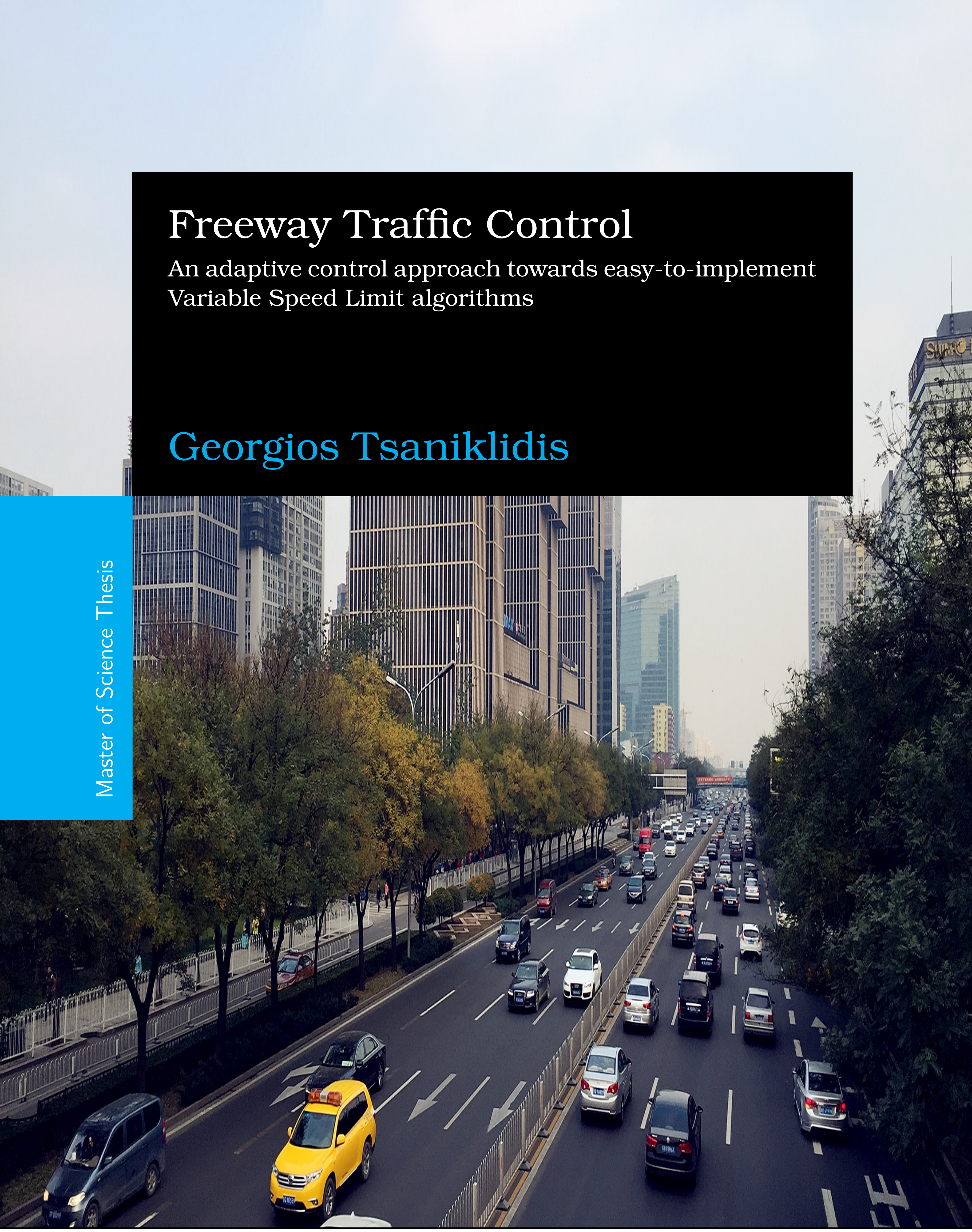

Freeway Traffic Control

An adaptive control approach towards easy-to-implement Variable Speed Limit algorithms

Georgios Tsaniklidis

Master of Science Thesis

Delft University of Technology

Delft Center for Systems and Control

Freeway Traffic Control

An adaptive control approach towards easy-to-implement Variable Speed Limit algorithms

Master of Science Thesis

For the degree of Master of Science in Systems and Control at Delft University of Technology

Georgios Tsaniklidis

January 2, 2023

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

Abstract

Nowadays, the high demand for road transportation often reaches a point where it exceeds the capacity of freeway traffic networks, resulting in congestion. Freeway traffic congestion is a major social problem, as it is the reason for increased time delays, higher accident risk and environmental pollution. There is, therefore, the need for a sustainable solution that can be implemented on the existing road infrastructure. Freeway traffic control is considered as such a solution. It uses different control measures, in order to improve the performance of the freeway traffic network, by influencing the drivers' behaviour.

The Variable Speed Limits (VSLs) are a traffic control measure that aims to increase traffic safety, improve traffic flow and reduce the environmental pollution. Towards the improvement of the freeway traffic flow, easy-to-implement VSL control algorithms are used as mainline metering control approach.

Two easy-to-implement VSL algorithms are reported, namely the Mainstream Traffic Flow Control (MTFC) and the Logic-Based control algorithm for Variable Speed Limits (LB-VSL). The algorithms are usually implemented in an non-adaptive framework. The main contribution of this thesis is that it proposes an adaptive framework for both algorithms, in which the critical density of the freeway traffic network at bottleneck's location is estimated on-line. This estimated critical density is used to adjust the controllers' parameters in real-time.

Three different estimation methods for the bottleneck's critical density are studied, namely the Parameter Estimation (Parameterschatter) method, the Simple Derivative Estimation (SDE) and the Kalman-Filter-based derivative Estimation (KFE) methods. All three methods focus on the real-time estimation of the derivative of the Fundamental Diagram (FD), in order to produce estimations of the critical density.

A case study is performed to evaluate the performance of the three algorithms. A hypothetical 12 km long freeway stretch is used, which contains two VSLs installations and a lane-drop bottleneck. In the first part of the case study an accident scenario is studied, which decreases the critical density at the bottleneck's location. The second part of the case study evaluates the three adaptive easy-to-implement VSL algorithms under the assumption of a decrease of the critical density across the whole network, due to bad weather conditions.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank Prof.dr.ir. Bart De Schutter for his feedback whenever it was needed and of course José Ramón Domínguez Frejo for his comments, advice, suggestions and support during my research.

Also, I would like to thank my parents Manoli, Alexia, and my sister Margarita, because they always supported me during my studies in every possible way and they had faith in my dreams, sometimes more than myself.

This thesis document is successfully completed, because my future wife Konstantina created the best conditions in my life to be able to focus on my research and the writing of the final report document.

Finally, I would like to thank all my friends from Athens, Crete and Delft, who were by my side and were always trying to make me feel capable of fulfilling my master degree. Thank you all, I am very lucky to have you in my life.

Delft, University of Technology
January 2, 2023

Georgios Tsaniklidis

To my family.

Chapter 1

Introduction

1-1 Overview and Motivation

One of the major problems plaguing today's societies and causing serious problems on people's quality of life, as well as on the environment, is freeway traffic congestion. It is an un-deniable fact that both the number of road users and the need for transportation grows systematically the last decades, leading, at least during rush hours, the demand to exceed the capacity of the existing infrastructure around metropolitan areas. Some negative cited examples of traffic congestion on freeways are the increased time delays, the higher accident risk and the waste of fuel, with the latter increasing the emissions of polluting gases into the atmosphere. According to the European Commission [10], "CO2 emissions from transport would remain one third higher than their 1990 level by 2050 and congestion costs will increase by about 50% by 2050".

An environmental-friendly and cost-effective solution to address the aforementioned problem is the freeway traffic network's dynamic traffic control. This work focuses on improving the performance of the freeway traffic network by means of maximum throughput by utilizing the Variable Speed Limits (VSLs) as appropriate traffic control measure.

The application of the VSLs contributes to the reduction of the traffic congestion by moderating the capacity drop phenomenon that reduces the flow at bottleneck locations. To achieve that, the VSLs limit the density at bottleneck locations, by decreasing the mainline flow using appropriate control algorithms. From the available VSL algorithms [20],[13],[12] this thesis focuses on easy-to-implement algorithms since their parameters are tuned with ease and are more suitable for real applications. In particular, the Mainstream Traffic Flow Control (MTFC) [4],[5] and the Logic-Based control algorithm for Variable Speed Limits (LB-VSL) [11], [16] algorithms are preferred since they carry the advantages of the other VSL schemes as well as they show robustness and a performance close to optimal one. The MTFC and LB-VSL algorithms improve the traffic efficiency by maximizing the traffic network's outflow. This is achieved by trying to maintain the density at the bottleneck location

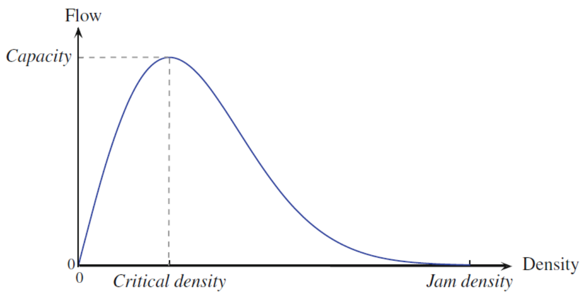

Figure 1-1: Example of fundamental diagram of traffic [8]

close to the critical density. However, the value of the critical density at the bottleneck location changes due to bad weather conditions and/or accidents. Thus, the on-line estimation of the critical density is expected to have a positive impact on the VSL algorithms.

To this end, three different estimation methods are put into work: the Parameter Estimator (PE) [23], the Simple Derivative Estimation (SDE) [32] and the Kalman-Filter-based derivative Estimation (KFE) [32],[24],[22] . Furthermore, some parameters of both controllers are updated on-line to increase their performance. Lastly, a critical comparison between the two control algorithms w.r.t. the aforementioned estimation methods is performed.

1-2 Freeway Traffic Control

1-2-1 Aim of freeway traffic control

The goal of freeway traffic control can be the reduction of the emissions produced by the usage of the freeway network, the increase of the network's performance in terms of throughput maximization or the increase of safety and the reduction of the accidents' rate. In this thesis, the objective of traffic control is to reduce the congestion and increase network's throughput.

In order to achieve traffic congestion reduction, first some basic notions of freeway traffic modeling are required. To describe the dynamics of freeway traffic in this thesis, traffic flow is expressed at an aggregate level, as collective vehicular flow. This means that a macroscopic viewpoint is used to describe the evolution of freeway traffic. For this representation, three aggregated variables are used to describe the traffic state of a network's section, namely the density ρ (expressed in $\frac{veh}{km \cdot lane}$), the flow q (expressed in $\frac{veh}{h}$) and the mean speed v (expressed in $\frac{km}{h}$).

1-2-2 Fundamental Diagram (FD)

The steady-state relationship between flow and density (or speed and density) is modelled using the Fundamental Diagram (FD). The FD represents the theoretical relation between the traffic flow q and density ρ at a given section of the freeway, in case of *homogeneous* and *stationary* traffic conditions [8], [2]. The FD is a curve with a characteristic shape, which can vary for each freeway section. An example of FD, which was initially described in

[18], can be seen in Figure 1-1. The FD differs from the flow-density diagram of the traffic network, which is obtained from measurements, since from measured data non-stationary traffic conditions of heterogeneous vehicles are obtained. However, the FD can be fitted to flow-density measurements, in order to obtain the traffic network's unknown parameters of the capacity, critical density and jam density.

Capacity is defined as the maximum value of the traffic flow that can be reached at a freeway section. The capacity is reached for a density called *critical density*. If the density is increased further, then the traffic flow decreases. The density at which the traffic flow becomes zero again is called *jam density*.

1-2-3 Capacity drop

A bottleneck is a location in the freeway, where congestion starts. A bottleneck can be active or inactive. A drop in the capacity occurs, whenever the bottleneck is activated. This means that the traffic flow exiting the bottleneck location is lower than its capacity, which was measured before the activation of the bottleneck. The capacity drop is defined as the difference between the free-flow capacity and the capacity of the bottleneck after its activation [8],[6].

To avoid the capacity drop phenomenon, specific control actions can be applied to the freeway traffic network, which can limit the density at the bottleneck, in order to keep this density below the critical one. The density's limitation can be achieved by the application of one or more traffic control measures.

1-2-4 Traffic control measures

The most common control measures used in dynamic traffic control applications are the *ramp metering* and the *variable speed limits*.

Ramp Metering (RM) is the most well-studied and commonly applied traffic control measure in freeways. RM controls, using a traffic light, the flow of vehicles that want to enter the freeway from an on-ramp. The amount of vehicles per time unit is determined by carefully selected red, green and amber light timings.

Variable Speed Limits (VSLs) is a freeway control measure, used to limit the maximum speed at the mainline, in order for specific performance, safety, or environmental criteria to be satisfied.

1-2-5 Variable Speed Limits (VSLs)

The working principle of the VSL systems depends on the objectives they have to fulfil, which are the safety improvement, the traffic flow improvement and the reduction of the environmental impact.

It can be easily understood that speed reduction on freeways can be associated with an improvement in safety [9],[31]. Less accidents happen when speeds are lower, both in adverse traffic conditions or extreme weather conditions. Also, VSLs may contribute in reducing collisions, when they are used as a means of warning in case of an incident happening nearby.

Furthermore, VSLs play an important role in traffic flow improvement. This can be achieved by preventing a traffic breakdown on a bottleneck, by temporarily metering the traffic flow in the mainline.

1-3 Contribution

This thesis aims to improve the performance of the easy-to-implement VSL algorithms by incorporating the effect of the on-line estimation of the critical density in their implementation. In particular, the adaptivity is accomplished by rendering the controller's gain dependence on the estimated critical density. The latter is obtained via Parameter Estimator (PE), Simple Derivative Estimation (SDE) and Kalman-Filter-based derivative Estimation (KFE). The control performance of all three methods is studied and compared under two realistic scenarios related with traffic jams, namely under rain and accident conditions.

1-4 Thesis Outline

The thesis is structured as follows. Chapter 2 presents the relevant literature on adaptive control for easy-to-implement Variable Speed Limit (VSL) algorithms. Chapter 3 proposes six adaptive, easy-to-implement VSL control algorithms for freeway traffic control. Chapter 4 presents a case study that evaluates the performance of the proposed adaptive algorithms. Finally, Chapter 5 concludes this thesis, discusses the obtained results and suggests areas for future work.

Chapter 2

Literature Review: Adaptive control for easy-to-implement Variable Speed Limit algorithms

This chapter outlines the related work on the application of Adaptive control for easy-to-implement Variable Speed Limit (VSL) algorithms. The reader is expected to be familiar with the basic notions of VSL algorithms for freeway traffic networks and the on-line estimation methods of critical density or occupancy at bottleneck's location. The chapter is structured as follows. Section 2-1 outlines the different approaches for freeway traffic modeling. Then, in section 2-2, the METANET model is analyzed. In section 2-3, VSL modeling is discussed. Subsequently, section 2-4 outlines two easy-to-implement VSL algorithms. In Section 2-5, adaptive control in freeway traffic control is discussed and in 2-6 the most well-studied adaptive control algorithms for RM are analyzed. Section 2-7 presents three different estimation methods for the derivative of the FD. In section 2-8, some conclusions are drawn and, finally, in section 2-9, the problem statement for this work is given.

2-1 Modeling of freeway traffic

A reliable mathematical model of traffic is a prerequisite to process traffic data reliably and accurately estimate or predict undesirable traffic conditions. Moreover, traffic flow modeling is essential during the design and the test phase of a control strategy.

Traffic models can be classified, with respect to the *level of detail* that they use to describe the dynamic behaviour of the traffic system, into microscopic, mesoscopic and macroscopic.

- Microscopic models describe the dynamics of each vehicle in the traffic stream and how it interacts with other vehicles and road infrastructure.

- Mesoscopic models use probability distribution functions to represent the behaviour of the drivers individually and an aggregate way to describe the changes in the traffic flow.
- Macroscopic models describe traffic flow at an aggregate level, using mean speed, flow and density as variables. These models can be further classified in *first-order*, *second-order* and higher order, according to the number of state variables they use.

Macroscopic traffic models are suitable for real-time, model-based control, because they have a relatively low computational complexity. They can also capture enough dynamics for real-time traffic control. Hence, in the majority of control application, macroscopic traffic models are used. The rest of this work uses a macroscopic modelling approach as well. In the following section an overview of macroscopic traffic modelling is given.

2-1-1 Macroscopic traffic flow models

Advantages of macroscopic traffic flow models

As already mentioned, macroscopic models are more suitable for control applications, compared to microscopic or mesoscopic models. Macroscopic models are less computationally intensive than microscopic models, which provides the benefit of fast simulation of the freeway network. This advantage of macroscopic traffic models can be useful in case of on-line predictive control applications. Furthermore, the cost function used in case of macroscopic models contains terms which are expressed using aggregate variables. That makes the cost that corresponds to a specific traffic state easy to be computed. A macroscopic traffic model has often fewer parameters than a microscopic model to estimate. Thus, the identification and the calibration of the traffic model is simpler. In addition, macroscopic models are expressed as state space models, which makes control design more intuitive and applicable.

Macroscopic models in freeway traffic flow modeling

Macroscopic traffic models can be differentiated between first-order and second-order, depending on the number of aggregate variables they use to describe the traffic dynamics. In the related literature, the most commonly used first-order traffic model is the Cell Transmission Model (CTM) [7]. On the other hand, the macroscopic second-order traffic model that is widely used in the context of freeway traffic modeling is the Modele d'Ecoulement de Trafic sur Autoroute NETworks (METANET) model [26].

The CTM includes a static speed-density relationship. METANET considers speed as a second state variable with its own state equation. In this way, METANET is able to capture additional dynamics. Thus, the most important reason for choosing METANET for this work is that it can reproduce the capacity drop phenomenon [33]. The METANET model is further analyzed in the following section.

2-2 The METANET model

The Modele d'Ecoulement de Trafic sur Autoroute NETworks (METANET) was firstly introduced in [26] as a macroscopic simulation tool, which included a deterministic, discrete, second-order traffic flow model. Since then, METANET [29] is the most used macroscopic, second-order traffic flow model, as it is suitable for modeling and control purposes. The model has a good accuracy level combined with adequate simulation speed.

The macroscopic nature of the model has to do with the definition of some variables that express the average behavior of the traffic at specific locations and time instances. The model is discretized in time and space.

K time intervals with the sample time denoted by T[h] constitute the time horizon. So, each time instant can be denoted by $t = k \cdot T, k = 0, 1, \ldots, K$. Concerning the space, the discretization refers to each link. Each link m is divided into N_m segments which have the same length denoted with L_m[km] and a number of lanes λ_m.

For the rest of this work, $N_m = 1 \forall m$. Therefore, it is not necessary to differentiate between links and segments. Thus, only the index i is used from now on.

Furthermore, for stability, it should hold that $L_i > T v_{\mathrm{f},i}$, where $v_{\mathrm{f},i}$ is the free flow speed in link i.

For each segment i, three variables are important, namely the *traffic density* $\rho_i(k)$ (veh/km/lane), the *mean speed* of the vehicles $v_i(k)(km/h)$ and the *traffic volume* or *traffic flow* $q_i(k)(veh/h)$.

2-2-1 Link equations

The following equations describe the system dynamics of METANET model. The first equation (Eq. (2-1)) describes the conservation of vehicles principle, while the second one (Eq. (2-2)) computes the traffic flow that leaves from each link during each time step.

$$\rho_i(k+1) = \rho_i(k) + \frac{T}{\lambda_i L_i} \left(q_{i-1}(k) - q_i(k) + q_{\mathrm{r},i}(k) - \beta_i(k) q_{i-1}(k) \right) \tag{2-1}$$

where $q_{\mathrm{r},i}(k)$ is the traffic flow entering the link i from the connected on-ramp (if any), $\beta_i(k)$ is the percentage of vehicles exiting the link i from the connected off-ramp (if any).

$$q_i(k) = \lambda_i \cdot \rho_i(k) \cdot v_i(k) \tag{2-2}$$

The third equation (Eq. (2-3)) gives the mean speed's dynamic behavior as a sum of four terms, namely the previous mean speed, a relaxation term, a convection term and an anticipation term. These terms are further analyzed in [14].

$$v_i(k+1) = v_i(k) + \frac{T}{\tau_i} \left(V(\rho_i(k)) - v_i(k) \right) + \frac{T}{L_i} v_i(k) \left(v_{i-1}(k) - v_i(k) \right) - \frac{\mu_i(k) T}{\tau_i L_i} \frac{\rho_{i+1}(k) - \rho_i(k)}{\rho_i(k) + K_i} \tag{2-3}$$

where τ_i(time constant),μ_i (anticipation constant) and K_i are model parameters that have to be identified, $V(\rho_i(k))$ is the desired speed for the drivers.

According to [19], the model should take two different values for the anticipation constant, in order to have better reproduction of shock waves and capacity drop. Therefore, the following equation (Eq. (2-4)) gives the anticipation constant.

$$\mu_i(k) = \begin{cases} \mu_{i,\mathrm{h}} & \text{for } \rho_{i+1}(k) \leq \rho_i(k) \\ \mu_{i,\mathrm{l}} & \text{Otherwise} \end{cases} \tag{2-4}$$

where $\mu_{i,\mathrm{h}}$ and $\mu_{i,\mathrm{l}}$ are model parameters.

The drivers' desired speed (Eq. (2-5)), without VSLs, is defined as follows:

$$V_i(k) = v_{\mathrm{f},i} e^{-\frac{1}{a_i}\left(\frac{\rho_i(k)}{\rho_{\mathrm{crit},i}}\right)^{a_i}} \tag{2-5}$$

where a_i is a parameter of the fundamental diagram, $v_{\mathrm{f},i}$ is the free flow speed, $\rho_{\mathrm{crit},i}$ is the critical density, i.e the density at which the maximum flow is achieved.

The above mentioned parameters are not the same for every freeway network and, therefore, the model needs to be simulated and validated in order for these values to be determined.

In cases that modeling of lane drops in the mainstream or modeling of merging phenomena caused by on-ramps is required, extra terms are added in Eq. (2-3).

For speed drop caused by merging phenomena:

$$-\frac{\delta_i T q_{\mathrm{r},i}(k) v_i(k)}{L_i \lambda_i \left(\rho_i(k) + K_i\right)} \tag{2-6}$$

where δ_i is a model parameter.

For speed drop caused by lane-drop:

$$-\frac{\phi_i T \Delta\lambda_{i-1} \rho_{i-1}(k) v_{i-1}^2(k)}{L_{i-1} \lambda_{i-1} \rho_{\mathrm{crit},i-1}} \tag{2-7}$$

where ϕ_i is a model parameter and $\Delta\lambda_{i-1} = \lambda_{i-1} - \lambda_i$

The last equation of the model defines the flow that enters the freeway from an on-ramp:

$$q_{\mathrm{r},i}(k) = \min\left(C_{\mathrm{r},i} r_i(k), D_i(k) + \frac{w_i(k)}{T}, C_{\mathrm{r},i} \frac{\rho_{\mathrm{m},i} - \rho_i(k)}{\rho_{\mathrm{m},i} - \rho_{\mathrm{crit},i}}\right) \tag{2-8}$$

where $\rho_{\mathrm{m},i}$ (jam density) and $C_{\mathrm{r},i}$ (on-ramp capacity) are model parameters, $r_i(k)$ is the ramp metering rate ($r_i(k) = 1 \forall k$ for unmetered on-ramp), $D_i(k)$ is the on-ramp demand, $w_i(k)$ is the length of the queue on the on-ramp.

The first term in Eq. (2-8) corresponds to the maximal flow allowed by the metering rate, the second term to the traffic that is willing to enter the freeway from the on-ramp at simulation step k and the third one to the maximal flow allowed by the mainstream conditions [20].

For the dynamics of the queue length on the on-ramp, a simple queue model is used:

$$w_i(k+1) = w_i(k) + T\left(D_i(k) - q_{\mathrm{r},i}(k)\right) \tag{2-9}$$

The same queue model can also be used for the queue length on the mainline origin, substituting the ramp demand with the mainstream demand and the ramp flow with the flow entering the mainline.

The Eq. (2-9) corresponds to the conservation of vehicles principle for an origin link and states that the number of vehicles waiting in the queue to enter the freeway will be the number of vehicles that waited in the queue the previous time period plus the difference between the demand flow at the origin at time period k and the outflow of that link at the same time period.

2-2-2 Boundary conditions

Due to the fact that the traffic conditions downstream and upstream of a freeway segment influence the traffic state of that segment, some boundary conditions have to be defined. These conditions correspond to the *upstream speed*, which is set equal to the speed of the first segment $v_1(k)$, the *upstream flow*, which is limited by an active speed limit or by the actual speed of the first segment, and the *downstream density*, which is set either equal to the density of the last segment or equal to the critical density.

2-3 Variable speed limit modeling

The METANET model analyzed in the previous section does not include the effects of VSLs. The effects of VSLs are described in [28]. The application of the VSLs has an effect on the three characteristic parameters of the FD, namely the free-flow speed, the critical density and the capacity. In the related literature, three different models are considered, to incorporate the effects of VSLs into the METANET model.

The first modeling approach, proposed by Hegyi et al. in [21], considers a modification of the drivers' desired speed equation whenever the VSLs are active. The model considers a parameter that reflects the drivers' compliance rate.

In the second modeling approach, proposed by Papamichail et al. in [30] and used mainly by Carlson et al. in [5] and [3], a speed ratio of the VSL value over the legal speed limit value when VSLs are inactive, is defined.

The third modeling approach, proposed by Frejo et al. in [15], combines the advantages of both previously mentioned methods. The effect of a VSL is modelled by modifying all three parameters of the FD, namely the critical density, the free flow speed and the capacity. Frejo et al. VSL model uses a calibration parameter for each one of the those parameters. However, its calibration procedure requires more time.

Choice of VSL model In this work, the model of Hegyi et al. is used. Not only is the simplest of the three, but also includes the drivers' compliance rate, which increases the model's accuracy [15].

In Hegyi et al. model, the drivers' desired speed is given by the equation Eq. (2-10):

$$V_i(k) = \min\left(v_{\mathrm{f},i} e^{-\frac{1}{a_i}\left(\frac{\rho_i(k)}{\rho_{\mathrm{crit},i}}\right)^{a_i}}, (1+\alpha_i)\, V_{\mathrm{c},i}(k) \right) \tag{2-10}$$

where α_i reflects the drivers' compliance rate, $V_{\mathrm{c},i}(k)$ is the speed limit value on link i

2-4 Variable speed limit algorithms

The VSLs algorithms for freeway traffic control focus mostly on one of the following three goals [15]: increase traffic safety, improve traffic efficiency, reduce fuel consumption and emissions.

This thesis focuses on the second goal, namely the improvement of traffic efficiency. This is achieved using mainline metering algorithms. Three different types of such algorithms are commonly used, which are the following: optimization-based algorithms, easy-to-implement algorithms and intermediate approaches.

In this work, the easy-to-implement VSL algorithms are chosen to be used, as they are simple, suitable for real-time applications (as their computation complexity is low), robust to measurement errors [11] and they depend less on model's accuracy compared to optimization-based algorithms [17].

Among the existing easy-to-implement VSL algorithms which focus on traffic efficiency improvement, the Mainstream Traffic Flow Control (MTFC), and the Logic-Based control algorithm for Variable Speed Limits (LB-VSL) algorithms are considered in this work. The main reasons for that choice are that they can be easily implemented, their parameters can be simply tuned and they are quite robust. Also, they both consider requirements connected with safety and practicality, making them suitable for simulations that approximate the reality.

2-4-1 VSL implementation constraints

Whenever the aforementioned VSL control algorithms are implemented in real-life applications, some implementation constraints should be considered. According to [13], two of these constraints are the temporal and the spatial constraint. Furthermore, for safety reasons, a minimum and a maximum allowable VSL value is usually used (VSL_{min}, VSL_{max}). Finally, the discrete nature of VSL values is also considered as a constraint.

Regarding the temporal constraint, the VSLs in real applications should change in a smooth way, to avoid accidents and a lack of comfort for the drivers. Thus, a maximum difference $\gamma(\frac{km}{h})$ in the VSL values that will be displayed to the drivers by the same gantry between two consecutive simulation time steps, is introduced. The Eq. (2-11) describes this constraint.

$$|V_{\mathrm{c},i}(k+l) - V_{\mathrm{c},i}(k+l-1)| \le \gamma \tag{2-11}$$

where $l = 0, 1, \ldots N_u - 1$, with N_u being the control horizon.

This constraint should be applied to every segment i that is controlled by a VSL.

As far as the spatial constraint is concerned, it corresponds to the difference that the value of the VSLs displayed in two adjacent segments during each simulation time step should have. It should be bounded by another constant value $\zeta(\frac{km}{h})$. Eq. (2-12) is used to represent this constraint.

$$|V_{\mathrm{c},i+1}(k+l) - V_{\mathrm{c},i}(k+l)| \leq \zeta \tag{2-12}$$

This constraint must be satisfied by all segments that are controlled by a VSL and have an adjacent segment which is also controlled by a VSL.

Regarding the discrete nature of VSLs, the displayed to the drivers values should have discrete values, which are usually limited by a predefined set $\mathbb{S}$. In the literature, two different approaches have been used in order to deal with that constraint.

The first one initially considers VSLs as continues variables, their continuous values are optimized and discretization is applied before the values are displayed to the drivers. In the second approach, the discrete characteristic of the VSL values should be directly considered during the optimization [13]. In this case, discrete optimization techniques are required.

2-4-2 Logic-Based control algorithm for Variable Speed Limits (LB-VSL)

The first easy-to-implement VSL algorithm used in this work is the Logic-Based control algorithm for Variable Speed Limits (LB-VSL), proposed in [11]. This algorithm is designed to reduce or avoid the congestion created at bottleneck locations. The controller computes the number of vehicles that have to be held back or released in order to maximize the bottleneck's outflow, by preventing the capacity drop phenomenon. This number is used to change the values of the VSLs either upwards or downwards.

The LB-VSL algorithm uses a feed-forward control structure that aims at the anticipation of the future evolution of the bottleneck's density and at the activation of the VSLs before the critical density is reached.

Among the advantages of this control algorithm are the easy tuning, the low computation time, the effectiveness and robustness.

Below are presented the six equations that produce all necessary variables for the implementation of the LB-VSL algorithm.

The first equation, Eq. (2-13), corresponds to the total number of vehicles that should be held back at time step k (at the most upstream segment).

$$V_{\text{hold },1}(k) = \max\left[0, T_{\text{ff}}(k)\left(Q_{\text{iB}}(k) - \overline{C_{\text{B}}}\right) - \lambda_{\text{B}} \mathrm{L}_{\text{B}}\left(\rho_{\text{crit,B}} - \rho_{\text{B}}(k)\right)\right] \tag{2-13}$$

where $T_{\text{ff}}(k)$ is the exact time that is needed for all vehicles that are located at most at distance L_A upstream of the bottleneck at time step k to reach the bottleneck's location, $Q_{\text{iB}}(k)$ is the average flow over all lanes that enters the bottleneck during $T_{\text{ff}}(k)$, $\overline{C_{\text{B}}}$ is a tuning parameter, usually around the capacity of the bottleneck, λ_{B} is the number of lanes at the bottleneck location, $\mathrm{L_B}$ is bottleneck's length, $\rho_{\mathrm{crit,B}}$ is bottleneck's critical density, $\rho_{\mathrm{B}}(k)$ is the bottleneck's density at time step k.

The time period $T_{\mathrm{ff}}(k)$ is given by the equation:

$$T_{\mathrm{ff}}(k) = L_{\mathrm{A}}/\hat{v}_{\mathrm{A}}(k) \tag{2-14}$$

in which L_{A} is the distance between the bottleneck and the most upstream segment with VSL installed and $\hat{v}_{\mathrm{A}}(k)$ is the weighted sum of all segments' speeds, given by the following equation:

$$\hat{v}_{\mathrm{A}}(k) = \frac{\sum_{i \in D_A} v_i(k)\hat{L}_i}{L_{\mathrm{A}}} \tag{2-15}$$

where $v_i(k)$ is the speed at segment i, $\hat{L}_i$ is the distance between two detectors, namely detectors i and $i+1$, D_A is the set of all detectors in the distance L_{A}.

In this work, it is assumed that detectors are present in all segments upstream the bottleneck [11],

The average flow $Q_{\mathrm{iB}}(k)$ is given as the weighted sum of all flows, by the equation that follows:

$$Q_{\mathrm{iB}}(k) = \frac{\sum_{i \in D_A} q_i(k)\hat{L}_i}{L_{\mathrm{A}}} \tag{2-16}$$

where $q_i(k)$ is the flow at segment i.

The second equation, Eq. (2-17), corresponds to the total number of vehicles that have to be released at time step k (at the most upstream segment).

$$V_{\mathrm{rel},1}(k) = \max\left[0, -T_{\mathrm{ff}}(k)\left(Q_{\mathrm{iB}}(k) - \underline{C_{\mathrm{B}}}\right) + \lambda_{\mathrm{B}} L_{\mathrm{B}}\left(\rho_{\mathrm{crit,B}} - \rho_{\mathrm{B}}(k)\right)\right] \tag{2-17}$$

where $\underline{C_{\mathrm{B}}}$ is a calibration parameter that has to be set between the capacity of the bottleneck and the congested outflow (capacity minus capacity drop) of the bottleneck [11].

In every time step, either one or both of the Eq. (2-13) and Eq. (2-17) should be zero.

The following equation, Eq. (2-18), which is the third equation of LB-VSL, gives the value of the VSL once it is decreased, which happens when $V_{\mathrm{hold},j}(k) > 0$.

$$VSL_j(k) = \max\left[\frac{L_j \lambda_j v_j(k) \rho_j(k)}{(1+\alpha_j)\left(L_j \lambda_j \rho_j(k) + V_{\mathrm{hold},j}(k)\right)}, \underline{VSL_j}\right] \tag{2-18}$$

where $VSL_j(k)$ is the speed limit at time step k in the freeway segment j, $v_j(k)$ is the speed at time step k in the freeway segment j, λ_j is the number of lanes in segment j, L_j is the length of segment j, α_j is the compliance parameter of segment j, $\underline{\mathrm{VSL}_j}$ is the minimum speed limit that can be applied.

Once the value of VSL is increased, which happens in the case of ($V_{\text{rel},j}(k) > 0$), the following equation, Eq. (2-19), which is the fourth equation of LB-VSL, holds:

$$\text{VSL}_j(k) = \begin{cases} \overline{\text{VSL}}_j & if \rho_j(k) \leq \frac{V_{\text{rel},j}(k)}{L_j\lambda_j} \\ \min\left[\frac{L_j\lambda_j v_j(k)\rho_j(k)}{(1+\alpha_i)\left(L_j\lambda_j\rho_j(k)-V_{\text{rel},j}(k)\right)}, \overline{\text{VSL}}_j\right] & \text{otherwise} \end{cases} \tag{2-19}$$

where $\overline{\text{VSL}_j}$ is the maximum speed limit that can be applied.

The fifth equation, Eq. (2-20), computes the remaining number of vehicles that still have to be held back after the VSL decrease in segment j:

$$V_{\text{hold},\ j+1}(k) = \max\left[0, \left(V_{\text{hold},\ j}(k) - V^{\text{hold}}_{\text{VSL},\ j}(k)\right]\right. \tag{2-20}$$

where $V^{\text{hold}}_{\text{VSL},\ j}(k)$ corresponds to the number of vehicles that are going to be held back by the VSL of segment j. This value is computed as follows:

$$V^{\text{hold}}_{\text{VSL},j}(k) = \lambda_j L_j \max\left(0, \frac{v_j(k)\rho_j(k)}{(1+\alpha_j)\,\text{VSL}_j(k)} - \rho_j(k)\right) \tag{2-21}$$

In a similar way, sixth equation, Eq. (2-4-2), computes the remaining number of vehicles that should be still released after the VSL increase in segment j.

$$V_{\text{rel},j+1}(k) = \max\left(0, \left(V_{\text{rel},j}(k) + V^{\text{rel}}_{\text{VSL},j}(k)\right)\right. \tag{2-22}$$

where $V^{\text{rel}}_{\text{VSL},\ j}(k)$ gives the number of vehicles that are going to be released by the VSL of segment j. This value is computed as follows:

$$V^{\text{rel}}_{\text{VSL},j}(k) = \lambda_j L_j \min\left(0, \frac{v_j(k)\rho_j(k)}{(1+\alpha_j)\,\text{VSL}_j(k)} - \rho_j(k)\right) \tag{2-23}$$

The are, therefore, two parameters that should be tuned for the LB-VSL algorithm, namely the $\overline{C_\text{B}}$ and $\underline{C_\text{B}}$.

2-4-3 Mainstream Traffic Flow Control (MTFC)

The second easy-to-implement VSL algorithm used in this work is the Mainstream Traffic Flow Control (MTFC) algorithm enabled via VSLs. The basic idea of MTFC is to enable the mainstream traffic flow upstream of a bottleneck to take values ordered by an appropriate control strategy, in order to establish optimal traffic condition for any appearing demand. The local aspect of the MTFC concept can be seen in Figure 2-1.

In [3], a VSL-based MTFC feedback cascade controller is proposed. The two loops of the feedback cascade control structure can be seen in Figure 2-2. This control structure is found to approximate the optimal controller's efficiency. It is also simpler, more robust and easily

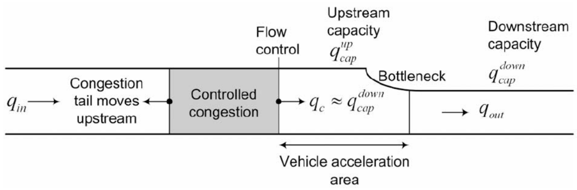

Figure 2-1: A local aspect for MTFC [4]

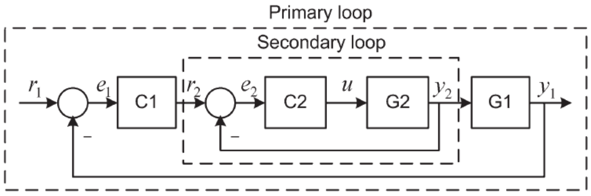

Figure 2-2: Two-loop feedback cascade control structure [3]

implementable. Also, the controller considers practical and safety requirements, which makes it suitable for field implementations.

For the implementation of the MTFC algorithm, two equations are required to be defined, as it can be seen in Figure 2-2.

In the first equation, Eq. (2-24), b corresponds to the VSL rate and is approximately equal to the displayed VSL divided by the legal speed limit without VSL [3]. It is actually an integral (I) regulator, and has the following form in the time domain:

$$b(k) = b(k-1) + K_I e_q(k) \tag{2-24}$$

where K_{I} is the integral gain, e_{q} is the flow control error (per lane), $e_{\mathrm{q}} = \hat{q}_{\mathrm{c}}(k) - q_{\mathrm{c}}(k)$.

The second equation, Eq. (2-25), computes the reference flow $\hat{q}_{\mathrm{c}}(k)$ and the measured density ρ_{out} is compared with the target density $\hat{\rho}_{\mathrm{out}}$. This target density, for throughput maximization, is taken equal to the critical density ρ_{crit}. It is a proportional-integral (PI) regulator, with the following form in the time domain:

$$\hat{q}_{\mathrm{c}}(k) = \hat{q}_{\mathrm{c}}(k-1) + \left(K'_{\mathrm{P}} + K'_I\right) e_\rho(k) - K'_{\mathrm{P}} e_\rho(k-1) \tag{2-25}$$

where K'_{P} is the proportional gain, K'_{I} is the integral gain, e_ρ is the density control error (per lane), $e_\rho = \hat{\rho}_{\mathrm{out}}(k) - \rho_{\mathrm{out}}(k)$.

For the computation of the value of the VSL, the following equation 2-26 is used, as already mentioned.

$$VSL_j(k) = v_f b(k) \tag{2-26}$$

where v_f is the legal speed limit without VSL.

The are, therefore, three gains that should be tuned for the MTFC algorithm. Those are the K'_{P} and K'_{I} for the PI controller (primary controller) and K_{I} for the I regulator (secondary controller).

2-5 Adaptive control in freeway traffic control

Based on [1], "An adaptive controller is a controller with adjustable parameters and a mechanism for adjusting the parameters". In the context of adaptive control, the controller's parameters are not constant, because some parameters of the plant's model are unknown, change unpredictably or vary slowly in time. These variations in the process parameters are caused by parameter disturbances, which affect the performance of the control system.

A freeway traffic network needs to be controlled with an adaptive control mechanism, because changes in traffic model's parameters can occur due to unpredictable external disturbances, like mainline or on-ramp demands, weather conditions, or incidents, as well as due to the implementation of control measures that affect the process dynamics and are not modeled accurately.

According to the literature, the adaptive control techniques for freeway traffic systems focus mainly on RM and on the estimation of the critical occupancy.

As far as the VSLs are concerned, to the best of the author's knowledge, the literature that uses them in an adaptive framework is quite limited, if not nonexistent.

In this chapter, the adaptive control approaches used in the literature for RM ([32], [24], [22]) are presented. The existing adaptive algorithms for RM are mostly variations of the ALINEA control algorithm. They use measurements of flow and occupancy.

2-6 Adaptive controllers for RM

In the recent literature, the ALINEA control algorithm (Eq. (2-27)) [27] and its variations are the most commonly used and well-studied strategies for traffic-responsive and local RM. Those strategies are also chosen by most of the researchers as the base algorithms in order to develop adaptive control approaches for RM.

The ramp metering strategy ALINEA is actually an integral regulator, which reads as follows:

$$r(k) = r(k-1) + K_{\mathrm{R}} \left[\hat{o} - o_{\mathrm{out}}(k) \right], \tag{2-27}$$

where K_{R} is a constant and positive parameter, $o_{\mathrm{out}}(k)$ is the (measured) downstream occupancy and $\hat{o}$ is a desired value for the downstream occupancy.

The goal of the adaptive control approaches for RM is the downstream flow (q_{out}) maximization. This goal is achieved when the set value (or target value) $\hat{o}$ is equal to the critical occupancy o_{crit} [24].

In the non-adaptive case, the critical occupancy is estimated before the application of the RM algorithm. However, it is proven that the measurements of flow and occupancy do not cover the area of critical occupancy. Also, "the critical occupancy may change in real time because of environmental conditions (rain, darkness), traffic composition (trucks), or other control measures (e.g., VSLs)" [24]. Thus, the on-line estimation of the critical occupancy is useful and studied by many researchers when RM is considered. The RM strategies, which contain algorithms for the estimation of the critical occupancy, are mentioned below.

AD-ALINEA The first algorithm is the AD(adaptive)-ALINEA algorithm, which is proposed in [32]. It uses real-time flow ($q_{\text{out}}(k-1)$) and occupancy ($o_{\text{out}}(k-1)$) measurements to generate estimates for the critical occupancy ($\tilde{o}_{\text{crit}}(k)$). Then, these estimates are used as set values in the ALINEA algorithm ($\hat{o}(k) = \tilde{o}_{\text{crit}}(k)$).

The estimation of the critical occupancy $\tilde{o}_{\text{crit}}(k)$ is based on the estimation of the derivative $D = \frac{dq_{\text{out}}}{do_{\text{out}}}$ at the current estimate $\tilde{o}_{crit}(k-1)$. Depending on the value of the derivative, the current estimate $\tilde{o}_{crit}(k-1)$ is either increased or decreased by Δ. In case of $D = 0$, the new estimate of the critical occupancy is set equal to the estimate of the previous time step. The same happens in the case that the derivative has a value in the interval $[D^-, D^+]$, where D^- and D^+ correspond to the down and upper limit for a change of the estimate of the critical occupancy.

Modified AD-ALINEA The second algorithm, proposed in [24], is the Modified AD-ALINEA. The modification of the original AD-ALINEA were made to increase its robustness and accuracy in real traffic conditions.

The modified AD-ALINEA algorithm is combined with the Simple Derivative Estimation (SDE) method (see 2-7-2).

The modifications correspond to the addition of some extra steps. Initially, an extra step reduces the value of $\tilde{o}_{\text{crit}}(k)$ by Δ every K time units, in order to guarantee that the decrease of the critical occupancy after an increase is properly tracked by the estimation algorithm, as explained in [24]. The next modification of the original AD-ALINEA algorithm is the addition of a threshold value for the modification of the critical occupancy estimation, in case two consecutive measurements are close to each other. Furthermore, the time derivative $\delta(k)$, which is based on two consecutive measurements is restricted to take values in the interval $[\delta_{\min}, \delta_{\max}]$. Finally, the value of $\tilde{o}_{\text{crit}}(k)$ is restricted to the interval $[\tilde{o}_{\text{crit}}^{\min}, \tilde{o}_{\text{crit}}^{\max}]$, in order to ensure that the estimate of the critical occupancy will not take very small or very high values.

Adaptive D-ALINEA algorithm in PPA Since 2013, RM, together with other traffic control measures, has begun to be implemented in a large Field Operational Test (FOT), the Amsterdam Practical Trial (APT) or PraktijkProef Amsterdam (PPA) [22].

In the PPA, an adaptive control algorithm is used for RM. This algorithm is based on the well-known algorithm for RM, D-ALINEA, which is a variation of the ALINEA algorithm. The difference between D-ALINEA and ALINEA is that the D-ALINEA algorithm uses the density instead of the occupancy as the measured variable and the target value [23], as it can be seen in the Eq. (2-28).

$$r(k) = r(k-1) + K_{\mathrm{R}}\left[\hat{\rho} - \rho_{\mathrm{out}}(k-1)\right] \tag{2-28}$$

The adaptive version of D-ALINEA, that is utilized in the PPA, uses a Parameter Estimator (PE) or Parameterschatter to "estimate the critical density and the capacity at the bottleneck location" [22]. Thus, the PE uses on-line measurements of the traffic state to estimate the critical density on-line. Then, the target density value is computed using $\hat{\rho} = \xi * \rho_{\mathrm{crit}}$, with $\xi \leq 1$. In this way, the implemented target density value never exceeds the estimated critical density. The estimation method adopted in the PPA for the critical density follows the AD-ALINEA algorithm described earlier, modified for density instead of occupancy.

AD-RMC In [23], a new ramp metering algorithm with parameter adaptation is proposed. The name given to this algorithm is Adaptive Ramp Metering Controller (AD-RMC). This algorithm uses a variation of the PI (Proportional-Integral)-ALINEA algorithm. The original PI-ALINEA algorithm uses the occupancy, while its variation used in [23] uses density measurements and density as target value. The variation of PI-ALINEA used in [23] can be seen in Eq. (2-29)

$$r(k) = r(k-1) - K_{\mathrm{P}}\left[\rho_{\mathrm{out}}\,(k-1) - \rho_{\mathrm{out}}\,(k-2)\right] + K_{\mathrm{I}}\left[\hat{\rho} - \rho_{\mathrm{out}}\,(k-1)\right] \tag{2-29}$$

The AD-RMC proposed in [23], which is presented in Eq. (2-30), uses time-dependent values for both the controller's gains K_{P}, K_{I}, and also for the target density $\hat{\rho}$.

$$r(k) = r(k-1) + K_P(k)\left[\hat{\rho}(k) - \rho_{\mathrm{out}}(k-1)\right] + K_I(k)\int_1^k \left(\hat{\rho}(k) - \rho_{\mathrm{out}}(k-1)\right)\mathrm{d}k \tag{2-30}$$

The adaptive law used in [23] for the controller's gains K_{P} and K_{I} is the *gradient method*, which is also used in [25] for updating the gains of a PI-controller for a real-time traffic management application.

The target density value ($\hat{\rho}(k)$) is determined adaptively using the equation $\hat{\rho}(k) = \xi * \hat{\rho}_{\mathrm{crit}}(k)$, with $\xi \leq 1$. Thus, the estimation of the critical density in every time step k ($\hat{\rho}_{\mathrm{crit}}(k)$) is needed. This estimation uses the derivative of the FD at time k ($D(k)$) and one of the following update rules for overcritical (Eq. (2-31)) or undercritical (Eq. (2-32)) traffic conditions.

The update rule for overcritical traffic conditions is chosen if the calculated derivative is on the "left" of the interval (β^-, β^+), while the update rule for undercritical traffic conditions is selected if the derivative is on the "right" of the same interval. In case the derivative is found to be "inside" the interval, the critical density is not updated.

For overcritical conditions $(D(k) < \beta^- < 0)$

$$\hat{\rho}_{\mathrm{crit}}(k) = \begin{cases} \alpha\hat{\rho}_{\mathrm{crit}}(k-1) + (1-\alpha)\rho_{\mathrm{out}}(k) & \text{if } \hat{\rho}_{\mathrm{crit}}(k-1) > \rho_{\mathrm{out}}(k) \\ \hat{\rho}_{\mathrm{crit}}(k-1) & \text{if } \hat{\rho}_{\mathrm{crit}}(k-1) \leq \rho_{\mathrm{out}}(k) \end{cases} \tag{2-31}$$

For undercritical conditions $(D(k) > \beta^+ > 0)$

$$\hat{\rho}_{\text{crit}}(k) = \begin{cases} \alpha\hat{\rho}_{\text{crit}}(k-1) + (1-\alpha)\rho_{\text{out}}(k) & \text{if } \hat{\rho}_{\text{crit}}(k-1) < \rho_{\text{out}}(k) \\ \hat{\rho}_{\text{crit}}(k-1) & \text{if } \hat{\rho}_{\text{crit}}(k-1) \geq \rho_{\text{out}}(k) \end{cases} \tag{2-32}$$

The constant weighing smoothing factor $\alpha \in (0,1)$ is used "to prevent oscillatory behaviour" [23]. Due to the non-symmetric form of the FD around the critical density, $|\beta^+| > |\beta^-|$, as it can be seen in Figure 2-3.

2-7 Estimation methods for the derivative of the FD

In the previous section, the state-of-the-art in adaptive ramp metering algorithms is reported. All the algorithms previously mentioned have one common step before calculating the critical density, the real-time estimation of the derivative of the FD. Three different methods are reported in the literature to calculate this quantity, namely the Parameter Estimator (PE) method used in [23] (or Parameterschatter), the Simple Derivative Estimation (SDE) method used in [32] and the Kalman-Filter-based derivative Estimation (KFE) method, which is reported in [32],[22] and in [24]. Both AD-ALINEA and AU-ALINEA algorithms are tested in [32], using the Simple Derivative Estimation (SDE) and Kalman-Filter-based derivative Estimation (KFE) methods for the derivative D determination, having similar results. Also, in [24], the modified AD-ALINEA algorithm is tested using both of these derivative estimation methods, resulting slightly better performance for the KFE algorithm. The adaptive version of D-ALINEA, that is utilized in the PPA, uses a Parameter Estimator (PE) or Parameterschatter to "estimate the critical density and the capacity at the bottleneck location" [22]. Also, in the PPA, the AD-ALINEA algorithm is used in combination with the KFE approach (modified for density instead of occupancy) for the estimation of the derivative. Finally, the AD-RMC uses the PE method for the estimation of the derivative of the FD.

2-7-1 Parameter Estimator (PE)

The first estimation method, namely the Parameter Estimator (PE), is based on the Least Squares (LS) method to determine the derivative of the FD. It uses measurements of flow (q) and density (ρ) taken the last T time steps. The equation for the derivative estimation in this case, follows:

$$D(k) = \frac{\sum_{i=k-T}^{T}(\rho(i) - \bar{\rho}) * (q(i) - \bar{q})}{\sum_{i=k-T}^{T}(\rho(i) - \bar{\rho})^2} \tag{2-33}$$

where $\bar{\rho}$ and $\bar{q}$ are the mean values of the density and flow, respectively.

2-7-2 Simple Derivative Estimation (SDE)

The second estimation method, namely the Simple Derivative Estimation (SDE), estimates the derivative of the FD at the simulation time step k ($D(k)$) using the exponential smoothing technique for the time derivatives $\delta(k)$, as it can be seen in Eq. (2-34). The time derivatives $\delta(k)$ are based on measurements of the flow and occupancy from the last two time steps.

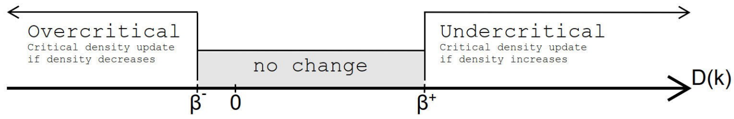

Figure 2-3: Critical density update in PE

$$\delta(k) = \frac{q_{\text{out}}\ (k-1) - q_{\text{out}}\ (k-2)}{o_{\text{out}}\ (k-1) - o_{\text{out}}\ (k-2)} \tag{2-34}$$

After having calculated the time derivatives $\delta(k)$, the exponential filter presented in Eq. (2-35) can be applied.

$$D(k) = \alpha\delta(k) + (1-\alpha)D(k-1) \tag{2-35}$$

where $\alpha \in (0,1)$ is a constant smoothing parameter.

2-7-3 Kalman-Filter-based derivative Estimation (KFE)

The third method for the estimation of the derivative of the FD is the Kalman-Filter-based derivative Estimation (KFE). This method also uses real-time measurements of flow ($q_{\text{out}}(k-1)$) and occupancy ($o_{\text{out}}(k-1)$). The occupancy measurement should be close to the current estimate of the critical occupancy. The idea is to fit a straight line to these measurements. This line should be expressed by the Eq. (2-36).

$$q_{\text{out}} = D\left(o_{\text{out}} - \tilde{o}_{\text{crit}}\right) + E \tag{2-36}$$

Therefore, it is needed to "recursively estimate the parameters D($\frac{\frac{veh}{h}}{\%}$) and E ($\frac{veh}{h}$) around the current estimate $\tilde{o}_{\text{crit}}$" [32].

To achieve this, the method uses a Kalman filter with state $\mathbf{x} = \begin{bmatrix} D \\ E \end{bmatrix}$. The method is further analyzed in chapter 3.

2-8 Conclusions

This chapter presents the relevant literature on adaptive control for freeway traffic control. In the first section, freeway traffic modeling is discussed. Section 2-1 presents the macroscopic traffic model METANET. This model is chosen to be used as the simulation model for this work, as it is the most suitable for control purposes. It provides a good trade-off between computational complexity and model accuracy and it can model capacity drop phenomenon. In the next section, three modeling approaches are presented, which incorporate the effects of VSLs into the METANET model. Hegyi et al. model is chosen to be used in this work to model the VSL effects. In the fourth section, the different goals of the VSL algorithms are briefly mentioned and two easy-to-implement VSL algorithms, which focus on the improvement of traffic efficiency, are chosen to be further analyzed. Those are the MTFC and LB-VSL. Section 2-5 discusses adaptive control in freeway traffic. The existing literature focuses mainly on the RM as control measure, and uses mostly the ALINEA control algorithm with its variations. Those algorithms improve traffic efficiency, by maximizing network's throughput. This is achieved by trying to reduce the capacity drop phenomenon near bottleneck locations. In section 2-6, the main adaptive control algorithms for RM are mentioned. The algorithms focus on downstream flow maximization, firstly by estimating on-line the critical occupancy or critical density at the bottleneck location and secondly by setting the control strategy's target density or occupancy equal to this estimated value. For the estimation of the critical density or critical occupancy, all three methods presented in 2-7 have the same goal, namely the estimation of the derivative of the FD. Three different methods are presented for the estimation of the derivative of the FD. Those are the PE, SDE and KFE.

2-9 Problem Statement

This work investigates how the easy-to-implement MTFC and LB-VSL control algorithms can be combined with the estimation methods PE, SDE and KFE for freeway traffic control with VSLs. The estimation methods are modified to incorporate density measurements and estimate the critical density. Also, the VSL algorithms are made adaptive, by adjusting their parameters using estimated values for the freeway traffic netwrok's critical density. The goal is to increase freeway traffic network's performance, in terms of throughput maximization.

Chapter 3

Online critical density estimation for adaptive easy-to-implement Variable Speed Limit algorithms

This chapter proposes six adaptive, easy-to-implement control frameworks for freeway traffic control with VSLs. Modified versions of the MTFC and LB-VSL easy-to-implement VSL control algorithms are combined with the PE and modified versions of the SDE and KFE estimation methods, to estimate on-line the freeway traffic network's critical density at bottleneck's location.

1. The SDE and KFE estimation methods are adjusted to include the critical density instead of the critical occupancy and the complete SDE and KFE algorithms are reported.
2. The estimation method PE is combined with the update rule for the critical density from the AD-RMC algorithm.
3. The MTFC algorithm is adjusted to have variable gains $K'_{\mathrm{P}}(t)$, $K'_{\mathrm{I}}(t)$ and $K_{\mathrm{I}}(t)$.
4. The LB-VSL algorithm is adjusted to have variable calibration parameters $\overline{C_{\mathrm{B}}}(t)$ and $\underline{C_{\mathrm{B}}}(t)$.

The notation and modelling concepts of METANET model are considered. The chapter is structured as follows. In Section 3-1, the framework used in this chapter is introduced. Section 3-2 discusses the three different approaches for the online estimation of the critical density. In section 3-3, the adaptivity added to the MTFC and LB-VSL algorithms is reported. The parameters of the estimation methods and control algorithms are summarized in 3-4. Finally, the conclusions are stated in section 3-5.

3-1 Framework introduction

This section provides an introduction to the framework used in this chapter. Freeway traffic control is achieved by the usage of different control measures. In this work, the VSLs are chosen to be used. The easy-to-implement VSL algorithms MTFC and LB-VSL are utilized, which aim at the improvement of traffic efficiency, by means of outflow maximization. To maximize the positive impact of those algorithms, the traffic network's density at the bottleneck's location should be as close as possible to the critical density.

However, the value of the critical density changes, due to many reasons, such as environmental conditions or incidents. Hence, the real-time estimation of the critical density is crucial. For that, three estimation methods, which produce an estimate of the network's critical density, are implemented. In those estimation methods, the calculation of the derivative of the FD is required. The estimation of the derivative of the FD is studied, by comparing three different algorithms, namely the PE, SDE and KFE. The last two methods are modified to include measurements of density. They are also adjusted to estimate the critical density instead of the critical occupancy. Furthermore, the parameters $\underline{C_\mathrm{B}}$ and $\overline{C_\mathrm{B}}$ of the LB-VSL and the controllers' gains K_P', K_I' and K_I of the MTFC algorithm are updated on-line to increase the controllers' performance.

3-2 Online critical density estimation

In chapter 2, the SDE and KFE estimation algorithms are explained. However, those algorithms use measurements of occupancy and estimate on-line the critical occupancy. In this work, the measurements of density are used instead and the algorithms produce estimates for the freeway traffic network's critical density at the bottleneck's location. Below are presented the modified versions of the SDE and KFE algorithms.

3-2-1 Modified Simple Derivative Estimation (SDE) algorithm

In the modified SDE algorithm, the calculation of the time derivatives $\delta(k)$ is based on measurements of flow and density from the last two time steps.

$$\delta(k) = \frac{q_\mathrm{out}\,(k-1) - q_\mathrm{out}\,(k-2)}{\rho_\mathrm{out}\,(k-1) - \rho_\mathrm{out}\,(k-2)} \tag{3-1}$$

After having calculated the time derivatives $\delta(k)$, the exponential filter can be applied.

$$D(k) = \alpha\delta(k) + (1-\alpha)D(k-1) \tag{3-2}$$

where $\alpha \in (0, 1)$.

In the existing SDE algorithm, the critical density $\tilde{\rho}_\mathrm{crit}$ is used instead of the critical occupancy. Depending on the value of the derivative D, the critical density is either increased or decreased by Δ. If $D = 0$ or if the derivative has a value in the interval $[D^-, D^+]$, the new

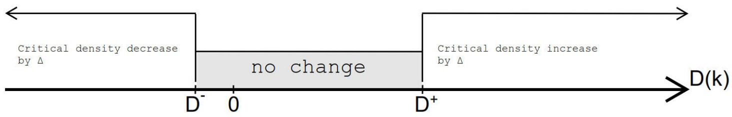

Figure 3-1: Critical density update in SDE

estimate of the critical density is set equal to the estimate of the previous time step. Due to the non-symmetric form of the FD around the critical density, $|D^+| > |D^-|$, as it can be seen in Figure 3-1

The modified SDE algorithm is presented below.

a. Initialize $D(0) = 0, k = 1, \tilde{\rho}_{\text{crit}}(0) = \rho_{\text{crit}}^{\text{min}}$
b. Enter the new measurements $q_{\text{out}}(k-1), \rho_{\text{out}}(k-1)$
c. Reduce $\tilde{\rho}_{\text{crit}}(k)$ every K time units by Δ, unless $\tilde{\rho}_{\text{crit}}(k) = \rho_{\text{crit}}^{\text{min}}$
d. If $|\tilde{\rho}_{\text{crit}}(k-1) - \rho_{\text{out}}(k-1)| > P$, then $\tilde{\rho}_{\text{crit}}(k) = \tilde{\rho}_{\text{crit}}(k-1)$; go to Step k
e. If $|\rho_{\text{out}}\ (k-1) - \rho_{\text{out}}\ (k-2)| \geq \epsilon$ calculate $\delta(k)$
otherwise, set $\tilde{\rho}_{\text{crit}}(k) = \tilde{\rho}_{\text{crit}}(k-1)$ and go to Step k
f. If $\delta(k) \geq \delta_{\text{max}}$ then $\delta(k) = \delta_{\text{max}}$; if $\delta(k) \leq \delta_{\text{min}}$ then $\delta(k) = \delta_{\text{min}}$
g. Update the value of the derivative $D(k)$
h. If $D(k) > D^+$ then set $s(k) = +1$; if $D(k) < D^-$then set $s(k) = -1$;
otherwise set $s(k) = 0$
i. Calculate the quantity $\tilde{\rho}^{\text{c}}_{\text{crit}}(k) = \tilde{\rho}_{\text{crit}}(k-1) + s(k)\Delta$; update the critical density estimate according to

$$\tilde{\rho}_{\text{crit}}(k) = \begin{cases} \tilde{\rho}^{\text{c}}_{\text{crit}}(k) & \text{if } \tilde{\rho}^{\text{c}}_{\text{crit}}(k) \in (\rho_{\text{crit}}^{\text{min}}, \rho_{\text{crit}}^{\text{max}}) \\ \rho_{\text{crit}}^{\text{min}} & \text{if } \tilde{\rho}^{\text{c}}_{\text{crit}}(k) \leq \rho_{\text{crit}}^{\text{min}} \\ \rho_{\text{crit}}^{\text{max}} & \text{if } \tilde{\rho}^{\text{c}}_{\text{crit}}(k) \geq \rho_{\text{crit}}^{\text{max}} \end{cases}$$

j. If $s(k) \neq 0$, set $D(k) = 0$
k. Set $k := k + 1$; go to Step b.

In step a, the necessary initializations are introduced. Step c reduces the value of the estimate of the critical density by Δ every K time units, in order to guarantee that the decrease of the critical occupancy after an increase is properly tracked by the estimation algorithm, as explained in [24]. In step e, a threshold value is introduced for the modification of the critical density estimation, in case two consecutive measurements are close to each other. In step f, the time derivative $\delta(k)$, which is based on two consecutive measurements, is restricted to take values in the interval $[\delta_{\text{min}}, \delta_{\text{max}}]$. Finally, in step i, the value of the critical density estimate is restricted to the interval $[\rho_{\text{crit}}^{\text{min}}, \rho_{\text{crit}}^{\text{max}}]$, in order to ensure that the estimate of the critical density will not take very small or very high values.

3-2-2 Modified Kalman-Filter-based derivative Estimation (KFE) algorithm

The KFE method uses real-time measurements of flow ($q_{\text{out}}(k-1)$) and density ($\rho_{\text{out}}(k-1)$). The flow and density measurements that are used for the estimation of the derivative are sufficiently close to the current estimate of the critical density, in order to fit a straight line to these measurements. This line should be expressed by the Eq. (3-3).

$$q_{\text{out}} = D\left(\rho_{\text{out}} - \tilde{\rho}_{\text{crit}}\right) + E \tag{3-3}$$

Therefore, it is now needed to estimate the parameters D($\frac{km}{h}$) and E ($\frac{veh}{h}$) around the current estimate $\tilde{\rho}_{\text{crit}}$.

The state of the Kalman filter and the state equation remains the same as $\mathbf{x} = \begin{bmatrix} D \\ E \end{bmatrix}$ and

$$\mathbf{x}(k) = \mathbf{x}(k-1) + \mathbf{z}(k) \tag{3-4}$$

where $\mathbf{z}$ is the system noise with covariance $\mathbf{Z}$. Only table c changes in the output equation, as follows:

$$y(k) = \mathbf{c}(k)\mathbf{x}(k) + w(k) \tag{3-5}$$

where

- y(k) $= q_{\text{out}}(k)$
- $\mathbf{c}(k) = \begin{bmatrix} (\rho_{\text{out}} - \tilde{\rho}_{\text{crit}}) & 1 \end{bmatrix}$
- w(k) is the output noise with covariance matrix W.

Based on the above, the resulting Kalman filter now reads:

$$\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k-1) + \mathbf{H}(k-1)[y(k) - \mathbf{c}(k)\hat{\mathbf{x}}(k-1)] \tag{3-6}$$

where

- $\hat{\mathbf{x}}(k)$ is the estimate of the state vector $\mathbf{x}$
- $\mathbf{H}(k-1)$ is the gain vector
- $y(k) - \mathbf{c}(k)\hat{\mathbf{x}}(k-1)$ is the output error

The filter gain vector $\mathbf{H}(k)$ is calculated using the following equation:

$$\mathbf{H}(k-1) = [\mathbf{\Pi}(k-1) + \mathbf{Z}]\,\mathbf{c}^T(k)\left\{\mathbf{c}(k)\,[\mathbf{\Pi}(k-1) + \mathbf{Z}]\,\mathbf{c}^T(k) + W\right\}^{-1} \tag{3-7}$$

where $\mathbf{\Pi}$, the error covariance matrix, is updated according to the Eq. (3-8).

$$\mathbf{\Pi}(k) = [\mathbf{\Pi}(k-1) + \mathbf{Z}] - \mathbf{H}(k-1)\mathbf{c}(k)\,[\mathbf{\Pi}(k-1) + \mathbf{Z}] \tag{3-8}$$

For the initial values of vectors $\hat{\mathbf{x}}$ and $\mathbf{\Pi}$ the following equations are used:

$$\mathbf{\Pi}(0) = \mathbf{Z} \tag{3-9}$$

$$\hat{\mathbf{x}}(0) = \begin{bmatrix} 0 \\ q_{\text{cap}}^{\text{est}} \end{bmatrix} \tag{3-10}$$

where $q_{\text{cap}}^{\text{est}}$ is the estimate of the capacity of the freeway downstream of the bottleneck.

The following algorithm represents the modified KFE algorithm for the critical density estimation.

a. Initialize $D(0) = 0, k = 1, E(0) = q_{\text{cap}}^{\text{est}}, \tilde{\rho}_{\text{crit}}(0) = \rho_{\text{crit}}^{\text{min}}, \mathbf{\Pi}(0) = \mathbf{Z}$.
b. Enter the new measurements $q_{\text{out}}(k-1), \rho_{\text{out}}(k-1)$
c. Reduce $\tilde{\rho}_{\text{crit}}(k)$ every K time units by Δ, unless $\tilde{\rho}_{\text{crit}}(k) = \rho_{\text{crit}}^{\text{min}}$
d. If $|\tilde{\rho}_{\text{crit}}(k-1) - \rho_{\text{out}}(k-1)| > P$, then $\tilde{\rho}_{\text{crit}}(k) = \tilde{\rho}_{\text{crit}}(k-1)$; go to Step i
e. Calculate $\mathbf{H(k\text{-}1)}, D(k), E(k), \mathbf{\Pi}(k)$ in this order
f. If $D(k) > D^+$ then set $s(k) = +1$; if $D(k) < D^-$ then set $s(k) = -1$; otherwise set $s(k) = 0$.
g. Calculate the quantity $\tilde{\rho}_{\text{crit}}^{\text{c}}(k) = \tilde{\rho}_{\text{crit}}(k-1) + s(k)\Delta$; update the critical density estimate according to

$$\tilde{\rho}_{\text{crit}}(k) = \begin{cases} \tilde{\rho}_{\text{crit}}^{\text{c}}(k) & \text{if } \tilde{\rho}_{\text{crit}}^{\text{c}}(k) \in (\rho_{\text{crit}}^{\text{min}}, \rho_{\text{crit}}^{\text{max}}) \\ \rho_{\text{crit}}^{\text{min}} & \text{if } \tilde{\rho}_{\text{crit}}^{\text{c}}(k) \le \rho_{\text{crit}}^{\text{min}} \\ \rho_{\text{crit}}^{\text{max}} & \text{if } \tilde{\rho}_{\text{crit}}^{\text{c}}(k) \ge \rho_{\text{crit}}^{\text{max}} \end{cases}$$

h. If $s(k) \neq 0$, set $E(k) := E(k) + D(k)\Delta$ and $D(k) = 0$
i. Set $k := k + 1$; go to Step b.

Almost all the steps of the above algorithm are same as in the case of SDE method, except for the calculation of the derivative of the FD, which is performed in step e in above algorithm.

3-2-3 Critical density estimation and PE

In this work, the aforementioned estimation methods for the derivative of the FD, SDE and KFE, are compared with the PE estimation method. The PE method is presented in 2-7-1. This estimation method is combined with the update rule for the critical density estimation from AD-RMC algorithm, as described in (2-31) and (2-32).

3-3 Adaptive easy-to-implement Variable Speed Limit algorithms

In this section, adaptivity is added to the previously mentioned easy-to-implement VSL algorithms MTFC and LB-VSL.

3-3-1 Adaptive Mainstream Traffic Flow Control (MTFC)

In section 2-4-3, the MTFC algorithm is reported. The controller consists of one integral (I) regulator and one proportional-integral (PI) regulator in a two-loop feedback cascade control structure. The I regulator contains the gain K_I, while the PI regulator contains the gains K'_{P} and K'_{I}. In this work, those gains are made variable. This means that the controllers' gains are updated in each controller step. The update of the controllers' gains is based on the value of the estimated critical density produced by the aforementioned estimation methods PE, SDE and KFE. The equations of the adaptive MTFC control algorithm are presented below.

In the first equation, Eq. (3-11), b corresponds to the VSL rate and is equal to the displayed VSL divided by the legal speed limit without VSL, that is $b(k-1) = \frac{VSL(\tau-1)}{v_f}$.

$$b(k) = b(k-1) + K_{\mathrm{I}}(\tau)[\hat{q}_{\mathrm{VSL}}(k) - q_{\mathrm{VSL}}(k)] \tag{3-11}$$

where

- $K_{\mathrm{I}}(\tau)$ is the adaptive integral gain updated in every controller's step,
- $\hat{q}_{\mathrm{VSL}}(k)$ is the estimated flow exiting the segment with the VSL installation and
- $q_{\mathrm{VSL}}(k)$ is the measured flow exiting the segment with the VSL installation,

The temporal VSL implementation constraint is also applied, as it can be seen is Eq. (3-12), together with a minimum and maximum allowable limit for the applied VSL, as presented in Eq. (3-13).

$$|VSL(\tau) - VSL(\tau-1)| \leq \gamma \tag{3-12}$$

$$\frac{VSL_{min}}{v_f} \leq b(k) \leq \frac{VSL_{max}}{v_f} \tag{3-13}$$

For the adaptive gain $K_{\mathrm{I}}(\tau)$, the following equation has been used.

$$K_{\mathrm{I}}(\tau) = K_{\mathrm{I}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)}) \tag{3-14}$$

The second equation, Eq. (3-15), gives the reference flow $\hat{q}_{\mathrm{VSL}}(k)$ using the measured density $\rho_{\mathrm{VSL}}(k)$ and the estimated critical density $\hat{\rho}_{\mathrm{crit}}(\tau)$.

$$\hat{q}_{\mathrm{VSL}}(k) = \hat{q}_{\mathrm{VSL}}(k-1) + \left(K'_{\mathrm{P}}(\tau) + K'_{I}(\tau)\right)\left(\hat{\rho}_{\mathrm{crit}}(\tau) - \rho_{\mathrm{VSL}}(k)\right) - K'_{\mathrm{P}}(\tau)\left(\hat{\rho}_{\mathrm{crit}}(\tau) - \rho_{\mathrm{VSL}}(\tau)\right) \tag{3-15}$$

where

- $K'_{\mathrm{P}}(\tau)$ is the adaptive proportional gain updated in every controller's step and
- $K'_{\mathrm{I}}(\tau)$ is the adaptive integral gain updated in every controller's step

For the adaptive gains $K'_{\mathrm{P}}(\tau)$ and $K'_{\mathrm{I}}(\tau)$, the following equations have been used.

$$K'_{\mathrm{P}}(\tau) = K'_{\mathrm{P}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)}) \tag{3-16}$$

$$K'_{\mathrm{I}}(\tau) = K'_{\mathrm{I}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)}) \tag{3-17}$$

3-3-2 MTFC gains' equations

For each one of the adaptive gains' equations of the MTFC algorithm, Eq. (3-14), Eq. (3-16) and Eq. (3-17), three different equations were evaluated. All possible combinations were checked and the selected ones were shown to result in the lowest value of TTS for all considered scenarios. The three different equations used for each adaptive gain of the MTFC algorithm are presented below.

For the adaptive gain $K_{\mathrm{I}}(\tau)$, the following equations were evaluated:

- $K_{\mathrm{I}}(\tau) = K_{\mathrm{I}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)})$
- $K_{\mathrm{I}}(\tau) = K_{\mathrm{I}}(\frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)})$
- $K_{\mathrm{I}}(\tau) = K_{\mathrm{I}}(1 - \zeta(1 - \frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)}))$, where $\zeta \in [0.1, 0.9]$

Similarly, for the adaptive proportional gain of the PI controller of the MTFC algorithm, the following equations were evaluated:

- $K'_{\mathrm{P}}(\tau) = K'_{\mathrm{P}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)})$
- $K'_{\mathrm{P}}(\tau) = K'_{\mathrm{P}}(\frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)})$
- $K'_{\mathrm{P}}(\tau) = K'_{\mathrm{P}}(1 - \zeta(1 - \frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)}))$, where $\zeta \in [0.1, 0.9]$

Finally, for the adaptive integral gain of the PI controller of the MTFC algorithm, the following equations were evaluated:

- $K_{\mathrm{I}}'(\tau) = K_{\mathrm{I}}'(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)})$
- $K_{\mathrm{I}}'(\tau) = K_{\mathrm{I}}'(\frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)})$
- $K_{\mathrm{I}}'(\tau) = K_{\mathrm{I}}'(1 - \zeta(1 - \frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)}))$, where $\zeta \in [0.1, 0.9]$

In all different cases the same value was used for the initial critical density value. Also, the same values have been used for the constant gains K_{I}, K_{P}' and K_{I}'.

3-3-3 Adaptive Logic-Based control algorithm for Variable Speed Limits (LB-VSL)

The LB-VSL algorithm is presented in section 2-4-2. The controller uses a feed-forward control structure, which aims to prevent the congestion at bottleneck's location. The algorithm computes the total number of vehicles that should be held up or released, in order to maximize the outflow at the bottleneck's location. The are two parameters which need to be tuned for the implementation of the LB-VSL algorithm. Those are the $\overline{C_{\mathrm{B}}}$ and $\underline{C_{\mathrm{B}}}$. The former usually takes values around the capacity of the bottleneck, while the latter is set between the capacity of the bottleneck and the congested outflow of the bottleneck. In this section, those two parameters are updated adaptively. The update rule is based on the value of the estimated critical density produced by the PE, SDE and KFE estimation methods. Below are presented the equations of the adaptive LB-VSL control algorithm.

In the first equation, Eq. (3-18), $V_{\mathrm{hold}\,,1}$ corresponds to the total number of vehicles that should be held back.

$$V_{\mathrm{hold,1}}(k) = \max\left[0, T_{\mathrm{ff}}\,(k)\left(Q_{\mathrm{iB}}\,(k) - \overline{C_{\mathrm{B}}}(\tau)\right) - \lambda_{\mathrm{B}}\mathrm{L}_{\mathrm{B}}\left(\hat{\rho}_{\mathrm{crit,B}}(\tau) - \rho_{\mathrm{B}}(k)\right)\right] \tag{3-18}$$

where

- $\overline{C_{\mathrm{B}}}(\tau)$ is updated on-line in each controller's time step and
- $\hat{\rho}_{\mathrm{crit,B}}(\tau)$ is the estimate of bottleneck's critical density

The rest of the parameters mentioned in Eq. (3-18) are explained in section 2-4-2.

For the adaptive parameter $\overline{C_{\mathrm{B}}}(\tau)$, the following equation has been used.

$$\overline{C_{\mathrm{B}}}(\tau) = \overline{C_{\mathrm{B}}}(1 - 0.4(1 - \frac{\hat{\rho}_{\mathrm{crit,B}}(\tau)}{\rho_{\mathrm{crit,B}}(0)})) \tag{3-19}$$

In the second equation, Eq. (3-20), $V_{\mathrm{rel,1}}$ corresponds to the total number of vehicles that have to be released.

$$V_{\mathrm{rel,1}}(k) = \max\left[0, -T_{\mathrm{ff}}(k)\left(Q_{\mathrm{iB}}(k) - \underline{C}_{\mathrm{B}}(\tau)\right) + \lambda_{\mathrm{B}}L_{\mathrm{B}}\left(\hat{\rho}_{\mathrm{crit,B}(\tau)} - \rho_{\mathrm{B}}(k)\right)\right] \tag{3-20}$$

where

- $\underline{C_{\mathrm{B}}}(\tau)$ is updated on-line in each controller's time step and
- $\hat{\rho}_{\mathrm{crit,B}}(\tau)$ is the estimate of bottleneck's critical density

The rest of the parameters mentioned in Eq. (3-20) are explained in section 2-4-2.

For the adaptive parameter $\underline{C_{\mathrm{B}}}(\tau)$, the following equation has been used, after trial-and-error.

$$\underline{C_{\mathrm{B}}}(\tau) = \underline{C_{\mathrm{B}}}(1 - 0.4(1 - \frac{\hat{\rho}_{\mathrm{crit,B}}(\tau)}{\rho_{\mathrm{crit,B}}(0)})) \tag{3-21}$$

The same temporal VSL implementation constraint as in the case of MTFC algorithm is also applied for the LB-VSL controller. This constraint can be seen is Eq. (3-12). The minimum and maximum allowable limit for the applied VSL is already included in the implementation of the original LB-VSL algorithm, as it can be seen in Eq. (2-18) and Eq. (2-19).

3-3-4 LB-VSL parameters' equations

Similarly to the MTFC case, for each one of the adaptive parameters' equations of the LB-VSL algorithm, Eq. (3-19) and Eq. (3-21), three different equations were evaluated. All combinations were considered and the selected ones were shown to result in the lowest value of TTS for all different scenarios. The three different equations used for each adaptive parameter of the LB-VSL algorithm are presented below.

For the adaptive parameter $\overline{C_{\mathrm{B}}}(\tau)$, the following equations were evaluated:

- $\overline{C_{\mathrm{B}}}(\tau) = \overline{C_{\mathrm{B}}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)})$
- $\overline{C_{\mathrm{B}}}(\tau) = \overline{C_{\mathrm{B}}}(\frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)})$
- $\overline{C_{\mathrm{B}}}(\tau) = \overline{C_{\mathrm{B}}}(1 - \zeta(1 - \frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)}))$, where $\zeta \in [0.1, 0.9]$

Finally, for the adaptive parameter $\underline{C_{\mathrm{B}}}(\tau)$, the following equations were evaluated:

- $\underline{C_{\mathrm{B}}}(\tau) = \underline{C_{\mathrm{B}}}(\frac{\hat{\rho}_{\mathrm{crit}}(\tau)}{\rho_{\mathrm{crit}}(0)})$
- $\underline{C_{\mathrm{B}}}(\tau) = \underline{C_{\mathrm{B}}}(\frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)})$
- $\underline{C_{\mathrm{B}}}(\tau) = \underline{C_{\mathrm{B}}}(1 - \zeta(1 - \frac{\rho_{\mathrm{crit}}(0)}{\hat{\rho}_{\mathrm{crit}}(\tau)}))$, where $\zeta \in [0.1, 0.9]$

As in the case of the MTFC algorithm, the same value was used for the initial critical density value. Also, the same values have been used for the constant parameters $\overline{C_{\mathrm{B}}}$ and $\underline{C_{\mathrm{B}}}$.

3-4 Control Parameters

In this section, the control parameters are reported. The parameters which need to be selected correspond to the estimation methods for the online critical density estimation, as well as to the adaptive easy-to-implement VSL algorithms and the VSL implementation constraints. Below tables summarize all parameters.

Method	Control Parameter	Units	Description
PE	T	–	interval length for the derivative
	β^-	$\frac{km}{h}$	lower threshold value of derivative for critical density update
	β^+	$\frac{km}{h}$	upper threshold value of derivative for critical density update
	α	–	weighing smoothing factor
SDE	$\rho_{\text{crit}}^{\text{min}}$	$\frac{veh}{km \cdot lane}$	minimum value for the critical density
	$\rho_{\text{crit}}^{\text{max}}$	$\frac{veh}{km \cdot lane}$	maximum value for the critical density
	K	sec	interval for critical density reduction
	Δ	$\frac{veh}{km \cdot lane}$	value for the critical density reduction every K seconds
	P	$\frac{veh}{km \cdot lane}$	threshold value for critical density estimation update
	ϵ	$\frac{veh}{km \cdot lane}$	threshold value for difference in two consecutive density measurements
	δ_{max}	$\frac{km}{h}$	threshold value for time derivative
	δ_{min}	$\frac{km}{h}$	threshold value for time derivative
	α	–	constant smoothing parameter
	D^+	$\frac{veh}{km \cdot lane}$	threshold value for derivative estimate
	D^-	$\frac{veh}{km \cdot lane}$	threshold value for derivative estimate
KFE	$q_{\text{cap}}^{\text{est}}$	$\frac{veh}{h}$	estimate of the bottleneck's capacity
	$\mathbf{Z}$	-	covariance matrix of system noise
	$\mathbf{W}$	-	covariance matrix of output noise
	$\rho_{\text{crit}}^{\text{min}}$	$\frac{veh}{km \cdot lane}$	minimum value for the critical density
	$\rho_{\text{crit}}^{\text{max}}$	$\frac{veh}{km \cdot lane}$	maximum value for the critical density
	K	sec	interval for critical density reduction
	Δ	$\frac{veh}{km \cdot lane}$	value for the critical density reduction every K seconds
	P	$\frac{veh}{km \cdot lane}$	threshold value for critical density estimation update
	D^+	$\frac{veh}{km \cdot lane}$	threshold value for derivative estimate
	D^-	$\frac{veh}{km \cdot lane}$	threshold value for derivative estimate

Table 3-1: Estimation methods' parameters

Method	Control Parameter	Units	Description
Adaptive MTFC	v_f	$\frac{km}{h}$	free-flow speed
	K_{I}	$\frac{h}{veh \cdot lane}$	gain of I controller
	K'_{P}	$\frac{km}{h \cdot lane}$	proportional gain of PI controller
	K'_{I}	$\frac{km}{h \cdot lane}$	integral gain of PI controller
	$\rho_{\mathrm{crit}}(0)$	$\frac{veh}{km \cdot lane}$	initial guess for critical density
	q_{min}	$\frac{km}{h}$	lower threshold for flow
	q_{max}	$\frac{km}{h}$	upper threshold for flow
Adaptive LB-VSL	α_j	-	compliance parameter of segment j
	L_j	km	length of segment j
	λ	-	number of lanes in the VSL application area
	$\hat{L}_i$	km	distance between detectors
	L_A	km	distance between bottleneck and the most upstream segment with VSL installed
	λ_{B}	-	number of lanes at the bottleneck location
	$\mathrm{L_B}$	km	bottleneck's length
	$\underline{C_{\mathrm{B}}}$	$\frac{veh}{h}$	tuning parameter between the congested outflow of the bottleneck (capacity minus capacity drop) and the capacity of the bottleneck
	$\overline{C_{\mathrm{B}}}$	$\frac{veh}{h}$	tuning parameter (around bottleneck's capacity)
	$\rho_{\mathrm{crit}}(0)$	$\frac{veh}{km \cdot lane}$	initial critical density guess
VSL Constraints	γ	$\frac{km}{h}$	maximum difference in the VSL values displayed by the same gantry between two consecutive simulation time step
	VSL_{min}	$\frac{km}{h}$	minimum allowable VSL value
	VSL_{max}	$\frac{km}{h}$	maximum allowable VSL value

Table 3-2: Controllers' parameters

3-5 Conclusions

In this chapter an introduction to the framework used in this chapter is provided. The well-known easy-to-implement VSL algorithms MTFC and LB-VSL are modified in an adaptive framework, in order to maximize bottleneck's outflow. The critical density at the bottleneck location and the controllers' parameters are estimated on-line.

More specifically, the critical density at the bottleneck's location is estimated online using three estimation methods. The first one uses the PE method together with the updated rule for the critical density from the AD-RMC algorithm. The second method uses a modified version of the SDE algorithm, in which the critical occupancy is replaced by the critical density. Similar approach is followed for the third estimation method, which is a modified version of the KFE algorithm.

Regarding the adaptivity added to the MTFC algorithm, the three controllers' gains, K_{I}, K'_{P} and K'_{I}, are considered to be updated online is each controller's time step, using the estimated value of the critical density and an initial guess of the critical density.

For the LB-VSL algorithm, the two tuning parameters $\overline{C_{\mathrm{B}}}$ and $\underline{C_{\mathrm{B}}}$ are chosen to be updated adaptively in a similar way, using again the estimated value of the critical density.

Finally, all control parameters are summarized.

Chapter 4

Case study

The adaptive easy-to-implement Variable Speed Limit algorithms proposed in Chapter 3 are evaluated in this chapter. This is done by comparing their performance under two different scenarios, namely under accident and rain conditions. The chapter is structured as follows. Section 4-1 discusses the case study details. Section 4-2 describes the parameters of the estimation methods that need to be defined. The next two sections describe the case studies in detail. In Section 4-3 the accident scenario is studied, while in section 4-4 rain conditions are considered in the traffic network. Lastly, the chapter is concluded in section 4-5.

4-1 Case study details

In this section the details of the case study are discussed. Firstly, the freeway traffic network used in the simulations is described. Then, the performance criteria used to evaluate the control algorithms are presented. Subsequently, the computer specifications used to run the simulations are mentioned. Lastly, the no-control simulation result is presented to be used as benchmark.

4-1-1 Freeway traffic network

The freeway traffic network used in [11], shown in figure 4-1, is used in this work to evaluate the performance of the proposed control algorithms. It is an hypothetical 12 km long freeway stretch, consisted of N=12 segments and λ_i=3 lanes. Each segment has a length of L_i=1 km. The 4th segment has an on-ramp and the 11th has a lane drop. Segment 11, which has 2 lanes, is considered as a bottleneck, in which congestion is created once the demand is high enough. Two segments of the considered traffic network are equipped with VSLs, namely segment 5 and 6. The VSLs can only take a limited number of discrete values in the range of {40, 50, 60, 70, 80, 90, 100}. Also, they can either increase or decrease by 10 km/h for each controller time step. For those implementation constraints the MATLAB function floor has

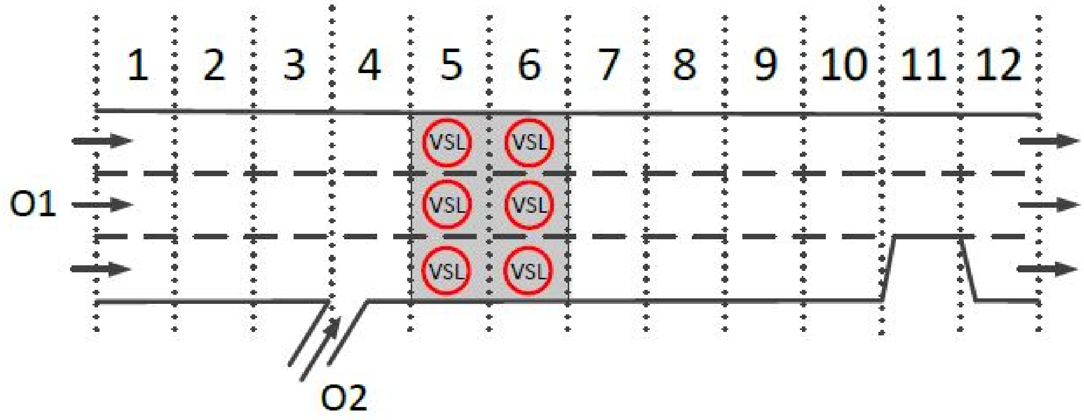

Figure 4-1: Freeway traffic network used

α	μ_H	μ_L	ϕ	K	δ	a
2	80	20	0.1	40	0.01	0.1

Table 4-1: METANET parameters

been used in the equations 2-18 and 2-19 for the LB-VSL controller and in the equation 2-26 for the MTFC controller.

The considered network is simulated using the METANET model. The VSL model proposed in [20] is used to include the effects of VSLs. All network's segments have the same METANET parameters. The simulation time is chosen to be three hours. The length of the controllers' time step is $T_c = 60s$, while the length of the simulation time step is $T = 10s$. So, 180 controller sample steps and 1080 simulation steps are presented.

The on-ramp of segment 4 has a capacity of $C_r = 2000\frac{veh}{h}$, the free-flow speed is $v_f = 110\frac{km}{h}$, the critical density is $\rho_c = 32\frac{veh}{(km \cdot lane)}$, the maximum density is $\rho_m = 180\frac{veh}{(km \cdot lane)}$, and $\tau = 18s$ is the time constant. In Table 4-1, the reader can find the rest of METANET's parameters.

As shown in figure 4-2, the demand scenario used for the ramp has a constant value of $500\frac{veh}{h}$, while for the mainline, a demand scenario which creates congestion for the second hour of the simulation is used. The used demand scenario creates no queues on the on-ramp or on the mainline.

For the implementation of the LB-VSL controller, flow and speed detectors are considered for segments 5 to 10, while for MTFC controller, segment's 11 density and segment's 6 flow are measured.

The initial guess for the critical density of the bottleneck is considered $32\frac{veh}{km \cdot lane}$. Then, it is estimated on-line. The initial guess is done using METANET's fundamental diagram.

4-1-2 Performance criterion

The goal of this thesis is to develop adaptive easy-to-implement VSL control algorithms for freeway traffic networks, by estimating on-line the bottleneck's critical density, in order to

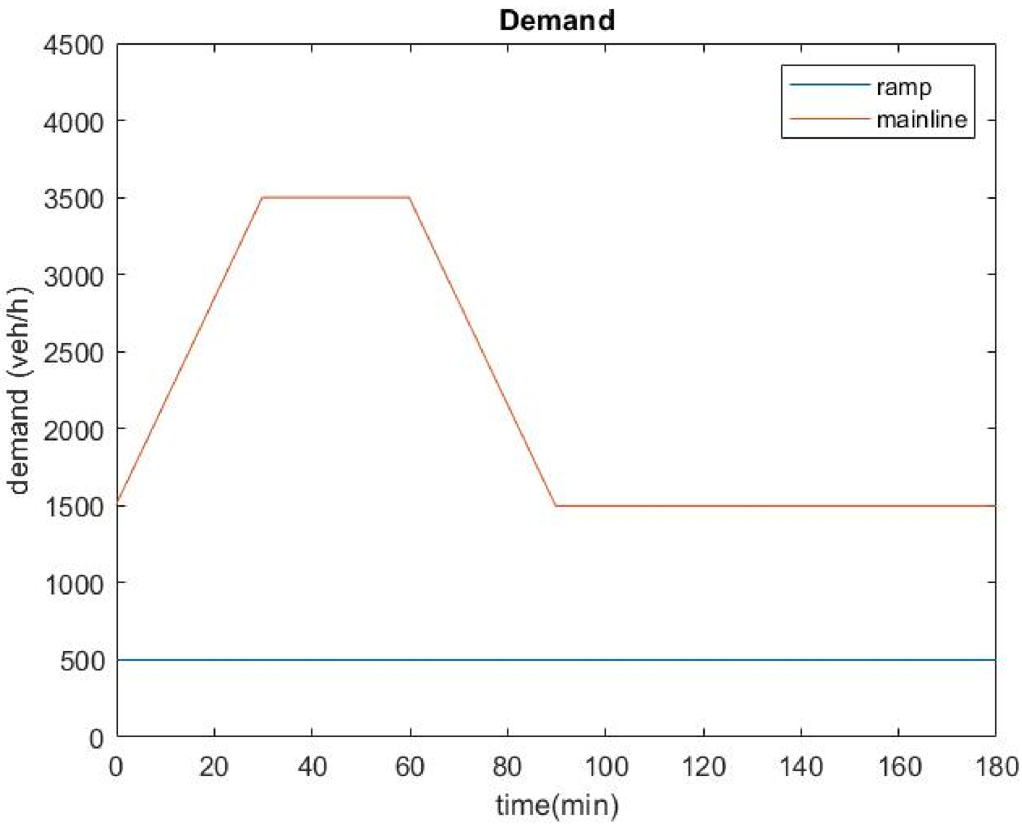

Figure 4-2: Mainline and on-ramp demands

increase freeway traffic efficiency. To measure the efficiency of the controllers, the performance indicator used is the Total Time Spent (TTS) by all drivers in the network, including the waiting time experienced on the on-ramp and the mainline queue. It is computed using the following equation 4-1 [20].

$$TTS = \sum_{k=1}^{180}(T\sum_{j=1}^{2} w_j(k) + T\sum_{i=1}^{12}(\rho_i(k)L\lambda)) \tag{4-1}$$

Furthermore, to evaluate the methods used to estimate online the critical density at the bottleneck location, the mean absolute estimation error was used. For the calculation of this quantity, the equation 4-2 was used.

$$MAER = \frac{1}{180}\sum_{k=1}^{180}(|\rho_{crit}^{real} - \tilde{\rho}_{\text{crit}}|) \tag{4-2}$$

4-1-3 Computer specifications

All the simulations in this work are conducted on a Toshiba Satellite $P50 - C$, containing an Intel Core i7 processor and 12GB of RAM. The simulations are evaluated in MATLAB R2019b.

4-1-4 Congestion scenarios

Two different congestion scenarios are considered in this case study. In the first scenario, an accident on segment 11 (bottleneck) of the traffic network is simulated. The accident happens at the start of the second simulation hour and lasts for 30mins ($t = 60$ to 90 min). For the accident's simulation, the critical density of segment 11 is reduced from $32\frac{veh}{km \cdot lane}$ to $22\frac{veh}{km \cdot lane}$. This is affecting the driver's desired speed and creates congestion at the bottleneck location. In the second scenario, heavy rain is considered in the whole traffic network, during the second hour of the simulation ($t = 60$ to 120 min). To simulate the rain conditions, the critical density of all network's segments is reduced during the second hour from $32\frac{veh}{km \cdot lane}$ to $20\frac{veh}{km \cdot lane}$. Due to the rain conditions, higher delays are observed for the network's drivers, who drive at lower speeds.

4-1-5 No-control simulation result

When simulating the traffic network without any control measure, congestion appears in the freeway for both scenarios, as it can be seen in the figures 4-3, 4-4, 4-5, 4-6. The congestion appears during the second hour of the simulation and is resolved during the third hour. So, during the third hour, free flow conditions prevail in the whole traffic network. The TTS for the no-control case can be found in the table 4-2.

Figure 4-3 shows the speeds in all the freeway traffic network's segments in the accident case, while figure 4-4 presents the densities. As it can be observed, congestion appears in segments 7 to 12.

No-control simulation scenario	TTS $(veh \cdot hours)$
scenario accident	1244.23
scenario rain	1335.05

Table 4-2: TTS comparison, no-control in accident and rain scenarios

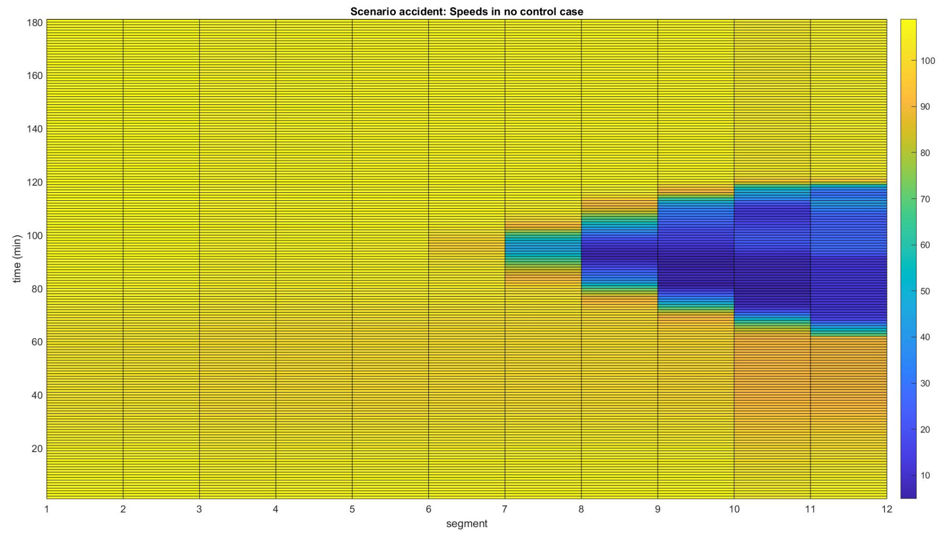

Figure 4-3: Scenario accident - speeds in no-control case

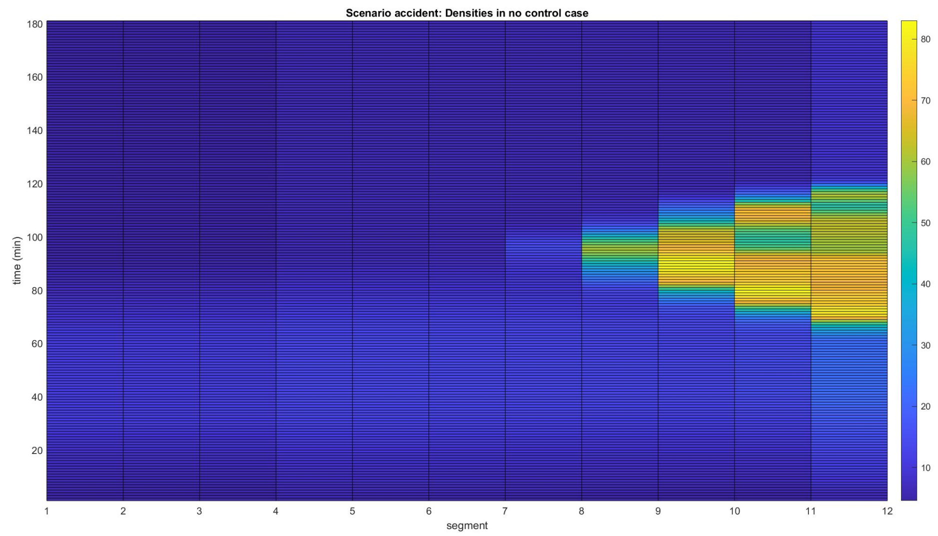

Figure 4-4: Scenario accident - densities in no-control case

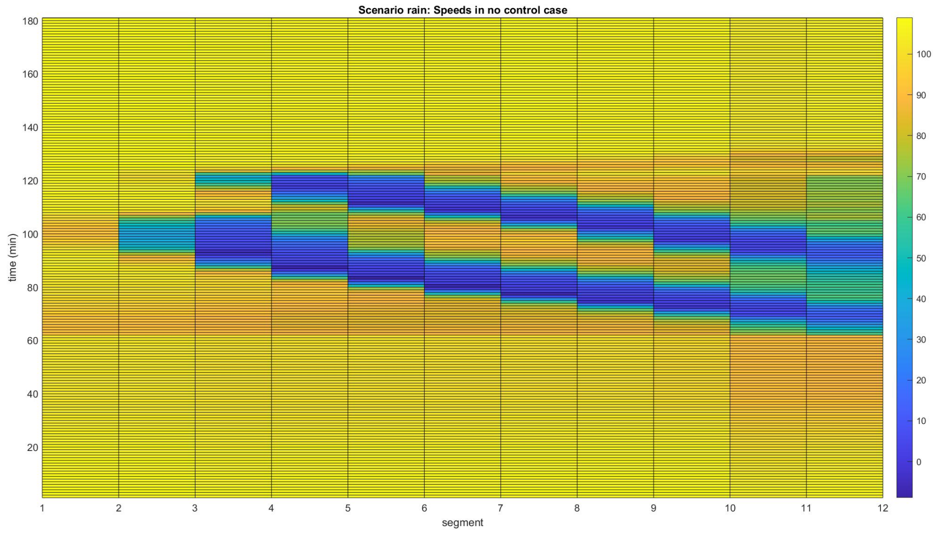

Figure 4-5: Scenario rain - speeds in no-control case

Similarly, for the rain scenario, figures 4-5 and 4-6 present the speeds and densities respectively in the whole network when no control is applied. As it can be seen, congestion appears in all the segments of the freeway traffic network.

4-2 Parameters of the estimation methods

In this section, the choices made for the parameters of the estimation methods are reported. The choices can be different based on the congestion scenario.

PE For the PE estimation methods, four parameters have to be decided. In both accident and rain scenarios all parameters are same for the MTFC and LB-VSL controllers. Below table presents the values chosen for the PE algorithm.

Parameter	Value
T	6
β^-	$-10\frac{km}{h}$
β^+	$80\frac{km}{h}$
α	0.5

Table 4-3: PE parameters

The choice of the above parameters for PE was made by trial-and-error. According to [32], the absolute value of β^- should be smaller than the value of β^+, because of the asymmetric shape of the FD. As for the smoothing parameter α, its value depends on the value of the time interval T, or else on the number of the last density and flow measurements used [23]. The value of the interval length of the derivative, T, is chosen to be determined over six

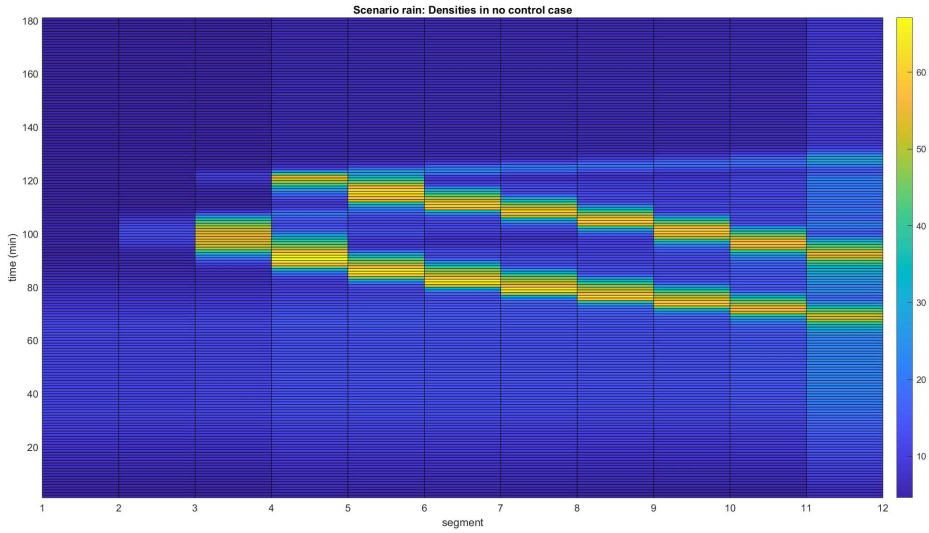

Figure 4-6: Scenario rain - densities in no-control case

preceding measurements, which corresponds to 1 min time, given that the simulation time step is 10 sec. The same value was also used in [23].

Simple Derivative Estimation (SDE) In the case of the SDE method, all eleven parameters are considered the same for all four scenario cases. Below table presents the values chosen for the SDE algorithm.

Parameter	Value
α	0.99
D^+	$20\frac{veh}{km{\cdot}lane}$
D^-	$-10\frac{veh}{km{\cdot}lane}$
$\rho_{\text{crit}}^{\text{min}}$	$20\frac{veh}{km{\cdot}lane}$
$\rho_{\text{crit}}^{\text{max}}$	$40\frac{veh}{km{\cdot}lane}$
K	$600s$
Δ	$5\frac{veh}{km{\cdot}lane}$
P	$2\frac{veh}{km{\cdot}lane}$
ϵ	$0.1\frac{veh}{km{\cdot}lane}$
δ_{min}	$-100\frac{km}{h}$
δ_{max}	$100\frac{km}{h}$

Table 4-4: SDE parameters

As in the case of PE, the parameters of the SDE algorithm were chosen by trial-and-error. However, the logic behind the selection of the parameters was based on [23], in which occupancy measurements are used in the algorithm instead of density measurements. For the values of D^+ and D^-, the rule $|D^+| > |D^-|$ was used, which is due to the non-symmetric form of the FD around the critical density. Low threshold values were used for both the upper and lower limit of the derivative estimates. The lower and upper limit for the value of the critical density were selected based on reasonable limits for free-flow and congested states, respectively, of the selected freeway traffic network. A high value was selected for α, smoothing parameter for the exponential filter, as "a strong smoothing calls for lower threshold magnitude and vice versa" [24]. Regarding the values of δ_{min} and δ_{max}, although $\delta(k)$ was never found to reach these limits, it was deemed necessary to introduce the limits to prevent $D(k)$ from taking unreasonably high or low values in the case of corrupted data. The value of P, threshold value for the critical density estimation update, was selected to be the same for both SDE and KFE algorithms. A selection of a small value for P indicates that there will be no derivative update (and critical density estimation) if the current density measurement and the last estimate of the critical density are not close to each other, which seems reasonable so that density measurements that are not close to critical density are excluded. Regarding the selection of the values for the parameters K and Δ, $K = 600sec$ and $\Delta = 5\frac{veh}{km \cdot lane}$ were considered as appropriate. Finally, a very small value was selected for ϵ, as "no modification in the estimation of the critical density should be allowed when two consecutive density measurements are very close to each other, because in this case the time derivative $\delta(k)$ may not be a reliable estimate of the actual derivative" [24].

Kalman-Filter-based derivative Estimation (KFE) For the KFE algortihm, ten parameters have to be chosen. As it can be seen in the following table, the parameters' values are same for all different scenarios. The output noise $w(k)$ is considered to be zero, so the covariance matrix is $W = 0$. The system noise follows normal distribution with mean value 0 and standard deviation 0.5.

Parameter	Value
$q_{\text{cap}}^{\text{est}}$	$6000\frac{veh}{h}$
Z	$Z \sim \mathcal{N}(0, 0.5)$
W	0
$\rho_{\text{crit}}^{\text{min}}$	$20\frac{veh}{km \cdot lane}$
$\rho_{\text{crit}}^{\text{max}}$	$40\frac{veh}{km \cdot lane}$
K	$600s$
Δ	$5\frac{veh}{km \cdot lane}$
P	$2\frac{veh}{km \cdot lane}$
D^+	$20\frac{veh}{km \cdot lane}$
D^-	$-10\frac{veh}{km \cdot lane}$

Table 4-5: KFE parameters

The parameters of the KFE algorithm which are also used in the SDE algorithm were selected to be the same in both algorithms. The logic behind the selection procedure is explained in the previous paragraph. For the initial value of E, $E(0)$, which is equal to an estimate of the bottleneck's capacity $q_{\text{cap}}^{\text{est}}$, it was selected to be used the same value used as maximum flow limit in the MTFC algorithm.

4-3 Case study A : Accident scenario

In this section, the results of the first case study are presented. An accident in segment 11 is simulated for 30 minutes, during the first half of the second hour of the simulation. The critical density at the bottleneck location is reduced for 30 mins from $32\frac{veh}{km \cdot lane}$ to $22\frac{veh}{km \cdot lane}$. The initial density guess at the bottleneck location is $32\frac{veh}{km \cdot lane}$.

In the first subsection, the MTFC algorithm is simulated with the full knowledge of the critical density at each simulation step. In each one of the next three sections, one estimation method is combined with the MTFC algorithm. Then, the same structure is followed for the case of the LB-VSL algorithm. First the LB-VSL algorithm is simulated using the real values of the critical density, while in the last three sections the controller is combined with the three estimation methods. The chosen parameters are given below for each controller.

The TTS by all drivers in the network is presented in each subsection. Also, speed and density figures are used to capture the controllers' effect on the freeway traffic network. Finally, the absolute estimation error is calculated for all estimation methods.

The MTFC controller's parameters in the accident case can be found in table 4-6.

Controller	Control Parameter
Adaptive MTFC	$v_f = 100\frac{km}{h}$ $K_{\mathrm{I}} = 0.0707\frac{h}{veh \cdot lane}$ $K'_{\mathrm{P}} = 109.49\frac{km}{h \cdot lane}$ $K'_{\mathrm{I}} = 2.6502\frac{km}{h \cdot lane}$ $\hat{\rho}_{\mathrm{crit}}(0) = 32\frac{veh}{km \cdot lane}$ $q_{min} = 1000\frac{km}{h}$ $q_{max} = 6000\frac{km}{h}$

Table 4-6: Accident scenario: MTFC Controller's parameters

The LB-VSL controller's parameters in the accident case can be found in table 4-7.

Controller	Control Parameter
Adaptive LB-VSL	$\alpha_j = 0.1$ $L_j = 1km$ $\lambda = 3$ $\hat{L}_i = 1km$ $L_A = 6km$ $\lambda_{\mathrm{B}} = 2$ $\mathrm{L_B} = 1km$ $\underline{C}_{\mathrm{B}} = 3171.8\frac{veh}{h}$ $\overline{C}_{\mathrm{B}} = 4514.2\frac{veh}{h}$ $\hat{\rho}_{\mathrm{crit}}(0) = 32\frac{veh}{km \cdot lane}$

Table 4-7: Accident scenario: LBVSL Controller's parameters

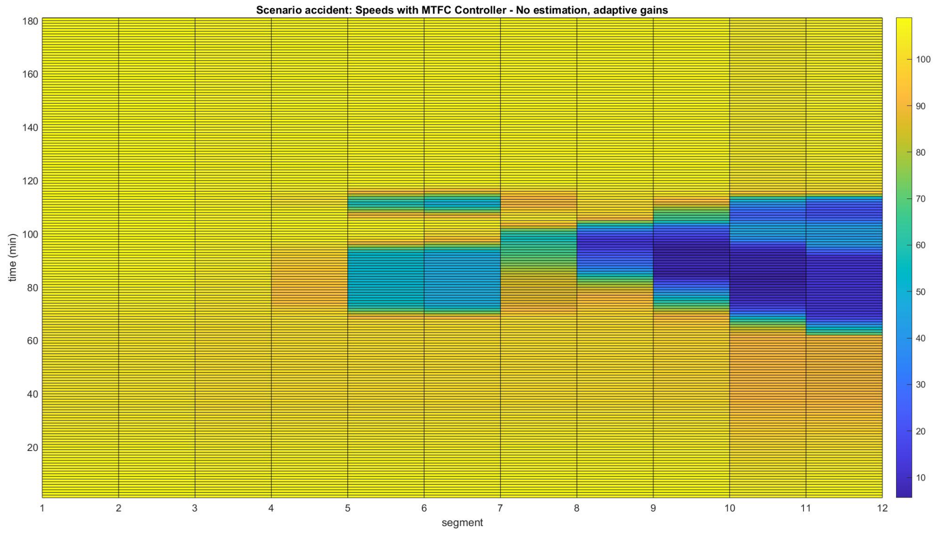

Figure 4-7: Scenario accident - speeds with adaptive MTFC - full knowledge of ρ_{crit}

4-3-1 Mainstream Traffic Flow Control (MTFC) + No estimation

In this section, the MTFC algorithm uses the real values of the critical density at each instant. The results are presented below.

First, the speeds of all network's segments are shown in figure 4-7. As it can be observed, compared to the no-control case, the congestion is resolved earlier in time. However, when the adaptive MTFC algorithm is applied, the congestion appears also in segments 4,5 and 6.

Figure 4-8 presents network's densities with the adaptive MTFC controller knowing the critical density during the whole simulation time. As in the case of the speeds' figure, the adaptive MTFC controller is able to provide free-flow conditions earlier in the traffic network, compared to the no-control case.

The TTS for all the drivers in the traffic network was found to be $1200.71 veh \cdot hours$, which is a 3.49% decrease compared to the no-control case.

4-3-2 Mainstream Traffic Flow Control (MTFC) + Parameter Estimator (PE)

In this section, the PE method is combined with the MTFC algorithm and the results are presented below. In the figure 4-9, it can be observed that, using an accurate initial guess for the bottleneck's critical density, the PE estimation method seems to be able to track the reduction of the critical density after the first simulation hour. However, the estimation algorithm cannot track properly the critical density increase after the decrease, which leads to undercritical conditions, meaning that the estimated critical density is lower than the real one.

Figures 4-10 and 4-11 shows the speeds and densities for all the segments of the traffic network, when the adaptive MTFC algorithm is combined with the estimation method PE. We can observe lighter congestion at the bottleneck location, compared to the no-control case.

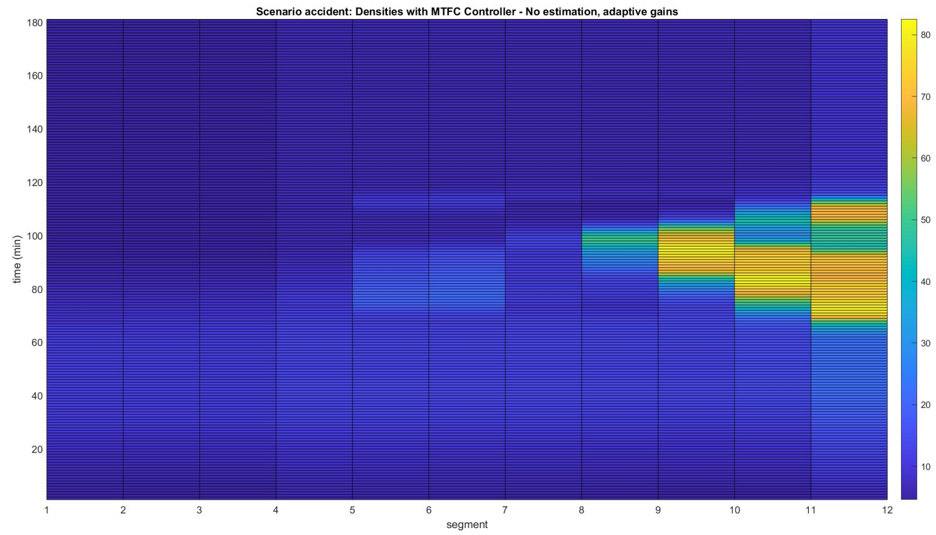

Figure 4-8: Scenario accident - densities with adaptive MTFC - full knowledge of ρ_{crit}

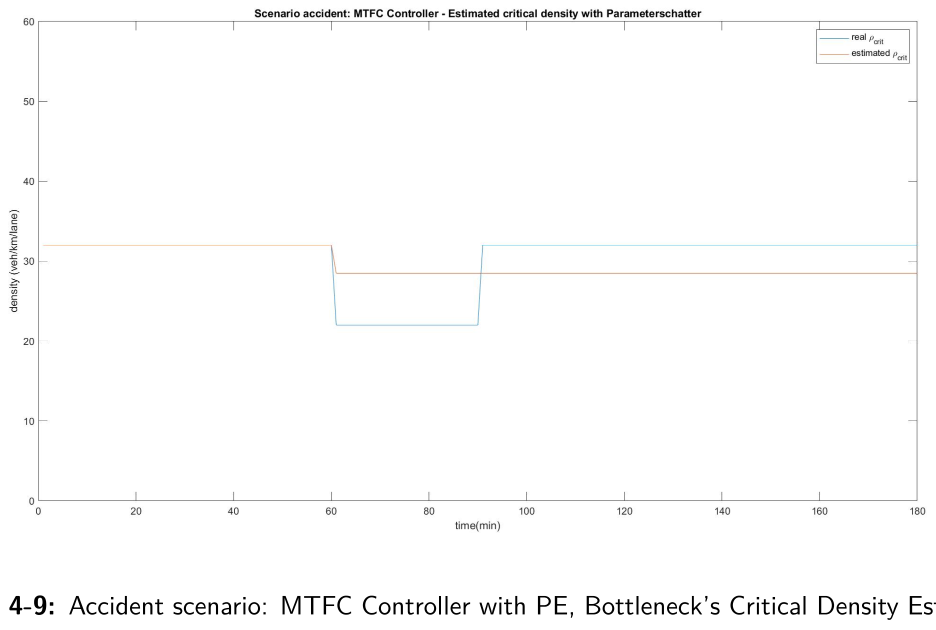

Figure 4-9: Accident scenario: MTFC Controller with PE, Bottleneck's Critical Density Estimation

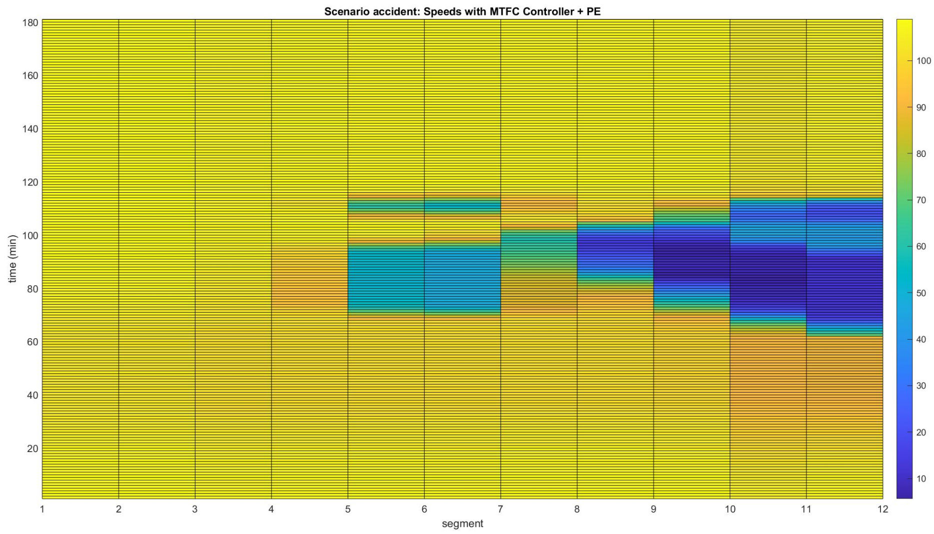

Figure 4-10: Scenario accident: speeds with MTFC Controller and PE

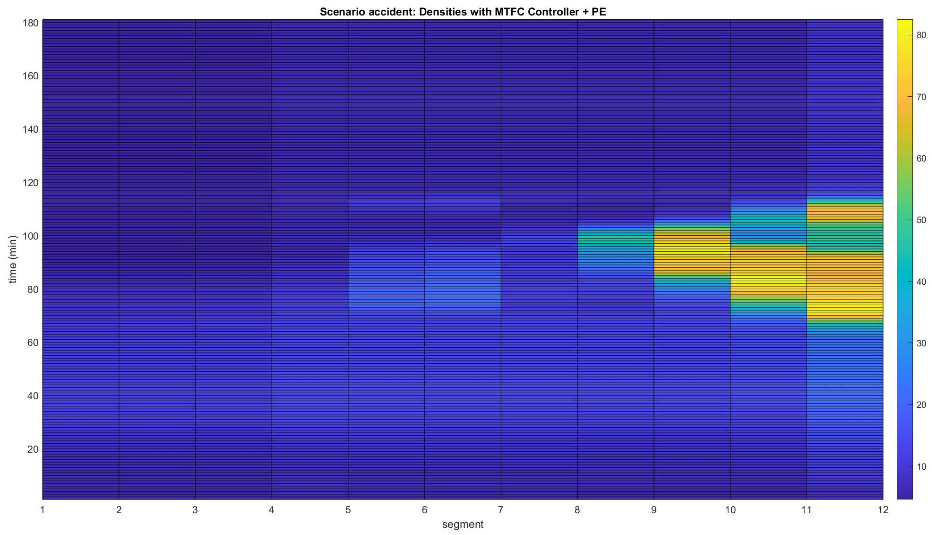

Figure 4-11: Scenario accident: densities with MTFC Controller and PE

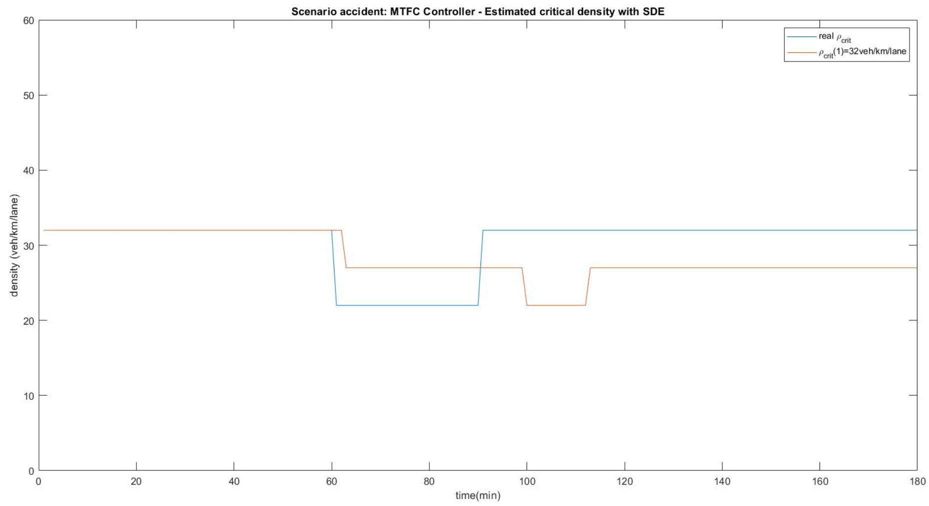

Figure 4-12: Accident scenario: MTFC Controller with SDE, Bottleneck's Critical Density Estimation

The TTS in this case was found to be $1198.38 veh \cdot hours$, which corresponds to a 3.68% decrease compared to the no-control case.

Regarding the mean absolute estimation error it was found to be $2.84 \frac{veh}{km \cdot lane}$.

4-3-3 Mainstream Traffic Flow Control (MTFC) + Simple Derivative Estimation (SDE)

In this section, the SDE method is combined with the MTFC algorithm. The results can be seen in figure 4-12.

As it can be seen in figure 4-12, the SDE method can track better, compared to the PE method, the changes of the critical density at the bottleneck location. It can properly track the critical density's decrease during the second hour of the simulation, but the increase once the congestion starts to resolve cannot be identified.

In figures 4-13 and 4-14, the speeds and densities for all the segments of the traffic network are presented, when the adaptive MTFC algorithm is combined with the estimation method SDE. Both figures show very similar results with the PE case.

The TTS in this case was found to be $1199.06 veh \cdot hours$, which corresponds to a 3.63% decrease compared to the no-control case, and slightly lower decrease than the PE case. In addition, the mean absolute estimation error was found to be $3.75 \frac{veh}{km \cdot lane}$, which is higher than the PE case.

4-3-4 Mainstream Traffic Flow Control (MTFC) + Kalman-Filter-based derivative Estimation (KFE)

The KFE method is combined with the MTFC algorithm and the results are presented in this section. Figure 4-17 shows the estimated critical density when the KFE method is combined

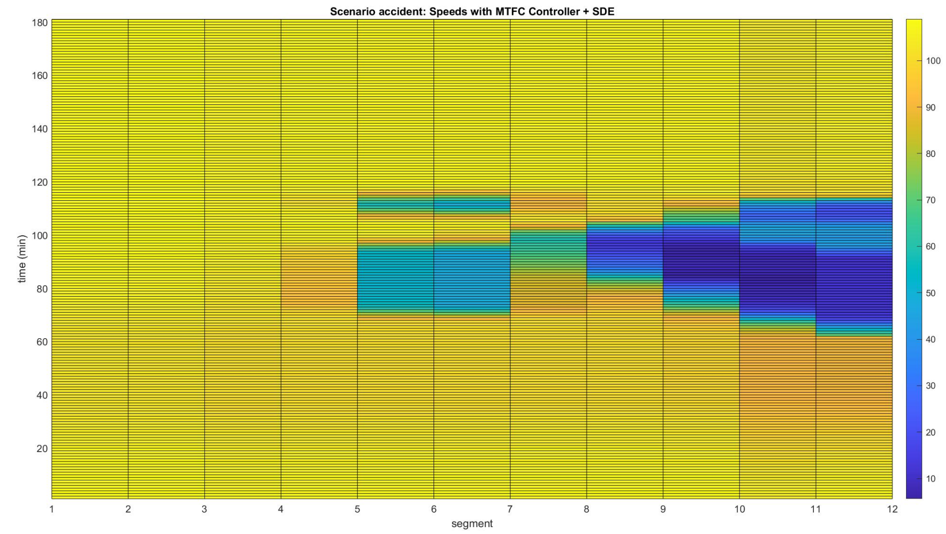

Figure 4-13: Scenario accident: speeds with MTFC Controller and SDE

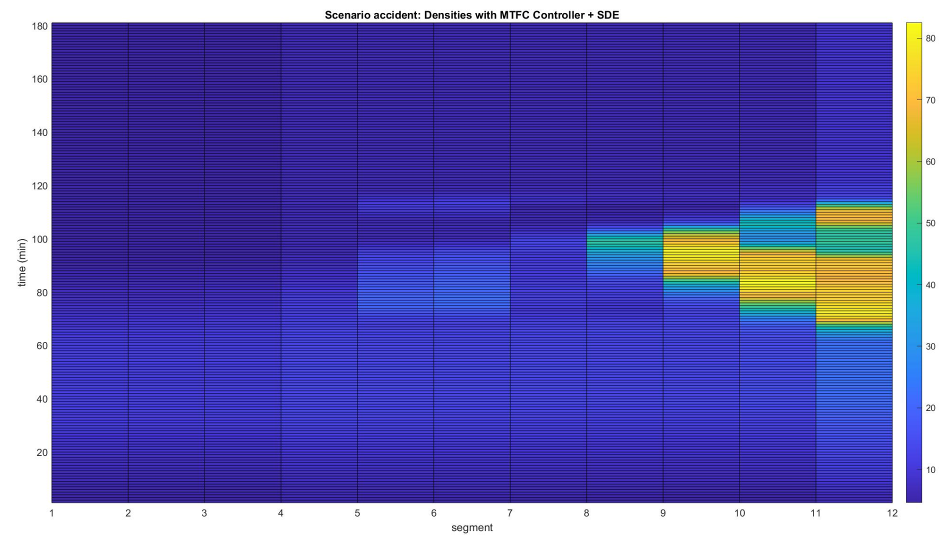

Figure 4-14: Scenario accident: densities with MTFC Controller and SDE

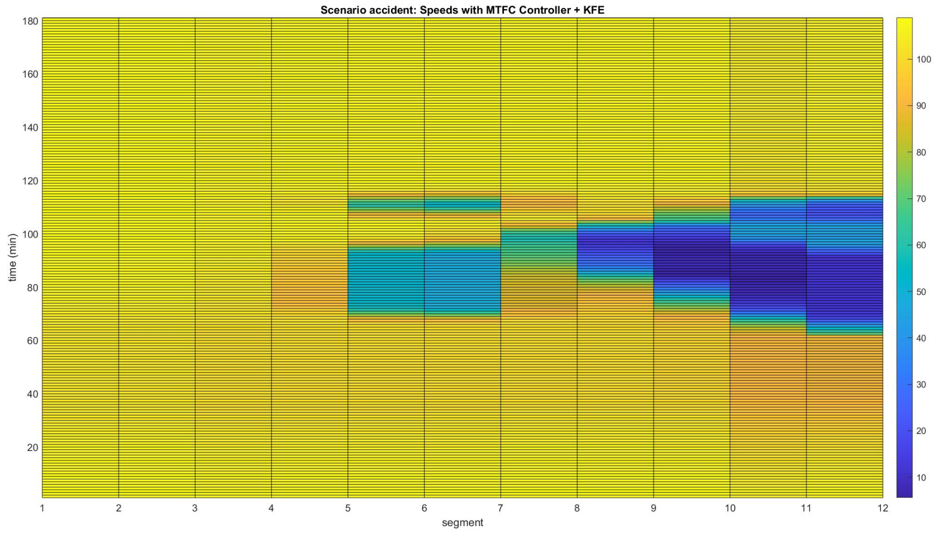

Figure 4-15: Scenario accident: speeds with MTFC Controller and KFE

with the MTFC algorithm for the accident scenario.

The speeds and densities for all network's segments are also presented in figures 4-15 and 4-16 , respectively, in which similar results as in the previous cases can be observed.

The TTS in this case was found to be $1196.27 veh{\cdot}hours$, which corresponds to the lowest value for the accident case with the MTFC algorithm. A 3.85% decrease is observed, compared to the no-control case. Furthermore, the mean absolute estimation error for this case was $3.22\frac{veh}{km{\cdot}lane}$, which is higher than the PE case, but lower than the SDE case.

As it can be seen in the figure 4-17, the KFE algorithm seems not able to track the reduction of the critical density after the first simulation hour. However, the estimation algorithm approximates the critical density after the decrease, during the last hour of the simulation. This happens due to step c of the KFE algorithm, which reduces the estimate every K time units.

4-3-5 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + No estimation

In this section, the LB-VSL algorithm has full knowledge over the real values of the critical density at each instant. The results are presented below.

Figures 4-18 and 4-19 present the speeds and densities, respectively, in all the segments of the freeway traffic network. As it can be observed from the figures, the LB-VSL algorithm shows slightly better results at the bottleneck location, compared to the MTFC algorithm. However, when the MTFC algorithm is used, congestion is created in the VSL application area, which later reduces the inflow to the bottleneck, resulting in better overall results.

The TTS achieved when the adaptive LB-VSL algorithm was used is $1242.94 veh \cdot hours$, which corresponds to a decrease of 0.1%, compared to the no-control case.

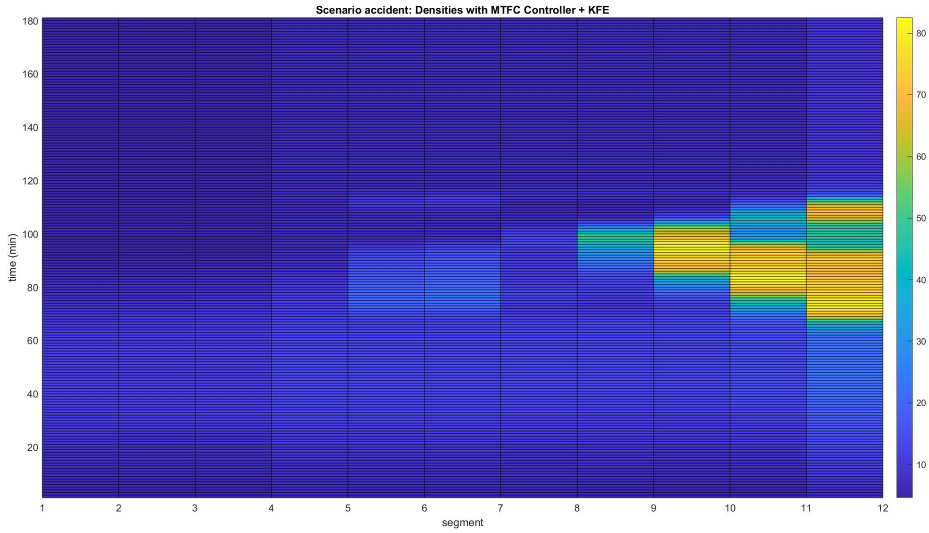

Figure 4-16: Scenario accident: densities with MTFC Controller and KFE

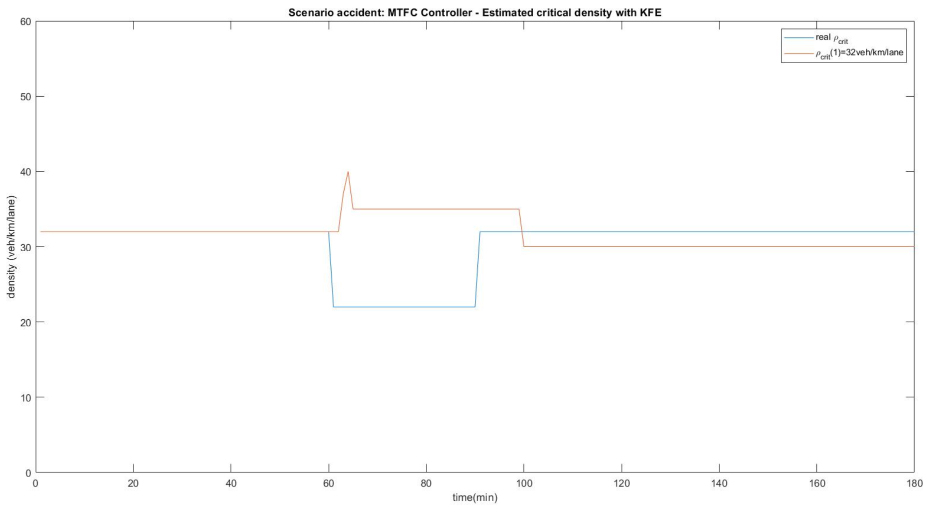

Figure 4-17: Accident scenario: MTFC Controller with KFE, Bottleneck's Critical Density Estimation

4-3-6 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Parameter Estimator (PE)

The PE method is combined with the LB-VSL algorithm and the results are presented below. In the figure 4-20, the bottleneck's critical density is estimated using the PE method. It can be observed that the algorithm can track the reduction of the critical density during the second hour of the simulation. However, the algorithm overestimates the critical density during the second half of the simulation time, resulting in higher values of critical density, compared to the real values.

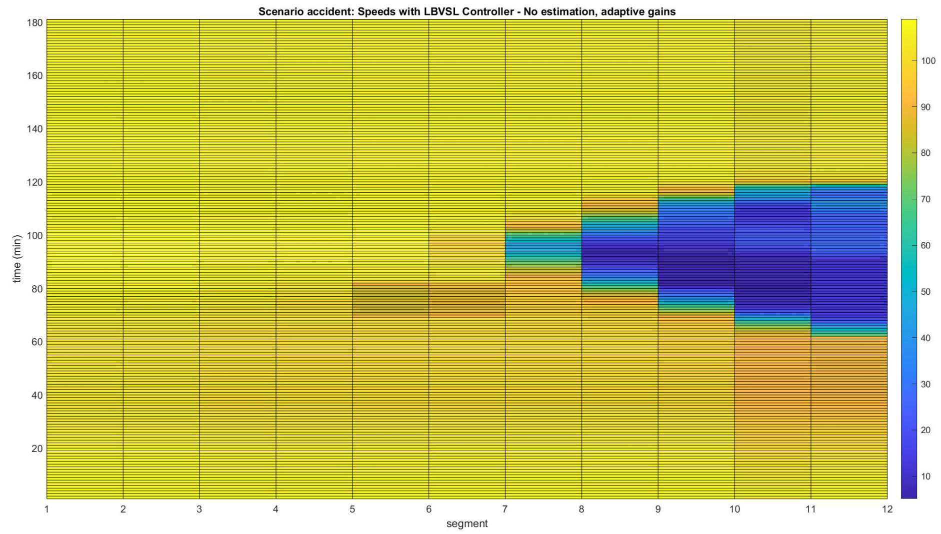

Figure 4-18: Scenario accident - speeds with adaptive LBVSL - full knowledge of ρ_{crit}

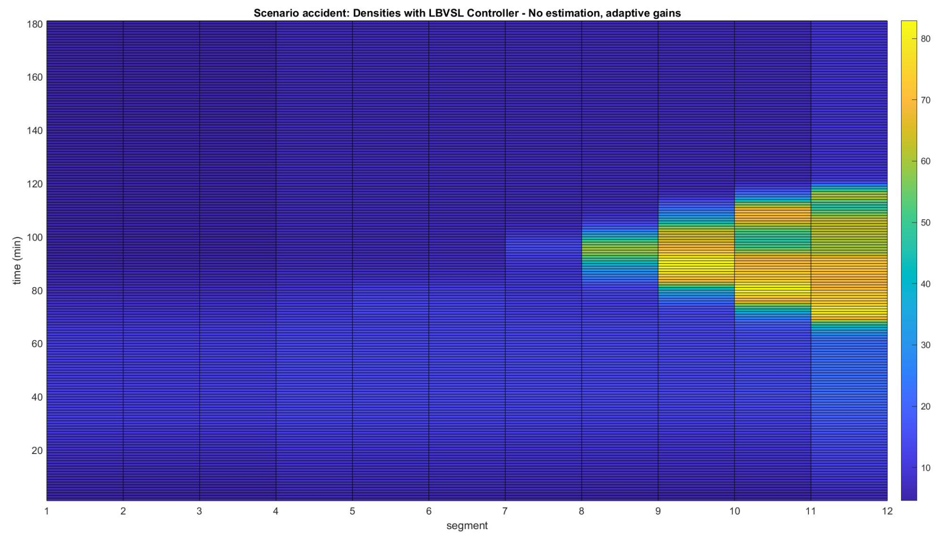

Figure 4-19: Scenario accident - densities with adaptive LBVSL - full knowledge of ρ_{crit}

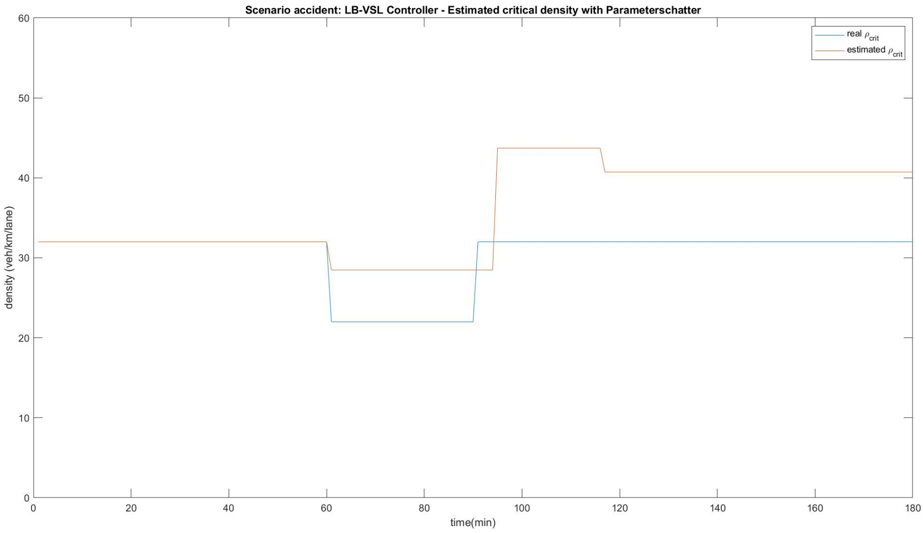

Figure 4-20: Accident scenario: LB-VSL Controller with PE, Bottleneck's Critical Density Estimation

The traffic network's speeds in all the segments, when the PE estimation method is used, are shown in figure 4-21, while the densities are presented in figure 4-22. Compared to the no-estimation case, it can be seen that the PE slows down the vehicles' speeds in segments 5 and 6 after the first simulation hour.

This speed reduction has a positive impact on the TTS, which was found to be reduced by 0.36%, compared to the no-control case, at the value of $1239.70 veh \cdot hours$.

Finally, the mean absolute estimation error it was found to be $5.70 \frac{veh}{km \cdot lane}$, which is the highest observed for the accident scenario case.

4-3-7 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Simple Derivative Estimation (SDE)

In this section the SDE method is combined with the LB-VSL algorithm. Once again, the SDE algorithm seems to be able to track the reduction of the critical density during the accident time, but not the increase once the congestion starts resolving. As it can be seen in figure 4-23, the estimation is quite similar to the case of the MTFC algorithm, and it results to a TSS of $1236.05 veh \cdot hours$, which corresponds to a 0.66% decrease compared to the no-control case.

Figures 4-24 and 4-25 present the speeds and densities, respectively, in all the segments of the freeway traffic network, when the SDE method is selected. As it can be observed from the figures, the LB-VSL algorithm, when combined with the SDE method, has very similar results, in terms of the speeds and densities of all the vehicles in all the segments, with the case of the PE method.

The mean absolute estimation error was found to be $3.89 \frac{veh}{km \cdot lane}$, which is smaller compared to the PE case.

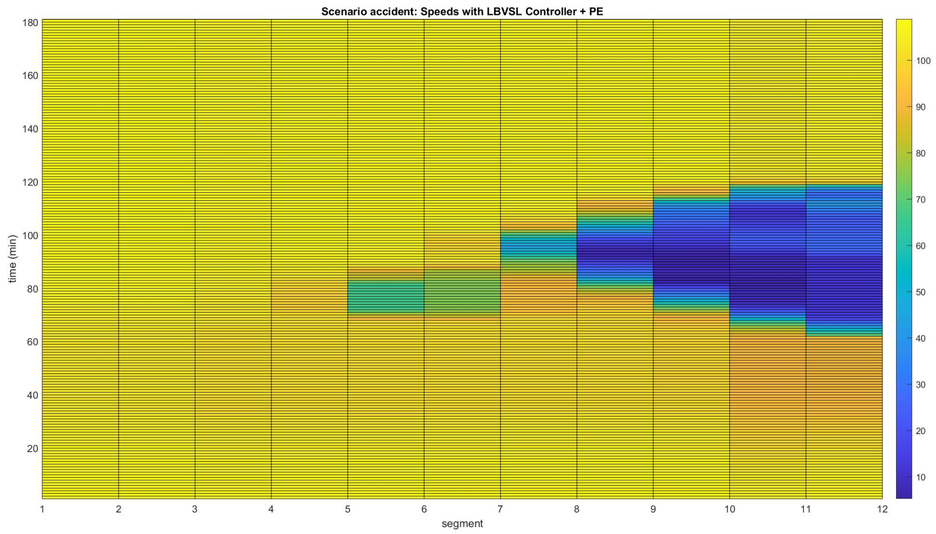

Figure 4-21: Scenario accident: speeds with LB-VSL Controller and PE

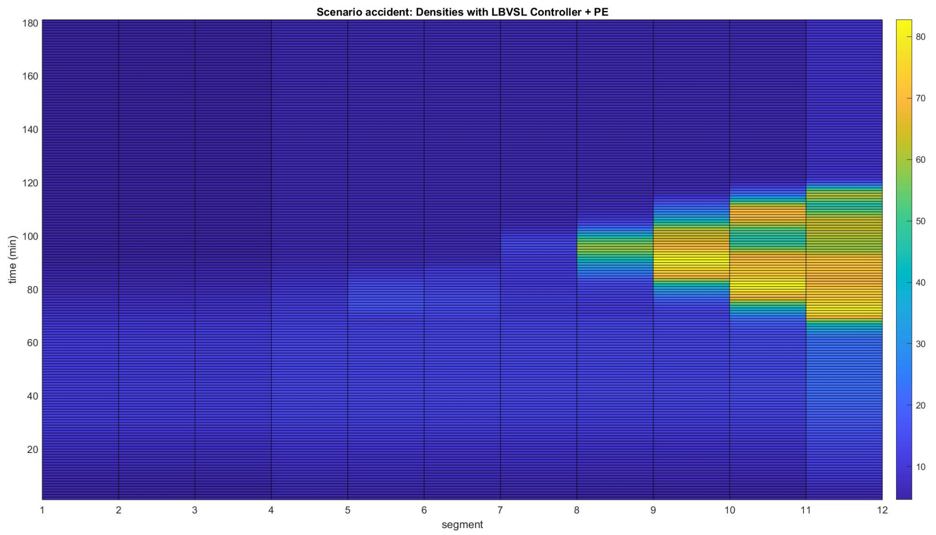

Figure 4-22: Scenario accident: densities with LB-VSL Controller and PE

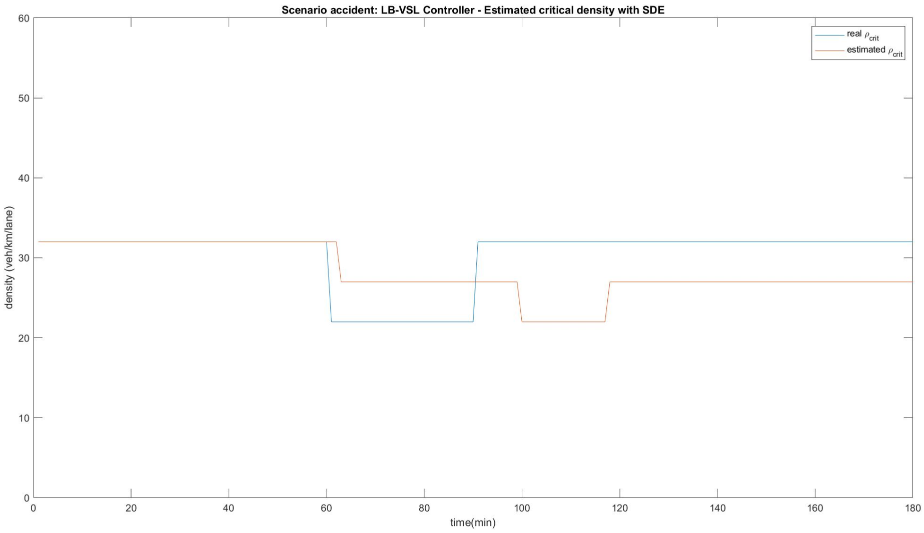

Figure 4-23: Accident scenario: LB-VSL Controller with SDE, Bottleneck's Critical Density Estimation

4-3-8 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Kalman-Filter-based derivative Estimation (KFE)

The KFE method is combined with the LB-VSL algorithm and the results are presented in this section. Similarly to the MTFC case, the KFE algorithm cannot track the reduction of the critical density after the first simulation hour. However, the estimation algorithm approximates the real critical density after the decrease, during the last hour of the simulation. This happens due to step c of the KFE algorithm, which reduces the estimate every K time units. The TTS in this case was found to be $1244.23 veh \cdot hours$, which shows no change compared to the no control case.

The traffic network's speeds in all network's segments, when the KFE estimation method is used, are shown in figure 4-27, while figure 4-28 shows the corresponding densities.

Finally, the mean absolute estimation error it was found to be $3.22 \frac{veh}{km \cdot lane}$, which is the lowest observed for the accident scenario case, same as in the case of the MTFC algorithm when combined with the KFE estimation method.

4-4 Case study B : Rain scenario

The results of the second case study are presented in this section. Rain in the whole network is simulated for 60 minutes, during the second hour of the simulation. The critical density of the whole network is reduced for one hour from $32 \frac{veh}{km \cdot lane}$ to $20 \frac{veh}{km \cdot lane}$. As in the accident scenario case, the initial density guess at the bottleneck location is $32 \frac{veh}{km \cdot lane}$.

The first subsection presents the MTFC algorithm when simulated with the full knowledge of the critical density at each simulation step. The next three sections show the results of the

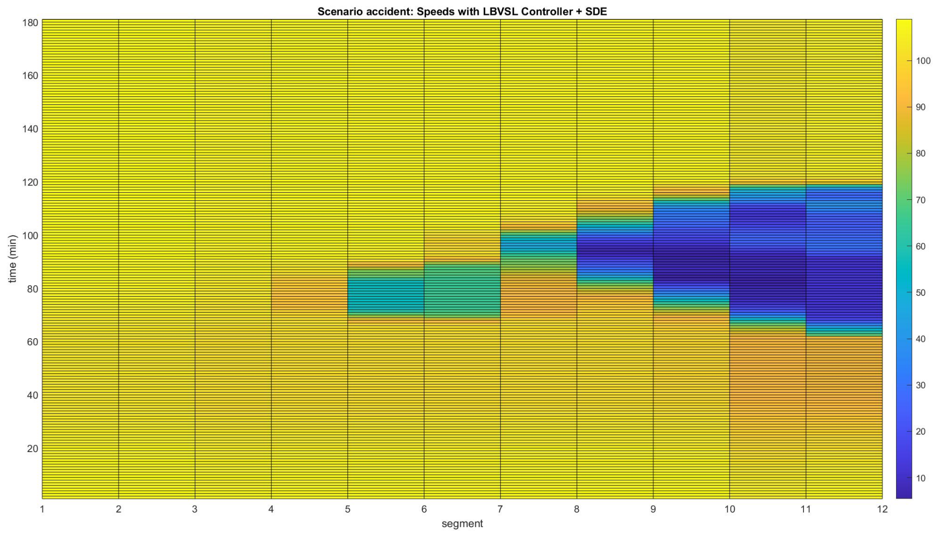

Figure 4-24: Scenario accident: speeds with LB-VSL Controller and SDE

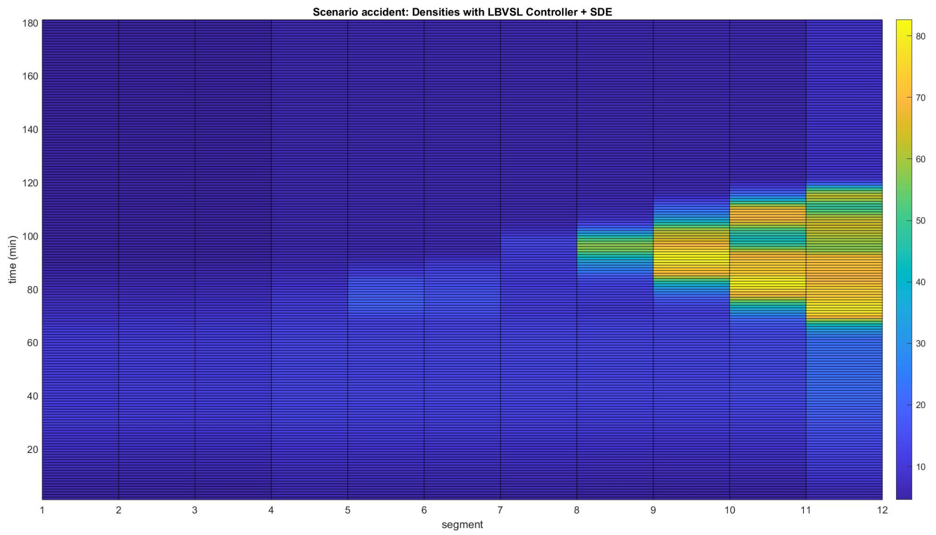

Figure 4-25: Scenario accident: densities with LB-VSL Controller and SDE

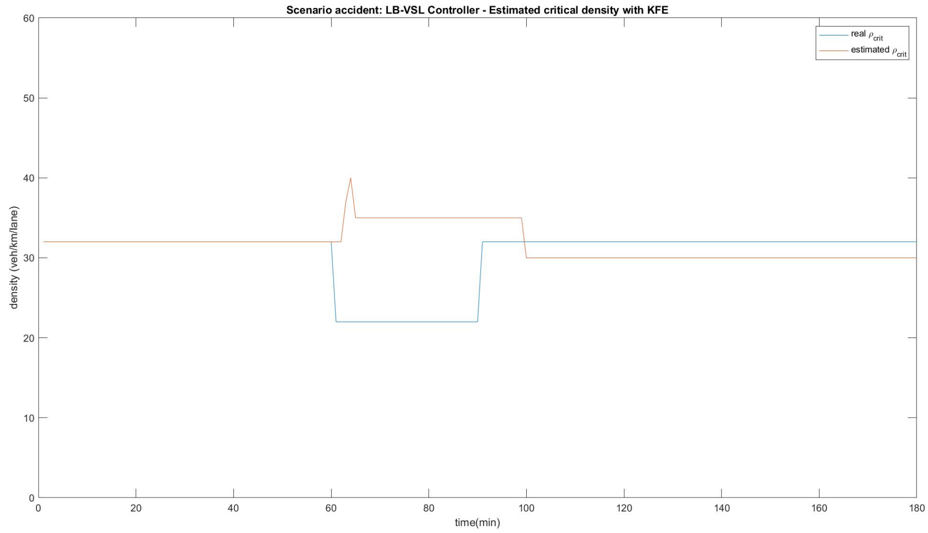

Figure 4-26: Accident scenario: LB-VSL Controller with KFE, Bottleneck's Critical Density Estimation

combination of each one of the three estimation methods with the MTFC algorithm. Then, the results of the simulation of the LB-VSL algorithm are presented. First the LB-VSL algorithm is simulated using the real values of the critical density, while in the last three sections the controller is combined with the three estimation methods.

In each subsection, the TTS by all drivers in the network is presented. Speed and density figures are also used to capture the controllers' effect on the freeway traffic network. Finally, the absolute estimation error is presented for all estimation methods.

The parameters of the MTFC algorithm are the same as in the case of accident scenario. Their values can be found in the table 4-6.

The LB-VSL controller's parameters in the rain scenario case are also same as in the accident case and can be found in table 4-7.

4-4-1 Mainstream Traffic Flow Control (MTFC) + No estimation

In this section, the MTFC algorithm has the full knowledge of the real values of the critical density at each instant. The results are presented below.

Initially, the speeds and densities in all the segments of the freeway traffic network are presented in the figures 4-29 and 4-30, respectively.

As it can be observed in figure 4-29, compared to the no-control case, the adaptive MTFC algorithm shows no improvement.

Similar situation can be seen in figure 4-30, which presents network's densities with the adaptive MTFC controller knowing the critical density during the whole simulation time. As

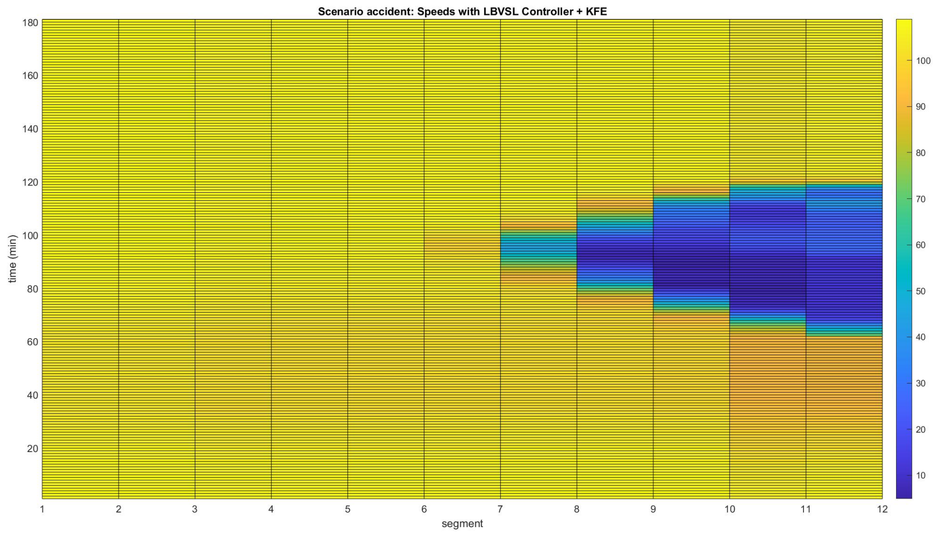

Figure 4-27: Scenario accident: speeds with LB-VSL Controller and KFE

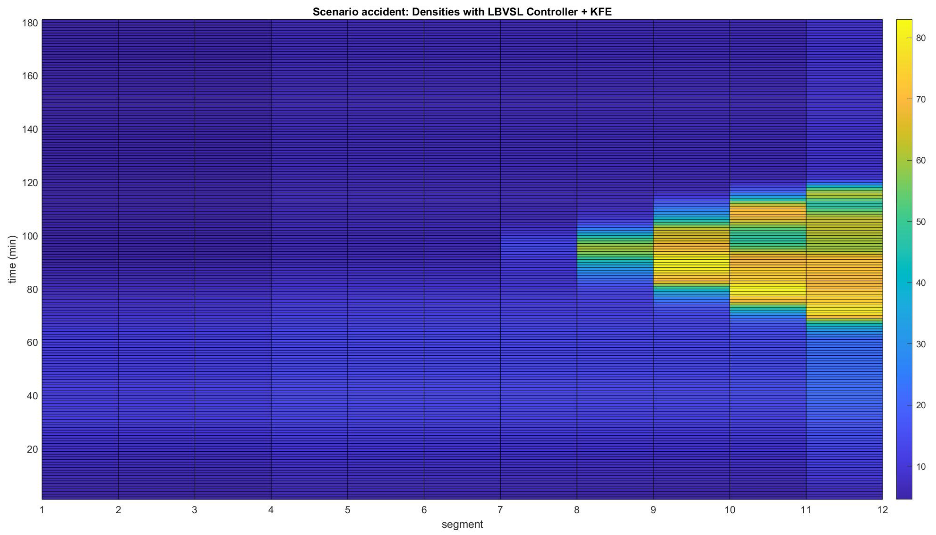

Figure 4-28: Scenario accident: densities with LB-VSL Controller and KFE

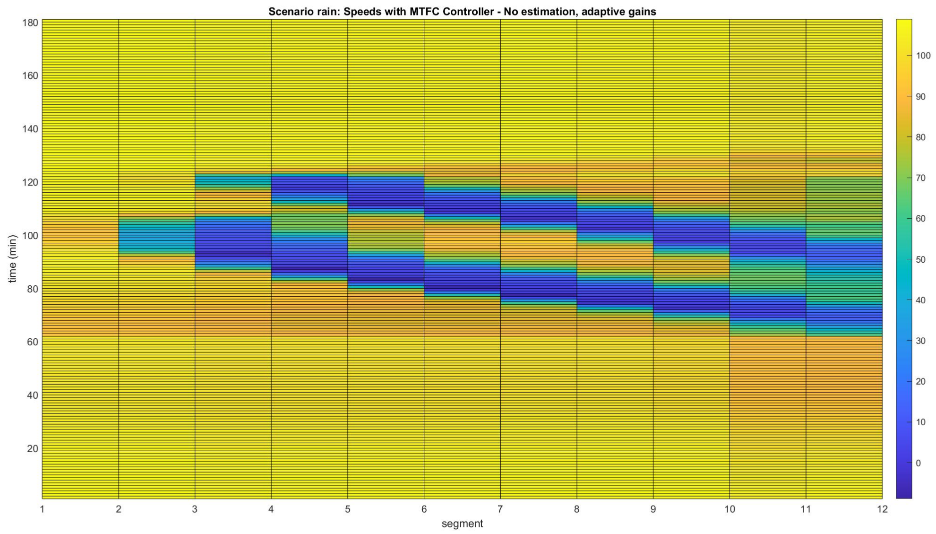

Figure 4-29: Scenario rain - speeds with adaptive MTFC - full knowledge of ρ_{crit}

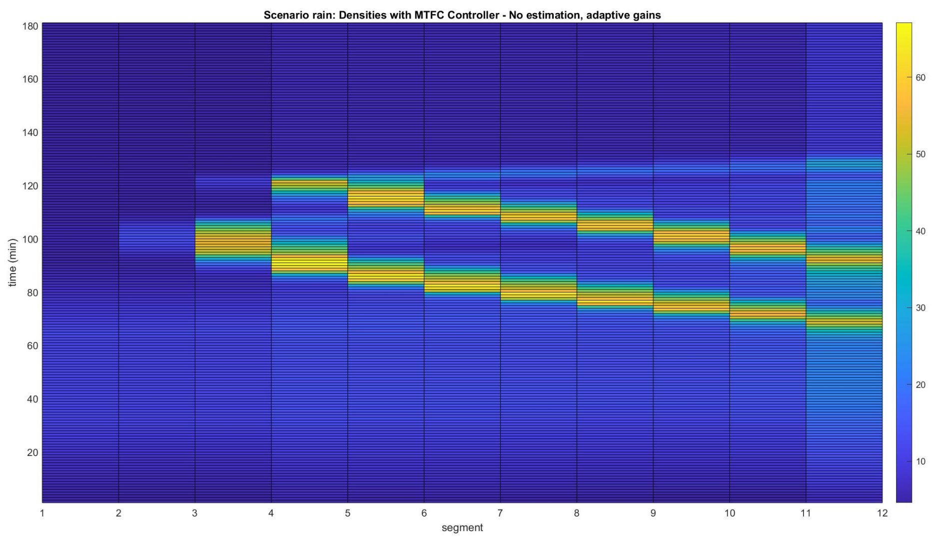

Figure 4-30: Scenario rain - densities with adaptive MTFC - full knowledge of ρ_{crit}

in the case of the speeds' figure, the adaptive MTFC controller is not able to provide better results compared to the no-control case.

The computed TTS in the rain scenario without any estimation was $1335.05 veh \cdot hours$, which is same to the no-control case.

4-4-2 Mainstream Traffic Flow Control (MTFC) + Parameter Estimator (PE)

In this section, the PE method is combined with the MTFC algorithm and the results are presented below. Figure 4-31 shows that the PE algorithm is able to accurately track bottleneck's critical density during the first two hours of the simulation. However, the algorithm cannot properly identify the increase of the critical density once the rain stops.

The TTS in this case was found to be $1334.16 veh \cdot hours$, which corresponds to a 0.07% decrease compared to the no-control case.

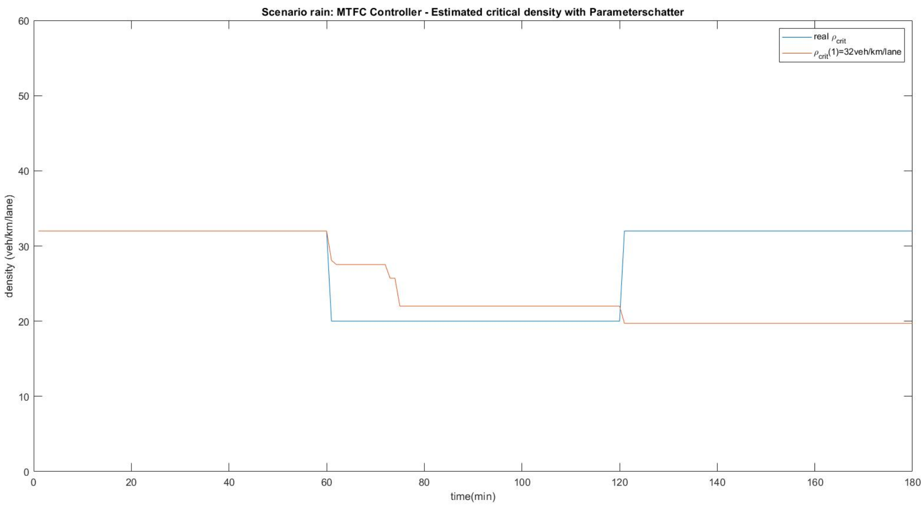

Figure 4-31: Rain scenario: MTFC Controller with PE, Bottleneck's Critical Density Estimation

Figures 4-32 and 4-33 show the speeds and densities for all the segments of the traffic network, when the adaptive MTFC algorithm is combined with the estimation method PE. Slightly lighter congestion can be observed at the bottleneck location, compared to the no-control case.

Regarding the mean absolute estimation error, it was found to be $5.18 \frac{veh}{km \cdot lane}$.

4-4-3 Mainstream Traffic Flow Control (MTFC) + Simple Derivative Estimation (SDE)

In this section the SDE method is combined with the MTFC algorithm. The SDE method seems to be able to track the decrease of bottleneck's critical density when the rain starts. Moreover, as it can be seen in the figure 4-34, the SDE algorithm can also identify the increase

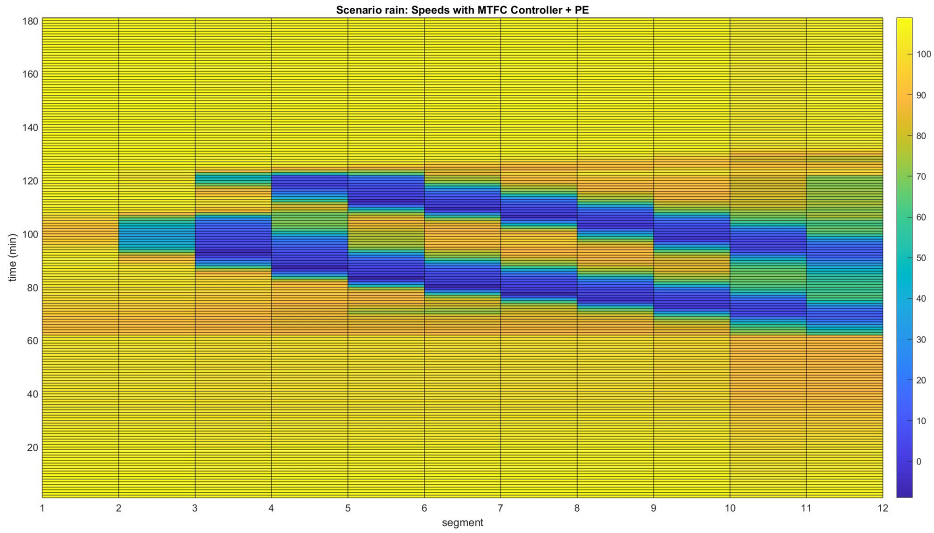

Figure 4-32: Scenario rain: speeds with MTFC Controller and PE

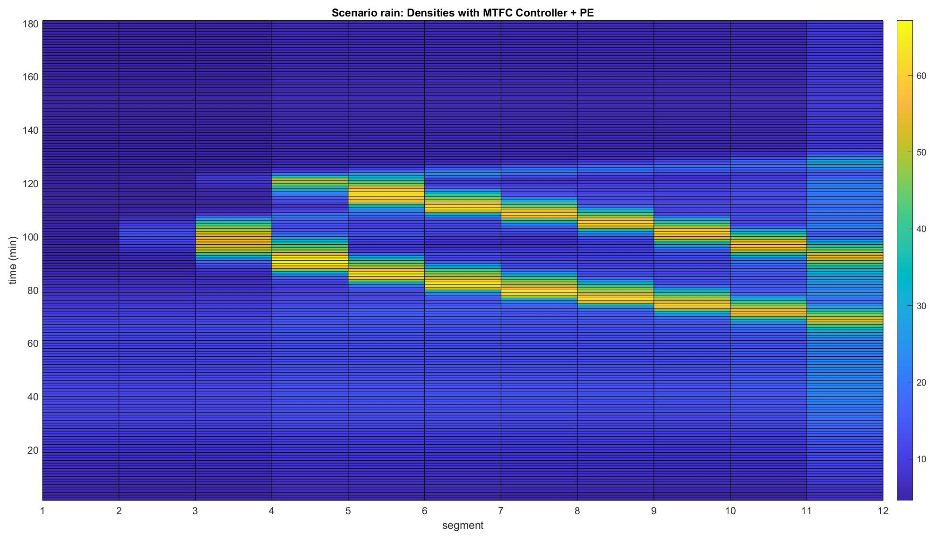

Figure 4-33: Scenario rain: densities with MTFC Controller and PE

of the critical density after rain stops. However, the final estimated value of the critical density is higher than the real one.

In figures 4-35 and 4-36, the speeds and densities for all the segments of the traffic network can be found. When the adaptive MTFC algorithm is combined with the estimation method SDE, we can observe no difference, which can also be confirmed by the TTS calculation.

The TTS by all drivers in the network was found to be $1334.16 veh \cdot hours$, which corresponds, as in the case of the PE algorithm, to a 0.07% decrease compared to the no-control case.

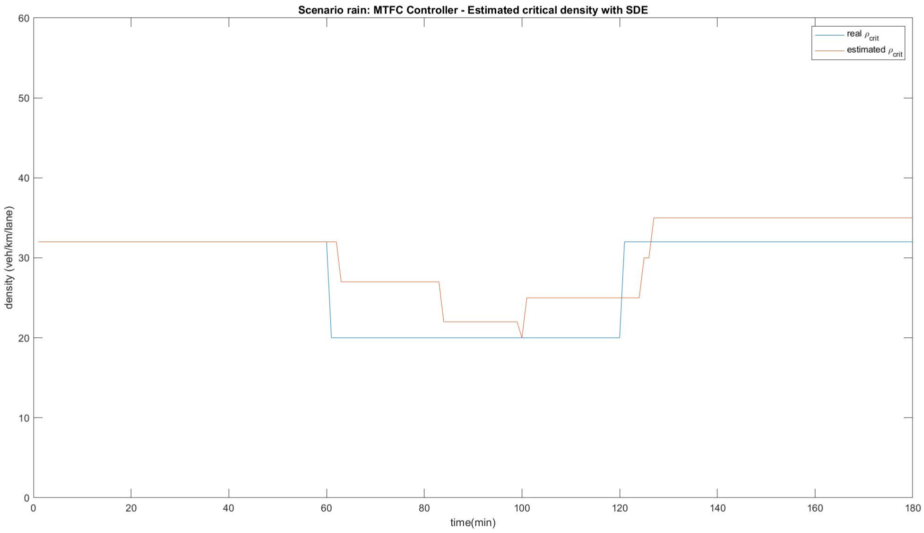

Figure 4-34: Rain scenario: MTFC Controller with SDE, Bottleneck's Critical Density Estimation

Regarding the mean absolute estimation error, it was found to be $2.76\frac{veh}{km \cdot lane}$, which is the lowest value for the MTFC algorithm in the rain scenario.

4-4-4 Mainstream Traffic Flow Control (MTFC) + Kalman-Filter-based derivative Estimation (KFE)

The KFE method is combined with the MTFC algorithm for the rain scenario and the results are presented in this section. The KFE method, as it can be seen in figure 4-37, can adequately track the decrease of the critical density. However, once the rain stops and the critical density is increased, the algorithm is not able to follow the change.

The speeds and densities for all network's segments are also presented in figures 4-38 and 4-39 , respectively, in which similar results as in the previous cases can be observed.

The TSS in the rain scenario and the MTFC algorithm with the KFE estimation method was found to be $1334.23 veh \cdot hours$, which corresponds to a 0.06% decrease compared to the no-control case.

Finally, the mean absolute estimation error it was found to be $5.91\frac{veh}{km \cdot lane}$, which is the highest for the rain scenario case when the MTFC algorithm is used.

4-4-5 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + No estimation

In this section, the LB-VSL algorithm is simulated, knowing the real values of the critical density at each instant. The results are presented below.

Figures 4-40 and 4-41 present the speeds and densities, respectively, in all the segments of the freeway traffic network.

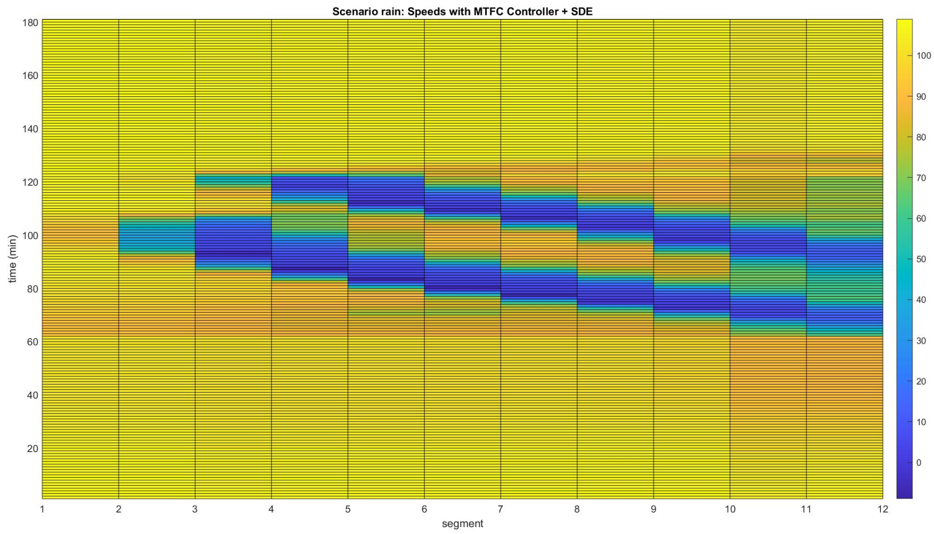

Figure 4-35: Scenario rain: speeds with MTFC Controller and SDE

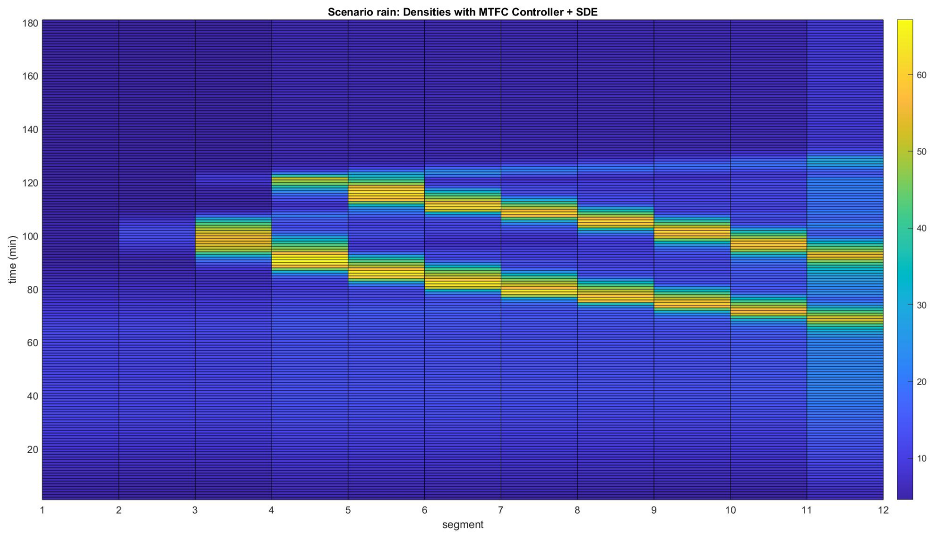

Figure 4-36: Scenario accident: densities with MTFC Controller and SDE

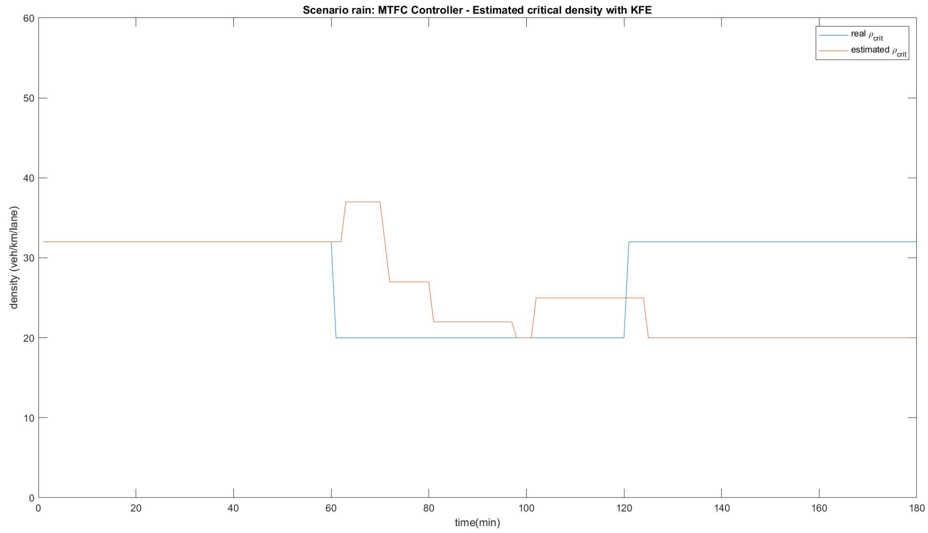

Figure 4-37: Rain scenario: MTFC Controller with KFE, Bottleneck's Critical Density Estimation

As it can be observed from the figures, there is no major difference between the LB-VSL algorithm and the MTFC algorithm. However, when the MTFC algorithm is used, slightly better results are observed, in terms of TTS reduction.

The TTS achieved when the adaptive LB-VSL algorithm was used is $1335.04 veh \cdot hours$, which corresponds to no decrease, compared to the no-control case.

4-4-6 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Parameter Estimator (PE)

In this section, the PE method is combined with the LB-VSL algorithm. The bottleneck's critical density is estimated using the PE method. Based on the results presented in the figure 4-42, it can be observed that the algorithm can track the reduction of the critical density during the second hour of the simulation. However, the algorithm underestimates the critical density during the second half of the simulation time, resulting in lower values of critical density, compared to the real values.

The TSS in the rain scenario and the LB-VSL algorithm with the PE method was found to be $1336.70 veh \cdot hours$, which corresponds to a 0.12% increase compared to the no-control case.

Figures 4-43 and 4-44 shows the speeds and densities for all the segments of the traffic network, when the adaptive LB-VSL algorithm is combined with the estimation method PE.

Regarding the mean absolute estimation error it was found to be $5.27 \frac{veh}{km \cdot lane}$.

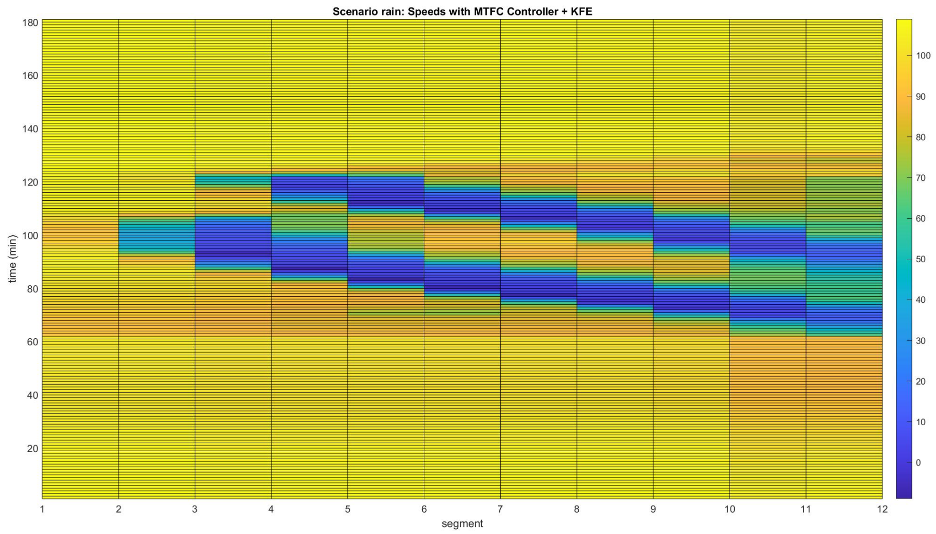

Figure 4-38: Scenario rain: speeds with MTFC Controller and KFE

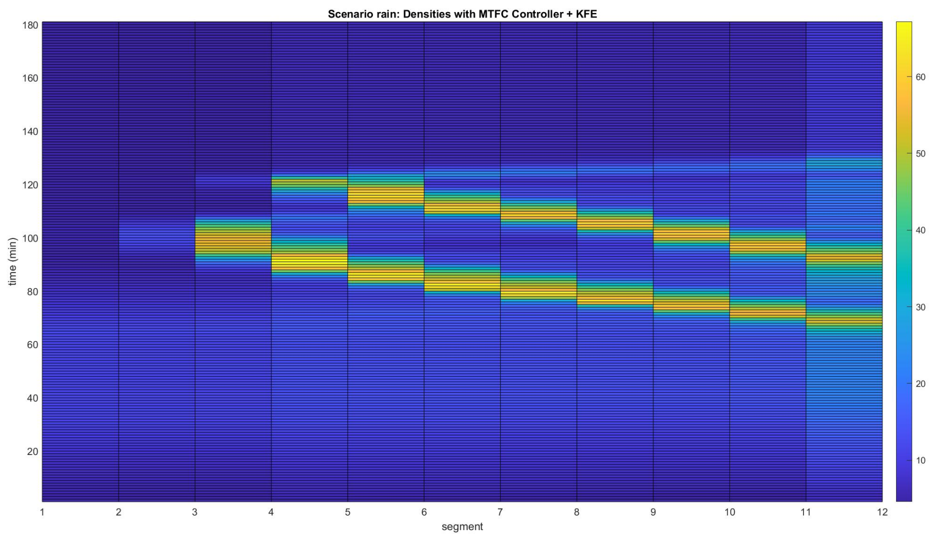

Figure 4-39: Scenario rain: densities with MTFC Controller and KFE

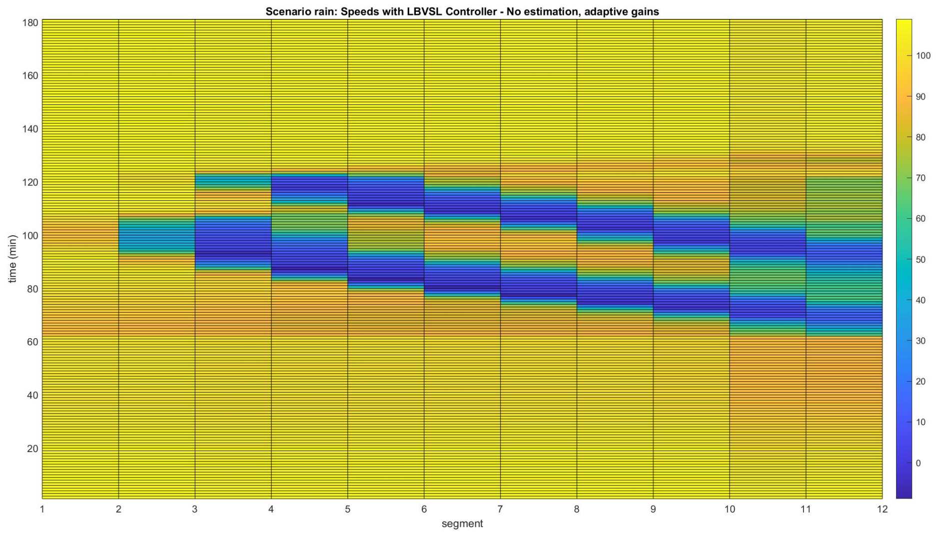

Figure 4-40: Scenario rain - speeds with adaptive LBVSL - full knowledge of ρ_{crit}

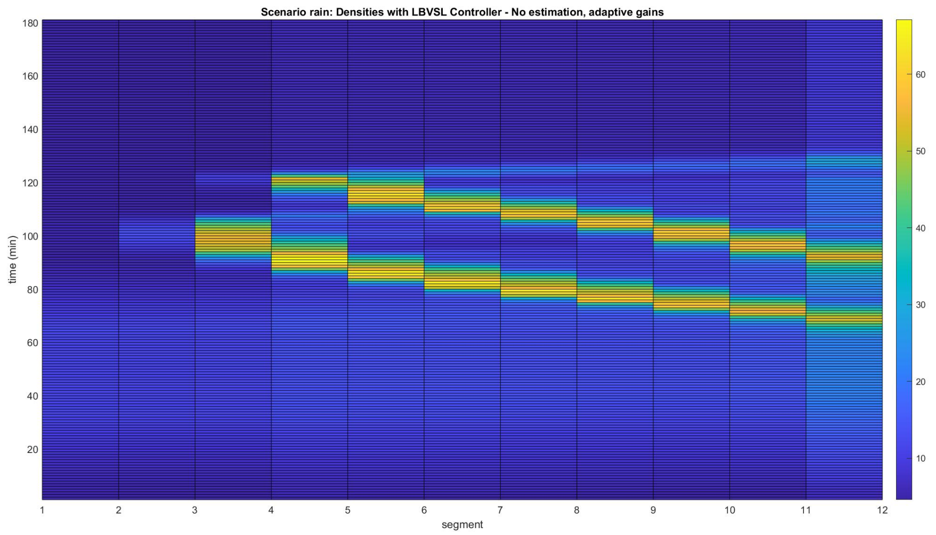

Figure 4-41: Scenario rain - densities with adaptive LBVSL - full knowledge of ρ_{crit}

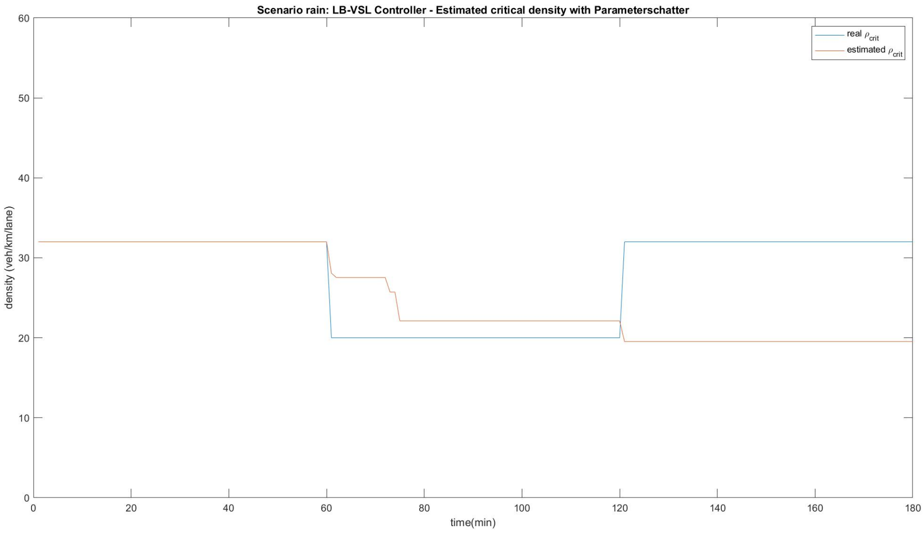

Figure 4-42: Rain scenario: LB-VSL Controller with PE, Bottleneck's Critical Density Estimation

4-4-7 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Simple Derivative Estimation (SDE)

In this section the SDE method is combined with the LB-VSL algorithm. As it can be seen in the figure 4-45, the SDE algorithm can accurately track the critical density at the bottleneck's location. The algorithm is able to follow both the decrease of the critical density once the rain starts and also the increase of the critical density after the rain stops.

The TSS in this case was found to be $1335.42 veh \cdot hours$, which corresponds to a 0.03% increase compared to the no-control case.

Figures 4-46 and 4-47 present the speeds and densities, respectively, in all the segments of the freeway traffic network, when the SDE method is selected.

The mean absolute estimation error was found to be $1.94 \frac{veh}{km \cdot lane}$, which is the smallest for the rain scenario case.

4-4-8 Logic-Based control algorithm for Variable Speed Limits (LB-VSL) + Kalman-Filter-based derivative Estimation (KFE)

The KFE method is combined with the LB-VSL algorithm for the rain scenario and the results are presented in this section. The KFE algorithm can accurately track the critical density at the bottleneck's location. The algorithm is able to track the decrease of the critical density once the rain starts, with a small delay, but it can also identify the increase of the critical density after the rain stops more accurately.

The TTS by all drivers in the network was found to be $1335.42 veh \cdot hours$, as in the case of the SDE algorithm, which corresponds to a 0.03% increase compared to the no-control case.

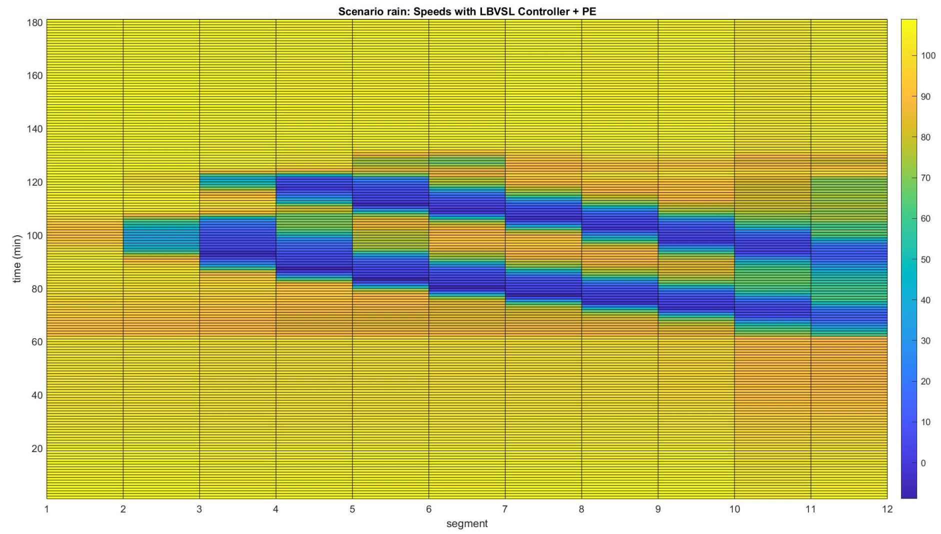

Figure 4-43: Scenario rain: speeds with LB-VSL Controller and PE

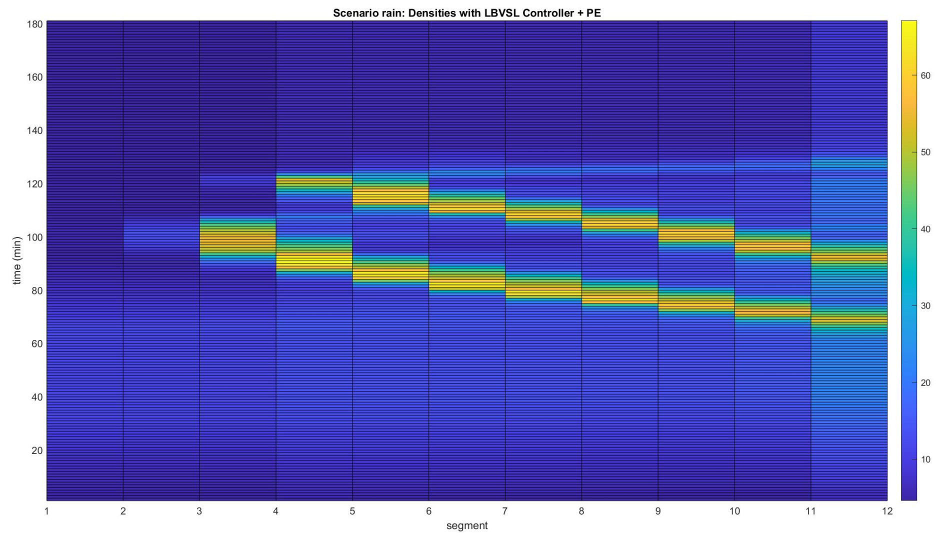

Figure 4-44: Scenario rain: densities with LB-VSL Controller and PE

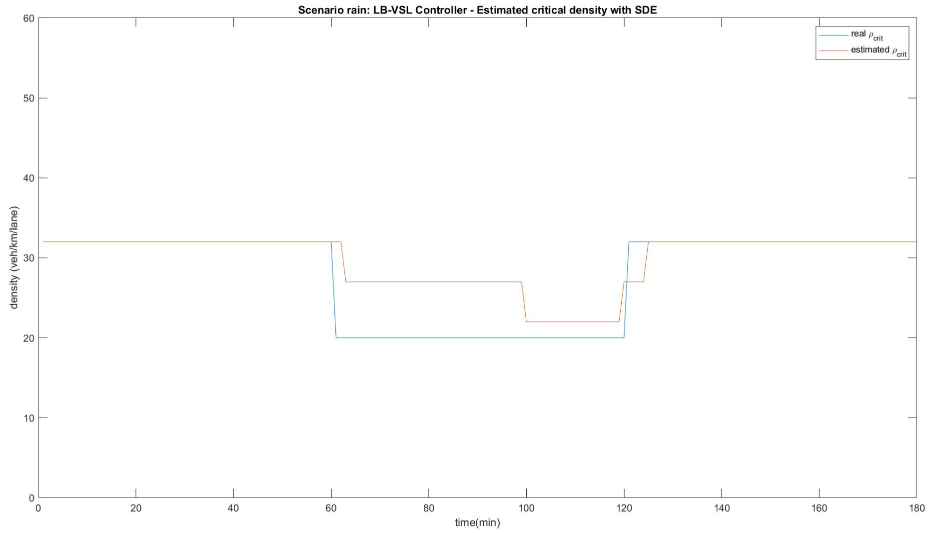

Figure 4-45: Rain scenario: LB-VSL Controller with SDE, Bottleneck's Critical Density Estimation

The traffic network's speeds in all network's segments, when the KFE estimation method is used, are shown in figure 4-49, while figure 4-50 shows the corresponding densities.

Finally, the mean absolute estimation error it was found to be $2.27\frac{veh}{km \cdot lane}$.

4-5 Conclusions

This chapter evaluates the adaptive easy-to-implement VSL algorithms Adaptive-MTFC and Adaptive-LB-VSL. Both algorithms are combined with the PE, SDE and KFE estimation methods, which estimate online the critical density at the freeway traffic network's bottleneck location. The evaluation is performed under two different scenarios, namely a rain and an accident scenario. To measure the performance of the adaptive frameworks, the TTS by all drivers in the network is calculated. Also, the mean absolute estimation error was calculated to compare the performance of the estimation methods. Below table 4-8 summarizes the results for the TTS for all different scenarios. Table 4-9 presents a comparison (in terms of TSS reduction) with no control, full knowledge of the critical density (no estimation) and the mean absolute estimation error among all the estimation methods for both the control algorithms MTFC and LB-VSL.

Based on the results, it can be concluded that the highest reduction in the TTS is achieved for the accident scenario when the MTFC algorithm is used, in combination with the KFE method for the estimation of the critical density. The same combination achieves a smaller decrease of the TTS in the rain scenario. Regarding the LB-VSL controller, it seems to have better results in the accident scenario at the bottleneck location, compared to the rain scenario at the whole network. Concerning the different estimation methods, it seems that the SDE method provides better results in general, as it provides satisfactory reduction of

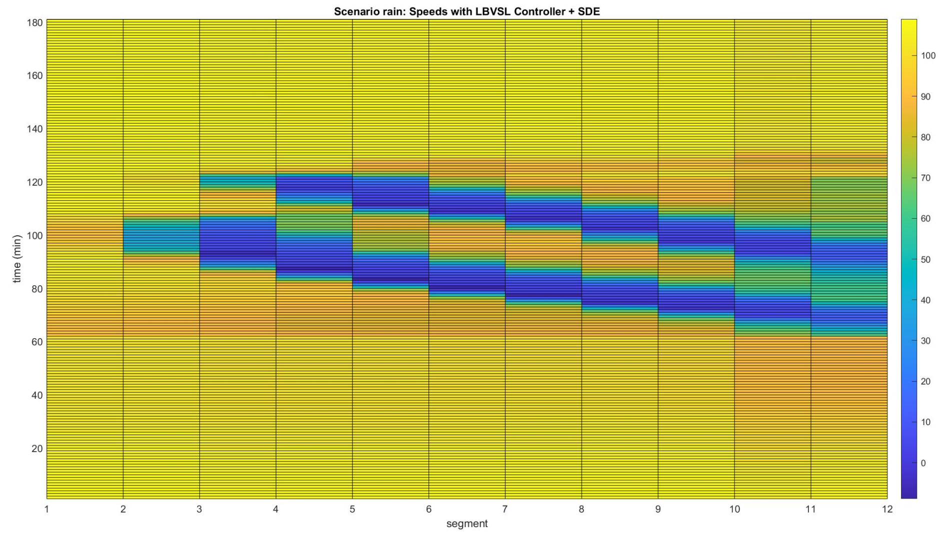

Figure 4-46: Scenario rain: speeds with LB-VSL Controller and SDE

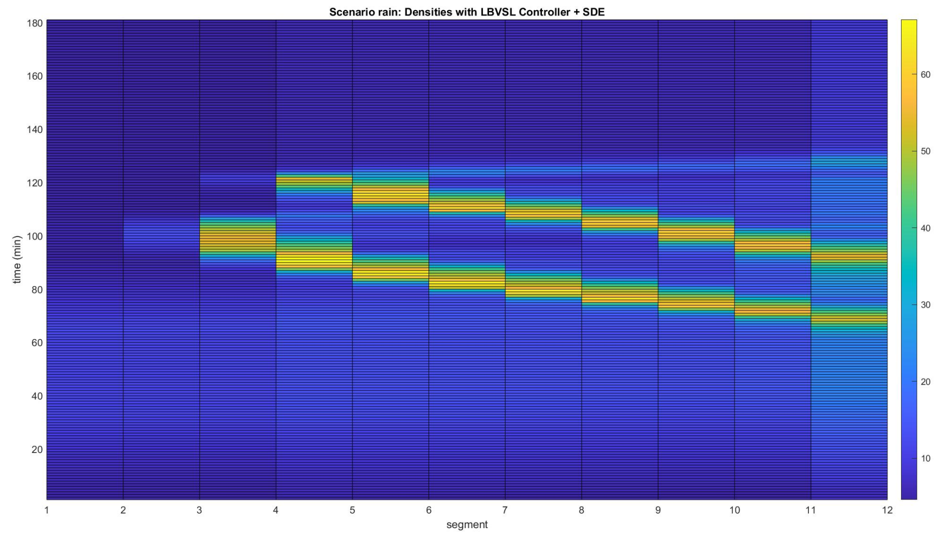

Figure 4-47: Scenario rain: densities with LB-VSL Controller and SDE

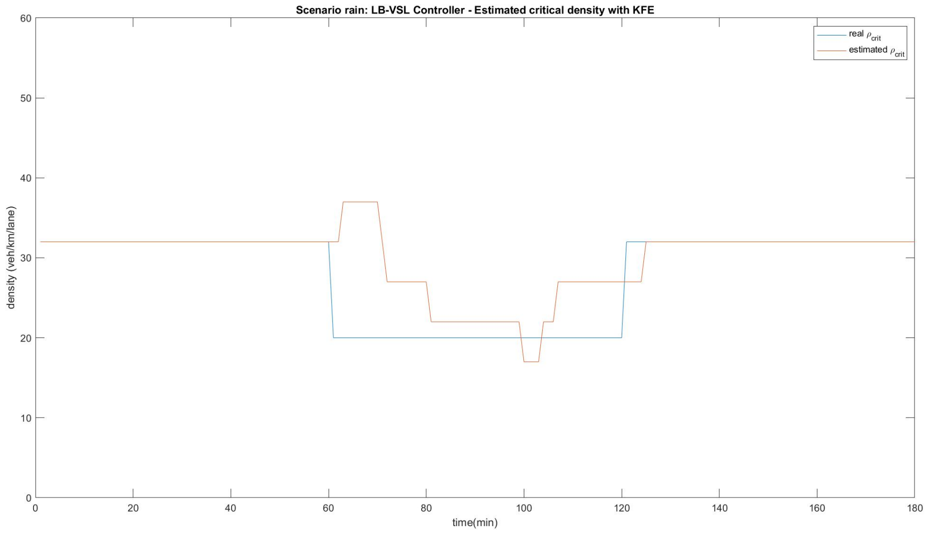

Figure 4-48: Rain scenario: LB-VSL Controller with KFE, Bottleneck's Critical Density Estimation

the TSS in all cases and an adequate tracking of the real critical density.

In terms of the estimation error, it seems that, for the rain scenario, the lowest error is achieved in the case of the LB-VSL controller, when combined with the SDE method. The other estimation methods have higher estimation errors for both controllers. Regarding the accident scenario, the KFE method shows the best results in terms of the estimation error for both the MTFC and LB-VSL adaptive controllers.

Scenario	Controller	Estimation Method	TTS
Accident	MTFC	PE	$1198.38 veh \cdot hours$
	MTFC	SDE	$1199.06 veh \cdot hours$
	MTFC	KFE	$1196.27 veh \cdot hours$
	LB-VSL	PE	$1239.70 veh \cdot hours$
	LB-VSL	SDE	$1236.05 veh \cdot hours$
	LB-VSL	KFE	$1244.23 veh \cdot hours$
Rain	MTFC	PE	$1334.16 veh \cdot hours$
	MTFC	SDE	$1334.16 veh \cdot hours$
	MTFC	KFE	$1334.23 veh \cdot hours$
	LB-VSL	PE	$1336.70 veh \cdot hours$
	LB-VSL	SDE	$1335.42 veh \cdot hours$
	LB-VSL	KFE	$1335.42 veh \cdot hours$

Table 4-8: TTS Estimation method comparison

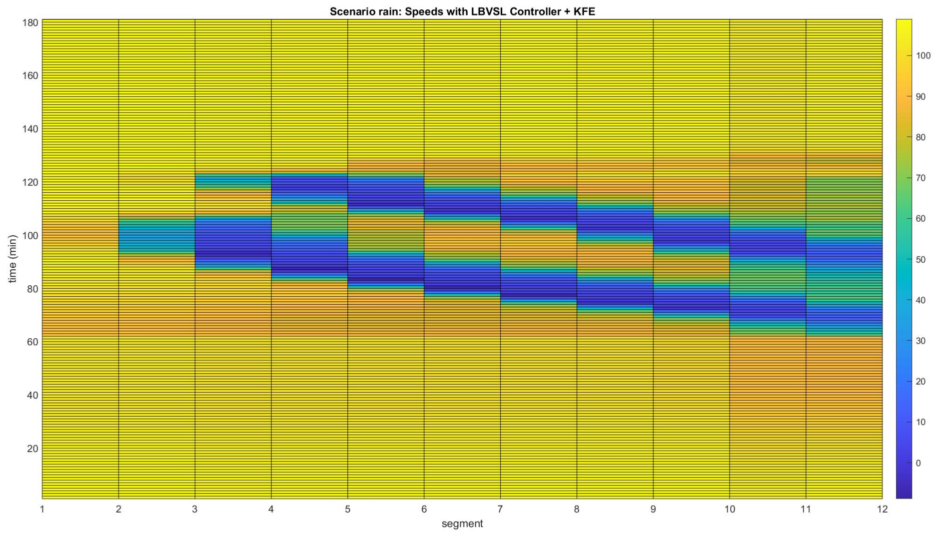

Figure 4-49: Scenario rain: speeds with LB-VSL Controller and KFE

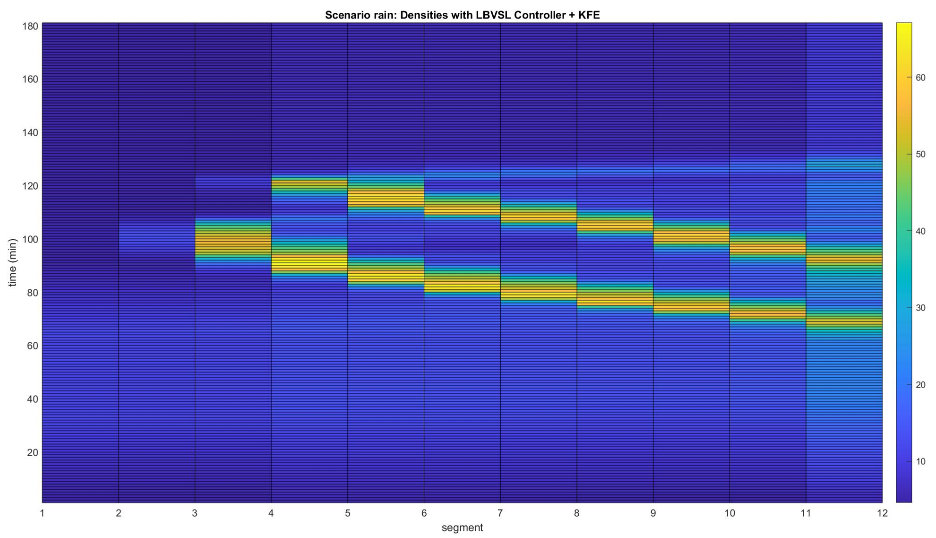

Figure 4-50: Scenario rain: densities with LB-VSL Controller and KFE

Scenario	Controller	Estimation Method	Comparison with no control	Comparison with full knowledge	Absolute estimation error
Accident	MTFC	PE	−3.68%	−0.19%	2.84
	MTFC	SDE	−3.63%	−0.14%	3.75
	MTFC	KFE	−3.85%	−0.37%	3.22
	LB-VSL	PE	−0.36%	−0.26%	5.70
	LB-VSL	SDE	−0.66%	−0.55%	3.89
	LB-VSL	KFE	0.00%	−0.10%	3.22
Rain	MTFC	PE	−0.07%	−0.07%	5.18
	MTFC	SDE	−0.07%	−0.07%	2.76
	MTFC	KFE	−0.06%	−0.06%	5.91
	LB-VSL	PE	+0.12%	+0.12%	5.26
	LB-VSL	SDE	+0.03%	+0.03%	1.94
	LB-VSL	KFE	+0.03%	+0.03%	2.27

Table 4-9: TTS Reduction and estimation error

Chapter 5

Concluding remarks

In this chapter the conclusions of this thesis are drawn. Furthermore, a discussion section has been included with the author's comments on the results obtained. Also, suggestions for future work on relevant areas are made. In section 5-1, the main conclusions are stated, while in section 5-2 the thesis results are discussed. Finally, in section 5-3, future research areas are reported.

5-1 Conclusions

The goal of this thesis is to study the performance of the easy-to-implement VSL algorithms MTFC and LB-VSL, in terms of the TTS by all drivers in the freeway traffic network, in an adaptive control framework. Both proposed adaptive algorithms use adjustable parameters, influenced by the estimated value of the critical density at bottleneck's location. The critical density is estimated on-line by three different estimation methods, the PE, SDE and KFE. The different adaptive control frameworks are tested under the assumption of rain and accident scenarios.

Under rain conditions, the MTFC shows to perform best under the SDE and PE. The worst performance is shown under the KFE. The LB-VSL controller seems to show increase instead of decrease of the TTS by all drivers in the network.

Under accident conditions, the MTFC performs the best in conjunction with KFE. PE and SDE provide adequate results in terms of the decrease of the TTS, however, all methods seem to delay to capture the density's variation. Further, the LB-VSL controller shows the best performance under the SDE approach. However, the PE again delays to capture the density's variation while the KFE methods shows no improvement over the no-control case.

More specifically, the following conclusions can be drawn:

- The well-known easy-to-implement VSL algorithms MTFC and LB-VSL have been updated to include adaptive gains $K_{\mathrm{I}}(\tau)$, $K'_{\mathrm{P}}(\tau)$, $K'_{\mathrm{I}}(\tau)$ and adaptive parameters $\overline{C_{\mathrm{B}}}(\tau)$ and $\underline{C_{\mathrm{B}}}(\tau)$ respectively.

- The critical density at the bottleneck's location is estimated online using three updated versions of the well-known estimation methods for the derivative of the Fundamental Diagram, namely the PE, SDE and KFE.
- The adaptive control frameworks are evaluated under the assumption of two different scenarios that affect the critical density of the bottleneck, namely a rain and an accident scenario.
- In the accident case, the higher decrease of the TSS is observed when the MTFC algorithm is combined with the KFE method. In this case the decrease is -3.85%
- In the accident case, the decrease of the TSS is maximized when the critical density at the bottleneck location is overestimated during the congestion period and underestimated while congestion resolves.
- In the rain scenario, the higher decrease of the TSS is observed when the MTFC algorithm is combined with the PE and SDE methods. In both cases the decrease is 0.07%.
- In the rain scenario, only the SDE method in the MTFC case and the SDE and KFE methods in the LB-VSL case are able to capture the changes of the critical density at the bottleneck location.
- In terms of the lowest mean absolute estimation error, for the accident case, the KFE method shows the best results, while for the rain scenario, the SDE method is the most capable to identify the changes of the critical density.

5-2 Discussion

In this section, the derived results are commented. The obtained decrease in the TTS by all drivers in the freeway traffic network is not very satisfactory. The higher decrease in the TTS was noticed in the case of the MTFC controller and the accident scenario, which was a local incident with shorter duration, which affected less the total delays in the freeway traffic network. Moreover, the real-time estimation of the critical density at the bottleneck location, which was added to the adaptive MTFC controller, did not improve much the situation, compared to the case with full knowledge of the critical density. It was, however, greater the improvement in the case of the accident, compared to the rain scenario. The rain scenario was simulated in the whole network and for longer time. The decrease of the critical density was higher and the total delays were bigger. It is therefore, reasonable to notice better results in the accident case.

Between the two different control algorithms, the MTFC algorithm showed better results in both scenarios affecting the critical density. It is reasonable to show better results in the case of the rain in the whole network, as the LB-VSL algorithm focuses on avoiding the congestion at bottleneck locations. During the rain conditions in the network, the algorithm is not able to activate the VSLs before the critical density is reached. For the accident case, the LB-VSL algorithm could possibly be tuned better to obtain more satisfactory results.

Regarding the small difference in terms of the TTS decrease when the adaptivity was added to both controllers, it seems that the logic for the update of the adaptive gains and parameters could be chosen in a more systematic way. Also, the selection of the controllers' constant parameters, as well as the selection of the parameters of the estimation methods, could have been made following an multi variable optimization method.

Moreover, the high values in the absolute estimation error are obtained due to the inability of the estimation algorithms to respond to the sudden changes in the critical density. In most of the scenario cases, the estimation algorithms are able to track the decrease of the critical density better compared to the case that the critical density increases after a previous decrease. When the critical density increases after a decrease, the estimation algorithms are not producing accurate estimates of the critical density, as the derivative calculation is not indicating undercritical conditions. For that reason, probably smaller fluctuations in the critical density could have been simulated.

Furthermore, based on the simulation results, the easy-to-implement VSL algorithms studied in this thesis do not seem to improve much the traffic conditions in case of incidents that highly affect the traffic network's critical density. A possible reason for that is because the VSLs are used to reduce the congestion downstream and close to a bottleneck location, to avoid or delay the activation of the bottleneck. However, the congestion should be delayed further upstream with possible multiple control measures in case of an accident or rain conditions in the freeway traffic network with predefined capacity.

Finally, to the best of the author's knowledge, all algorithms developed in MATLAB have been tested and checked for their correctness, to avoid any programming error that could have affected the results.

5-3 Future Work

In this last section some recommendations for future work are outlined.

- Multiple different demand scenarios for the ramp and the mainline could be used in the future for the simulations, to guarantee the robustness of the proposed adaptive frameworks.
- An investigation can be performed for the behavior of the controllers under different scenarios for drivers' compliance.
- The application of the proposed adaptive controllers in combination with the online estimation of the critical density at bottleneck's location could be introduced in a real traffic network.
- The controllers' parameters have been chosen based on trial-and-error. Another interesting approach for future work would be to be chosen based on optimization techniques.
- The freeway traffic network's capacity can be also estimated online and fed up to the adaptive easy-to-implement VSL algorithms.

- The estimation methods for the critical density also contain multiple parameters which could be optimally chosen based on multi-variable optimization techniques.
- Different values for the initial density guess could be used.
- It would be useful to be studied why the TTS reduction was found low.
- Finally, another case study should be used to compare the performance of the proposed controllers.

Bibliography

[1] Karl J Åström and Björn Wittenmark. *Adaptive control*. Courier Corporation, 2013.

[2] Tom Bellemans, Bart De Schutter, and Bart De Moor. Models for traffic control. *JOURNAL A*, 43(3/4):13–22, 2002.

[3] Rodrigo C Carlson, Ioannis Papamichail, and Markos Papageorgiou. Local feedback-based mainstream traffic flow control on motorways using variable speed limits. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1261–1276, 2011.

[4] Rodrigo C Carlson, Ioannis Papamichail, Markos Papageorgiou, and Albert Messmer. Optimal mainstream traffic flow control of large-scale motorway networks. *Transportation Research Part C: Emerging Technologies*, 18(2):193–212, 2010.

[5] Rodrigo C Carlson, Ioannis Papamichail, Markos Papageorgiou, and Albert Messmer. Optimal motorway traffic flow control involving variable speed limits and ramp metering. *Transportation Science*, 44(2):238–253, 2010.

[6] Michael J Cassidy and Robert L Bertini. Some traffic features at freeway bottlenecks. *Transportation Research Part B: Methodological*, 33(1):25–42, 1999.

[7] Carlos F Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994.

[8] Antonella Ferrara, Simona Sacone, and Silvia Siri. *Freeway traffic modelling and control*. Springer, 2018.

[9] National Research Council (US). Transportation Research Board. Committee for Guidance on Setting and Enforcing Speed Limits. *Managing speed: Review of current practice for setting and enforcing speed limits*, volume 254. Transportation Research Board, 1998.

[10] European Comission. Directorate-General for Mobility and Transport. *White Paper on Transport: Roadmap to a Single European Transport Area: Towards a Competitive and Resource-efficient Transport System*. Publications Office of the European Union, 2011.

[11] José Ramón D Frejo and Bart De Schutter. A logic-based speed limit control algorithm for variable speed limits to reduce traffic congestion at bottlenecks. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 198–204. IEEE, 2018.

[12] José Ramón D Frejo and Bart De Schutter. Spert: A speed limit strategy for recurrent traffic jams. *IEEE Transactions on Intelligent Transportation Systems*, 20(2):692–703, 2018.

[13] José Ramón D Frejo, Alfredo Núñez, Bart De Schutter, and Eduardo F Camacho. Hybrid model predictive control for freeway traffic using discrete speed limit signals. *Transportation Research Part C: Emerging Technologies*, 46:309–325, 2014.

[14] José Ramón D Frejo, Ioannis Papamichail, Markos Papageorgiou, and Eduardo F Camacho. Macroscopic modeling and control of reversible lanes on freeways. *IEEE Transactions on Intelligent Transportation Systems*, 17(4):948–959, 2015.

[15] José Ramón D Frejo, Ioannis Papamichail, Markos Papageorgiou, and Bart De Schutter. Macroscopic modeling of variable speed limits on freeways. *Transportation research part C: emerging technologies*, 100:15–33, 2019.

[16] José Ramón D. Frejo and Bart De Schutter. Logic-based traffic flow control for ramp metering and variable speed limits—part 1: Controller. *IEEE Transactions on Intelligent Transportation Systems*, 22(5):2647–2657, 2021.

[17] José Ramón D Frejo, Ioannis Papamichail, Markos Papageorgiou, and Bart De Schutter. A new macroscopic model for variable speed limits. *IFAC-PapersOnLine*, 51(9):343–348, 2018.

[18] BD Greenshields, Ws Channing, Hh Miller, et al. A study of traffic capacity. In *Highway research board proceedings*, volume 1935. National Research Council (USA), Highway Research Board, 1935.

[19] Andreas Hegyi. *Model predictive control for integrating traffic control measures.* Netherlands TRAIL Research School, 2004.

[20] Andreas Hegyi, Bart De Schutter, and Hans Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C: Emerging Technologies*, 13(3):185–209, 2005.

[21] Andreas Hegyi, Bart De Schutter, and J Hellendoorn. Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on intelligent transportation systems*, 6(1):102–112, 2005.

[22] Serge Hoogendoorn, Ramon Landman, Jaap van Kooten, and Marco Schreuder. Integrated network management amsterdam: Control approach and test results. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 474–479. IEEE, 2013.

[23] Victor L Knoop, Henk Taale, Michel Meulenberg, Paul BC van Erp, and Serge P Hoogendoorn. Ramp metering with real-time estimation of parameters. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 3619–3626. IEEE, 2018.

[24] Elias Kosmatopoulos, Markos Papageorgiou, Diamantis Manolis, Jill Hayden, Roger Higginson, Keith McCabe, and Nanu Rayman. Real-time estimation of critical occupancy for maximum motorway throughput. *Transportation research record*, 1959(1):65–76, 2006.

[25] Henry X Liu, Jeff X Ban, Wenteng Ma, and Pitu B Mirchandani. Model reference adaptive control framework for real-time traffic management under emergency evacuation. *Journal of urban planning and development*, 133(1):43–50, 2007.

[26] Albert Messmer and Markos Papageorgiou. Metanet: A macroscopic simulation program for motorway networks. *Traffic engineering & control*, 31(9), 1990.

[27] Markos Papageorgiou, Habib Hadj-Salem, Jean-Marc Blosseville, et al. Alinea: A local feedback control law for on-ramp metering. *Transportation Research Record*, 1320(1):58–67, 1991.

[28] Markos Papageorgiou, Elias Kosmatopoulos, and Ioannis Papamichail. Effects of variable speed limits on motorway traffic flow. *Transportation Research Record*, 2047(1):37–48, 2008.

[29] Markos Papageorgiou, Ioannis Papamichail, Albert Messmer, and Yibing Wang. Traffic simulation with metanet. In *Fundamentals of traffic simulation*, pages 399–430. Springer, 2010.

[30] Ioannis Papamichail, Katerina Kampitaki, Markos Papageorgiou, and Albert Messmer. Integrated ramp metering and variable speed limit control of motorway traffic flow. *IFAC Proceedings Volumes*, 41(2):14084–14089, 2008.

[31] Virginia P Sisiopiku. Variable speed control: Technologies and practice. In *Proceedings of the 11th Annual Meeting of ITS America*, pages 1–11, 2001.

[32] Emmanouil Smaragdis, Markos Papageorgiou, and Elias Kosmatopoulos. A flow-maximizing adaptive local ramp metering strategy. *Transportation Research Part B: Methodological*, 38(3):251–270, 2004.

[33] Kai Yuan, Victor L Knoop, and Serge P Hoogendoorn. Capacity drop: Relationship between speed in congestion and the queue discharge rate. *Transportation Research Record*, 2491(1):72–80, 2015.

Glossary

List of Acronyms

FD Fundamental Diagram

VSLs Variable Speed Limits

VSL Variable Speed Limit

LB-VSL Logic-Based control algorithm for Variable Speed Limits

MTFC Mainstream Traffic Flow Control

TTS Total Time Spent

RM Ramp Metering

ALINEA Asservissement Linéaire d'Entrée Autoroutière

METANET Modele d'Ecoulement de Trafic sur Autoroute NETworks

CTM Cell Transmission Model

PE Parameter Estimator

SDE Simple Derivative Estimation

KFE Kalman-Filter-based derivative Estimation

FOT Field Operational Test

APT Amsterdam Practical Trial

PPA PraktijkProef Amsterdam

AD-RMC Adaptive Ramp Metering Controller

LS Least Squares