\tilde{\rho}_i$  and boundary data  $\hat{\rho}_i$ .

We require the consistency condition

$$RS(RS(\tilde{\rho})) = RS(\tilde{\rho}).$$

Once a Riemann solver is assigned admissible solutions at  $J$  can be assigned in the following way.

**Definition 2.2.6** Assume a Riemann solver  $RS$  is assigned at junction  $J$ . Let  $\rho = (\rho_0, \dots, \rho_{n+m-1})$ ,  $\rho_i : [a_i, b_i] \times [0, +\infty) \rightarrow \mathbb{R}$  be such that  $\rho_i(t, \cdot)$  is of bounded variation for every  $t \geq 0$ . Then  $u$  is an admissible weak solution to equation 2.0.1 related to the  $RS$  at vertex  $J$  iff:

- i)  $\rho_i$  is a weak solution to equation 2.0.1 on the edge;
- ii) for almost every  $t$ , setting

$$\rho_J(t) = (\rho_0(b_0-, \cdot), \dots, \rho_{n-1}(b_{n-1}-, \cdot), \rho_n(a_n+, \cdot), \dots, \rho_{n+m-1}(a_{n+m-1}, \cdot))$$

we have

$$RS(\rho_J(t)) = \rho_J(t),$$

This general definition also includes non-physical cases. For some applications it may be required that the total mass on the network is conserved, meaning that the inflow into the junction equals the outflow (see Rankine-Hugoniot condition). As a second condition to ensure physically valid solutions one must ask for solutions to the IVBP to produce waves with negative characteristic velocities on incoming roads and waves with positive velocities on outgoing roads.

## 2.3 Modelling of junctions - the buffer model

In the following we introduce the buffer model derived by Bressan [9]. The basic idea states that vehicles not being able to pass the junction instantaneously towards outgoing roads instead have to queue and wait until conditions on the outgoing road permit them to leave the junction. A typical illustration for the concept of buffers are roundabouts (Figure 2.2).

### The setting

Consider a family of  $n + m$  roads, all joining at a single junction. The indices  $i \in \{1, \dots, m\} =: \mathcal{I}$  hereby denote *incoming roads* and indices  $j \in \{m + 1, \dots, m + n\} =: \mathcal{O}$  denote *outgoing roads*. Then the evolution of the density of cars  $\rho_k(x, t)$  on the  $k$ -th road can be described by the **conservation law**

$$\partial_t(\rho_k) + \partial_x f(\rho_k) = 0. \quad (2.3.1a)$$

accompanied by initial densities  $\rho_k(x, 0) = \rho_{k,0}(x)$  and external inflows  $f(\rho_{in,k})$  into the network defined by

$$f(\rho_k(a_k, t)) = f(\rho_{in,k}(t)) \quad \forall e_k \in \mathcal{E}^{in}, \quad (2.3.1b)$$

where  $\rho_{in,k}(t)$  denotes the incoming density into road  $e_k \in \mathcal{E}^{in}$ . Furthermore, the corresponding inflows and outflows of the junctions are given as

$$f(\rho_k(b_k, t)) = \gg f_k(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{out} \quad (2.3.1c)$$

$$f(\rho_k(a_k, t)) = f_k^{\gg}(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{in}, \quad (2.3.1d)$$

where the junction fluxes  $\gg f_k$  and  $f_k^{\gg}$  are defined later in equation (2.3.7).

For the roads leaving the network we also impose Dirichlet boundary conditions.

## The buffer model

In order to be useful for the analysis of global optimization, the used traffic flow model at junctions should provide two crucial properties:

- Well-posedness for  $\mathcal{L}^\infty$  data
- Continuous solution w.r.t. weak convergence

Due to the ill-posedness of the general junction model in [26] for certain input data (see Appendix B, we need to come up with a different approach.

In [9] a model is proposed where each intersection in the model includes a **buffer** with limited capacity. The current filling level of the buffer in front of the outgoing road  $j \in \mathcal{O}$  may be denoted as the *queue length*  $q_j(t)$ . The rate of flux at which cars from incoming roads enter the intersection is controlled by the current length of the queues. The outgoing fluxes are governed by the queue length and the personal preference destination of the individual drivers, as well as the maximum permitted fluxes on the designated roads.

The main results of their analysis can be stated as:

- I. If the queue lengths  $q_j$  for every outgoing road are given, the initial boundary value problems on each road become decoupled and can be solved individually, first on every incoming road, and secondly on every outgoing road. The densities  $\rho_k(t, x)$  on every road  $k = 1 \dots m+n$  can then be explicitly computed via a Lax type formula [25].
- II. Given the densities, the lengths  $q_j$  of the queues can be determined by balancing the influxes and outflows of the intersection. The queue lengths can finally be obtained as a fixed point of a contractive transformation  $q \rightarrow \Delta(q)$  where  $q$  needs to be Lipschitz continuous.
- III. The buffer model is thus well-posed at intersections for general  $\mathcal{L}^\infty$  data. It is also shown that the traffic flow model is continuous w.r.t. weak convergence.

The interested reader is referred to [9] for the detailed proofs.

Let the general setting be as earlier in this section. Further we include two realistic assumptions for the boundary values at the entrance and exit points of junctions:

- i. **Driver's preferences**  $\Theta_{i,j}$  define the fraction of cars on road  $i$  that want to exit the junction in direction of road  $j$ ,

$$\Theta_{i,j} \in [0, 1].$$

- ii. **Relative priorities**  $\eta_i$  given to incoming roads e.g. external effects like traffic lights.

For every junction  $J$  we can define the driver's preferences  $\Theta$ , i.e. the percentage of drivers going from one incoming to one outgoing road, as followed:

**Definition 2.3.1** *Given a junction  $J$  with  $n := |\delta^{in}(J)|$  incoming roads, say  $e_1, \dots, e_n$ , and  $m := |\delta^{out}(J)|$  outgoing roads, say  $e_{n+1}, \dots, e_{n+m}$ . Then the traffic distribution matrix  $\Theta$  is given by*

$$\Theta = \begin{pmatrix} \Theta_{n+1,1} & \cdots & \Theta_{n+1,n} \\ \vdots & \ddots & \vdots \\ \Theta_{n+m,1} & \cdots & \Theta_{n+m,n} \end{pmatrix}, \quad (2.3.2)$$

where  $0 \leq \Theta_{i,j} \leq 1$  for every  $i = 1, \dots, n$  and  $j = n+1, \dots, n+m$ . Furthermore, we impose the validity condition

$$\Theta_{i,j} \in [0, 1], \quad \sum_{j \in \delta^{out}(J)} \Theta_{i,j} = 1 \quad \forall i = 1, \dots, n.$$

In general,  $\Theta_{i,j} = \Theta_{i,j}(t, x)$  needs not to be necessarily constant, but can be time- and location-dependent. In the following it is assumed that the drivers' preferences are known in advance and that they do not change their itinerary throughout the network. Then the conservation law reads

$$\partial_t(\Theta_{i,j}\rho_i) + \partial_x[\rho_i\Theta_{i,j}v(\rho_i)] = 0.$$

Using product rule and reordering yields

$$\rho_i[\partial_t\Theta_{i,j} + v(\rho_i)\partial_x\Theta_{i,j}] + \Theta_{i,j}[\partial_t\rho_i + \partial_x(\rho_i v(\rho_i))] = 0.$$

The second term vanishes as the general conservation law still needs to be fulfilled. For zero density on the road this is fulfilled trivially. For  $\rho_i \neq 0$  we obtain the passive scalar transport equation along the flux:

$$\partial_t\Theta_{i,j} + v(\rho_i)\partial_x\Theta_{i,j} = 0. \quad (2.3.3)$$

A similar approach has been pursued by [10]. In their intersection model the queue capacity is arbitrarily big. Cars wanting to enter road  $j$  but exceeding the maximum outflow of the intersection are instead stored in the queue. As a consequence there is no backward propagation of queues and therefore no emergence of shocks on incoming roads.

The model by [9] extends this model. Consider a single junction and let  $M > 0$  be the maximum capacity of the queue. Then the incoming fluxes into the junction depend on the current filling level of the buffer, which is defined by

$$q = (q_j)_{j \in \mathcal{O}}, \quad q \in \mathbb{R}^n.$$

The Cauchy problem for traffic flow on a network can thus be formulated as

$$\begin{aligned}\partial_t \rho_i + \partial_x f_i(\rho_i) &= 0 \\ \partial_t(\Theta_{ij}) + v(\rho_i)\partial_x(\Theta_{i,j}) &= 0\end{aligned}$$

supplemented by suitable initial conditions.

We denote the boundary values at the junction  $J$  by

$$\left\{ \begin{array}{ll} \bar{\Theta}_{i,j}(t) := \lim_{x \rightarrow 0-} \Theta_{i,j} & e_i \in \delta^{in}(J), e_j \in \delta^{out}(J) \\ \bar{\rho}_i(t) := \lim_{x \rightarrow b_i-} \rho_i(x, t) & e_i \in \delta^{in}(J) \\ \hat{\rho}_j(t) := \lim_{x \rightarrow a_j+} \rho_j(x, t) & e_j \in \delta^{out}(J) \\ \gg f_i(t) := f_i(\bar{\rho}_i(t)) = \lim_{x \rightarrow b_i-} f_i(\rho_i(x, t)) & e_i \in \delta^{in}(J) \\ f_j^{\gg}(t) := f_j(\hat{\rho}_j(t)) = \lim_{x \rightarrow a_j+} f_j(\rho_j(x, t)) & e_j \in \delta^{out}(J). \end{array} \right. \quad (2.3.4)$$

The **maximum fluxes** that can enter and exit the intersection depend on the current filling level on the incoming and outgoing road, respectively. In particular, they are given by the following equations.

- Maximum fluxes on incoming roads:

$$\omega_i := \omega_i(\bar{\rho}) = \begin{cases} f(\bar{\rho}) & \bar{\rho} \leq \sigma \\ f(\sigma) & \bar{\rho} > \sigma. \end{cases} \quad (2.3.5a)$$

- Maximum fluxes on outgoing roads:

$$\omega_j := \omega_j(\hat{\rho}) = \begin{cases} f(\sigma) & \hat{\rho} \leq \sigma \\ f(\hat{\rho}) & \hat{\rho} > \sigma. \end{cases} \quad (2.3.5b)$$

This property ensures that Riemann problems on incoming roads are solved by waves with negative speed, and Riemann problems on outgoing roads are solved by waves with positive speed.

Based on the junctions fluxes, we can also derive the rate of change of the size of the junction buffer. In detail, conservation of mass yields the additional differential equation for the **evolution of the queue size**

$$\dot{q}_j = \sum_{i \in \mathcal{I}} \bar{\Theta}_{i,j} \gg f_i - f_j^{\gg}, \quad (2.3.6)$$

where  $\gg f_i, f_j \gg$  denote the boundary fluxes at the junction.

We are now ready to state two sets of equations regarding the incoming and outgoing junction fluxes depending on the drivers' choices  $\Theta_{ij}$  and the queue lengths  $q_j$ .

The first model provides a shared buffer of capacity  $M$  for every outgoing junction. Incoming cars can cross the intersection governed by the amount of free space left in the queue, regardless of the car's destination. Once within the junction, cars leave at the maximum rate allowed by the outgoing road of their choice.

**Single Buffer Junction model (SBJ):** Consider a constant  $M > 0$ , describing the maximum capacity of the junction at any given time, and constants  $c_i > 0, i \in \mathcal{I}$  accounting for priorities given to the different incoming roads.

We then require that the incoming fluxes  $\gg f_i$  satisfy

$$\gg f_i = \min \left\{ \omega_i, c_i(M - \sum_{j \in \mathcal{O}} q_j) \right\}, \quad i \in \mathcal{I}. \quad (2.3.7a)$$

Furthermore, the outgoing fluxes  $f_j \gg$  should satisfy

$$f_j \gg = \begin{cases} \omega_j & q_j > 0 \\ \min\{\omega_j, \sum_{i \in \mathcal{I}} \gg f_i \bar{\Theta}_{i,j}\} & q_j = 0. \end{cases} \quad (2.3.7b)$$

As seen, the outgoing fluxes are uniquely defined once the incoming fluxes are known in addition to the current status of the queue.

The second model uses  $n$  different buffers, one for every outgoing road. Once having entered the junction, cars are admitted to their desired road of destination depending on the length of the queue in front of the desired road.

**Multiple Buffer Junction model (MBJ):** Consider constants  $M_j > 0, j \in \mathcal{O}$ , describing the maximum capacities of the buffers in front of the  $n$  outgoing roads of the junction at any given time, and constants  $c_i > 0, i \in \mathcal{I}$  accounting for priorities given to the different incoming roads.

We then require that the incoming fluxes  $\gg f_i$  satisfy

$$\gg f_i = \min\{\omega_i, \frac{c_i(M_j - q_j)}{\Theta_{i,j}}, j \in \mathcal{O}\}, \quad i \in \mathcal{I}.$$

Furthermore, the outgoing fluxes  $f_j \gg$  should satisfy

$$f_j \gg = \begin{cases} \omega_j & q_j > 0 \\ \min\{\omega_j, \sum_{i \in \mathcal{I}} \gg f_i \bar{\Theta}_{i,j}\} & q_j = 0. \end{cases}$$

Now we can define the full **buffer model** by Bressan et al. [9]

$$\begin{aligned}\partial_t \rho_i + \partial_x f_i(\rho_i) &= 0 \quad , e_i \in \mathcal{E} \\ \dot{q}_j &= \sum_{i \in \mathcal{I}} \bar{\Theta}_{i,j} \gg f_i - f_j \gg \\ f(\rho_k(a_k, t)) &= f(\rho_{in,k}(t)) \quad \forall e_k \in \mathcal{E}^{in} \\ f(\rho_k(b_k, t)) &= \gg f_k(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{out} \\ f(\rho_k(a_k, t)) &= f_k \gg(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{in}\end{aligned}\tag{2.3.8}$$

with external inflows  $f(\rho_{in,k}(t))$  and junctions fluxes  $\gg f_k$  and  $f_k \gg(t)$  as defined in (2.3.7). Let  $\rho_k(x, t), e_k \in \mathcal{E}$  and  $q_j(t), e_j \in \mathcal{E} \setminus \mathcal{E}^{in}$  now be such that equations 2.3.7 and (2.3.6) are fulfilled. Then we say that the densities  $\rho_k(x, t)$  provide a solution to the Cauchy problem (2.3.8) close to the intersections.



Figure 2.2: Roundabouts as an illustration of the buffer model<sup>1</sup>.

### Remark 2

- In the following we restrict ourselves to the use of the single buffer junction model (SBJ).
- The buffer capacity  $M$  greatly influences the behavior and the quality of the model. If  $M$  is too small,  $M < 1$ , the junction hardly allows any flux through the junction. This can be interpreted in a way that every single car entering the junction fully congests it, not allowing for any other car to enter the junction until the first car has left it towards its desired destination. On the other hand having a buffer capacity close to infinite works as a sink for vehicles not able to instantaneously cross the junction (see also [10]).

<sup>1</sup>CC BY 3.0, <https://en.wikipedia.org/w/index.php?curid=29293425>, <https://www.flickr.com/photos/thienzieyung/15919564968>

- We also underline one limitations of our modelling approach:

– The buffer model in general violates the Rankine-Hugoniot condition 2.2.4:

$$\sum_{\substack{e_j \in \delta^{out}(J) \\ J \in \mathcal{N}}} f_j^{\gg} = \sum_{\substack{e_j \in \delta^{out}(J) \\ J \in \mathcal{N}}} \sum_{\substack{e_i \in \delta^{in}(J) \\ J \in \mathcal{N}}} \gg f_i \Theta_{i,j} - \sum_{\substack{e_j \in \delta^{out}(J) \\ J \in \mathcal{N}}} \dot{q}_j \quad (2.3.9)$$

$$= \sum_{\substack{e_i \in \delta^{in}(J) \\ J \in \mathcal{N}}} f_i^{\gg} - \sum_{\substack{e_j \in \delta^{out}(J) \\ J \in \mathcal{N}}} \dot{q}_j \quad (2.3.10)$$

as  $\sum_{\substack{e_j \in \delta^{out}(J) \\ J \in \mathcal{N}}} \Theta_{i,j} = 1$  for every  $i \in \delta^{in}(J), J \in \mathcal{N}$ . This does not fulfill the Rankine-Hugoniot condition as long as at least one queue  $q_j$  changes its length.

# Chapter 3

## Numerical approximations

Numerical approximations using finite difference schemes usually have difficulties near discontinuities (see Appendix C). In particular, they cannot reproduce typical phenomena like shocks as they try to approximate discontinuous solutions by smooth ones. To solve that problem we dedicate the next chapter to the introduction of the Godunov scheme, a numerical method which also preserves the underlying conservation law.

### 3.1 The Godunov scheme for nonlinear conservation laws

Finite difference schemes as discussed in the introductory part of this chapter provide *smooth* solutions, even if the initial datum of the conservation law contains discontinuities and jumps. Even to nonlinear problems these PDEs can often be linearized and therefore results from the linear FD methods can be applied in order to obtain convergence results for nonlinear problems [57]. There are several limitations to this approach:

Firstly, it is not guaranteed that the obtained numerical solution really converges towards the analytical discontinuous solution (see e.g. Burger's equation). Secondly the modelling of travelling waves, in particular the evolution of traffic jams and shocks, often emerge from initial discontinuities.

Secondly, and more applied to the modelling of traffic flow, the numerical solutions provided by finite difference schemes are built in a very different way than analytical ones. More precisely, they are not made of waves which are fundamental elements of the LWR theory. This becomes a problem once one wants to illustrate LWR theory with numerical solutions. Thus a different approach has to be pursued.

In the following section we will introduce the so-called *Godunov scheme* as originally introduced by S. K. Godunov in 1959 [27]. This method is based on the approximation of the fundamental flow-density relationship and on the explicit tracking of waves.

### 3.1.1 Theory of conservative numerical schemes

Let us consider the  $(x, t)$ -plane as the operating space. We can discretize this space by choosing a mesh with step size  $h := \Delta x$  and time step  $k := \Delta t$ ,  $\frac{h}{k} = C > 0$  fixed. In particular, the mesh points are given by

$$x_j = jh, \quad j = \dots, -1, 0, 1, \dots \quad (3.1.1)$$

$$t_n = nk, \quad n = 0, 1, \dots \quad (3.1.2)$$

We also define the midpoints in space

$$x_{j \pm \frac{1}{2}} := x_j \pm \frac{h}{2}. \quad (3.1.3)$$

Following a finite volume approach, we define the **cell averages** of the density  $\rho(x, t)$  on the mesh as

$$u_j^n := \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho(x, t_n) dx. \quad (3.1.4)$$

The idea of the **Godunov scheme** is to approximate the solution  $\rho(x, t_n)$  to conservation law (1.2.3) by a piecewise constant function

$$u^n(x, t_n) = u_j^n \quad \text{if } x_{j-\frac{1}{2}} \leq x < x_{j+\frac{1}{2}} \quad (3.1.5)$$

and by solving the Riemann problems at every cell boundary caused by the space discretization at every time step. p

Following from the integral form of conservation law (1.2.3), namely

$$\begin{aligned} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho(x, t_{n+1}) dx &= \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho(x, t_n) dx \\ &\quad - \left[ \int_{t_n}^{t_{n+1}} f(\rho(x_{j+\frac{1}{2}}, t)) dt - \int_{t_n}^{t_{n+1}} f(\rho(x_{j-\frac{1}{2}}, t)) dt \right] \end{aligned} \quad (3.1.6)$$

we obtain

$$u_j^{n+1} = u_j^n - \frac{1}{h} \left[ \int_{t_n}^{t_{n+1}} f(\rho(x_{j+\frac{1}{2}}, t)) dt - \int_{t_n}^{t_{n+1}} f(\rho(x_{j-\frac{1}{2}}, t)) dt \right] \quad (3.1.7)$$

with the **numerical flux function**

$$F(u_j^n, u_{j+1}^n) \simeq \frac{1}{k} \int_{t_n}^{t_{n+1}} f(\rho(x_{j+\frac{1}{2}}, t)) dt, \quad (3.1.8)$$

which plays the role of the average flux through  $x_{j+\frac{1}{2}}$  over the time interval  $[t_n, t_{n+1}]$ . Numerical methods of the form (3.1.7) are called *in conservative form* as they still fulfill the conservation law.

We also note that in practice equation 3.1.8 simplifies since  $u_j^n$  at point  $x_{j+\frac{1}{2}}$  is constant along the line  $[t_n, t_{n+1}]$  (see Figure 3.1). Therefore the numerical flux at point  $x_{j+\frac{1}{2}}$  only depends on the neighboring densities  $u_j^n$  and  $u_{j+1}^n$ . Denoting this value by  $u^*(u_j^n, u_{j+1}^n)$  the numerical flux reduces to

$$F(u_j^n, u_{j+1}^n) = f(u^*(u_j^n, u_{j+1}^n)). \quad (3.1.9)$$

Obviously the scheme is consistent, since  $F(u_j^n, u_j^n) = f(u_j^n)$ .

We mentioned earlier that  $u_j^n$  at  $x_{j+\frac{1}{2}}$  is constant along the line  $[t_n, t_{n+1}]$ . This only holds

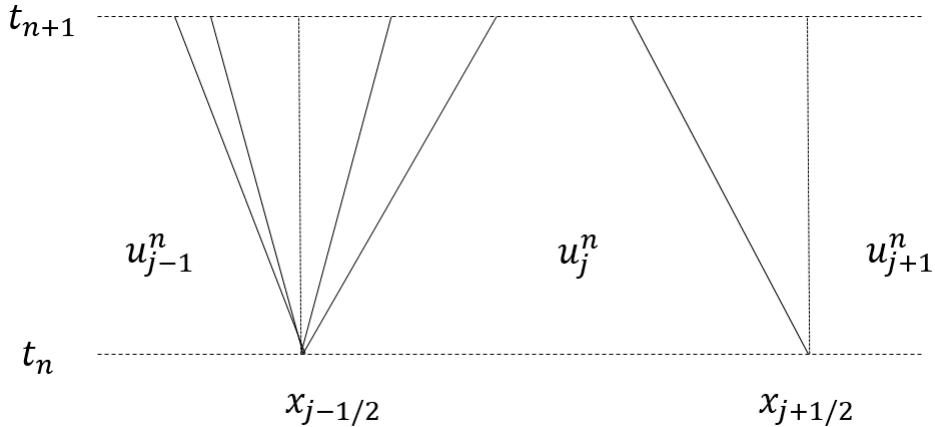


Figure 3.1: Illustration of the Godunov scheme. In this case there is a rarefaction wave through  $x_{j-\frac{1}{2}}^n$  and a shock with negative speed through  $x_{j+\frac{1}{2}}^n$ .

if the cells are "small enough" such that two shocks resulting from neighboring Riemann problems do not interact between two time steps. In general, the speed of the wave is bounded by the eigenvalues of  $Df(u)$ . Here we indicate that  $Df$  here corresponds to the Jacobian of  $f$ . Since we consider a scalar conservation law,  $Df(u)$  is scalar (in  $x$ ), thus it is enough to take  $\sup(f'(u))$ . In particular

$$C := \frac{k}{h} |\sup(f'(u))| \leq 1 \quad (3.1.10)$$

is called the Courant-number and serves as the one-dimensional CFL-condition [19].

The **Godunov method** now consists of separately solving every Riemann problem between every two neighboring cells. Let  $u_j$  and  $u_{j+1}$  be the densities on two neighboring cells. Then the flux  $f(u^*(u_j, u_{j+1}))$  is given by

$$f(u^*(u_j, u_{j+1})) = \begin{cases} \min \{f(u_j), f(u_{j+1})\} & \text{if } u_l \leq u_r \\ \max_{u_r \leq u \leq u_l} f(u) & \text{if } u_l \geq u_r. \end{cases} \quad (3.1.11)$$

This form of the resulting flux function is valid for any general scalar conservation laws, even for nonconvex fluxes. In the convex case we can distinguish 4 cases for the numerical flux (3.1.11), see Figure 3.2:

1.  $f'(u_j), f'(u_{j+1}) \geq 0$ : Here the solution consists of a rarefaction wave with  $u^*(u_l, u_r) = u_j$ .
2.  $f'(u_j), f'(u_{j+1}) \leq 0$ : The solution consists of a rarefaction wave with  $u^*(u_l, u_r) = u_{j+1}$ .
3.  $f'(u_{j+1}) < 0 \leq f'(u_j)$ : Then there is a shock through  $x_{j+\frac{1}{2}}$  and

$$u^*(u_j, u_{j+1}) = \begin{cases} u_j & : s \geq 0 \\ u_{j+1} & : s < 0 \end{cases}$$

with the shock speed

$$s = \frac{f(u_{j+1}) - f(u_j)}{u_{j+1} - u_j}.$$

4.  $f'(u_{j+1}) > 0 \geq f'(u_j)$ : In this case we have the so-called transonic rarefaction wave and  $u^*(u_j, u_{j+1}) = \sigma$ , with  $u = \sigma$  the unique point at which  $f'(u) = 0$  (cf. (1.2.2)).

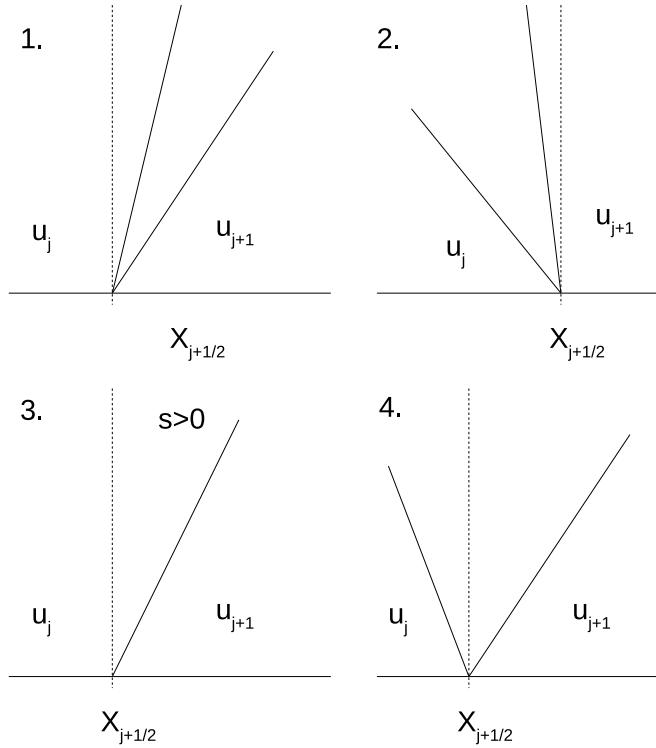


Figure 3.2: Different types of solutions for the Riemann problem dependent on densities  $u_j^n$  and  $u_{j+1}^n$ .

In the following we want to apply the Godunov scheme on the buffer model defined in section 2.3.

### 3.1.2 The Godunov scheme for the buffer model

Let us consider an arbitrary connected road network  $(\mathcal{N}, \mathcal{E})$  with  $n_R$  roads and  $n_J$  junctions. On every road  $e_i = [a_i, b_i]$  we apply the discretization

$$\begin{aligned} x_{i,j} &= a_i + jh, & j = 0, \dots, D_i, \\ x_{i,D_i h} &= b_i, \end{aligned}$$

with  $D_i = \frac{b_i - a_i}{h} - 1$  the number of cells on road  $e_i$ . Let us then denote  $u_{i,j}^n$  the density on the  $j$ -th cell of road  $e_i$  at time  $t = nk$ .

We can now define on all *inner cells* of every road  $e_i$  the Godunov scheme

$$u_{i,j}^{n+1} = u_{i,j}^n - \frac{k}{h} [f(u^*(u_{i,j}^n, u_{i,j+1}^n)) - f(u^*(u_{i,j-1}^n, u_{i,j}^n))], \quad e_i \in \mathcal{E}, j = 1, \dots, D_i - 1. \quad (3.1.12a)$$

We incorporate the *externally inflowing densities*  $q_i(t), e_i \in \mathcal{E}^{in}$  by introducing a ghost cell at the beginning of each road  $e_i$ . Then we define

$$u_{i,0}^{n+1} = u_{i,0}^n - \frac{k}{h} [f(u^*(u_{i,0}^n, u_{i,1}^n)) - f(u^*(v_{i,1}^n, u_{i,0}^n))], \quad i \in \mathcal{E}^{in} \quad (3.1.12b)$$

with the inflowing density  $v_{i,1}^n := \int_{t_n}^{t_{n+1}} \rho_{in,i}(t) dt$ .

The outfluxes leaving the network can be treated analogously.

For junctions we recall equations (2.3.7) for the junction fluxes in the single buffer model. Accordingly, we define for roads coming into junction  $J$

$$u_{i,D_i}^{n+1} = u_{i,D_i}^n - \frac{k}{h} [\gg f_i - f(u^*(u_{i,D_i-1}^n, u_{i,D_i}^n))], \quad i \in \delta^{in}(J). \quad (3.1.12c)$$

For outgoing roads of junction  $J$  we analogously define the scheme

$$u_{i,0}^{n+1} = u_{i,0}^n - \frac{k}{h} [f(u^*(u_{i,0}^n, u_{i,1}^n)) - f_i^\gg], \quad i \in \delta^{out}(J). \quad (3.1.12d)$$

#### Remark 3

- For the computation of the buffer, which is needed for the junction fluxes, we use a simple difference quotient to approximate  $\dot{q}$ ,

$$q_j^{n+1} = q_j^n + k \left[ \sum_{i \in \delta^{in}(J)} \bar{\Theta}_{i,j} \gg f_i - f_j^\gg \right], \quad J \in \mathcal{N}, j \in \delta^{out}(J) \quad (3.1.13)$$

- For convenience we condense equations (3.1.14)a-d on road  $e_i$  to the compact formulation for the Godunov scheme

$$u_i^{n+1} = G(u_i^n). \quad (3.1.14)$$

## 3.2 Numerical example

Let us consider a single junction with two incoming and one outgoing road set up as in Table 3.1. Also let the buffer have maximum capacity  $M = 1$  and let both incoming roads

road	length	initial density $\rho_0$	inflow $f(\rho_{in,i}(t))$ on $[0, 200]$
$e_0$	50	0.	$f(0.4)$
$e_1$	50	0.	$f(0.2)$
$e_2$	50	0.	-

Table 3.1: Setup for the network of example a).

have the same priority values  $c_0 = c_1 = 1$ , meaning that cars coming from both roads are equally allowed to pass the junction. It also means that they have to wait equally once the intersection is congested and buffer level rises.

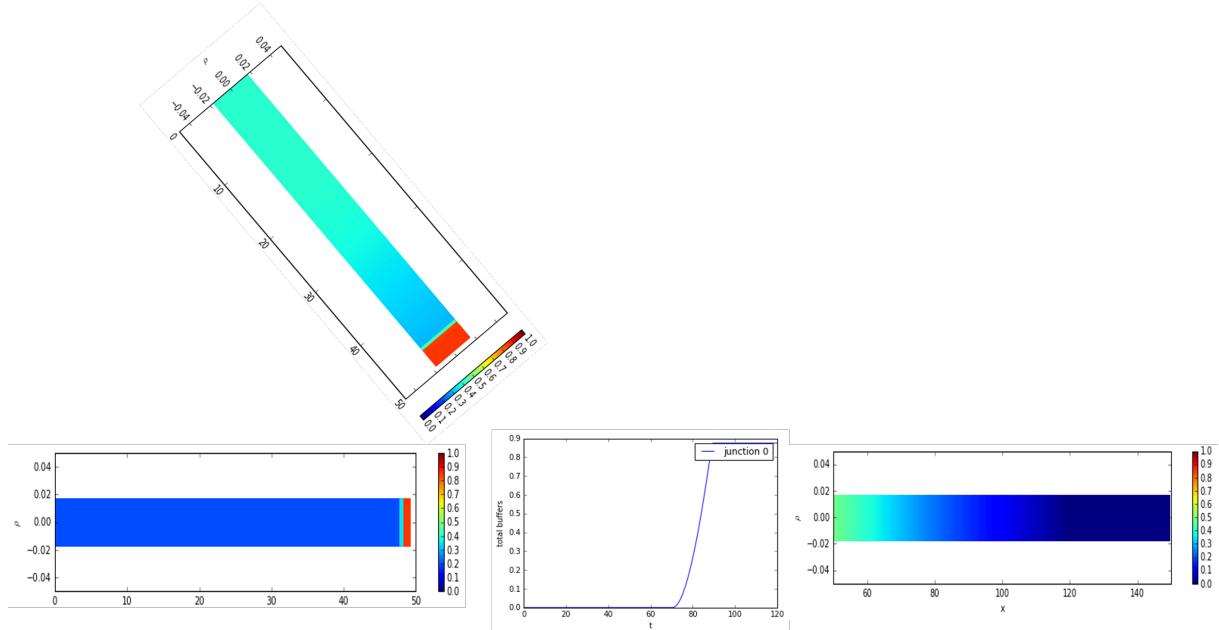


Figure 3.3: Resulting traffic densities after  $T = 120s$  and the evolution of the filling rate of the junction buffer (in the middle). The corresponding video *buffer\_model\_barplot.mp4* can be found on GitHub at [https://github.com/mstachl/Traffic\\_flow\\_optimization/Experiments/Buffer\\_model](https://github.com/mstachl/Traffic_flow_optimization/Experiments/Buffer_model).

**Comments on the video**

- As the network is initially empty it takes some time until the cars from the incoming roads  $e_0$  and  $e_1$  reach the junction.
- Once cars from both incoming roads reach the end of their roads, they pass the intersection. More and more cars try to cross the junction until the incoming fluxes (*supply*) is greater than the maximum road capacity of the outgoing road (in this example  $f(0.2) + f(0.4) = 0.16 + 0.24 = 0.4 > 0.25 = f(0.5)$ ).
- From this moment on the junction buffer starts to build up, causing the amount of cars crossing the junction to decrease and forcing cars trying to enter the junction to wait (red area).
- Subsequently, traffic jams emerge. We also observe the resulting shocks travelling to the left end of the incoming roads.

# Chapter 4

## Flux optimization via traffic light control

As discussed in section 2.3 we can model external effects, especially traffic lights, by assigning relative priorities

$$\eta_i \in \{0, 1\}, \quad e_i \in \mathcal{E} \setminus \mathcal{E}^{out}$$

to roads prior to intersections. These values have the effect that only a fraction of the flux  $f_i^{\gg}$  as defined in the buffer model can enter the junction and pass it towards outgoing roads. In particular, the binary values  $\eta_i = 1$  and  $\eta_i = 0$  for road  $e_i$  correspond to green and red phases, respectively, for the particular incoming road.

For every junction  $J$  the **feasibility condition** must hold. It is given by

$$\sum_{e_i \in \delta^{in}(J)} \eta_i(t) = 1, \quad (4.0.1)$$

assuring that cars from only one road can cross the junction at any time  $t$ .

This leads to a necessary update of the junction fluxes. In detail we define the **influx under applied traffic control** as

$$\gg f_k(t, \eta_k) := \min\{\omega_k, c_k(M - \sum_{j \in \mathcal{O}} q_j)\} \cdot \eta_k(t). \quad (4.0.2)$$

The corresponding **outflux under applied traffic control** is given by

$$f_j^{\gg}(t, \eta) := \begin{cases} \omega_j & q_j > 0 \\ \min\{\omega_j, \sum_{i \in \mathcal{I}} \gg f_i \bar{\Theta}_{ij}\} & q_j = 0. \end{cases} \quad (4.0.3)$$

For the conservation law

$$(\rho_k)_t + f(\rho_k)_x = 0 \quad \forall e_k \in \mathcal{E} \quad (4.0.4a)$$

$$f(\rho_k(b_k, t)) = \gg f_k(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{out} \quad (4.0.4b)$$

$$f(\rho_k(a_k, t)) = f_k^{\gg}(t) \quad \forall e_k \in \mathcal{E} \setminus \mathcal{E}^{in} \quad (4.0.4c)$$

with junction fluxes  $\gg f_k$  and  $f_k \gg$  on a given network  $(\mathcal{N}, \mathcal{E})$  the goal is to optimize the flow on every single road and through their intersections. The considered objective function - or also referred to as *cost functional* - can hereby vary depending on the goals of the simulation and the specific definition of the targeted problem. Typical **objective functions** for the optimization are introduced in the following.

### a) Mean travel time

From driver's point of view, the key quantity to determine the quality - and therefore the optimal value - of traffic is related to the time needed to reach the desired location. Taking into account the sum of every personal preference hence leads to the definition of the mean arrival time or **mean travel time** (cf. [17]).

Let  $x = 0$  be a point on the network. Then the mean travel time needed to reach point  $x = \bar{x} > 0$  can be described by

$$T(\bar{x}) := \frac{1}{Q_{in}} \int_{t_0}^{\infty} t f(\rho(\bar{x}, t)) dt,$$

where  $Q_{in} = \int_{t_0}^{t_{end}} q_0(t) dt$  is the accumulated influx  $q_0(t)$  into point  $x = 0$  on a compact time interval  $[t_0, t_{end}]$ .

#### Remark 4

- This approach expects compactly supported inflow into  $x = 0$ .
- On complex networks drivers can have distinct preferences of their respective arrival points and favoured routes. This means that cars, despite also crossing the point  $x = 0$ , might never reach the reference point  $x = \bar{x}$ , which complicates the computation of the average travel time between two points on a network.

### b) Cumulative traffic flux

From the traffic planner's perspective a more relevant quantity might be the overall flux on the entire network. Therefore the desired goal is to maximize the total number of cars travelling through the network over a certain time interval.

Following equation 4.0.4 we denote by  $\gg f$  and  $f \gg$  incoming fluxes into and outgoing fluxes out of junctions, respectively. Then we can formulate the **cumulated traffic flux** on the network during time  $t = [t_0, T]$  as

$$\begin{aligned} F_T(\eta) := & \sum_{i \in \mathcal{E}} \int_0^T \int_{a_i}^{b_i} f(\rho_i(x, t)) dx dt + \sum_{i \in \mathcal{E} \setminus \mathcal{E}^{out}} \int_0^T \gg f_i(t, \eta_i) dt \\ & + \sum_{j \in \mathcal{E} \setminus \mathcal{E}^{in}} \int_0^T f_j \gg(t, \eta) dt, \end{aligned} \quad (4.0.5)$$

where  $\gg f$  and  $f \gg$  are as defined as in equations 4.0.2 and 4.0.3 in the buffer model. (For the general definition of the cumulated flux on networks, in particular a functional to measure outgoing fluxes of junctions, see for instance [33]).

### c) Cumulated density on the network

From a more mathematical perspective minimizing travel time corresponds to the fastest delivery of a compact package of density from the start of a road to the end. Let  $G(t) : [0, \infty) \rightarrow [0, M_{total}]$  denote the cumulated mass on the network at time  $t$  with  $M_{total}$  the total mass of the package.

The cumulated density satisfies  $G(0) = G(T_G) = 0$  for some  $T_G > 0$  and  $\int_0^{T_G} G(t)dt = M_{total}$  (see Figure 4.1) with  $T_G$  denoting the total time needed for the mass  $M_{total}$  to pass

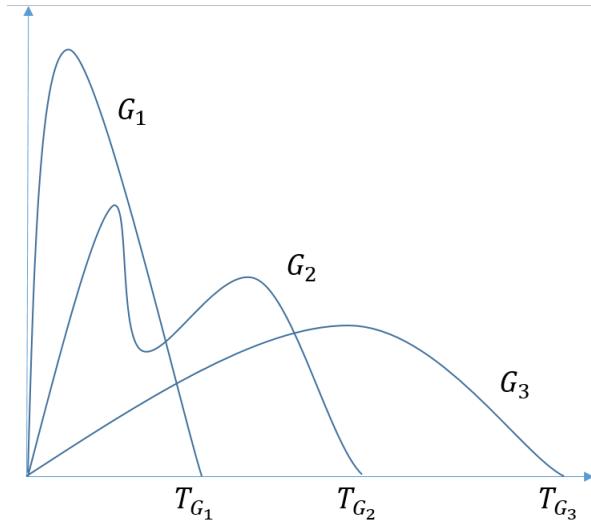


Figure 4.1: Different temporal distributions of the total mass  $M_{total}$ .

the network. Finding the optimal function  $G$  would then lead to an optimal solution with respect to applied controls.

One approach to find the optimal function  $G(t)$  could be to minimize the maximum value of the curvature, namely

$$\min_{controls} \left[ \max_t \frac{d^2}{dt^2} G(t) \right],$$

where the second derivative can also be written as

$$\frac{d^2}{dt^2} G(t) = \int_{network} \frac{d}{dt} \frac{d}{dx} f(\rho(x, t)) dx.$$

#### Remark 5

During the following optimization studies the cumulative traffic flux is the cost functional of choice.

Over the course of the remaining chapter different optimization approaches are used. Fixed strategies assume a fixed pattern for the switching of traffic lights. For a given pattern it aims to generate green waves over a sequence of traffic lights by delaying the given signal for every proceeding junction. Model predictive control on the other hand takes a given initial traffic distribution and tries to find an optimal traffic light setting forecasting future traffic behavior on the network using the underlying dynamics of the system. The flux functional serves as a measure for comparison of these two distinct approaches.

## 4.1 Green wave strategy: Traffic light coordination

Optimally tuned traffic lights provide a setting where drivers encounter a green wave, in particular a sequence of consecutive green lights. The distinction between synchronized and coordinated traffic lights is important. Synchronized traffic signals all switch at the same time and are hardly used in practice. On the other hand coordinated signals are controlled by a master controller. They are set up such that they progress (switch) in sequence in order to generate a green wave for crossing vehicles.

### 4.1.1 The model

Consider a sequence of two intersections with two incoming and one outgoing road (cf. figure 4.2) with inflows  $q_k(t)$  for  $e_k \in \mathcal{E}^{in}$ .

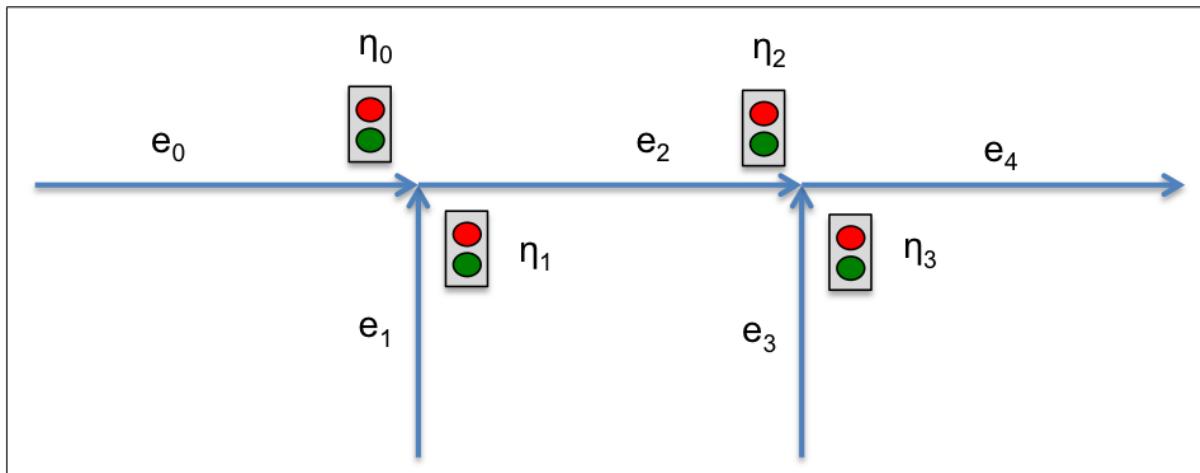


Figure 4.2: Example network consisting of two junctions and four controls.

Then we can refine the conservation law of 1.2.4 to

$$\partial_t \rho_k + \partial_x f(\rho_k) = 0 \quad \forall e_k, k = 0, \dots, 4 \quad (4.1.1a)$$

$$f(\rho_k(b_k, t)) = \gg f_k(t, \eta_k(t)) \quad \forall e_k, k = 0, 1, 2, 3 \quad (4.1.1b)$$

$$f(\rho_k(a_k, t)) = f_k^{\gg}(t, \eta(t)) \quad \forall e_k, k = 2, 4 \quad (4.1.1c)$$

accounting the restrictions on the fluxes induced by the traffic configuration  $\eta$ , where  $\eta(t) = (\eta_0(t), \dots, \eta_3(t)) \in [0, 1]^4$  is the vector containing all control values.

Impose now that the two traffic lights  $\eta_0, \eta_2$  have the same fixed frequency of red/green light (also called their lifetime)- say one time unit -, only set apart by a delay  $\tau$ , and recall the feasibility condition 4.0.1. Then the controls satisfy

$$\begin{aligned} \eta_0(t) &= \chi_{[0,1] \cap [2,3] \cap \dots} =: \eta_C(t) \\ \eta_1(t) &= 1 - \eta_C(t) \\ \eta_2(t) &= \eta_C(t - \tau) \\ \eta_3(t) &= 1 - \eta_C(t - \tau). \end{aligned}$$

We fix  $\eta_C(t)$ . The goal now is to find the optimal delay  $\tau$  in order to obtain the best value for  $F_T$ .

In particular, the optimization problem can be formulated as

$$\begin{aligned} F_T^* := \max_{\tau \in \mathbb{R}_0^+} & \sum_{i \in \mathcal{E}} \int_0^T \int_{a_i}^{b_i} f(\rho_i(x, t)) dx dt + \sum_{i \in \mathcal{E} \setminus \mathcal{E}^{out}} \int_0^T \gg f_i(t, \eta_i(t)) dt \\ & + \sum_{j \in \mathcal{E} \setminus \mathcal{E}^{in}} \int_0^T f_j^{\gg}(t, \eta(t)) dt \end{aligned} \quad (4.1.2)$$

and the optimal delay is given by

$$\begin{aligned} \tau^* := & \arg \max_{\tau \in \mathbb{R}} \sum_{i \in \mathcal{E}} \int_0^T \int_{a_i}^{b_i} f(\rho_i(x, t)) dx dt + \sum_{i \in \mathcal{E} \setminus \mathcal{E}^{out}} \int_0^T \gg f_i(t, \eta_i(t)) dt \\ & + \sum_{j \in \mathcal{E} \setminus \mathcal{E}^{in}} \int_0^T f_j^{\gg}(t, \eta) dt. \end{aligned} \quad (4.1.3)$$

#### 4.1.2 Experiments on coordinated light signals

##### Experiment a)

In the first study we want to examine the effect of the duration of the green and red phases on the total flux on the network. Here we consider a single junction with two incoming roads and one outgoing road, hence the traffic distribution matrix is given by

$$\Theta = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}.$$

Every road is filled initially with traffic density  $\rho_i(x, 0) = 0.2, i = 0, 1, 2$ . On the incoming roads  $e_0, e_1$  traffic of density  $\rho_{in,0} = 0.4, \rho_{in,1} = 0.2$  is constantly inflowing. Cars at the end of road  $e_2$  just exit the network and vanish. The traffic lights  $\eta_0, \eta_1$  at the end of the

road	length	initial density $\rho_0$	inflow $f(\rho_{in,i}(t))$	on [0, 200]
$e_0$	50	0.2	$f(0.4)$	
$e_1$	50	0.2	$f(0.2)$	
$e_2$	50	0.2	-	

Table 4.1: Setup for the network of example a).

incoming roads switch during time. The period, consisting of one green and red phase cycle, is fixed at  $T_s = 50s$ . The goal of this experiment is to find the optimal distribution of green times for both light signals based on the densities on the incoming roads. Using

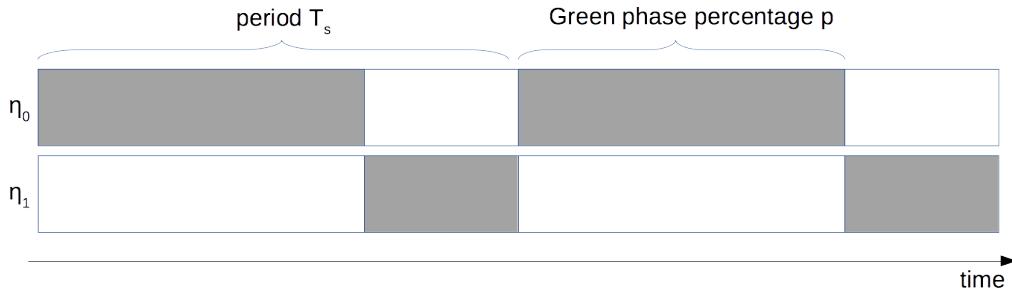


Figure 4.3: Switching traffic lights over time.

compactly supported inflow on both incoming roads in  $[0s, 200s]$  we compute the total flux  $F_T$  on the network up to  $T = 500s$  for varying green time percentages  $p_0$  (Note, that  $p_1 = 1 - p_0$  as it is a valid configuration). The optimal flux is eventually attained if  $\eta_0 = 1$  (meaning keeping its green phase) during 56% of its period (cf. Figure 4.4). We then obtain an optimal flux of  $F_T \sim 39382.84$ .

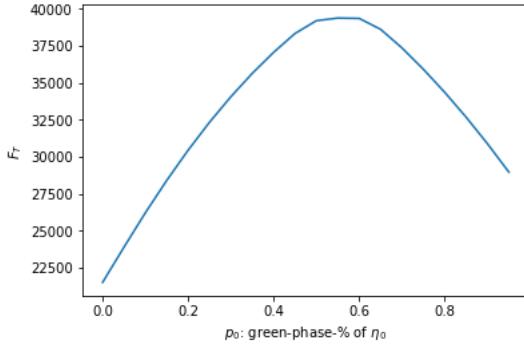


Figure 4.4: Total fluxes on the network dependent on the green time distribution  $p_0$ . The optimal flux is obtained if the green phase lasts  $0.56 T_s$ .

### Experiment b)

Consider the same network as provided in Figure 4.2, hence its traffic distribution matrix is given by

$$\Theta = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}.$$

Also we consider the initial setup given in Table 4.2.

road	length	initial density $\rho_0$	inflow $f(\rho_{in,i}(t))$	on $[0, 200]$
$e_0$	50	0.2	$f(0.4)$	
$e_1$	50	0.2	$f(0.2)$	
$e_2$	100	0.2		-
$e_3$	50	0.2	$f(0.2)$	
$e_4$	50	0.2		-

Table 4.2: Setup for the network of example b).

Furthermore, let the external inflows be constant over a compact interval  $[0s, 200s]$  and let the fixed green/red phase duration be set to 60s. Then the evaluation of our cost functional with respect to the delay  $\tau \in [0, 120]$  up to final time  $T = 500$ , according to equation 4.1.2, shows that an optimal solution is obtained once the traffic light of the second junction used the switching pattern of the first junction delayed by the optimal delay of  $\tau^* = 34s$ . This results in the optimal cumulated traffic flux  $F_T^* \sim 82851.38$ .

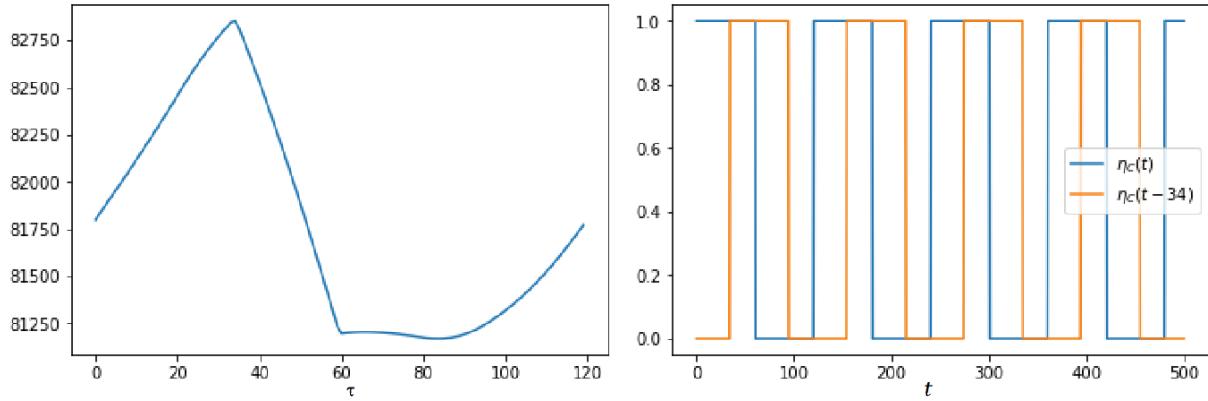


Figure 4.5: Left: Dependency of the cumulated traffic flux  $F_T$  on the value for the delay. In experiment b) the objective function attains its maximum  $F_T^* = 82851.38$  at  $\tau^* \sim 34$ . Right: Plotted controls with applied optimal delay  $\tau^*$ .

## 4.2 Optimization via Model Predictive Control

In contrary to chapter 4.1 the optimal master controller should not work on fixed green and red light periods but be able to adjust traffic lights based on the current - and ideally even future - traffic situation in front of the traffic lights. Based on the possible fluxes the master controller would assign traffic light configurations such that the total flow on the network is maximized over the full time horizon. While trying optimization over the full time horizon we run into two problems:

- In real networks complete knowledge of the future volume of traffic over the whole time horizon is usually not given (at best educated guesses about the volume of traffic and the occupation of the network can be made at certain times throughout the day).
- The computational effort needed to find the optimal solution might be too high to solve the optimization problem in reasonable time.

This is where **model predictive control** comes into play.

Model Predictive Control (MPC) is a method to control complex dynamic processes and has a wide range of applications. MPC algorithms use a model of the underlying system under consideration in order to find optimal control signals, taking the future behaviour of the system into account. This type of methods are suitable to control systems in which prediction is a key aspect.

The main advantage of MPC over non-predictive control is, that MPC methods inherently make a trade-off between immediate performance and future outputs [31, 14]. Model predictive control has been an area of high academic interest for the past decades, offering a wide range of applications, covering processes in industrial production as well as the

control of robots [24, 37] and clinical anaesthesia [50, 14].

MPC uses iterative optimization of an objective function based on the solution of a given model on predictions on a reduced time horizon.

The finite time interval used for MPC optimization is called **predictive horizon** denoted by  $n_p$ . The resulting optimal control sequence obtained after one optimization step serves as a feedback map and is applied as input for the light signals up to a **control horizon** denoted by  $n_c$ , when the optimization step repeats on the new time interval (Note that the condition  $n_c \leq n_p$  needs to be fulfilled). As the predictive horizon keeps being shifted forward MPC is also referred to as *receding horizon control*.

The idea of model predictive control is to utilize the known dynamics of a system in order to predict and optimize the future behavior of the system. In particular we use a discrete dynamical model of the form

$$x^{n+1} = Evo(x^n, \eta^n),$$

where  $Evo : X \times U \rightarrow X$  is a known and in general nonlinear map describing the evolution of the system and  $x^{n+1}$  denotes the successor of  $x^n$  under application of control  $\eta^n$  at time step  $t_n$ . The dynamics are utilized to optimize the system on a set  $U$  of valid controls with respect to a reasonable target function  $F$ , i.e.

$$\begin{aligned} & \min_{\eta \in U} F(x, \eta) \\ \text{s.t.} \quad & x^{n+1} = Evo(x^n, \eta^n) \end{aligned}$$

In the following we provide a detailed description of the MPC method [32]:

1. **Prediction.** The current state  $x(t_n)$  of the system, expected external influences and a planned control signal are used to predict the behavior of the considered system in the predictive horizon  $[t_n, t_n + n_p \Delta t]$ . For traffic flow, this involves the evaluation of a model to predict the future road conditions.
2. **Performance evaluation.** An objective function is used to evaluate the performance of the system under the planned control signal from the prediction phase. Typical functionals consist of the average travel time or the cumulated traffic flux on the network.
3. **Optimization.** In an optimization step, the optimal control signal is found over a finite time horizon. This control signal optimizes the objective function for the chosen time horizon of the prediction phase. Typical methods for this step use gradient-descent methods, LP solvers or least-squared methods.
4. **Model dynamics.** Given the optimal control signal from the previous step, the next control action is taken from the optimal control signal and subsequently applied

to the system on the control horizon  $[t_n, t_n + n_C \Delta t]$ . Recalculating optimal control signals proceeds using a receding horizon scheme.

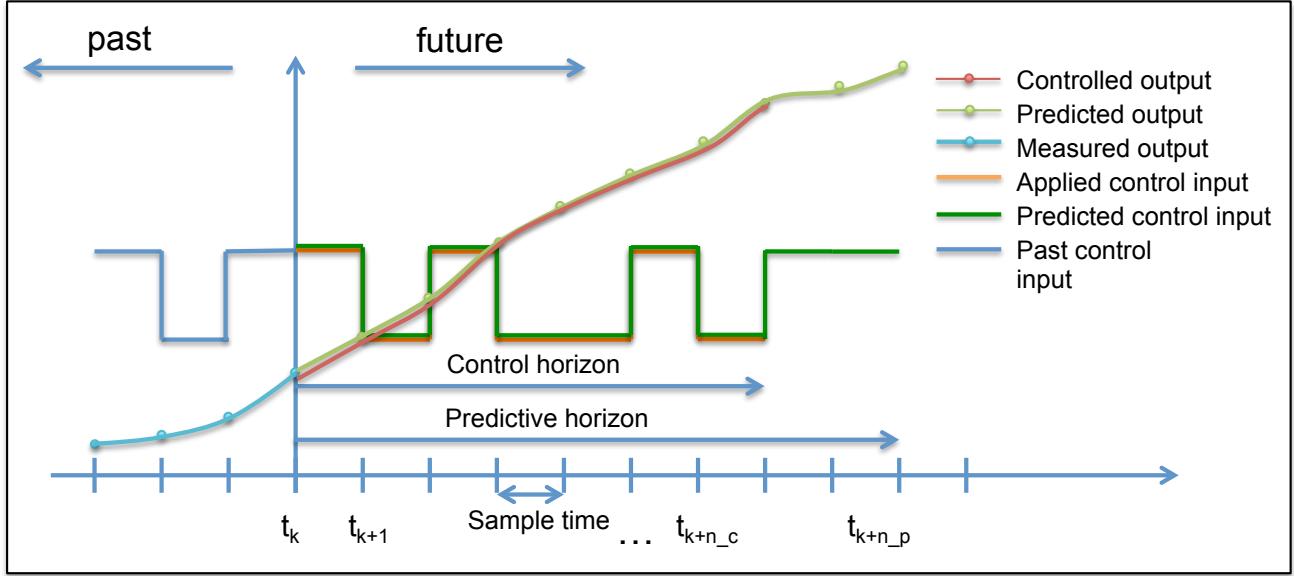


Figure 4.6: Illustration of an MPC step at time  $t_n$ .

Stability of solutions obtained by MPC, in particular with highly non-linear dynamics, in general cannot be guaranteed. Nevertheless, recent findings have shown that under certain conditions stability can be achieved using the target functional as Lyapunov function [49].

### Remark 6

In order to speed up the optimization we use the strategy of *relaxation*. Instead of finding binary controls ( $\eta \in \{0, 1\}$ ) and solving the integer programming problem, we remove the integrality constraint (relax the optimization problem) and solve the problem for such constraints considering the relaxation induced by solving the optimization problem for

$$\eta_i \in [0, 1], \quad e_i \in \mathcal{E} \setminus \mathcal{E}^{out}.$$

In the following, the theory on MPC for the previously defined buffer model will be provided, supplemented by numerical examples. Also a comparison with the results from the delay modelling in 4.1 will be given.

### 4.2.1 Problem formulation

In the following, we consider the **discrete optimal control problem** (OCP) over the predictive horizon  $n_P$  starting at  $t = t_n$ ,

$$F_{n_C}(t_n, u^*) - R(t_n, u^*) := \max_{\eta_i^j \in [0, 1]} F_{MPC}(\rho_i^n, t_n, \eta_i^n, \dots, \eta_i^{n+n_P-1}) \quad e_i \in \mathcal{E}, \quad (4.2.1a)$$

subject to initial conditions  $\rho_i^n$  at time  $t = t_n$  and the system dynamics given by the Godunov scheme (3.1.14) under application of control  $\eta^n$  at time step  $t = t_n$ ,

$$\rho^{n+1} = G(\rho^n, \eta_n). \quad (4.2.1b)$$

We also impose the feasibility condition (4.0.1) for every junction  $J$  at every time step  $t_n$ ,

$$\sum_{e_i \in \delta^i(J)} \eta_i^n \leq 1. \quad (4.2.1c)$$

Furthermore, let the discrete flux functional  $F_{MPC}$  over the time interval  $[t_n, t_{n+n_P}]$  be given by

$$\begin{aligned} F_{MPC}(\rho^n, t_n, \eta_n, \dots, \eta_{n+n_P-1}) &:= \sum_{j=n}^{n+n_P-1} \Delta t \left[ \sum_{i \in \mathcal{E}} \sum_{x=a_i}^{b_i} f(\rho_i(x, t_j)) \Delta x + \sum_{i \in \mathcal{E} \setminus \mathcal{E}^{out}} \gg f_i(t_j, \eta_i^j) \right. \\ &\quad + \sum_{j \in \mathcal{E} \setminus \mathcal{E}^{in}} f_j^{\gg}(t, \eta^j) - \epsilon \sum_{J \in \mathcal{N}} W_J(t, \eta_i) \\ &\quad \left. - \gamma \sum_{J \in \mathcal{N}} \sum_{i \in \delta^{in}(J)} (\eta_i(t_j) - \eta_i(t_{j-1}))^2 \right] \\ &= F_{n_C}(\rho_n, t_n, \eta) - R(t_n, \eta) \end{aligned}$$

with  $\epsilon, \gamma > 0$  and where

- $F_{n_C}(t_n, \eta)$  is the target functional referring to the cumulated flux travelled through the network during  $t \in [t_n, t_n + n_C \Delta t]$  and
- $R$  denotes the regularization terms
  - $W_J(t) := \sum_{i \in \delta^{in}(J)} \|\prod_k (\eta_i(t) - K_J^k)\|^2$  is a **multi-well function**. It is defined for every junction  $J \in \mathcal{N}$  and it works as a de-relaxation term meaning that the resulting controls  $\eta_i$  stay close to the binary values 0 and 1
  - and the term  $\sum_{j=n}^{n+n_P-1} (\eta_i(t_j) - \eta_i(t_{j-1}))^2$  which limits the total variation of  $\eta$ , meaning that

$$\gamma \int_{t_n}^{t_n + n_P \Delta t} \|\dot{\eta}\|^2 dt \sim \gamma \sum_{J \in \mathcal{N}} \sum_{i \in \delta^{in}(J)} (\eta_i(t_j) - \eta_i(t_{j-1}))^2$$

resembles the frequency with which traffic lights can switch.

The regularization term  $R$  is a generalization of the Modica-Mortola-functional which was first introduced in [45].

- and  $K_J \in \{0, 1\}^{|\delta^{in}(J)|}$  denotes a feasible discrete configuration for all traffic lights contained in junction  $J$  such that the feasibility condition 4.0.1 is met. Assuming that at any time only cars from one incoming road can pass the junction, we have that  $K_J^i \in \{0, 1\}$ , where  $K_J^i$  denotes the individual configuration of the  $j$ -th traffic light, and

$$\sum_i K_J^i = 1.$$

We then define  $\Omega_J := \{K_1(J), \dots, K_n(J)\}$  as the set of all **feasible discrete traffic light configurations**.

**Example 2** For a single junction with 3 incoming and a single outgoing road  $\Omega$  would be defined in a straight forward way as

$$\Omega := \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\},$$

supposing that only one signal can be green at every instant.

Although the penalization term  $R$  tries to shift the resulting controls towards binary states, they in general are not. This causes unfeasibility as in reality a traffic signal can either be green (=1) or red (=0). To get rid of possible non-binary controls, we perform a **quantization** step. Using the optimal sequence of controls  $u^*|_{n_C}(t_n)$  obtained from the OCP starting at time  $t_n$  we set for every control  $\eta \in u^*|_{n_C}(t_n)$

$$\eta_{bin} := Q(\eta) = \begin{cases} 1 & \eta \geq 0.5 \\ 0 & \eta < 0.5 \end{cases} \quad (4.2.2)$$

to obtain the seqence of binary controls  $u_{bin}^*|_{n_C}(t_n)$ . This sequence is then used as an input for the Godunov scheme on the MPC time interval to perform the resulting dynamics on the network.

These dynamics determine the resulting densities  $\rho_i(x, t)$  for  $t = t_{n+1}, \dots, t_{n+n_C}$  on the network (see Figure 4.7 ). The optimal flux on the interval  $[t_n, t_{n+n_C}]$  is then given by

$$F_{n_C}(t_n, u_{bin}^*|_{n_C}(t_n)).$$

This concludes the process of performing one single MPC step.

To find a global solution, the predictive horizon is then shifted to the new starting point  $t = t_{n+n_C}$  from which the computation of the new OCP again is performed. This process

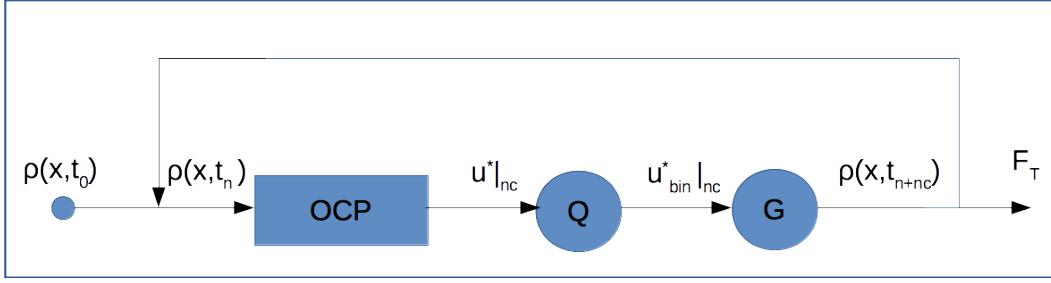


Figure 4.7: Closed loop diagram of the MPC algorithm.

is repeated until the final time  $T$  is reached and the total time horizon is covered (cf. Algorithm 1).

The **total flux** on the network over the entire time horizon  $[t_0, T]$  can finally be computed by summing up the local fluxes, i.e.

$$F_{MPC,T}(u_{bin}^*(t_0)) = \sum_{n=0}^{n_{MPC}} F_{n_C}(t_{n \cdot n_C}, u_{bin}^*|_{n_C}(t_n)), \quad (4.2.3)$$

where  $n_{MPC}$  denotes the number of MPC-steps computed.

A similar variation of algorithm 1 can be found in [28]. The solution of this algorithm converges towards a local optimum which is suboptimal with respect to the global optimum.

Suboptimality in this context means the following:

Let us denote the global optimum up to finite time  $T$  starting from time point  $t = t_n$  by

$$V_\infty(\rho_n, t_n, u^*(t_n)) := \max F_T(\rho_n, t_n, \eta) \quad (4.2.5)$$

with  $u^*(t_n) \in [0, 1]^{n_T - n}$  and  $n_T$  the total time horizon. Then we can define the Dynamic Programming Principle.

**Definition 4.2.1** Let  $F_{MPC}(\rho_0, t_0, \eta_{n_C})$  be defined as the OCP in 4.2.1. Then the dynamic programming principle states that

$$V_\infty(\rho_0, t_0, u^*(t_0)) = \max_{\eta} \{ F_{MPC}(\rho_0, t_0, \eta|_{n_C}) + V_\infty(\rho_{n_C}, t_{n_C}, \eta|_{[n_C+1, n_T]}) \}. \quad (4.2.6)$$

Due to the use of the quantization function  $Q(\eta)$  we obtain a suboptimal solution with the property

$$\begin{aligned} \max_{\eta} \{ F_{MPC}(\rho_0, t_0, \eta|_{n_C}) + V_\infty(\rho_{n_C}, t_{n_C}, \eta|_{[n_C+1, n_T]}) \} &\geq \\ F_{MPC}(\rho_0, t_0, Q(u^*|_{n_C})) + \max_{\eta} V_\infty(\rho_{n_C}, t_{n_C}, \eta|_{[n_C+1, n_T]}). \end{aligned} \quad (4.2.7)$$

**Algorithm 1** (*MPC*)

For every sampling time  $t_n, n = 0, n_c, 2n_c \dots$  do the following steps 1.-4.:

1. Measure the densities  $\rho_i(x, t_n)$  on every road  $e_i$  of the network
2. Solve the OCP 4.2.1 and denote the resulting optimal control sequence for TL  $i$  by  $\eta_i^* := \{\eta_i(t_n), \eta_i(t_{n+1}), \dots, \eta_i(t_{n+n_p})\}$ .
3. Apply the quantization function

$$Q(\eta) := \begin{cases} 1 & \eta \geq 0.5 \\ 0 & \eta < 0.5 \end{cases} \quad (4.2.4)$$

to the optimal control sequence  $u_i^*|_{n_c} := \{\eta_i(t_n), \eta_i(t_{n+1}), \dots, \eta_i(t_{n+n_c})\}$  in order to obtain a (sub-)optimal binary control sequence  $u_{i,bin}^*|_{n_c}$ .

4. Take  $u_{bin}^*|_{n_c} := \{u_{0,bin}^*|_{n_c}, u_{1,bin}^*|_{n_c}, \dots, u_{n_J,bin}^*|_{n_c}\}$  as an input for the Godunov scheme to obtain densities  $\rho_i(x, t_{n+n_c})$  and to compute the (sub-)optimal flux  $F_{n_C}(t_n, u_{bin}^*|_{n_c})$ . Proceed with 1.
5. Compute the total flux  $F_T(u_{bin}^*)$  based on equation 4.2.3.

Figure 4.8: The MPC algorithm.

How the quantization affects the optimality is also shown in Table 4.3. In any case increasing the predictive horizon in both cases - with and without quantization - does not worsen the final solution.

**Remark 7**

- The MPC algorithm produces suboptimal solutions per se. This suboptimality is further extended by quantizing the results obtained by the relaxed MPC, see Figure 4.3. By stating this we accept that the obtained results can be highly suboptimal and that by no means we are converging to the global optimum.
- Furthermore, the functional  $F_T$  in general is not convex. We refer to methods like Douglas-Rachford splitting [22] or the concave-convex procedure (CCCP) [62] to improve the optimization results. Stochastic methods like multi-start [5] can also help to find local optima that are closer to the global optimum.

Table 4.3: Resulting cumulated traffic fluxes for functionals with and without quantization for  $\epsilon = 5, \gamma = 10$ .

$n_P = n_C$	flux w/o quantization	flux w/ quantization
1	29036	27424
5	39361	38983
10	39594	39406
20	40408	39819

#### 4.2.2 Parameter reduction: minimum green phase time

The model proposed in section 4.2.1, despite the introduction of regularization term  $R$ , allows for changes in the controls at every time step  $t_j$ . This leads to several facts: Firstly, the computational time needed to solve the OCP increases rapidly with an expansion of the predictive horizon considered. Let us denote by  $n_P$  the predictive horizon and by  $n_{TL}$  the number of traffic lights on the network. Then for every optimization step we have to solve an OCP with  $n_P n_{TL}$  degrees of freedom (DOFs). Over the full time horizon  $[0, T]$  this equals  $\frac{T}{\Delta t n_C} n_P n_{TL}$  DOFs.

Secondly, traffic lights can still switch very frequently which in no sense represents realistic control behavior. Realistic solutions require a longer predictive horizon, which overstrains the computational power of conventional computers already for very small networks (see Figure 4.4, top).

$n_c$	$n_p$			
	1	5	10	20
1	29036 (40)	38635 (334)	39388 (2073)	39390 ( $\sim 1d$ )
5		39384 (114)	39389 (559)	39390 ( $\sim 4h$ )
10			39408 (322)	39390 (3225)
15				39390 (1650)
20				39408 (1491)

Table 4.4: Evaluation of the original MPC approach in Section 4.2.1. Optimal flux  $F_T$  and total computational cost in [s] (in brackets) for different choices for the predictive horizon  $n_p$  and the control horizon  $n_c$  on a network with two incoming roads and one outgoing road.

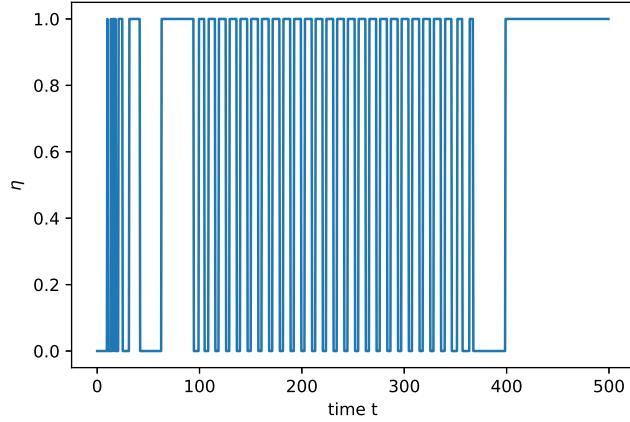


Figure 4.9: Control  $\eta_0$  for the first incoming road with  $n_P = n_C = 20$  and  $\gamma_1 = \gamma_2 = 10$ .

#### Comments on the original approach:

- The general observation states that the optimal flux increases with increasing the predictive horizon. Nevertheless, the size of the control horizon and the computational cost are indirect proportional to each other since the OCP has to be performed more often for smaller  $n_c$  as the predictive horizon is receded by a smaller time interval.
- For very short predictive horizons the future dynamics of the system does not play a major role. The traffic light will initially choose the road with a higher flux to let through the junction and will not change its signal once one road becomes congested due to the lack of an incentive (as the flux on the congested road is significantly lower than on the free road). We also refer to this setup as *instantaneous control*.
- For larger predictive horizons the future dynamics are of more and more importance. Nevertheless, we still observe very frequent switches and short periods (see Figure 4.4, bottom).

In the following, we derive a variation of the optimal control problem (4.2.1) which is supposed to be very responsive to changes in urban traffic but avoids the prohibitive computational expense of the original approach.

The most frequent methods to measure traffic volume on a road section use inductive-loop sensors (see for example RIITS in [52]). These sensors are integrated on freeways and turn on and off as cars pass over them. They typically work on a sampling rate of 1 reading/(sensor \* minute) [3]. This knowledge suggests that road signals are not allowed to switch at any time point but only on predefined **decision points**  $d_i$  where new sensor data is available. Between two decision points the traffic signal retains its current setting (either red or green). We assume that the decision points are equally distributed

over the full time horizon. Then we can refer to the duration between two successive decision points  $T_{min} = d_{i+1} - d_i$  as the *minimum green phase time*.

Let us now consider the following variables:

$d_i$ :	decision points with $d_0 = t_0, d_{n_D-1} = T$
$D_i := [d_i, d_{i+1})$ :	phase interval, $\cup_{i=0}^{n_D-1} = [t_0, T]$
$\mathcal{D}$ :	set of all phase intervals
$n_D$ :	number of decision points on $[t_0, T]$
$T_{min} = d_{i+1} - d_i$ :	minimum green/red phase time
$n_{sh}$ :	$n_{sh} = \frac{T_{min}}{\Delta t}$ signal horizon, number of time points per phase interval
$k_P$ :	number of predictive phases
$k_C$ :	number of control phases
$n_P$ :	$n_P = k_P n_{sh}$ predictive horizon
$n_C$ :	$n_C = k_C n_{sh}$ control horizon.

This leads us to the definition of the **minimum green phase optimal control problem** (MGPOCP)

$$\begin{aligned}
F_{n_C}(t_n, u^*) - R(t_n, u^*) &:= \max_{\eta_i^j \in [0,1]} F_{MPC}(\rho_i^n, t_n, \eta_n, \dots, \eta_i^{n+n_P-1}) & e_i \in \mathcal{E} \\
\text{s.t.} & \rho^{n+1} = G(\rho^n, \eta_n) \\
& \sum_{e_i \in \delta^i(J)} \eta_i^n \leq 1 \\
& \eta_j = \eta_k & \text{if } t_j, t_k \in D_i \forall j, k \in \{0, \dots, n_T\}, \\
& & i \in \{0, \dots, n_D - 1\}
\end{aligned} \tag{4.2.8}$$

where the first two constraints are carried over from the original OCP and the last constraint is added to ensure that the signals remain constant during phase intervals. By doing so, we enforce that the signal remains green (or red) for at least the duration of one phase, so does the minimum green phase time. We optimize over  $k_P$  control phases and apply to resulting control sequence on the dynamics up for the following  $k_C$  phase intervals. This avoids rapid changes for the light signals and enforce the traffic lights to only switch after a certain period of time. As a second benefit we reduce the number of free parameters (DOF) to  $\frac{T}{T_{min} k_C} k_P n_{TL} = \frac{T}{\Delta t n_{sh} n_C} n_P n_{TL}$ , a reduction by  $\frac{1}{n_{sh}}$  parameters compared to the original OPC ( $\frac{T}{\Delta t n_C} n_P n_{TL}$  DOFs). This greatly reduces the necessary computational effort while obtaining reasonable results for the optimal flux (Section 4.2.3).

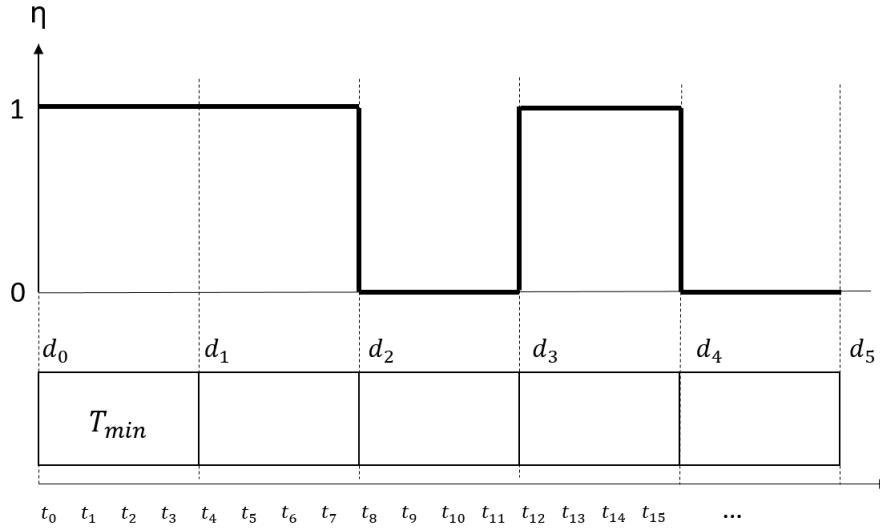


Figure 4.10: Schematic visualization of the minimum green phase approach.

#### Comments on the minimum green phase approach:

- We believe that results obtained from solving the MGPOCP are closer to the ones from the green wave model in section 4.1.2. Nevertheless we predict slightly better solutions due to higher responsivity obtained from MPC.
- The computational effort should significantly decrease compared to the original OCP.

#### 4.2.3 Numerical experiments

To solve the optimization problems (4.2.8) we use a sequential least squares programming optimizer (SLSQP) implemented as part of the SciPy module<sup>1</sup> in Python 2.7<sup>2</sup>. The SLSQP uses the Han-Powell quasi-Newton method with a Broyden-Fletcher-Goldfarb-Shanno (BFGS) update [36]. In particular we choose the convergence accuracy  $\epsilon = 10^{-6}$  for the optimization. All simulations were performed on a 8 GB DDR3 RAM Apple MacBook Pro with a 2,6 GHz Intel Core i7 processor.

For the discretization of our operating space we use  $\Delta x = \Delta t = 0.5$  and compute solutions up to final time  $T = 500$ .

<sup>1</sup>Jones E, Oliphant E, Peterson P, et al. SciPy: Open Source Scientific Tools for Python, 2001-, <http://www.scipy.org/>

<sup>2</sup>Python Software Foundation. Python Language Reference, version 2.7. Available at <http://www.python.org>

a) Evaluation of the MPC algorithm on a single junction:

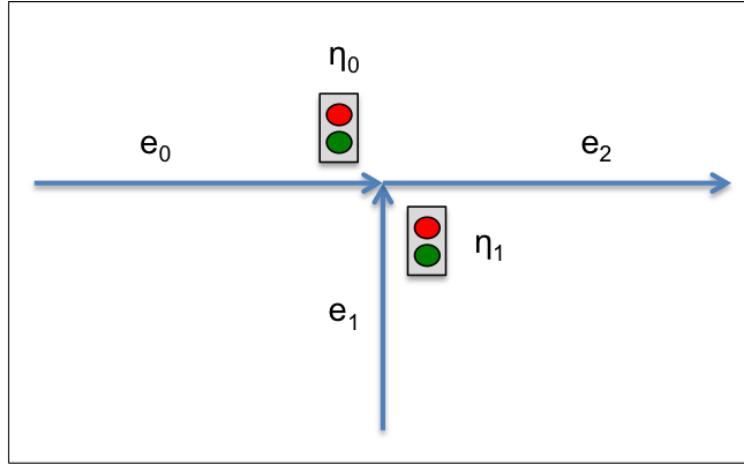


Figure 4.11: Two junction merge.

First we evaluate the computational cost and overall performance of the MPC algorithm. Therefore we use a simple junction (see Figure 4.11) consisting of two incoming roads and one outgoing road with initial values as in Table 4.1 and compare the different results which are given in Table 4.5.

$k_C = k_P$	signal horizon $n_{sh}$			
	10	20	50	125
1	32244 (10.9)	39247 (30.9)	39386 (37.5)	40440 (47.0)
2	39887 (42.5)	40730 (49.4)	40162 (91.4)	40313 (109.9)
4	40157 (119.8)	39753 (145.9)	39351 (262.4)	38543 (392.8)

Table 4.5: Optimal flux  $F_T$  and total computational cost in [s] (in brackets) for different choices for the signal horizon  $n_{sh}$  and  $k_P$ .

**Comments and comparison to the delay model:**

Using the MPC approach with minimal green phase on a single junction we obtain similar results compared to the optimal green time distribution example (experiment a) in section 4.1.2). Taking very short signal and control horizons, we observe very frequent switches in the control at almost every time step and we observe similar results to the original MPC approach (see Figure 4.4).

For appropriately chosen signal horizon and predictive horizon we observe less frequent switches and we obtain better results with respect to the cumulated flux compared to

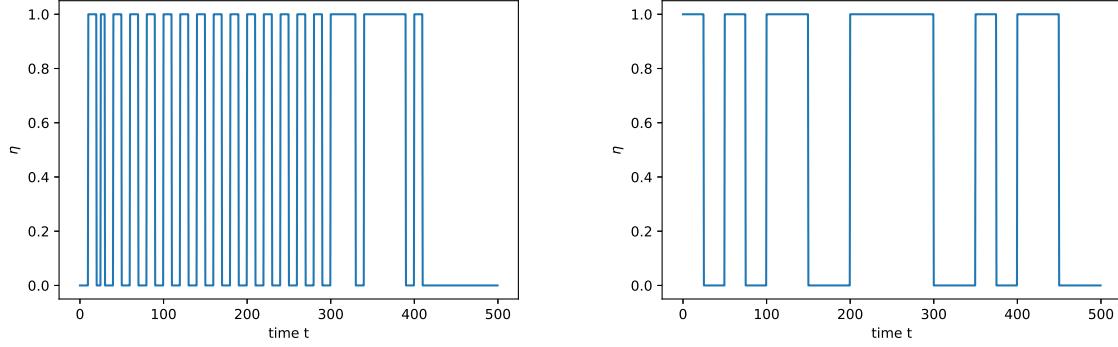


Figure 4.12: Controls  $\eta_0(t)$  applied at the first incoming road of the junction up to  $T = 500$  (for  $\epsilon = 10, \gamma = 5$ ) for  $k_C = k_P = 1, n_{sh} = 10$  (left) and for  $k_C = k_P = 2, n_{sh} = 20$  (right).

the optimal green time example (where  $F_T \sim 39382$ ). Of course the experiments were conducted with steady inflow. For varying inflow the improvement would be greater as the traffic lights can react to changing traffic on the network. The obvious advantage is the drop in computational effort to the original MPC approach (see Figure 4.4). Nevertheless, we observe an increase for longer signal horizons, so the period where we take a given control setting and do not change it. This is due to the increased cost of simulating the dynamics of the system as it needs to be simulated over the full predictive horizon for every optimization step. We also observe that the switching patterns of traffic lights appear to be more realistic and applicable with respect to real traffic situations due to the imposed conditions.

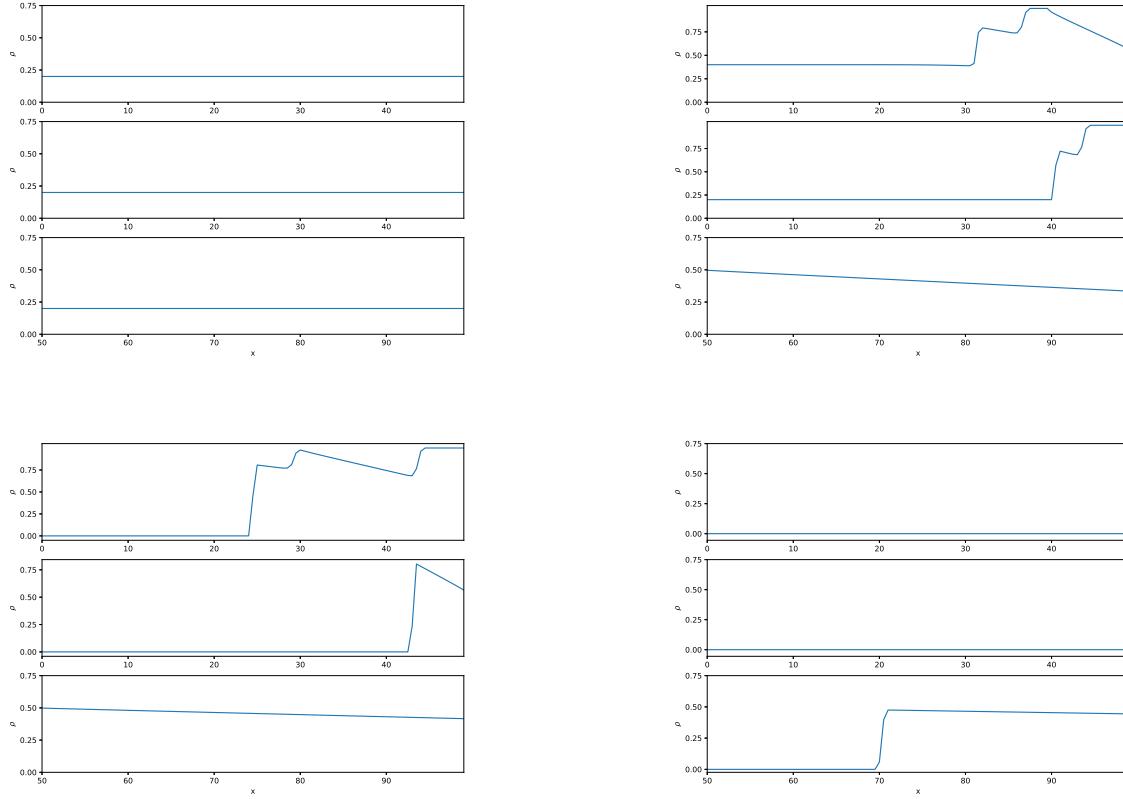


Figure 4.13: Density plots for incoming roads  $e_0$ ,  $e_1$  and outgoing road  $e_2$  (top down) for  $k_C = k_P = 2$ ,  $n_{sh} = 10$  at  $t=0$  (top left),  $t=150$  (top right),  $t=300$  (bottom left) and  $t=450$  (bottom right). The corresponding video *MPC\_single\_junction\_movie\_10\_2\_2.mp4* can be found on GitHub at [https://github.com/mstachl/Traffic\\_flow\\_optimization/tree/master/Experiments/Optimization/Model\\_Predictive\\_Control/After/MPC\\_decision\\_points](https://github.com/mstachl/Traffic_flow_optimization/tree/master/Experiments/Optimization/Model_Predictive_Control/After/MPC_decision_points).

### b) Network with two junctions:

Let us consider a network consisting of five roads and two junctions (see Figure 4.14) and impose the same initial setup as in 4.1.2. At the first junction roads  $e_0$  and  $e_1$  merge together to build a single road  $e_2$ . The traffic through the junction is controlled by traffic lights  $\eta_0$  and  $\eta_1$ . A second junction is built at the end of road  $e_2$  with incoming road  $e_3$ , merging into road  $e_4$ . The traffic at the second junction is controlled by traffic lights  $\eta_2$  and  $\eta_3$ . The goal is now to see how the MPC model at two consecutive junctions performs in comparison to the strategy with fixed periods for the traffic signals (see section 4.1).

#### Comments:

- Comparing solutions obtained through the MPC approach to the fixed strategy from section 4.1.2 ( $F_{T, \text{fixed}} \sim 83174.7$ ) we can achieve a 6.0 – 11.2% increase in

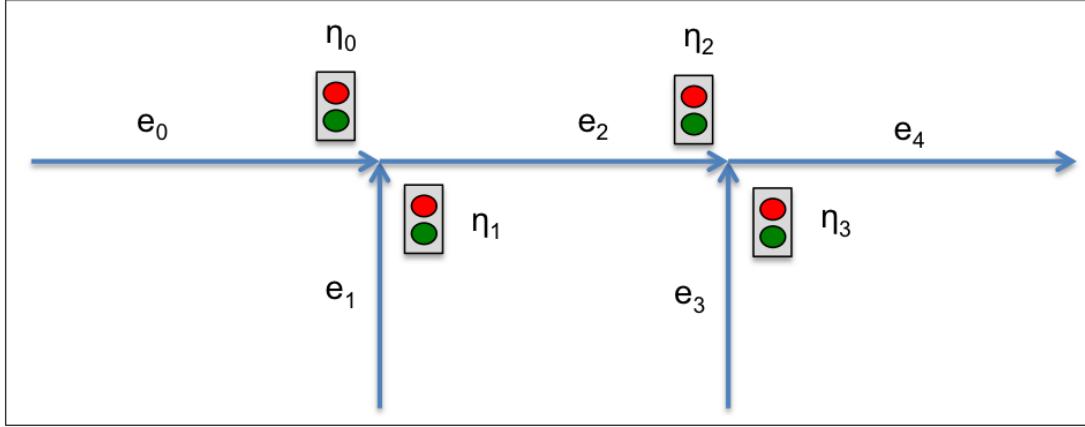


Figure 4.14: Network of two successive junctions.

$k_C = k_P$	signal horizon $n_{sh}$		
	10	20	50
1	88638 (109.9) [3905]	87480 (119.5) [2184]	87126 (144.5) [1132]
2	88375 (297.9) [5499]	90148 (364.8) [3682]	90616 (538.9) [2007]
4	89835 (905.1) [8077]	93253.9 (1081.2) [5863]	89097 (1627.4) [2667]
10	90793 (4685.1) [16112]	91688 (3881.5) [7903]	91837 (8257.7) [5730]

Table 4.6: Optimal flux  $F_T$ , total computational cost in [s] (in brackets) and number of function evaluations needed for solving the gradient descent for the OCP [in square brackets] for different choices for the signal horizon  $n_{sh}$  and  $k_P$  up to  $T = 500$  (for  $\epsilon = 10, \gamma = 5$ ).

the network flux depending on the chosen signal and predictive horizon. In this examples we used constant inflows. For fluctuating inflows, for instance heavy traffic due to peak travel times and lower traffic during non-busy periods, we expect even bigger improvements due to the increased responsivity of the traffic signals.

- The total number of function evaluations needed for performing the gradient descent decreases for enlarging the signal horizon  $n_{sh}$ . This is as the optimization has to be performed less frequently. Computational time and feval increase for increasing numbers of predictive periods  $n_P$  as the optimization becomes more and more costly.
- We also observe that the computational time increases with enlarging the signal horizon. While at first counter-intuitive, this suggests that computing the system dynamics becomes more costly for larger horizons.

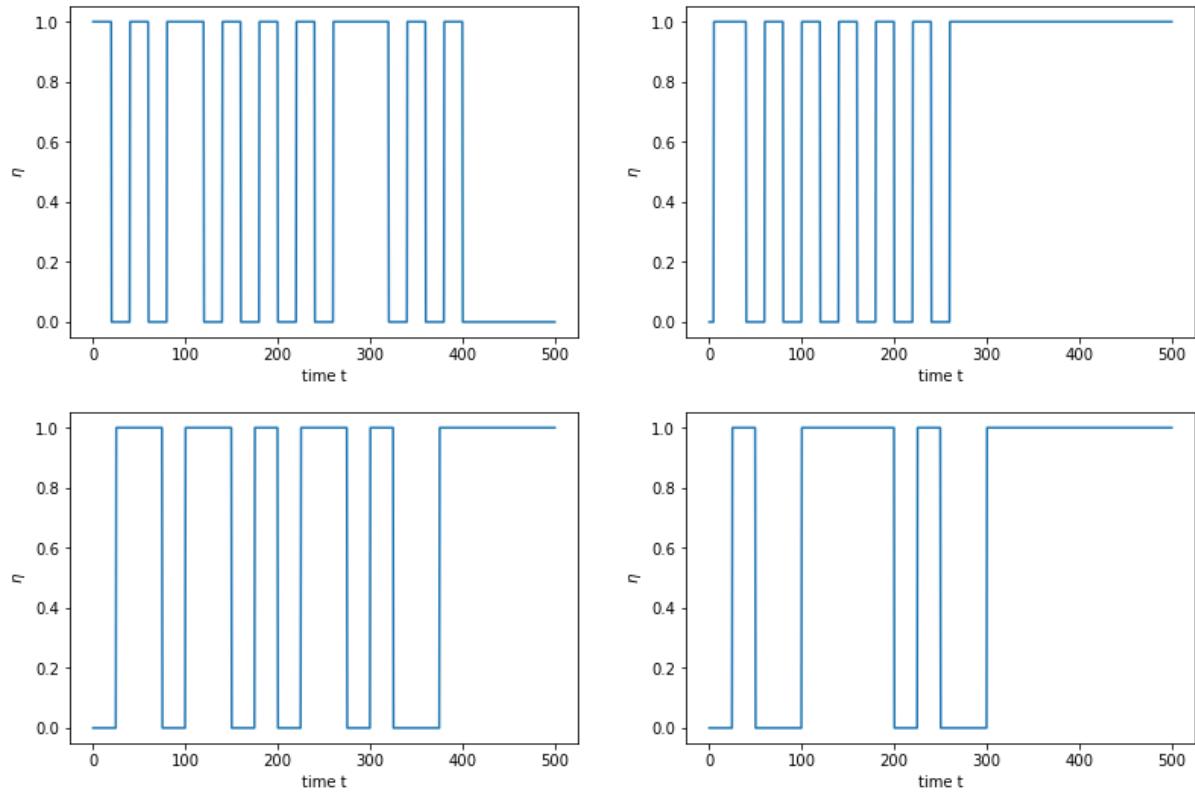


Figure 4.15: Resulting controls for traffic lights  $\eta_0$  and  $\eta_2$  at the end of roads  $e_0$  (left) and  $e_2$ , respectively, for  $k_C = k_P = 4$  and with short non-prediction periods  $n_{sh} = 10$  (top) and long non-prediction periods  $n_{sh} = 50$  (bottom).

The corresponding videos `2j_long_periods.mp4` and `2j_short_periods.mp4` can be found on github at [https://github.com/mstachl/Traffic\\_flow\\_optimization/Experiments/Optimization/Model\\_Predictive\\_Control/After/MPC\\_dp\\_b](https://github.com/mstachl/Traffic_flow_optimization/Experiments/Optimization/Model_Predictive_Control/After/MPC_dp_b).

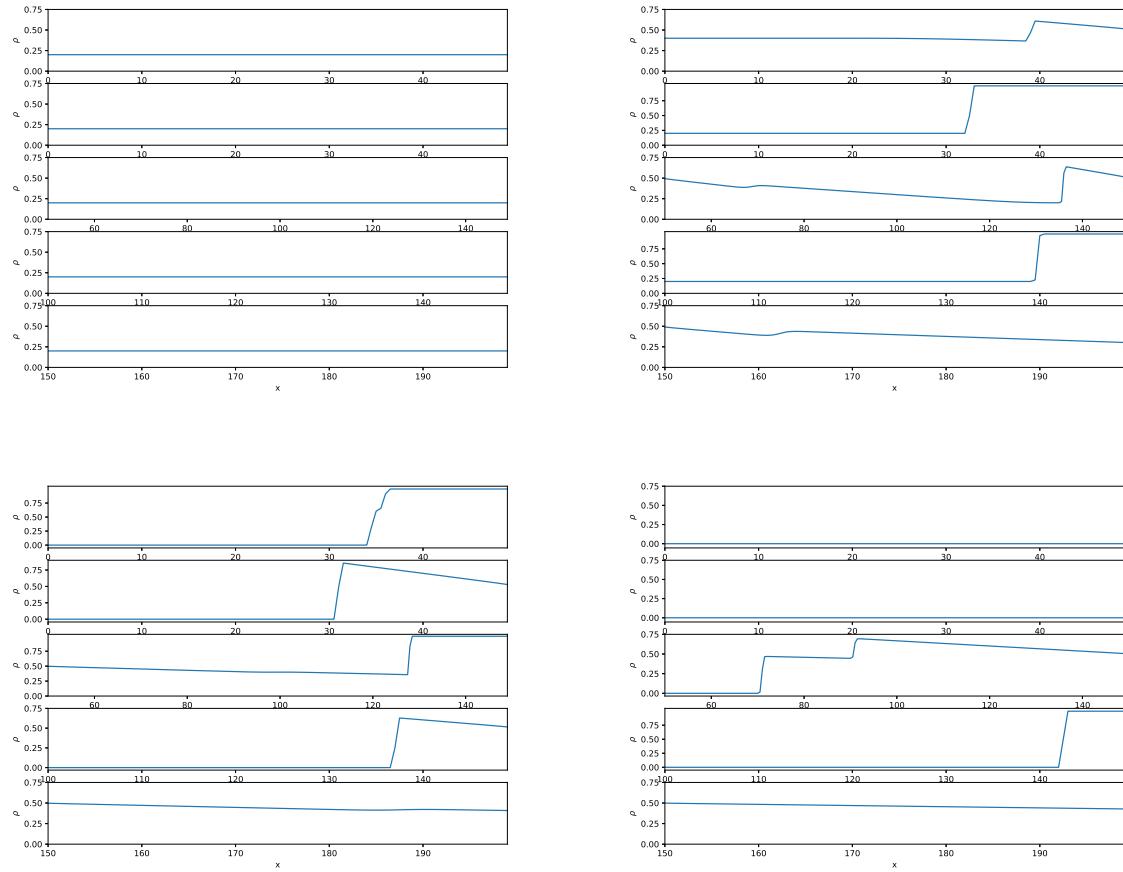


Figure 4.16: Density plots for the long non-prediction period at  $t = 0$  (top left),  $t = 150$  (top right),  $t = 300$  (bottom left) and  $t = 450$  (bottom right).

# Chapter 5

## Pollution modelling

The term pollution regards any substance released into the air. This can be solid particles (as the heavily discussed diesel particulates) as well as liquid droplets and gases (mainly carbon monoxide, the gas from motor vehicle exhaust, but also ammonia, Chlorofluorocarbons (CFC) and other substances) that can cause harmful effects on humans. The substances causing air pollution are collectively known as air pollutants.

The main sources responsible for releasing pollutants into the atmosphere can be classified into two major categories:

- anthropogenic sources, which are mostly related to burning fuel (like smoke stacks of power plants, factories and mobile sources like motor vehicles, trucks etc.), and
- natural sources including smoke from wildfires, methane from cattle or emissions from volcanic eruptions.

Due to the emissions from vehicles the transportation sector is a major contributor to air pollution in urban areas.

As our final aim is related to atmospheric pollution caused by vehicular traffic, we use the carbon monoxide ( $CO$ ) concentration as an indicator - nevertheless, we could also take nitrogen dioxide ( $N0_2$ ), particulate matter or other emitted substances instead. The general framework of this chapter follows [1].

### 5.1 CO emission in urban traffic: an instantaneous model

We will assume here that vehicle emissions are proportional to the amount of fuel consumed by the vehicle during its movement. Fuel consumption is expressed as the ratio of fuel consumed in liter per unit of distance travelled measured in  $\frac{liter}{100km}$ . It depends on many parameters, including its engine parameters, texture and gradient of the road and aerodynamic drag.

## A minimal model for fuel consumption

Depending on the motion dynamics of a car the amount of fuel and therefore the generated pollution will vary highly. In the following, we examine the behavior of a single, so-called, 'default' car. The vehicle can be in two driving modes: Cruise, if the vehicle is moving at a certain speed, or idle if the car is standing with its engine running. The generated pollution of the default vehicle is then composed of the following parts (see [61]):

$$P_{car} = \alpha(FC_c + FC_i), \quad (5.1.1)$$

where

$FC_c = \psi_c(v(t))v(t)$  fuel consumption rate in cruising state

$FC_i = \psi_0$  constant fuel consumption rate in idle state,

both measured in  $\frac{\text{liters}}{\text{s}}$  and with  $\alpha$  the CO emission level measured in grams/liter. The fuel consumption rate in idle mode is highly dependent on the average velocity  $v$  of the vehicle during a certain time interval. In particular,  $\psi_c$  denotes the cruise fuel consumption (see Figure 5.1, left) and  $\psi_0$  accounts for the consumption of the engine while working in idle mode (see for instance [6]). Typical representations of the fuel consumption curve can be found in [42, 58]. The main difference to the pollution model proposed by [1] lies in the nature of the emission curve: Alvarez et al. assume a linear relation between flux and emitted pollutants. In our approach a non-linear function  $\psi(v)$  is used.

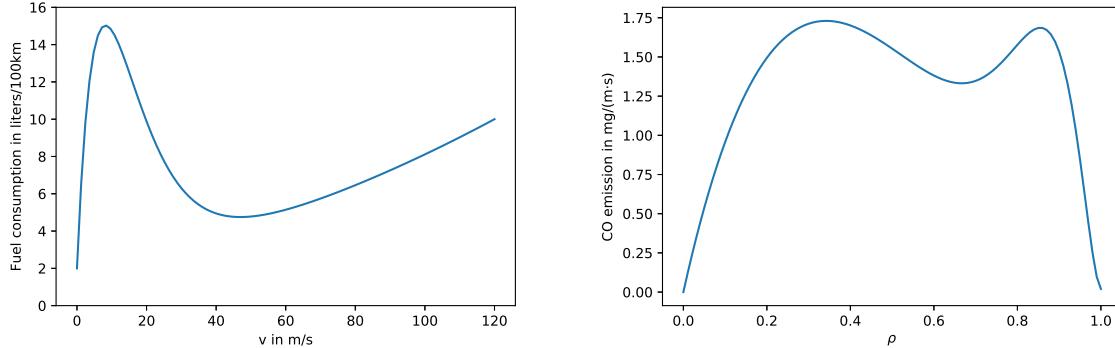


Figure 5.1: Left: Fuel consumption curve. Right: Pollution curve of a single car at a time instance  $t$ .

Under optimal conditions for combustion the entire fuel is converted to water ( $H_2O$ ) and carbon dioxide ( $CO_2$ ). The amount of CO created from fuel combustion is solely dependent on technical specifications of the car (e.g. type on injection, engine design etc.). Thresholds for CO emissions are regulated by so-called EURO norms. In Germany the currently active (2017) EURO 6b norm legally limits the CO emissions from gasoline-engine cars by  $1\frac{g}{km}$  and from diesel-engined cars by  $0.5\frac{g}{km}$  [18].

### Comments on the fuel consumption curve:

- The instantaneous model determines fuel consumption only by the average speed on the road. It does not account for rapid acceleration or de-acceleration phases.
- Fuel consumption rates can be rather diverse for different vehicles depending on engine specifications or weight and model of the vehicle. The fuel consumption curve (Figure 5.1, left) is given for a 'default car' whose specifications are given in [6, Section B.4.1]
- Standing cars consume fuel as their engines operate in engine idle mode. Therefore  $\psi(0) = \psi_0 > 0$ . Typically idling engines consume around  $1.5 \frac{\text{liters}}{\text{hour}}$ .
- For small average velocities (attained for instance in stop-and-go phases on congested roads in urban traffic), fuel consumption increases rapidly.
- Having a steady pace of moderate velocity is ideal with respect to fuel consumption. We denote this value as  $\psi_{opt} := \psi(v_{opt})$  with  $\psi'(v_{opt}) = 0$  and  $\psi''(v_{opt}) > 0$ . Further increasing the speed leads to increased air resistance and therefore increases fuel consumption.

### CO emission on a network

We now extend our knowledge of traffic-induced pollution of a single vehicle to the generated pollution of traffic densities on urban networks.

Let  $e = [a_e, b_e]$  be a road with starting point  $a_e$  and end point  $b_e$ . We propose to model the source of **instantaneous pollution** on road  $e$  due to vehicular traffic on road  $e$  by

$$P_e(t) = \int_{a_e}^{b_e} \alpha (\psi_c(v(x, t))v(x, t) + \psi_0) \rho(x, t) dx \quad (5.1.2)$$

at time  $t \in [0, T]$  with  $\rho(x, t)$  the traffic distribution on the edge. Here  $\alpha$  denotes the CO emission level measured in grams/liter and  $\psi(x, t) = \psi_c(v(x, t))v(x, t) + \psi_0$  represents the **fuel consumption rate** on the road measured in liters/s which is dependent on the average velocity  $v$ .  $\psi_c$  here denotes the cruise fuel consumption (see Figure 5.1, left) and  $\psi_0$  accounts for the consumption of the engine while working in idle mode (see for instance [6]). The instantaneous CO pollution  $P_e(t)$  on road  $e$  is now measured in  $\frac{\text{grams}}{\text{second}}$ .

Instantaneous fuel consumption models can be used to assess the impacts of proposed traffic management schemes on small road sections or urban networks where instantaneous traffic data are available.

Following this we measure the instantaneous pollution  $P_e(t)$  on the edge at time  $t$  in gram/s.

Depending on the chosen modeling approach and the complexity of the network, cars not being able to directly cross the junction may be temporarily stored in buffers. Even though buffers in the buffer model do not occupy physical space, we assume that they contribute to the total pollution in the following way. Let  $J$  be a junction on the network. Then  $M_J$  is the total capacity of this buffer and  $Q_J(t) := \sum_{i \in \delta^{out}(J)} q_i(t) < M_J$  the current capacity level of this buffer, where  $q_i$  denotes the queue length in front of road  $e_i$ . We then compute the pollution generated by junction  $J$  by

$$P_J(t) = \psi_c(\sigma)v(\sigma)Q_J(t) \quad (5.1.3)$$

As  $Q_J$  is without any unit we measure the pollution rate at a junction  $J$  in gram/s.

The buffer model tries to maximize the flux through the network, thus permits cars to leave the buffer at maximum flux. We therefore assume that vehicles inside of the buffer (the roundabout) travel with maximum flux velocity  $v(\sigma)$ .

Finally, the total pollution generated on the network over a finite time interval  $[0, T]$  can finally be computed via

$$P_{network} = \sum_{e \in \mathcal{E}} \int_0^T P_e(t)dt + \sum_{J \in \mathcal{N}} \int_0^T P_J(t)dt. \quad (5.1.4)$$

We measure the cumulated pollution in  $[P_{network}] = \text{gram}$ .

### Example:

In this example we use our toy network consisting of a single junction with two incoming roads and one outgoing, and with initial setup given in Table ???. We also assume that the network initially is empty. For a quantitative comparison between the standard buffer model without traffic lights and optimized solutions using fixed strategies and model predictive control, we apply a steady inflow for  $t \in [0, 200]$  and compare the emitted pollution up to time  $T = 550$ . As we can see in Figure 5.2 cars spend the least time on the network for the junction without traffic lights. This time is denoted by  $T_{end}$ . Nevertheless, we see that the difference in the generated pollution is very small (Figure 5.1) and can be neglected, as they are strongly depending on the chosen fuel consumption curve and therefore the specifications of the car. Furthermore, the chosen package of mass can leave the network after only a short period of time. The junction does hardly become congested during the simulation due to the size of external influxes. As almost all cars from both incoming roads can cross the junction at any time, this makes the introduction of traffic lights on this particular junction unnecessary.

### Remark 8

We could also include the pollution into the optimization model 4.2.8. Instead of maximizing the fluxes on the network we would then minimize the differences between network

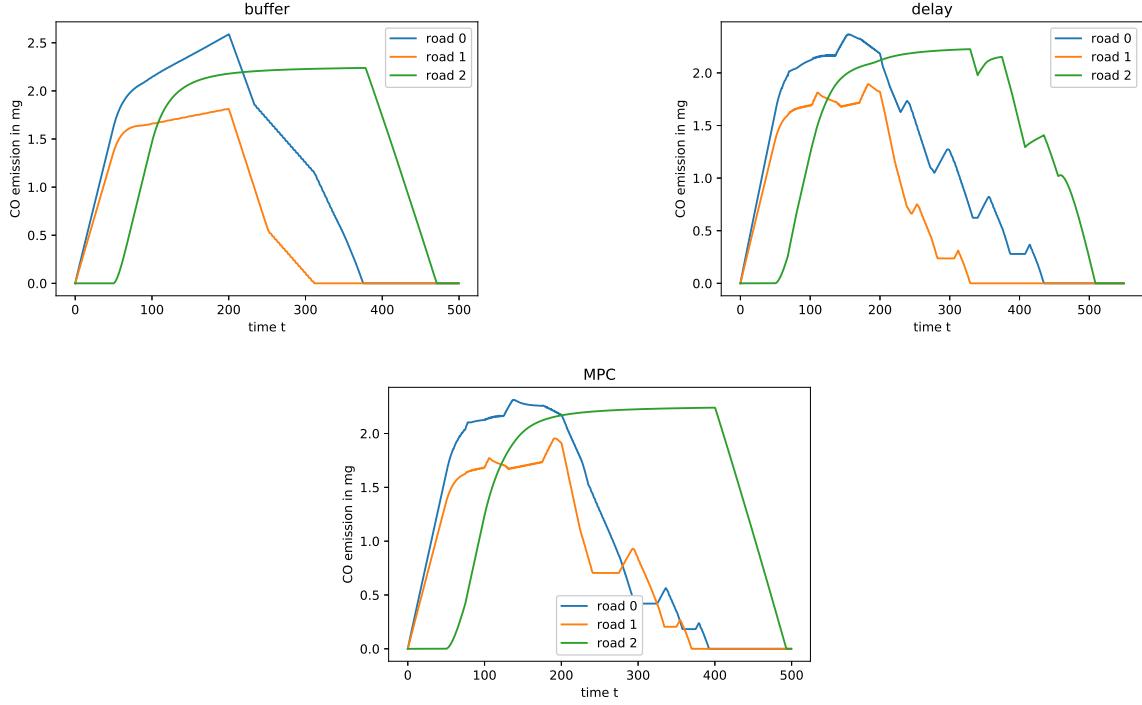


Figure 5.2: Generated pollution on the roads over time. Left: buffer model without optimization, middle: optimal lifetime, right: MPC with  $n_P = n_C = 4$  and  $n_{sh} = 50$ .

fluxes and the flux attained at the optimal flux for pollution,  $f(v_{opt}(x, t))$ . In particular, we would then define the minimization problem as

$$\begin{aligned} \min_{\eta} \sum_{e_i \in \mathcal{E}} \int_{t_n}^{t_{n+n_P}} \int_{a_i}^{b_i} |f(x, t) - f(v_{opt})| dx dt + \sum_{e_i \in \mathcal{E} \setminus \mathcal{E}^{out}} \int_{t_n}^{t_{n+n_P}} \gg f(t)_i - f(v_{opt}) | dt \\ + \sum_{e_j \in \mathcal{E} \setminus \mathcal{E}^{in}} \int_{t_n}^{t_{n+n_P}} |f_j^{\gg}(t) - f(v_{opt})| dt + \epsilon \int_{t_n}^{t_{n+n_P}} \sum_{J \in \mathcal{N}} W_J(t, \eta_i) dt \quad (5.1.5) \\ + \gamma \sum_{e_i \in \mathcal{E} \setminus \mathcal{E}^{out}} \int_{t_n}^{t_n + n_p \Delta t} \|\dot{\eta}_i\|^2 dt. \end{aligned}$$

For simplicity we use  $v_{opt} = v(\sigma)$ , which corresponds to the optimal control problem 4.2.8.

## 5.2 The 2D diffusion model

### Setting

Let us consider an urban domain  $D \in \mathbb{R}^2$  including a road network composed of  $n_R$  unidirectional roads that meet at  $n_J$  intersections. We impose that the endpoints of each

Table 5.1: Cumulated generated pollution on the network after  $T = 550s$ .

method	pollution in mg	$T_{end}$ in s
no opt	879.1	468
opt lifetime	878.4	503
mpc	866.2	491

road are either on the boundary of  $D$  or connected to one of the junctions. Each road is mapped onto the domain  $D$  via

$$\begin{aligned} \nu_i : [a_i, b_i] &\rightarrow A_i \subset D \\ s &\mapsto \nu_i(s) = (x_i(s), y_i(s)). \end{aligned} \quad (5.2.1)$$

Here  $\sigma_i$  denotes the parametrization of the segment  $A_i$  and mirrors the movement of cars on the road. In particular, let  $p_i, q_i \in A_i$  be start and end points of the segment  $A_i$  and  $s$  a point on road  $e_i$ . Then we can write the map  $\nu(s)$  as the convex combination of  $p_i$  and  $q_i$ , namely

$$\nu_i(s) = \left(1 - \frac{s - a_i}{b_i - a_i}\right) p_i + \frac{s - a_i}{b_i - a_i} q_i, \quad (5.2.2)$$

where  $\frac{s - a_i}{b_i - a_i} = N(s)$  denotes the normalization of  $[a_i, b_i]$  onto  $[0, 1]$ . We also define its inverse as  $l_i : p_i \bar{q}_i \rightarrow [a_i, b_i]$ ,  $l_i = \nu_i^{-1}$ .

We study the CO transport in a urban area  $D$  with a simple two-dimensional transport diffusion model

$$\partial_t \Phi - \nabla_x \cdot \mu \nabla_x \Phi = P(\nu, t), \quad (5.2.3a)$$

where  $\Phi(\nu, t)$  denotes the CO concentration at a point  $\nu = (x, y) \in D$  at the time moment  $t \in [0, T]$ .  $P(\nu, t)$  describes the CO emission rates based on the solution of the LWR-model (see later.....) and  $\Delta$  is the two-dimensional gradient.  $\mu(\nu, t)$  denotes the diffusion coefficient describing the particle movement on the domain. In general the coefficients  $\mu(\nu, t)$  are rather complicated functions and difficult to determine. To simplify the problem these functions are approximated by constant values  $\mu(\nu, t) = \mu$ .

As initial and boundary conditions we take

$$\Phi(r, 0) = \Phi_0(\nu) \quad \text{at } t = 0 \quad (5.2.3b)$$

$$\mu \frac{\delta}{\delta n} \Phi = 0 \quad \text{on } \delta D \quad (5.2.3c)$$

with  $\delta D$  the boundary of the domain  $D$ .

In contrast to [53, 54, 1] this is a simplistic model which only simulates the diffusion

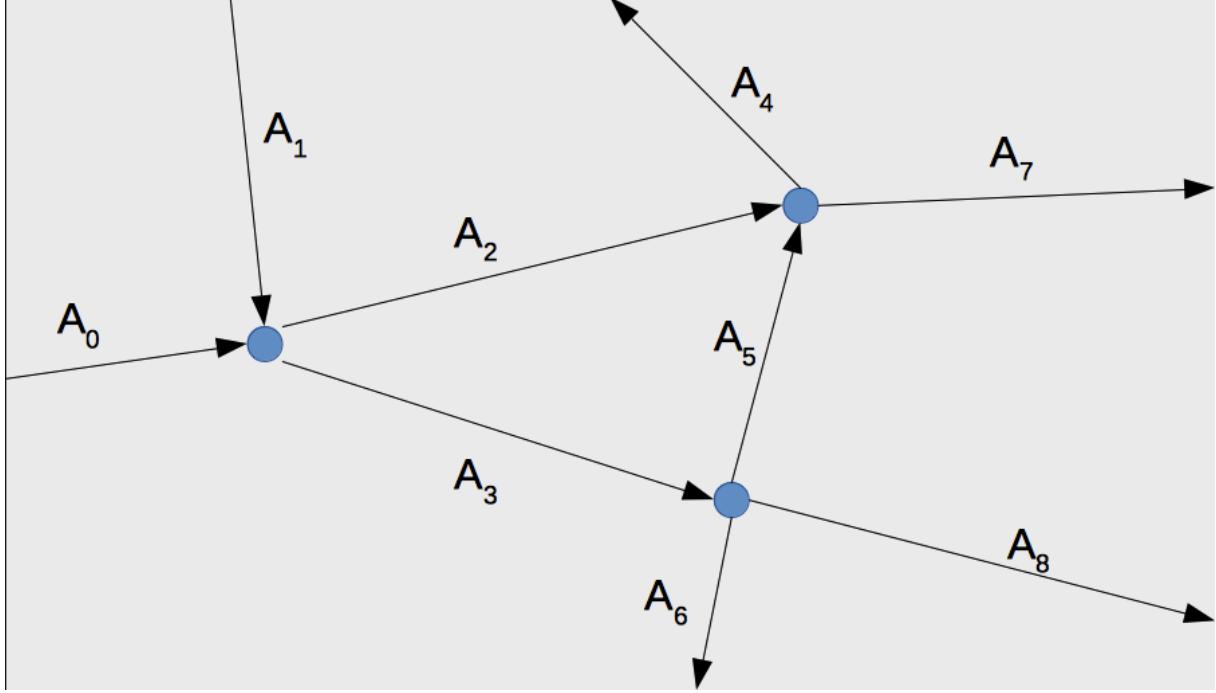


Figure 5.3: Schematic diagram of a 2D domain containing a network of  $n_R = 9$  unidirectional roads and  $n_J = 3$  junctions.

of particles emitted by travelling cars. It does not account for advection caused by atmospheric wind or other natural phenomena like interparticle reactions.

As the CO sources we consider vehicular emissions along  $n_R$  roads:

$$P(r, t) = \begin{cases} P_i(\nu, t) & \text{if } \nu \in A_i \\ 0 & \text{else.} \end{cases} \quad i = 0, \dots, n_R - 1 \quad (5.2.4)$$

We will assume here that the vehicle emissions are dependent on the velocity and the density of cars on the road, which will be known from the solution of the buffer model (2.3). In particular, we propose to model the source of pollution due to vehicular traffic by

$$P_i(\nu, t) = \alpha \psi_c(v(\nu_i(s), t)) v(\nu_i(s), t) \rho_i(\nu_i(s), t), \quad (5.2.5)$$

where  $\alpha$  is the contamination rate and  $\psi_c(v(\nu, t))$  the velocity dependent fuel consumption rate of a single vehicle on the segment  $A_i$ . We impose that  $\psi_c(v)$  is a Lipschitz function in  $v$ . A typical representation of  $\psi$  can be found in [42, 58] (see Figure 5.1).

For an arbitrary section  $[\nu_i(x), \nu_i(y)] \subset A_i, x, y \in e_i$  we can then define the cumulated pollution at time  $t$  as

$$\int_{\nu_i(x)}^{\nu_i(y)} P_i(\nu, t) d\nu = \int_x^y \alpha \psi_c(v(s, t)) \rho_i(s, t) v(s, t) \|v'(s)\| ds. \quad (5.2.6)$$

Noting that  $\|\nu'_i(s)\| = \frac{1}{b_i - a_i} \|p_i - q_i\|$  we observe that the lengths of segment  $a_i \subset \mathbb{R}^2$  coincide with the length of road  $e_i$  whenever  $\|\nu'_i(s)\| = 1$ .

## Analytical solution

From the fact that  $\rho_i(x, \cdot) \in L^1_{loc}$  and  $\xi(v)$  Lipschitz in  $v$  and assuming that  $\phi_0 \in L^2(D)$ , we can define weak solutions for problem (5.2.3).

**Definition 5.2.1** Given  $r, p \in [1, 2)$  such that  $\frac{2}{r} + \frac{2}{p} > 3$ . Then we say that a function  $\phi \in L^r(W^{1,p}(D); 0, T)$  is a weak solution of problem (5.2.3) if for all testfunctions  $v \in C^1(\bar{D} \times [0, T])$  with  $v(\cdot, T) = 0$  the following equality is verified:

$$\begin{aligned} - \int_0^T \int_D [\Phi(t, \nu) \partial_t v(x, t) + \mu \nabla \Phi \nabla v] d\nu dt &= \int_D \phi(x, 0) v(x, 0) dx \\ &+ \sum_{i=0}^{n_R} \int_{a_i}^{b_i} \alpha \psi_c(v(\rho_i(s, t))) \rho_i(s, t) v(s, t) \|\nu'(s)\| ds. \end{aligned}$$

Now we can prove the following existence and uniqueness result:

**Theorem 5.2.1** Problem (5.2.3) has a weak solution  $\phi$ , which is unique. Moreover,  $\phi \in L^r(W^{1,p}(D); 0, T)$ ,  $\forall r, p \in [1, 2)$  such that  $\frac{2}{r} + \frac{2}{p} > 3$ .

*Proof:* The existence and uniqueness of solution can be obtained arguing as in [15], Th 6.3.

### Remark 9

The solution to problem 5.2.3 can be computed numerically by using e.g. finite elements methods (FEM) (see e.g. [55, 7]). There the two-dimensional domain  $D$  is approximated by a triangulation  $D_h$  with grid size  $h > 0$ . On every element  $T_h$  the problem is solved separately. The global solution is then obtained by systematically combining the local solutions on the elements into a global system. This global system can be solved with initial values of the original problem.

### 5.3 A case study: Munich - Schleißheimer Straße

We want to conclude the thesis with a practical study to demonstrate an area of application.

In the following we consider a 2km part of the Schleißheimer Straße (see Figure 5.4), a main road through the inner city of Munich, the most populous city of the state Bavaria and the fourth biggest city in Germany at about 1.5 million people (2017). That area is chosen due to its high traffic volume during peak traffic periods accompanied by a high emergence rate of traffic jams. Due to the frequent stop-and-go phases this area is suitable for optimization of the resulting pollution generated by vehicular traffic.

The considered part begins at the Petuelring in the north, a busy area 4 km north of the center of Munich. From there the street leads to the south. In the following we consider a sequence of two junctions. At the first junction, the Schleißheimer Straße merges with the Lerchenauer Straße coming from the west. The second junction is build by a crossing of the Schleißheimer Straße in north-south direction, Ackermannstraße coming from the west and Karl-Theodor-Straße in the east. We therefore consider a road network formed by  $N_R = 6$  unidirectional avenues, with  $N_J = 2$  junctions.

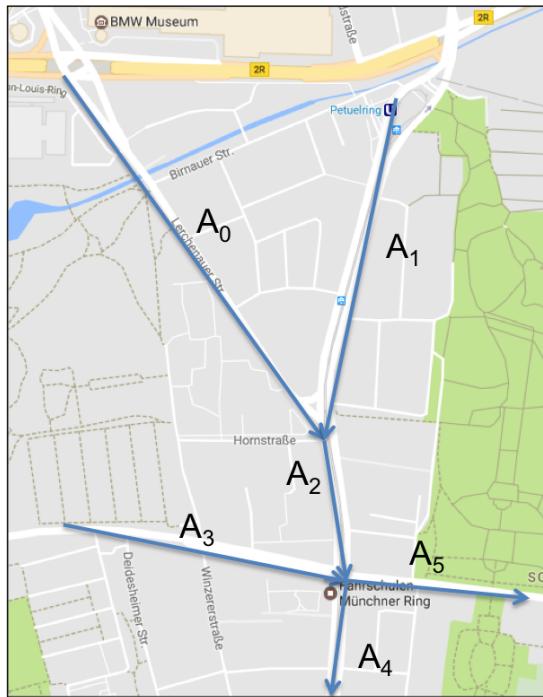


Figure 5.4: Schleißheimer Straße and adjacent roads.

#### The LWR model

To solve the system 2.3.8 under consideration, we first need to define the fundamental diagram. Taking equation 1.2.5 we consider the relative density  $\rho$  on the network and a

maximum density  $\rho_{max} = 1$ . For the maximum velocity allowed in urban areas we take  $v_{max} = 50 \frac{km}{h}$ , resulting in the fundamental diagram

$$f(\rho) = 50 \frac{km}{h} \rho (1 - \rho)$$

We consider that the avenues presented in this study show similar behavior, so we admit the same theoretical flux on all of them. For the discretization we choose  $\Delta x = 10m$  and  $\Delta t = 0.5s$  (Note that the CFL condition 3.1.10 is met). In the following experiments we assume that the network is initially empty (meaning that  $\rho_i(x, 0) = 0, i = 0, \dots, 9$ ) and that a finite number of cars can enter avenues  $A_0$ ,  $A_1$  and  $A_3$ . We want to observe how

Table 5.2: Initial setup for the case study.

road	$A_0$	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$
length in m	400	500	500	850	400	400
initial density	0	0	0	0	0	0

this portion of traffic distributes on the network over time for different conditions at the junctions. For the experiments we assume that the package of traffic is initially stored at the beginning of the incoming avenues (see Figure 5.5). We observe how the travelling waves caused by the inflow move along the roads, and we choose a simulation time of  $T = 3.5h$ . For the distribution matrix at the first junctions we choose

$$TD_0 = \begin{pmatrix} 1 & 1 \end{pmatrix}.$$

resembling the merging of two incoming roads into one outgoing. At the second junction we choose the distribution matrix

$$TD_1 = \begin{pmatrix} 0.8 & 0.6 \\ 0.2 & 0.4 \end{pmatrix}$$

meaning that a vehicle driving on the Schleißheimer Straße will stay on it with 80% probability and a car from a side road will turn off onto the Schleißheimer Straße 60% of the time. For the buffer sizes we uniformly choose  $M_J = 4$  for every junction and equal priorities  $c_i$  for all incoming roads  $e_i$ .

### Traffic behavior at different types of junctions under different optimization schemes

In this experiment we consider junctions of different types. In the first case we consider a junction with less outgoing roads than incomings. In particular the considered junction

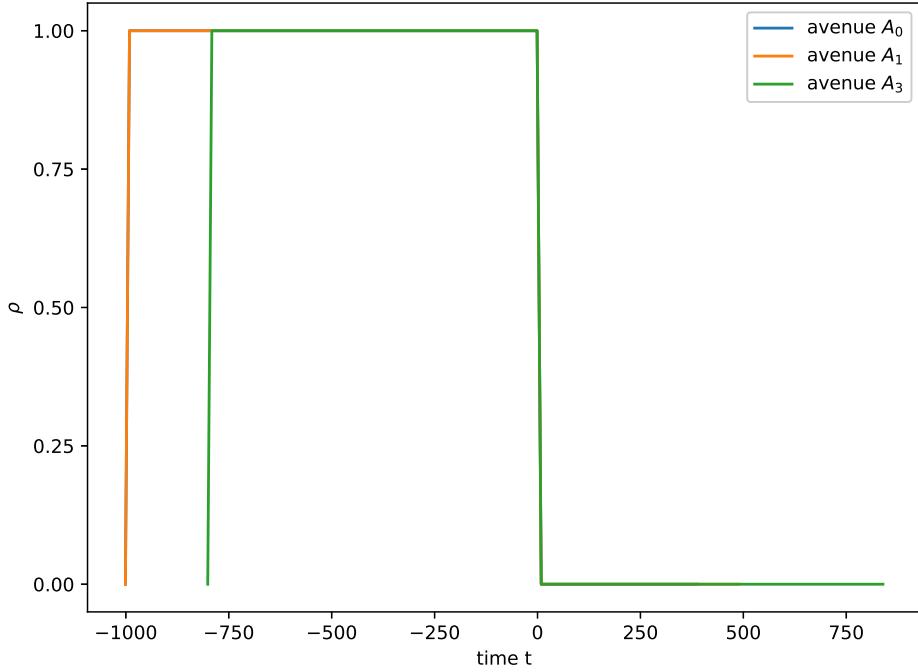


Figure 5.5: Incoming density into the network.

consists of two incoming roads and a single outgoing road ( $|\delta^{in}(J_0)| = 2 > 1 = |\delta^{out}(J_0)|$ ). At the second junction the number of incoming avenues equals the number of outgoing roads ( $|\delta^{in}(J_1)| = 2 = |\delta^{out}(J_1)|$ ). We analyze the behavior of the solutions in each one of the junctions using the different modelling approaches described previously, namely the buffer model resembling an approach without traffic lights, the delay model using a fixed switching strategy and the MPC model using a responsive switching. In particular for the fixed strategy we use a fixed green (and red) phase duration of 30s where switching of the second junction is delayed by 10s compared to the first. For the MPC a signal horizon of 25s is used with  $k_P = k_C = 2$ .

a) **Merge** (with incoming roads  $A_0, A_1$  and outgoing road  $A_2$ ):

- no optimization:

In the absence of an optimization scheme, and therefore the absence of traffic lights, the inflow of the two incoming roads into the junction is not regulated. Cars that cannot be handled by the junction are instead put in the queue and influence further inflow into the junction. Once the sum of the influxes into the junction exceed the maximum carrying capacity of the outgoing road, buffer builds up (see Figure 5.6, blue line). This causes shock waves with negative speed on the two incoming roads (see Figure 5.7). After some time the buffer level stabilizes. This happens once the

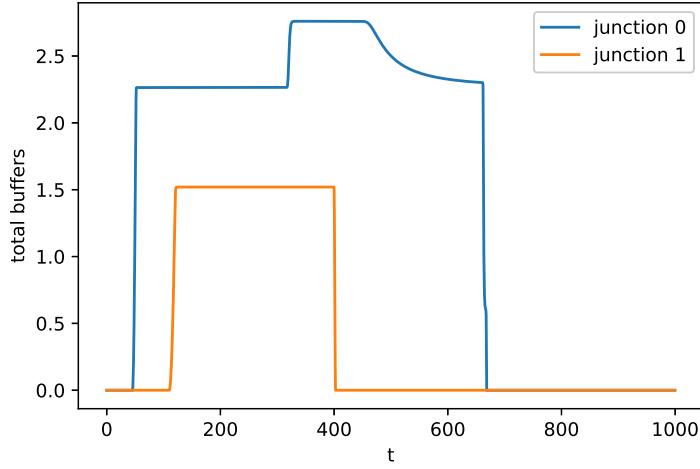


Figure 5.6: Junction buffers for the two junctions in the network.

inflow equals the maximum carrying flux on the outgoing road,

$$0 = \dot{q} = \sum_{i=0}^1 \gg f_i \bar{\Theta}_{i,2} - f_2 \gg$$

Cars stored in the buffer eventually leave it once the influxes drop.

- fixed strategy:

In the fixed strategy model we assume fixed green and red phase durations of 30s. Once the control signal at incoming road  $A_0$  changes from red to green (0 to 1) the signal at avenue  $A_1$  switches to red, and vice versa (see Figure 5.9). Every switching process triggers a discontinuity in the incoming fluxes, resulting in shock waves propagating with negative speed on the incoming road with red light and a rarefaction wave on the road with green light. At the outgoing road we observe cars leaving the junction at maximum flux and distributing on the road based on their density dependent velocities. In general using this model leaves junction buffers empty as long as the outgoing road remains in free state ( $\bar{\rho} \leq \sigma$ ), see for instance junction 1 in Figure 5.8. If the outgoing road becomes congested ( $\bar{\rho} \geq \sigma$ ), for example due to backwards propagating traffic jams caused by a second junction, we observe that buffer builds up and drops periodically (see 5.8, blue line) corresponding to the switching patterns of the traffic signal.

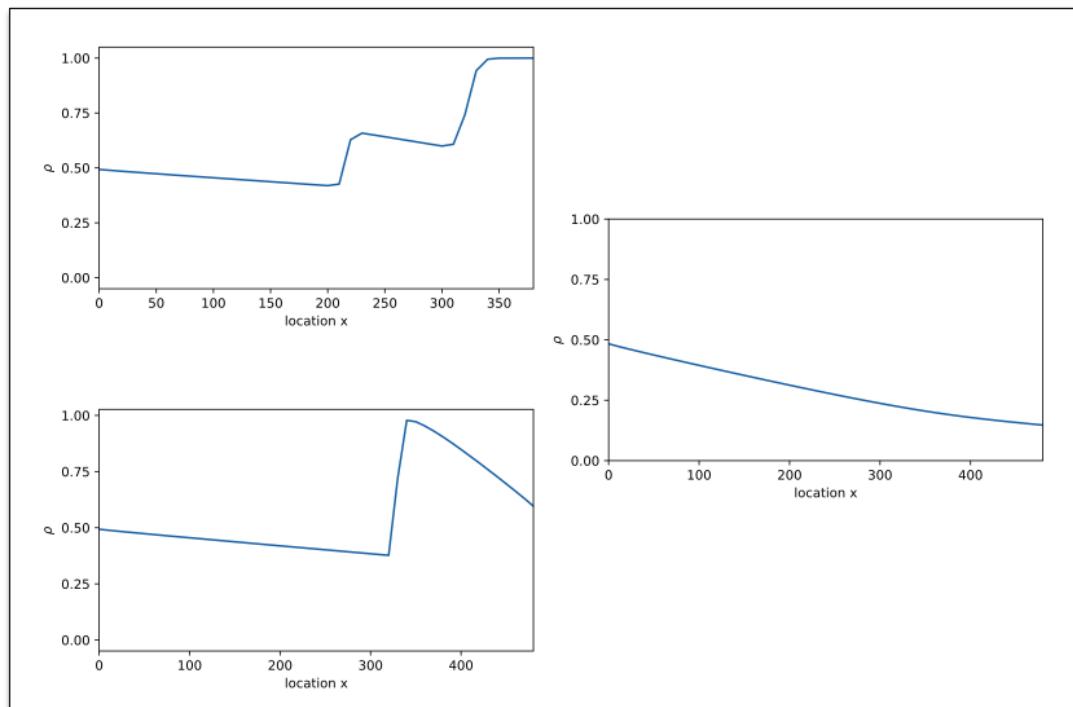


Figure 5.7: Behavior of the solution at junction  $J_0$  using no optimization: Vehicular density on the incoming avenues  $A_0$  (top left) and  $A_1$  (bottom left) and the outgoing road  $A_2$  (right) at time  $t = 100s$ .

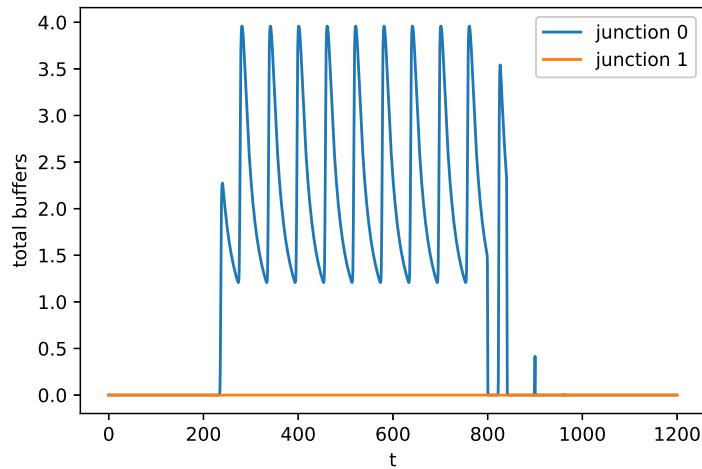


Figure 5.8: Junction buffers for the two junctions in the network.

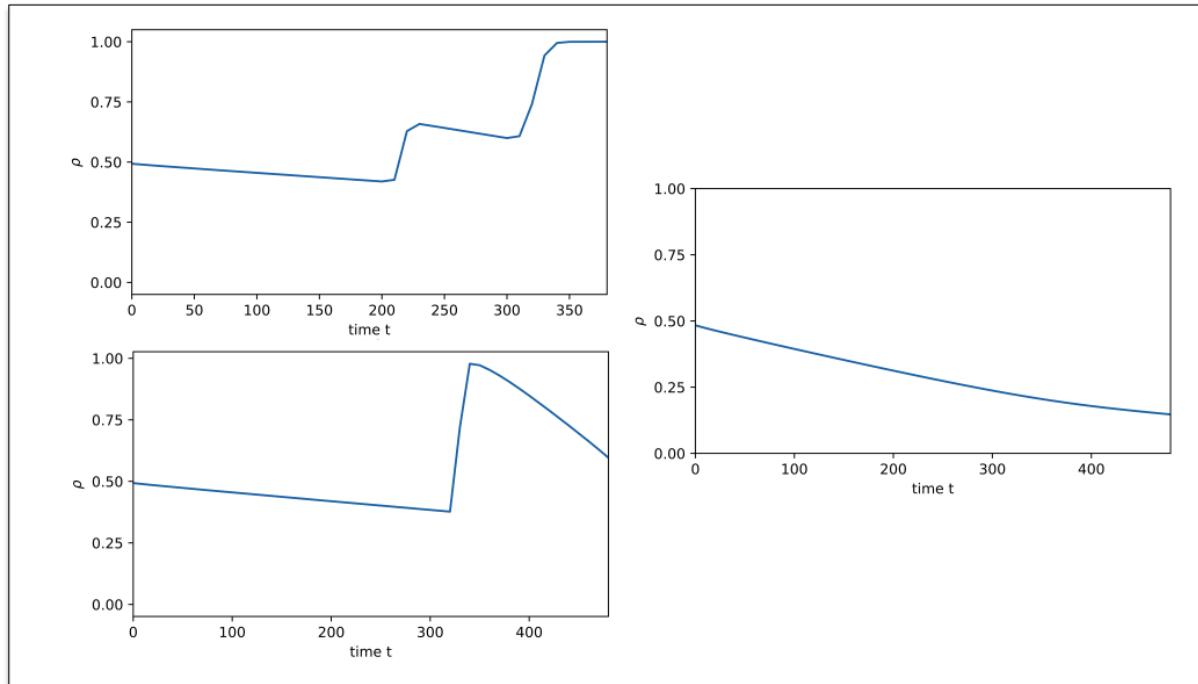


Figure 5.9: Behavior of the solution at junction  $J_0$  using the fixed strategy: Vehicular density on the incoming avenues  $A_0$  (top left) and  $A_1$  (bottom left) and the outgoing road  $A_2$  (right) at time  $t = 100s$ .

- MPC:

The distribution of the traffic obtained by MPC is similar to the solution using a fixed strategy. Red lights cause shock waves with negative speed traveling backwards (Figure 5.10). Compared to the fixed strategy the buffer at this junction gets filled less due to the higher responsivity of the MPC approach (Figure 5.11) The peak on the right end of the outgoing road is caused by a red phase at the successive junction. The switching pattern of junction 0 is shown in Figure 5.12.

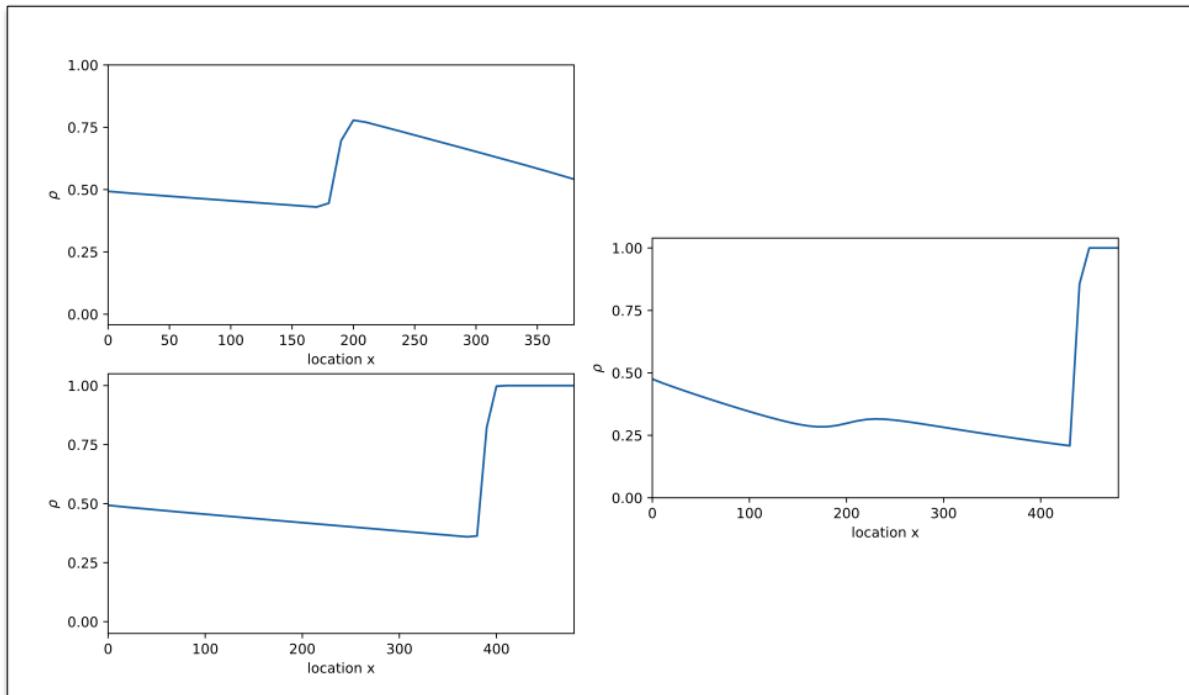


Figure 5.10: Behavior of the solution at junction  $J_0$  using MPC: Vehicular density on the incoming avenues  $A_0$  (top left) and  $A_1$  (bottom left) and the outgoing road  $A_2$  (right) at time  $t = 100s$ .

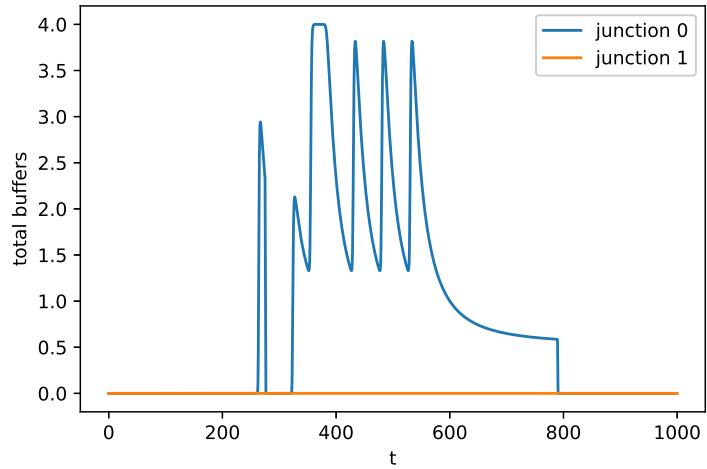


Figure 5.11: Junction buffers for the two junctions in the network using MPC.

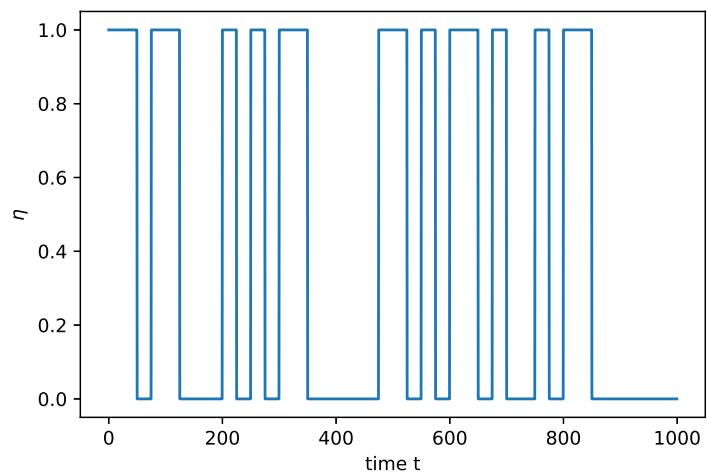


Figure 5.12: Traffic light pattern at the end of avenue  $A_0$ . The pattern at the second incoming road  $A_1$  is the inverted signal of  $A_0$ .

**b) Crossroad** (with incoming roads  $A_2, A_3$  and outgoing roads  $A_4, A_5$ ):

- No optimization:

The numerical solution achieved by the buffer model reflects the turning preferences  $TD_1$  as defined previously in this section. It shows that avenue  $A_4$  is loaded with high level of traffic while avenue  $A_5$  deals with a lower level (see Figure 5.13). On both incoming roads we observe the reduced inflow to the junction caused by the junction buffer (see 5.6, orange line). The left drop in density on the road  $A_3$  is caused by the end of the external inflow.

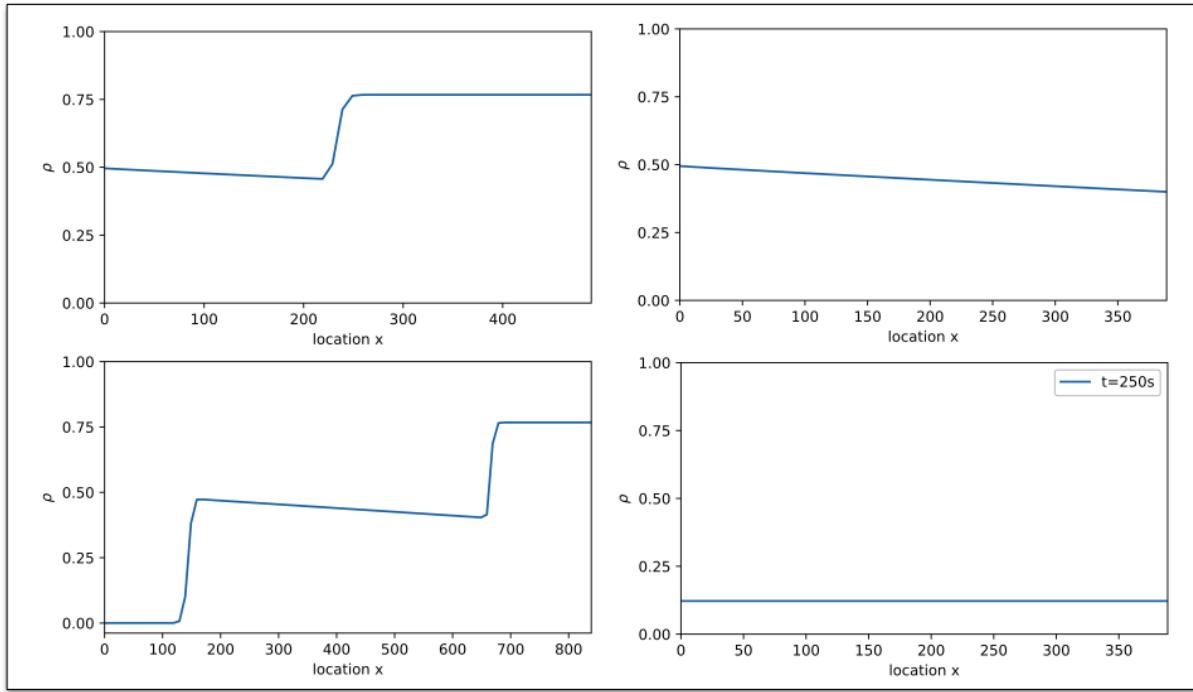


Figure 5.13: Behavior of the solution at junction  $J_1$  without optimization: Vehicular density on the incoming avenues  $A_2$  (top left) and  $A_3$  (bottom left) and the outgoing roads  $A_4$  (top right) and  $A_5$  (bottom right) at time  $t = 250s$ .

- fixed strategy:

Due to the uniform switching of the traffic signals at the crossroad, waves caused by traffic jam travel backwards. This causes incoming road  $A_2$  to congest (see Figure 5.14). We also see smooth changes in the outgoing densities triggered by different turning preferences for both in- and outgoing roads. As in the evaluation of the buffer model, the first drop of the density at incoming road  $A_3$  is caused by the end of the external inflow. We also note that the uniform switching pattern does not change even when one incoming road becomes empty at some point. This can force cars to wait in front of the traffic light, even though the junction is empty.

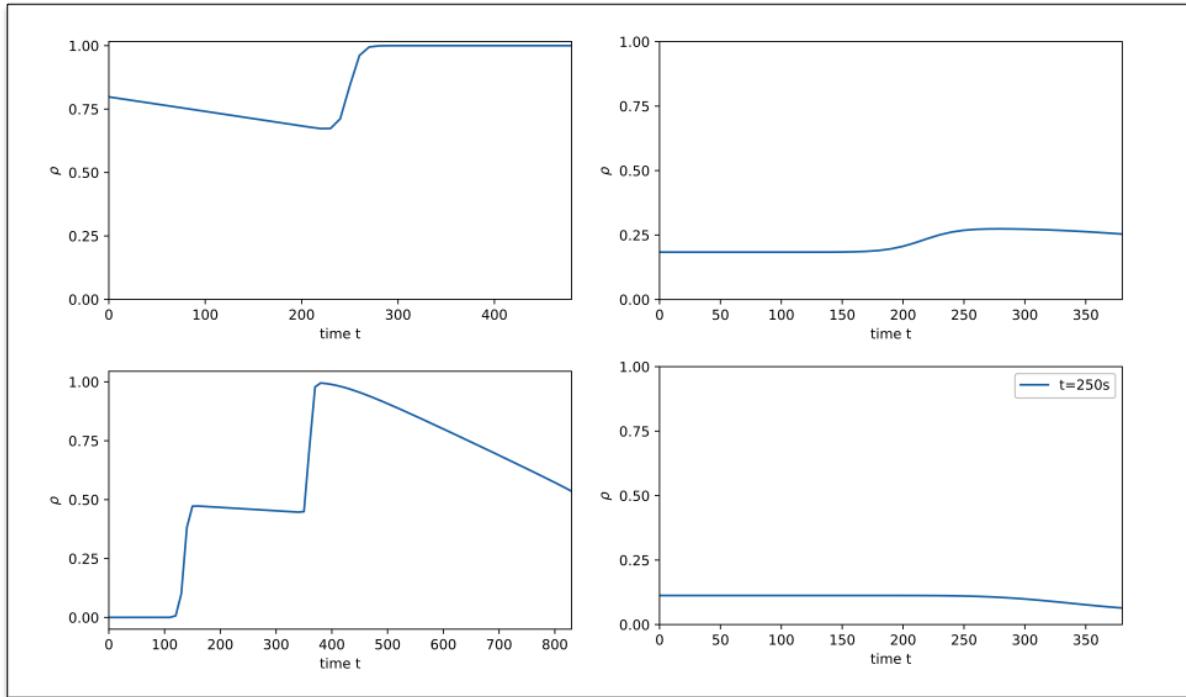


Figure 5.14: Behavior of the solution at junction  $J_1$  using the fixed strategy: Vehicular density on the incoming avenues  $A_2$  (top left) and  $A_3$  (bottom left) and the outgoing roads  $A_4$  (top right) and  $A_5$  (bottom right) at time  $t = 250s$ .

- MPC:

The numeric solution obtained via MPC shows the same qualitative behavior as the one obtained using a fixed strategy (see Figure 5.15). Again traffic waves propagate backwards on incoming roads at red light. Compared to the fixed strategy example the traffic lights always permit cars on incoming roads to cross the junction once the other incoming roads are empty (see Figure 5.16).

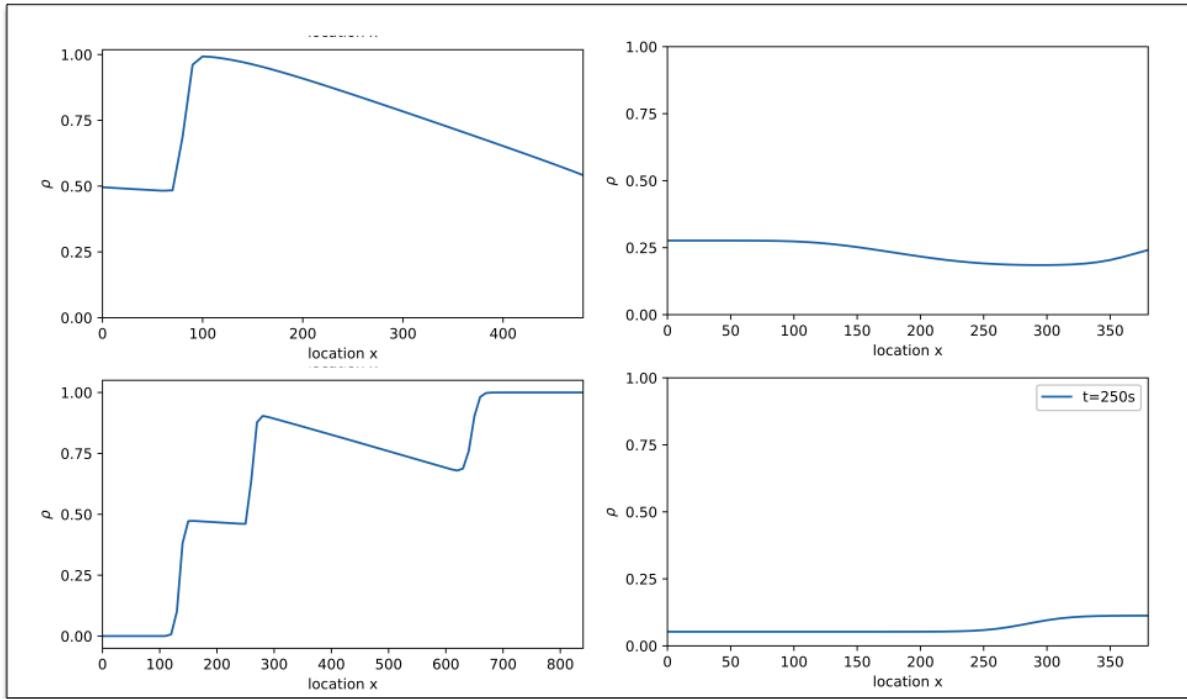


Figure 5.15: Behavior of the solution at junction  $J_1$  using MPC: Vehicular density on the incoming avenues  $A_2$  (top left) and  $A_3$  (bottom left) and the outgoing roads  $A_4$  (top right) and  $A_5$  (bottom right) at time  $t = 250s$ .

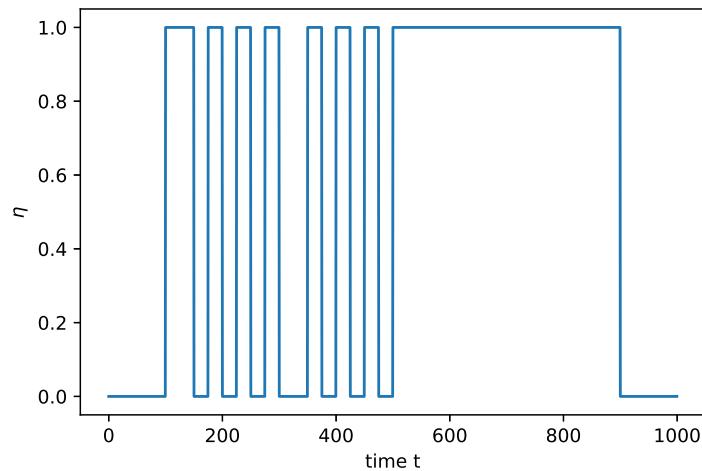


Figure 5.16: Traffic light pattern at the end of avenue  $A_2$ . The pattern at the second incoming road  $A_3$  is the inverted signal of  $A_2$ .

## Pollution

In this section we use the data obtained from the three models in the previous section to compute the total pollution emitted by the transport of a package of vehicles through the network.

The city of Munich provides a speed limit for all vehicles in the urban area. In the following we assume that cars can only move as fast as the maximum allowed velocity of  $50 \frac{km}{h}$ , assuming that drivers stick to the allowed speed limit. Therefore only the left part of the fuel consumption curve in Figure 5.1, right, needs to be considered. Due to the changes in the function of the fuel consumption also the pollution curve depending on the density on the road changes its shape (see Figure 5.17, right).

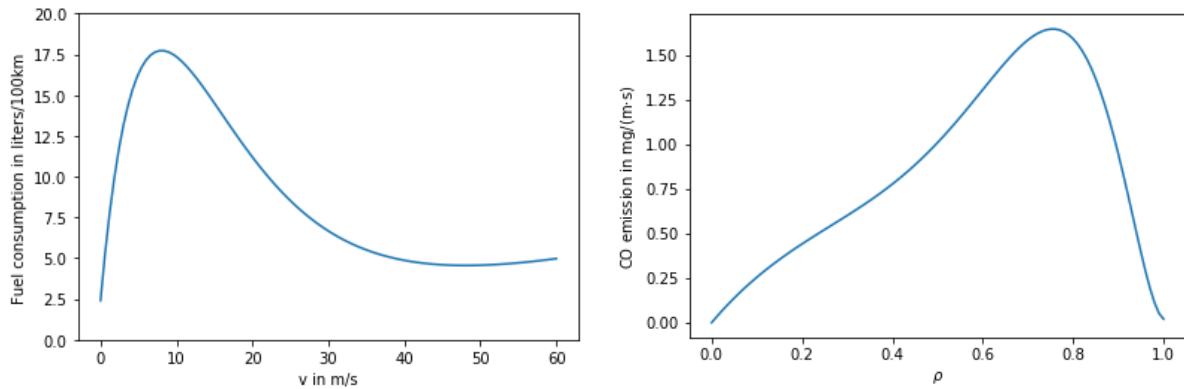


Figure 5.17: Left: Fuel consumption curve for urban traffic. Right: Rate of a single car dependent on the congestion of the road.

We consider again the buffer model, the model with fixed strategy and the MPC model. The cumulated flux as well as the time needed for the vehicles to travel through the network and the generated pollution is given in Table 5.3. As all simulations are run until every car has left the network, the cumulated flux is approximately the same for every model. The main difference can be seen in the needed travel time. In the general buffer model a portion of cars from all incoming roads can cross the junction at any time, keeping the traffic flux flowing at any time. Thus, the total transport time  $T_{end}$  is the shortest when roundabouts are used. Nevertheless, the buffer is filled to a considerable level through most of the simulation, causing a lot of pollution through cars moving in the buffer (imagine roundabouts). Using traffic lights with fixed switching patterns can cause a massive increase in the travel time for some vehicles. Since the fixed strategy cannot adapt to changing traffic situations, it can force cars to wait in front of the junction although all other incoming roads are free. This also causes the number of emitted pollutants to increase. This phenomenon can be relaxed through the predictive feature of model predictive control, leading to a shorter total travel time. As the buffers are also empty through longer periods the total pollution drops below the level when using no optimization. The detailed distribution of generated pollution on the roads can be seen in Figure 5.18.

Table 5.3: Computed cumulated flux, total transport time  $T_{end}$  and total generated pollution on the network for the buffer model without optimization, traffic lights with fixed control strategy and optimized control via MPC.

Method	$F_T$	$T_{end}$ in s	$P_{network}$ in mg
No optimization	1026177.00	792	387070.93
Fixed strategy	1010089.28	>1200	393296.07
MPC	1018152.98	948	371771.30

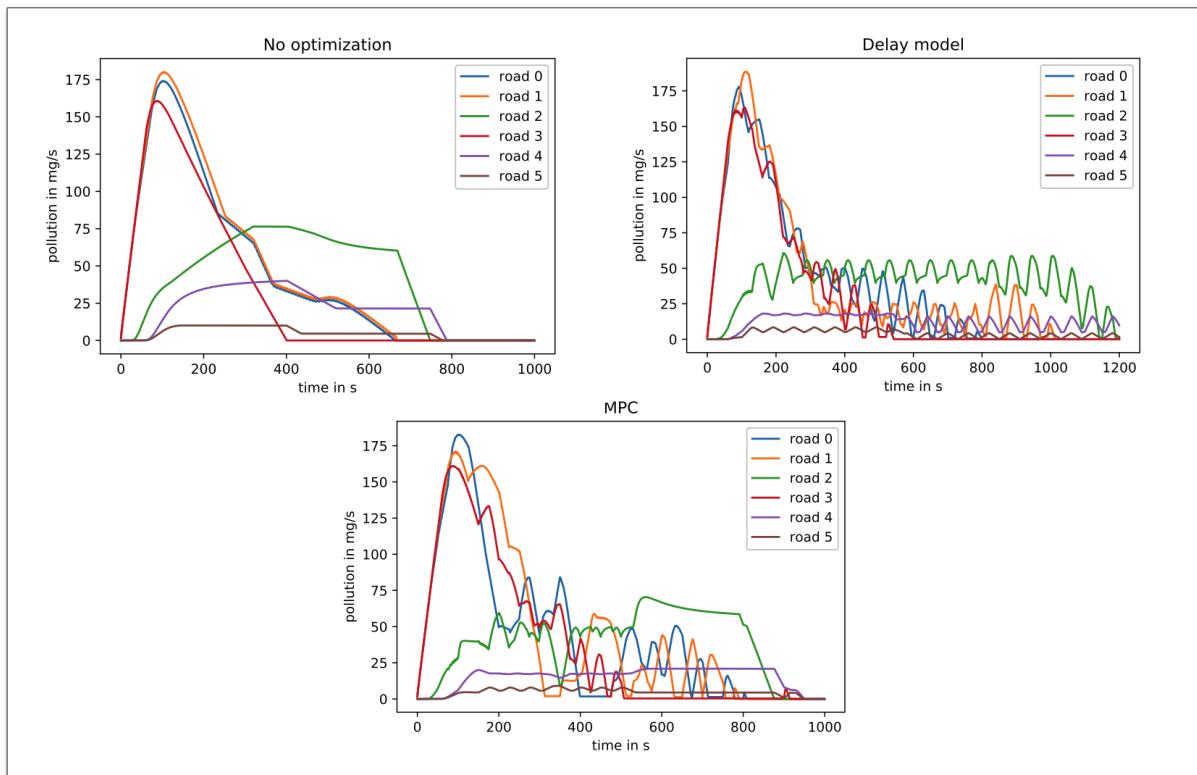


Figure 5.18: Generated pollution on the roads using the buffer model without traffic lights (top left), traffic lights with fixed control strategies (top right) and traffic lights and MPC (bottom).

# Chapter 6

## Discussion and further research

In this thesis the author has provided a numerical scheme for the solution of the buffer model on networks proposed by Bressan [9]. In order to control the traffic flux through junctions the concept of traffic lights has been introduced. For the switching patterns of these traffic controls two optimization schemes have been derived: The first one uses a fixed strategy where the control pattern of a single traffic light is given and successive controls are optimized with respect to the given control. In the second approach a flux functional is derived which is optimized eventually using Model Predictive Control (MPC). The solution obtained from this approach is suboptimal to the global optimum but serves as a reference point for comparison. Finally a novel model to simulate pollution caused by urban traffic is introduced. This model consists of the buffer model coupled with a non-linear pollution function. The results of the pollution model are then compared using data obtained from three optimization schemes of the buffer model: no optimization with traffic lights (signifying the use of roundabouts), traffic lights following a fixed control strategy and optimization via MPC.

The results obtained from the traffic flow simulation on the partial road network of Munich suggest that in terms of flux optimization roundabouts perform better as they permit cars to constantly cross the junction for all incoming and outgoing roads. On the other hand traffic lights can actually reduce the pollution generated at a junction as long as they are controlled by a master controller having sufficient information about the current traffic situation. No generalization can be made about the preference of either traffic lights or roundabouts. Nevertheless, un- or ill-controlled traffic lights offer worse traffic flow as well as higher amounts of emitted pollution, as they do not react to changing traffic situations in the correct manner.

Further research can be conducted towards the general optimization process as the solution obtained by MPC is highly suboptimal to the global optimum. Possible methods of improvement in this area can be particle swarm optimization (PSO) methods in order to iteratively improve candidate solutions with regard to a given measure of quality or convex-concave-procedure (CCCP) methods to deal with the non-convexity of the flux

functional. Additional methods worth mentioning for optimization on a finite control set can be compass search or generic algorithms where controls are changed randomly in order to find an optimal solution. Further refinement can be done in the computation of the generated pollution as acceleration and deceleration processes intensively affect the volume of burnt fuel in the process. This can have high impact on stop-and-go phases for instance caused by traffic lights.

We by no means have tried to solve the global traffic problem as there is no generic solution for every traffic situation. In particular, we want to point out that traffic lights can actually worsen traffic jams and counteract smooth travelling if they are installed without proper control.

# Appendices

# Appendix A

## Solutions to scalar hyperbolic equations

Let us consider the nonlinear scalar conservation law

$$\partial_t \rho + \partial_x f(\rho) = 0$$

with strictly concave and continuously differentiable flux  $f(\rho)$  supplemented by discontinuous initial conditions

$$\rho(x, 0) = \begin{cases} \rho^- & x \leq 0 \\ \rho^+ & x > 0 \end{cases}$$

and the discontinuity located at  $x = 0$ . A problem of this form with  $\rho^-, \rho^+ \in \mathbb{R}$  is called a **Riemann problem**. We now consider three cases:

1. Assume that a traffic light is positioned at  $x = 0$ . We consider the initial condition

$$\rho^- = \rho_{max}, \rho^+ = 0$$

meaning that the road in front of the light is full of cars and empty behind the light. This datum typically corresponds to the situation in which the traffic light shows red and cars in front of the light have to queue. Assume now that the traffic light switches to green at time  $t = 0$ . Then we can describe the resulting evolution by

$$\rho(x, t) = \begin{cases} \rho_{max} & x < f'(\rho_{max})t, \\ (f')^{-1}\left(\frac{x}{t}\right) & f'(\rho_{max}t) < x < f'(0)t \\ 0 & x > f'(0)t \end{cases}$$

For a fixed time  $t > 0$  the solution is equal to  $\rho_{max}$  if  $x < f'(\rho_{max})t$ , meaning that there is still a queue at this point. The density at points  $x > f'(0)t$  on the road vanishes, meaning that no cars have yet reached that point. The density function on the area in between these points is described by a decreasing function which is the effect of progressive acceleration of cars at the green light. This phenomenon is typically referred to as *rarefaction wave*.

2. Let the initial data be such that  $\rho^- < \rho^+$  and set  $s := \frac{f(\rho^-) - f(\rho^+)}{\rho^- - \rho^+}$ . Then the weak solution of the problem is given by

$$\rho(x, t) = \begin{cases} \rho^- & x < st \\ \rho^+ & x > st \end{cases}$$

where  $s \in \mathbb{R}$  denotes the characteristic shock speed. Solutions of this form are called *traveling shock wave*.

We also note that solutions of this form fulfil the entropy condition by Oleinik.

**Definition A.0.1** *A weak solution  $u : \mathbb{R} \times (0, T) \rightarrow \mathbb{R}$  satisfied the entropy condition by Oleinik iff along each discontinuity curve  $x = \psi(t)$  we have*

$$\frac{f(\rho^-) - f(u)}{\rho^- - u} \leq \psi'(t) =: s \leq \frac{f(\rho^+) - f(u)}{\rho^+ - u}$$

for all  $t \in (0, T)$  and  $\rho^- < u < \rho^+$

## Appendix B

# Limitations of the junction model by Garavello

Garavello's approach [26] consists of maximizing the flux through the junction. We now show that under certain circumstances this approach does not produce unique solutions.

For simplicity let us consider a single intersection consisting of 2 incoming and 2 outgoing roads and initial densities  $\rho_k, k = 0, 1, 2, 3$ . Garavello now defines, as Bressan does in the buffer model, the maximum junction fluxes as follows. For an incoming road  $e_i, i = 0, 1$  the maximum influx (supply) is given by

$$\omega_i(\rho_i) = \begin{cases} f(\rho_i) & \rho_i \in [0, \sigma] \\ f(\sigma) & \rho_i \in (\sigma, 1] \end{cases}$$

For an outgoing road  $e_j, j = 2, 3$  the maximum outflux (demand) is given by

$$\omega_j(\rho_j) = \begin{cases} f(\sigma) & \rho_j \in [0, \sigma] \\ f(\rho_j) & \rho_j \in (\sigma, 1] \end{cases}$$

These quantities represent the maximum flux that can be obtained by a single wave solution on each road.

Let us now consider a traffic distribution matrix  $TD \in \mathbb{R}^{2 \times 2}$  with

$$TD = \begin{pmatrix} \alpha_{0,2} & \alpha_{0,3} \\ \alpha_{1,2} & \alpha_{1,3} \end{pmatrix}$$

The goal is to maximize the flux through the junction, so we set

$$\gamma_j^> = \min \{\alpha_{0,j}\omega_0(\rho_0) + \alpha_{1,j}\omega_1(\rho_1), \omega_j(\rho_j)\}$$

where  $\gamma_j^>$  denotes the effective outflow of the junction into outgoing road  $e_j$  consisting of influxes from roads  $e_0, e_1$ .

We observe that every combination of influxes  $(\gamma_0, \gamma_1)$  should fulfill that  $\alpha_{0,j}\gamma_0(\rho_0) + \alpha_{1,j}\gamma_1(\rho_1) \leq \gamma_j^>$ . This concludes that the possible combinations of feasible influx builds a closed convex set  $\Omega$  (see Figure B.1) with

$$\Omega := \{(\gamma_0, \gamma_1) : 0 \leq \gamma_i \leq \omega_i(\rho_i), i = 0, 1, 0 \leq \alpha_{0,j}\gamma_0(\rho_0) + \alpha_{1,j}\gamma_1(\rho_1) \leq \gamma_j^>, j = 2, 3\}$$

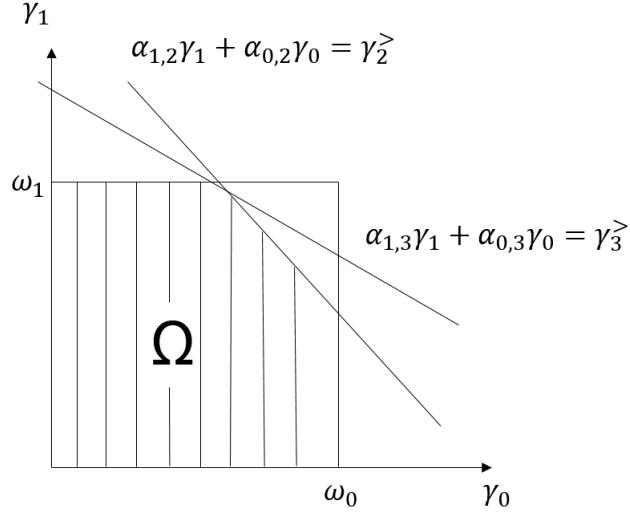


Figure B.1: The feasible set  $\Omega$  for a single junction with two incoming and two outgoing roads.

The final solution  $(>\gamma_0, >\gamma_1)$  is then obtained by optimizing the influx sum over the region  $\Omega$ , i.e.

$$(>\gamma_0, >\gamma_1) = \max_{(\gamma_0, \gamma_1) \in \Omega} \gamma_0 + \gamma_1$$

### Example for non-unique solution

The final optimization step can conflict with the desire of unique solutions. Let us for example consider again a junction with two incoming and two outgoing roads and initial densities  $\rho_i = 0.5, i = 0, 1, 2, 3$ . and regard the traffic distribution matrix

$$TD = \begin{pmatrix} 0.4 & 0.6 \\ 0.4 & 0.6 \end{pmatrix}$$

Then we obtain for the maximum fluxes  $\omega_i(\rho_i) = f(\sigma) = 0.25, i = 0, 1, 2, 3$ . Sketching again the set of feasible solutions we see that we obtain a line of infinitely many optimal solutions (see Figure B.2).

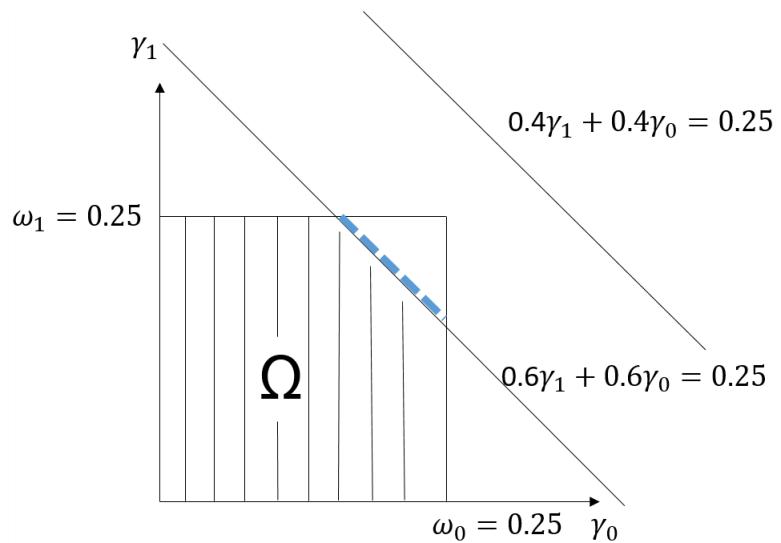


Figure B.2: The feasible set  $\Omega$  for a single junction with two incoming and two outgoing roads. The fat dashed line denotes the set of infinitely many optimal solutions.

# Appendix C

## Finite difference schemes

In order to solve conservation laws of the form (1.2.4) we usually assume smooth initial conditions  $\rho(x, 0)$ . We are naturally interested in the difficulties caused by discontinuities in the solution. Straight-forward numerical methods like the finite difference approximation usually have difficulties near the discontinuity. Consider for instance the scalar (linear) advection equation

$$\begin{aligned} \partial_t u + A \partial_x u &= 0, \quad -\infty < x < \infty, t \geq 0, \\ u(x, 0) &= \begin{cases} 1 & x < 0 \\ 0 & x > 0. \end{cases} \end{aligned} \tag{C.0.1}$$

Obviously the analytical solution obtained by the method of characteristics is given by a travelling shock wave  $u(x, t) = u_0(x - At)$  with wave speed  $A$ . Unfortunately the finite difference approach is inconsistent, as the finite difference approximation to  $\partial_x u$  at the discontinuity  $x = 0$  will not approach 0, e.g.

$$\partial_x u \sim \frac{u_0(At + h) - u_0(At - h)}{2h} = \frac{0 - 1}{2h} \rightarrow -\infty, \text{ as } h \rightarrow 0.$$

In particular, the *local truncation error*

$$L_k(0, t) := \frac{1}{k} [u(0, t + k) - u(0, t)] - a \frac{u_0(At + h) - u_0(At - h)}{2h} \tag{C.0.2}$$

for the finite difference scheme does not vanish for  $h, k \rightarrow 0$  (see also [38]). A numerical method of **order p** fulfills that  $L_k(x, t) = o(k^p)$ . This means that the finite difference scheme for discontinuous data does neither provide local nor global convergence towards the analytical solution.

For linear systems of the form (C.0.1) a wide variety of convergent numerical methods using adjusted finite difference schemes are given. Some possibilites are stated in Table C.1. First order methods (Upwind, Lax-Friedrich) typically produce flattened solutions compared to the original solutions, whereas second order methods (Lax-Wendroff and Beam-Warming) give oscillations. These phenomena are displayed in Figure C.1.

---

<sup>1</sup>Image taken from [38].

Name	Difference equation	Order
Lax-Friedrich	$\begin{aligned} u(x, t + k) &= \frac{1}{2}(u(x + h, t) + u(x - h, t)) \\ &\quad - \frac{k}{2h} A(u(x + h, t) - u(x - h, t)) \end{aligned}$	1
Upwind ( $A > 0$ )	$u(x, t + k) = u(x, t) - \frac{k}{2h} A(u(x, t) - u(x - h, t))$	1
Lax-Wendroff	$\begin{aligned} u(x, t + k) &= u(x, t) - \frac{k}{2h} A(u(x + h, t) - u(x - h, t)) \\ &\quad + \frac{k^2}{2h^2} A^2 (u(x + h, t) - 2u(x, t) + u(x - h, t)) \end{aligned}$	2
	$\begin{aligned} u(x, t + k) &= u(x, t) - \frac{k}{2h} A(3u(xh, t) - 4u(x - h, t) \\ &\quad + u(x - 2h, t)) + \frac{k^2}{2h^2} A^2 ((u(xh, t) \\ &\quad - 2u(x - h, t) + u(x - 2h, t))) \end{aligned}$	2

Table C.1: Finite difference schemes for the linear advection problem (C.0.1).

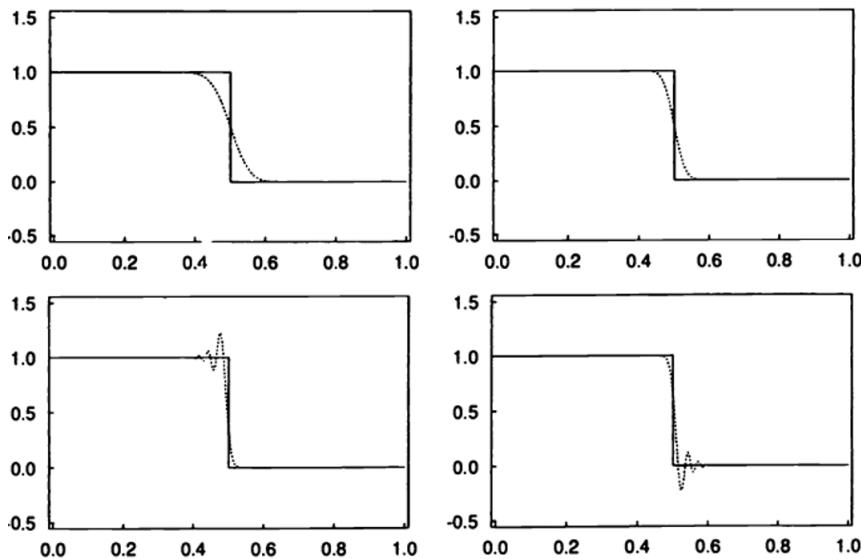


Figure C.1: Numerical and exact solution of conservation law (C.0.1) at  $t = 0.5$  and step size  $h = 0.0025$  using the following methods (top left to bottom right) (a) Lax-Friedrich, (b) Upwind, (c) Lax-Wendroff, (d) Beam-Warming<sup>1</sup>.

# Appendix D

## Traffic modelling toolbox

README file, written 2017-04-12

\*\*\*\*\*

How to use the traffic flow modelling toolbox?

#####

The toolbox permits traffic flow simulation on arbitrary networks using three different approaches:

- buffer\_model.py: buffer model by Bressan without optimization
- buffer\_mode\_delay.py: optimisation via fixed strategy for traffic lights
- MPC\_fixed\_decision\_points: optimisation using Model Predictive Control and fixed minimum green phase

The toolbox also permits computation of generated pollution on the network (without temporal development) via post-processing of the traffic data:

- pollution.py

#####

1. Traffic Flow Modelling:

### 1.1 Parameters

In all functions, several parameters have to be adjusted:

- dt,dx: step sizes in time and space (in s, m) of type double
- vmax: maximum allowed velocity on the road (in m/s) of type double

Important: check that CFL condition is fulfilled

- rhomax: maximum allowed density on the road of type double
- tend: final simulation time (in s) of type Integer
- network: (n x n)-traffic distribution matrix, n number of roads, ndarray

- $_X$ : 2d vector consisting of starting and end point of the roads and points of density discontinuities of initial data
- $_Rho$ : vector of lists with initial densities on the road,  $\dim(_Rho[i])=\dim(_X[i])-1$  for all  $i=1\dots n$
- $_ext\_in$ : n-dim vector of Doubles with incoming densities into the roads
- $M$ :  $n_J$ -dim array of lists of buffer capacities for all junctions, one list consists of buffer capacities for all outgoing roads separately
- $c$ :  $n_J$  dim array of priority values for incoming roads at junctions

only applicable for MPC:

- $\gamma$ : cost of control changes
- $\epsilon$ : weight of quantisation by double-well
- $n_p$ : Integer number of predictive signal intervals
- $n_c$ : Integer number of control signal intervals
- $signal\_horizon$ : number of time steps in one signal interval

## 1.2 Output

The pickle-module is used for storing output. The resulting dictionaries are stored in non-human-readable txt files and need to be imported first for further processing.

The output dictionary consists of the following elements:

- “ $t$ ”: time steps
- “ $x$ ”: space discretization
- “ $\rho$ ”: densities at time steps at every road
- “ $fluxes$ ”: resulting Godunov fluxes at time steps at every road
- “ $total\ flux$ ”: sum of all fluxes
- “ $total\ time$ ”: total computational time needed
- “ $junctions$ ”: dictionary of numbered junctions with following elements:
  - # “ $in$ ”: list of indexes of incoming roads
  - # “ $out$ ”: list of indexes of outgoing roads
  - # “ $matrix$ ”: traffic distribution matrix of this junction,  $(n_{in} \times n_{out})$ -dim
  - # “ $M$ ”: buffer capacity
  - # “ $c$ ”: priority vector for incoming roads
  - # “ $Buffer$ ”: list of filling rates of the buffer at time steps
- only for MPC and fixed strategy optimisation:
  - # “ $\eta$ ”: initial control
- Only for MPC and fixed strategy:
  - “ $controls$ ”: control vector
- Only for MPC
  - “ $feval$ ”: number of function evaluations

### 1.2.1 Buffer model:

Output file is named “buffer\_data.txt”

### 1.2.2 Fixed strategy:

Output file is named “buffer\_delay.txt”

### 1.2.3 MPC:

```
Output file is named 'output_buffer_dp_{}_{ }_{ }_{ }.txt',
    .format(signal_horizon,n_p,n_c,gamma)
```

## 1.3 Import

To import a generated text file of name filename use pickle module in the following:

```
with open(filename, 'rb') as handle:
    data = pickle.load(handle)
```

The data can be accessed by dict commands, e.g. X= data[‘x’]

## 1.4 Functions

### 1.4.1 General Functions

- velo(rho): computes density-dependent velocity
- f(rho): fundamental diagram
- init(x,rho,dx): generates initial setup on the network
- godunovFlux(f,rhol,rhor): computes flux between two cells
- godunovFluxes(f, rho): computes Godunov fluxes on a road
- getMaxInflux(f, rho): computes maximum influx into junction based on equation 2.3.5a
- getMaxOutflux(f, rho): computes maximum outflux of junction based on equation 2.3.5b
- getIncomingFluxesSB(Buffer, maxIn, junction): computes junction influxes by current buffer level using SBJ
- getIncomingFluxesMB(Buffer, maxIn,junction): computes junction influxes by current buffer level using MBJ
- getOutgoingFluxes(Buffer,maxOut,incomingFluxes): computes junction outfluxes
- updateBuffer(junction,incomingFluxes,outgoingFluxes): computes changes in the buffer level using junction fluxes
- godunovStep(fluxes,rho): computes one Godunov step based on equations 3.1.12

- `network2junctions(network, c,M,mode)`: builds junction dictionary based on transition matrix
- `network2Adjescency(network)`: builds adjacency matrix based on transition matrix
- `getJunctionFluxes(Buffer,rho_in,rho_out,mode)`: computes incoming and outgoing junction fluxes
- `getOverallFlux(fluxes)`: computes total flux based on flux array
- `plot2D(x,rho)`: plots densities at time tend on all roads
- `plot2DAtTime(x, rho,s)`: plots densities at times t in s, s list of time points
- `plotControls(control,t,index)`: plots controls of junction index
- `plotBuffers(t, junctions)`: plots buffer levels over time for all junctions
- `plotTotalDensity(t,rho)`: plots total density on the network over time
- `doAnimationNew(x, rho,filename)`: produces .mp4 file of the dynamics

#### 1.4.2 Buffer specific functions

- `solveArbitraryNetwork(tend, network, c,M, initX,initRho,ext_in)`: main function

#### 1.4.3 Fixed strategy specific functions

- `getJunctionFluxesWithDelay(rho_in,rho_out,junction,tau,t,mode)`: computes junction fluxes based on a signal delay of tau
- `solveNetworkWithDelay(tend, network,_c,_M,initX,initRho,ext_inflow)`: main function

#### 1.4.4 MPC specific functions

- `dynamicEvolution(rho_init,control,t0,n_p)`: simulated the dynamics of the system over the predictive horizon
- `MPC_functional(u,t0)`: functional used for minimization
- `adjescency2constraint`: computes inequality constraints for controls based on feasibility condition 4.0.1
- `solveNetworkWithMPC(_X,_Rho, network)`: main function

## 2. Pollution modeling

### 2.1 Parameters

- `buffer_file`: path to output file of the buffer model
- `delay_file`: path to output file of fixed strategy model
- `mpc_file`: path to output file of mpc model
- `vmax`: maximum velocity for pollution plot
- `vmax_2`: maximum velocity allowed on network
- `psi_0`: fuel consumption in idle mode (in l/h)

## 2.2 Functions

- `vel(rho)`: computes velocity from density
- `getFuelConsumptionByDensity(rho)`: computes fuel consumption given the density at some point
- `getPollutionOnNetwork(rho_vector)`: computes pollution given the density on the network
- `getBufferPollution(junctions)`: computes pollution generated at the junctions

## 2.3 Output

- pollution distribution on roads
- total pollution

# Glossary

## Abbreviations

BFGS	Broyden-Fletcher-Goldfarb-Shanno
CCCP	concave-convex-procedure
CFC	chlorofluorocarbons
CFL	Courant-Friedrichs-Lowy
CO	carbon oxide
CO <sub>2</sub>	carbon dioxide
CVN	cumulated vehicle number
DOF	degrees of freedom
FD	finite differences
FEM	finite element method
IVBP	initial value boundary problem
LP	linear programming
LWR	Lighthill-Whitham-Richards
MBJ	multiple buffer junction
MGPOCP	minimum green phase optimal control problem
MPC	model predictive control
NO <sub>2</sub>	nitrogen dioxide
ODE	ordinary differential equation
OCP	optimal control problem
PDE	partial differential equation

PM	particulate matter
RS	Riemann solver
SBJ	Single buffer junction
SLSQP	Sequential least squared programming
TL	traffic light

## Notations

$\mathbb{R}$	set of real numbers
$\rho_{max}$	maximum density allowed on the road
$\sigma$	maximum flux density
$f^{max}$	maximum flux allowed on the network
$v_{max}$	maximum velocity allowed on the road
$e_i$	road with index $i$
$a_i$	start point of road $e_i$
$b_i$	end point of road $e_i$
$\mathcal{E}$	set of all edges of a network
$\mathcal{N}$	set of all junctions of a network
$n_R$	number of roads in a network
$n_J$	number of junctions in a network
$\delta^{in}(J)$	set of roads incoming to junction $J$
$\delta^{out}(J)$	set of roads outgoing of junction $J$
$\mathcal{E}^{in}$	incoming roads of the network
$\mathcal{E}^{out}$	outgoing roads of the network
$\Theta$	transition matrix
$\Theta_{i,j}$	probability of leaving a junction towards road $e_j$ coming from road $e_i$

## Variables

$\rho(x, t)$	traffic density at time $t$ at location $x$
$f(x, t)$	traffic flux at time $t$ at location $x$
$f(\rho)$	fundamental diagram
$f(t, x, v)$	probability function for the mesoscopic model describing the density probability of a car located at $x$ at time $t$ and velocity $v$
$\rho_0(x)$	initial density
$v$	velocity
$t$	time variable
$\phi$	test function
$\gg f_k$	junction inflow of road $e_k$
$f_k \gg$	junction outflow into road $e_k$
$q_j(t)$	queue length in front of road $e_j$
$\bar{\Theta}_{i,j}(t)$	limit of the driver's preference
$\bar{\rho}_i$	traffic density on road $e_i$ in front of the junction
$\hat{\rho}_j$	traffic density on road $e_j$ behind the junction
$\omega_i$	maximum provided influx of road $e_i$ into the junction
$\omega_j$	maximum demanded outflux of the junction into road $e_k$
$c_i$	priorities given to incoming roads of junctions
$M$	maximum capacity of a single buffer
$h$	step size of the discretization in space

$k$	step size of the discretization in time
$x_{j+\frac{1}{2}}$	midpoint of the $j$ -th cell of the space discretization
$u_j^n$	average density on the $j$ -th cell at time $t_n$
$F(u_j^n, u_{j+1}^n)$	numerical flux function
$f(u^*(u_j, u_{j+1}))$	exchanged flux between the $j$ -th and $j + 1$ -th cell
$C$	Courant number
$s$	shock speed
$G$	Godunov scheme
$\eta_i$	influx control for road $e_i$
$T$	mean travel time
$Q_{in}$	total inflow
$q_0(t)$	inflow at time $t$
$M_{total}$	mass of a fixed density package
$\tau$	delay used for traffic coordination
$\tau^*$	optimal delay
$\eta_C$	given reference control pattern
$F_T$	total flux on the network over the time horizon $[0, T]$
$F_T^*$	optimal flux over the time horizon $[0, T]$
$n_P$	predictive horizon used for MPC
$n_C$	control horizon used for MPC

$F_{n_C}$	optimal flux obtained during one MPC step
$R$	regularization terms
$F_{MPC}$	value of the target functional for a single MPC step
$W_J$	multi-well function used for regularization
$K_J$	feasible configuration of controls at junction $J$
$\epsilon$	weight for $W_J$
$\gamma$	weight for the total variation
$u_{bin}^*$	optimal binary control sequence
$Q(\eta)$	function used for quantization of a control signal
$\eta_{bin}$	quantized control sequence
$u_{bin}^* _{n_C}$	optimal quantized control sequence restricted to the control horizon
$V_\infty$	global optimum of the flux functional
$n_{TL}$	number of traffic lights in the network
$n_{sh}$	signal horizon; number of discretization points per phase interval
$d_i$	decision points
$D_i$	phase interval
$\mathcal{D}$	set of all phase intervals
$n_D$	number of decision points
$T_{min}$	minimum green phase time
$k_P$	number of predictive phases
$k_C$	number of control phases

$P$	pollution on the road over a certain time horizon
$FC_c$	fuel consumption of the car in cruising mode
$FC_i$	fuel consumption of the car in idle mode
$\alpha$	conversion rate of fuel to carbon oxide
$\psi$	velocity dependent function for fuel consumption
$v_{opt}$	velocity for (locally) minimal pollution
$P_e$	pollution on road $e$
$P_J$	pollution on junction $J$
$P_{network}$	total pollution generated on the network
$M_J$	total buffer capacity of junction $J$
$Q_J$	current buffer level of junction $J$
$\nu_i$	mapping of road $e_i$ onto the 2D area
$A_i$	avenue on the 2D area
$TD_J$	traffic distribution matrix at junction $J$
$T_{end}$	total transport time needed from a starting distribution until the network is empty again

# Bibliography

- [1] L. Alvarez-Vázquez, N. García-Chan, and A. Martínez. Numerical simulation of air pollution related to traffic flow in urban networks. 2016.
- [2] A. Aw and M. Rascle. Resurrection of second order models of traffic flow. *SIAM Journal on Applied Mathematics*, 60(3):916–938, 2000.
- [3] F. Banaei-Kashani, C. Shahabi, and B. Pan. Discovering patterns in traffic sensor data. *IWGS '11 Proceedings of the 2nd ACM SIGSPATIAL International Workshop on GeoStreaming*, pages 10–16, 2011.
- [4] N. L. Biggs. *Algebraic graph theory*. Cambridge mathematical library. Cambridge Univ. Press, Cambridge, 2. ed. edition, 1993.
- [5] C. G. E. Boender, Rinnooy Kan, A. H. G., G. T. Timmer, and L. Stougie. A stochastic method for global optimization. *Mathematical Programming*, 22(1):125–140, 1982.
- [6] D. Bowyer, R. Akçelik, and D. Biggs. Guide to fuel consumption analysis for urban traffic management. *Special Report SR*, (32), 1985.
- [7] S. Brenner and R. Scott. *The Mathematical Theory of Finite Element Methods*. Texts in Applied Mathematics. Springer New York, 2007.
- [8] A. Bressan. Hyperbolic conservation laws: An illustrated tutorial. *Rendiconti del Seminario Matematico della Università di Padova*, pages 217–235, 2009.
- [9] A. Bressan. Conservation law models for traffic flow on a network. 2014.
- [10] A. Bressan and K. Han. Existence of optima and equilibria for traffic flow on networks. *Networks and Heterogeneous Media*, 8(3):627–648, 2013.
- [11] A. Bressan and F. Yu. Continuous riemann solvers for traffic flow at a junction. submitted, 2014.
- [12] G. Bretti, M. Briani, and E. Cristiani. An easy-to-use algorithm for simulating traffic flow on networks: Numerical experiments. *Discrete and Continuous Dynamical Systems - Series S*, 7(3):379–394, 2014.

- [13] D. J. Buckley. A semi-poisson model of traffic flow. *Transportation science : the publication of the Transportation Science Section, Operation Research Society of America*, 2 (1968):107–133, 1968.
- [14] N. Cardoso and J. M. Lemos. Model predictive control of depth of anaesthesia: guidelines for controller configuration. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, 2008:5822–5825, 2008.
- [15] E. Casas. Pontryagin’s principle for state-constrained boundary control problems of semilinear parabolic equations. *SIAM Journal on Control and Optimization*, 35(4):1297–1327, 1997.
- [16] B. D. Coensel, A. Can, B. Degraeuwe, I. D. Vlieger, and D. Botteldooren. Effects of traffic signal coordination on noise and air pollutant emissions. *Environmental Modelling & Software*, 35:74–83, 2012.
- [17] R. M. Colombo, P. Goatin, and M. D. Rosini. On the modelling and management of traffic. *ESAIM: Mathematical Modelling and Numerical Analysis*, 45(5):853–872, 009 2011.
- [18] Council of European Union. Council regulation (EU) no 2016/646: Emissions from light passenger and commercial vehicles (EURO 6), 2016. Available at <http://data.europa.eu/eli/reg/2016/646/oj>, Accessed: 2017-03-15.
- [19] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen differenzengleichungen der mathematischen physik. *Mathematische Annalen*, 100(1):32–74, 1928.
- [20] C. M. Dafermos. Polygonal approximations of solutions of the initial value problem for a conservation law. *Journal of Mathematical Analysis and Applications*, 38, 1972.
- [21] S. S. E. Cristiani. On the micro-ta-macro limit for first-order traffic flow models on networks. *arXiv:1505.01372 [math.DS]*, 2015.
- [22] J. Eckstein and D. P. Bertsekas. On the douglas—rachford splitting method and the proximal point algorithm for maximal monotone operators. *Mathematical Programming*, 55(1-3):293–318, 1992.
- [23] EPA Air Emission Inventories. Air emission sources - national summary of carbon monoxide emissions, 2014. Available at: [https://www3.epa.gov/cgi-bin/broker?polchoice=C0&\\_debug=0&\\_service=data&\\_program=dataprogs.national\\_1.sas](https://www3.epa.gov/cgi-bin/broker?polchoice=C0&_debug=0&_service=data&_program=dataprogs.national_1.sas). Accessed: 2017-04-02.
- [24] T. Erez, K. Lowrey, Y. Tassa, V. Kumar, S. Kolev, and E. Todorov. An integrated system for real-time model predictive control of humanoid robots. In *Humanoid*

*Robots (Humanoids), 2013 13th IEEE-RAS International Conference on*, pages 292–299. IEEE, 2013.

- [25] L. C. Evans. *Partial differential equations*, volume Volume 19 of *Graduate studies in mathematics*. American Mathematical Society, Providence, Rhode Island, second edition, reprinted with corrections edition, 2015.
- [26] M. Garavello and B. Piccoli. *Traffic flow on networks: Conservation laws model*, volume v. 1 of *AIMS series on applied mathematics*. American Institute of Mathematical Sciences, Springfield, MO, 2006.
- [27] S. K. Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.
- [28] L. Grüne and J. Pannek. *Nonlinear model predictive control: Theory and algorithms*. Communications and Control Engineering. Springer, Cham, Switzerland, second edition edition, 2017.
- [29] Göttlich, S. and Herty, M. and Ziegler, U. Modeling and optimizing traffic light settings in road networks. *Computers & Operations Research*, 55:36 – 51, 2015.
- [30] Göttlich, S. and Potschka, A. and Ziegler, U. Partial outer convexification for traffic light optimization in road networks. *AMS*, 2014.
- [31] A. Hegyi. *Model Predictive Control for Integrating Traffic Control Measures*. PhD thesis, Technische Universiteit Delft, 2004.
- [32] S. Hein. *MPC/LQG-Based Optimal Control of Nonlinear Parabolic PDEs*. PhD thesis, TU Chemnitz, 2009.
- [33] M. Herty and A. Klar. Modeling, simulation, and optimisation of traffic flow networks. *SIAM J. Sci. Comput.*, 25(3), 2003.
- [34] P. Hunt, D. Robertson, R. Winton, and R. Bretherton. Scoot-a traffic responsive method of coordinating signals. *Technical report*, 1981.
- [35] B. Kelly. A ‘green wave’ reprieve. *Traffic Engineering and Control*, 53(2):55–58, 2012.
- [36] D. Kraft. *A software package for sequential quadratic programming*, volume 88-28 of *Forschungsbericht / Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt*. DFVLR, Köln, als ms. gedr edition, 1988.
- [37] V. Kumar, Y. Tassa, T. Erez, and E. Todorov. Real-time behaviour synthesis for dynamic hand-manipulation. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6808–6815. IEEE, 2014.

- [38] R. J. LeVeque. *Numerical methods for conservation laws*. Lectures in mathematics. Birkhäuser, Basel, 2. ed. edition, 1992.
- [39] P. Lowrie. Scats, sydney coordinated adaptive traffic system: a traffic responsive method of controlling urban traffic. *Technical report*, 1990.
- [40] M. Madireddy, B. D. Coensel, A. Can, B. Degraeuwe, B. Beusen, I. D. Vlieger, and D. Botteldooren. Assessment of the impact of speed limit reduction and traffic signal coordination on vehicle emissions using an integrated approach. *Transportation Research Part D: Transport and Environment*, 16(7):504 – 508, 2011.
- [41] R. Mahnke and R. Kühne. Probabilistic description of traffic breakdown. *Traffic and granular flow '05*, pages 527–536, 2007.
- [42] T. V. Mathew. Fuel consumption and emission studies. [https://www.civil.iitb.ac.in/tvm/1111\\_nptel/583\\_FuelEmi/plain/plain.html](https://www.civil.iitb.ac.in/tvm/1111_nptel/583_FuelEmi/plain/plain.html), 2014. Accessed: 2017-03-03.
- [43] A. D. May. *Traffic flow fundamentals*. Prentice Hall, Englewood Cliffs, N.J., 1990.
- [44] P. Mazaré, C. Claudel, and A. Bayen. Analytical and grid-free solutions to the lighthill-whitham-richards traffic flow model. submitted to Elsevier, 2010.
- [45] L. Modica and S. Mortola. Il limite nella  $\gamma$ -convergenza di una famiglia di funzionali ellittici. *Boll. Un. Mat. Ital. A* (5), 14(3):526–529, 1977.
- [46] M. Monache, B. Piccoli, and F. Rossi. Traffic regulation via controlled speed limit. *arXiv:1603.04785*, 2016.
- [47] H. J. Payne. *Models of freeway traffic and control*. Simulation Councils, Inc, La Jolla, Calif., 1971.
- [48] L. A. Pipes. *An operational analysis of traffic dynamics*, volume no. 25 of *Institute of Transportation and Traffic Engineering*. Institute of Transportation and Traffic Engineering, University of California, [Berkeley, Calif.], 1953.
- [49] J. Rawlings and D. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Pub., 2009.
- [50] S. Rezvanian, F. Towhidkhah, N. Ghahramani, and A. Rezvanian. Increasing robustness of the anesthesia process from difference patient's delay using a state-space model predictive controller. *Procedia Engineering*, 15:928–932, 2011.
- [51] P. I. Richards. *Shock waves on the highway*, volume 4. 1956.
- [52] RIITS. Regional integration of intelligent transportation systems, 2006. <http://www.riits.net>, Accessed: 2017-03-05.

- [53] Y. N. Skiba and V. Davydova-Belitskaya. On the estimation of impact of vehicular emissions. *Ecological Modelling*, 166(1-2):169–184, 2003.
- [54] Y. N. Skiba and D. Parra-Guevara. Control of emission rates. *Atmósfera*, 26(3):379–400, 2013.
- [55] B. Sportisse. *Fundamentals in Air Pollution: From Processes to Modelling*. Springer Netherlands, 2009.
- [56] M. Stachl. Traffic flow optimization on networks, 2017. [https://github.com/mstachl/Traffic\\_flow\\_optimization](https://github.com/mstachl/Traffic_flow_optimization).
- [57] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Num. Anal.*, 5:506–517, 1968.
- [58] A. Tiwary and J. Colls. *Air pollution: Measurement, modelling and mitigation*. Routledge, London and New York, 3rd ed. edition, 2010.
- [59] Umweltbundesamt. Nationale Trendtabellen für die deutsche Berichterstattung atmosphärischer Emissionen seit 1990, Emissionsentwicklung 1990 bis 2014 (Stand 03/2016), 2016. Available at: [http://www.umweltbundesamt.de/sites/default/files/medien/384/bilder/dateien/3\\_tab\\_emi-ausgew-luftsadst\\_2016-05-03.pdf](http://www.umweltbundesamt.de/sites/default/files/medien/384/bilder/dateien/3_tab_emi-ausgew-luftsadst_2016-05-03.pdf). Accessed: 2017-04-02.
- [60] F. Wageningen-Kessels, H. van Lint, K. Vuik, and S. Hoogendoorn. Genealogy of traffic flow models. *EURO Journal on Transportation and Logistics*, 4(4):445–473, 2015.
- [61] L. Wu, Y. Ci, J. Chu, and H. Zhang. The influence of intersections on fuel consumption in urban arterial road traffic: A single vehicle test in harbin, china. *PLOS ONE*, 10(9):1–10, 09 2015.
- [62] A. Yuille and A. Rangarajan. The concave-convex procedure (cccp). *Neural Computation*, 15(4):915–936, 2002.