System Engineering and Automation Department
University of Seville

Model Predictive Control for Freeway Traffic Networks

José Ramón D. Frejo

Supervisor: Eduardo F. Camacho

Submitted in part fulfilment of the requirements for the degree of Doctor of Philosophy in Automation, Robotics and Telematics of the University of Seville.

"Crisis is produced when the Old does not finish to die, and the New does not end to born."

Bertolt Brecht

Abstract

Traffic congestion on freeways is a critical problem due to higher delays, waste of fuel, a higher accident risk probability, negative impact on the environment, etc. Variable speed limits, ramp metering, and reversible lanes are some of the most often used examples of freeway traffic measures that can be used to dynamically control traffic.

Nowadays, most of the dynamic traffic control systems operate according to a linear and local control loop. As explained in the thesis, the use of appropriate non-local and multivariable techniques can considerably improve the reduction in the total time spent by the drivers and other traffic performance indexes. Nonlinear centralized Model Predictive Control (MPC) is probably the best control algorithm choice for a small network as can be seen on previous references.

The main practical problem of nonlinear centralized MPC is that the computational time quickly increases with the size of the network making difficult to apply centralized MPC for large networks. Therefore, completely centralized control of large networks is viewed by most practitioners as impractical and unrealistic. The main objective of this thesis is the proposal of MPC techniques which can be applied, in practice, to real large traffic networks.

Possible solutions are the use of distributed MPC (considering the network as a set of subsystems controlling each subsystem by one independent MPC), hybrid MPC (splitting the problem in a continuous optimization for the ramp metering signals and in a discrete optimization for speed limits) or genetic algorithms (finding the fittest individuals within a generation, applying genetic operators for the recombination of those individuals, and generating a good offspring). This thesis proposes and analyses these solutions.

Other open problem in freeway traffic control is the dynamic operation of reversible lanes. Despite the long history and widespread use of reversible lanes worldwide, there have been few quantitative evaluations and research studies conducted on their performance. To address this problem, this thesis proposes a macroscopic model for reversible lanes and on-line controllers for the operation of reversible lanes.

Moreover, a MPC controller for freeway traffic requires a model to make accurate and reliable predictions of the traffic flow. On the other hand, this model is required to be fast enough, so that it can be used for on-line based control applications. Therefore, it is imperative to select or develop appropriate models, i.e., models that are fast and that provide accurate predictions. In this thesis, the METANET model and its extensions have been selected to be used for the prediction of the traffic flow and, based on this model, new advances in freeway traffic modeling for optimal control strategies are proposed.

Acknowledgements

First and foremost, I offer my sincerest gratitude to my supervisor, Eduardo Fernández Camacho, who has supported me throughout my thesis with his patience and knowledge. I always felt his full support in my work and was lucky for so many wonderful opportunities he gave me. Moreover, he has shown me that it is possible to bring top level research to Seville, instead of emigrating searching for it. I am very proud to have been his PhD student and I think he is a model of how a full professor should be. I hope our cooperation will continue for years.

I would like to extend thanks to the many people, in many countries, who so generously contributed to the work presented in this thesis. I want to thank three prominent figures in freeway traffic control for hosting me in their research groups:

- My thanks to Prof. Markos Papageorgiou, for hosting me at the Dynamic Systems and Simulation Laboratory, University of Crete. I really appreciate your scientific guidance and the sharing of your knowledge. I also want to thank Ioannis Papamichail for the common work whose result are in Chapter 5 and Section 2.5.

- My thanks to Prof. Bart de Schutter for hosting me at the Delft Center Systems and Control, Delft University of Technology. I was impressed by the quality of the feedback that Bart gave me for our common works, which are included in this thesis. I also want to thank Andreas Hegyi, whose PhD dissertation was the starting point of this thesis, and Alfredo Núñez, because Chapter 5 and Section 3.3 would have not been possible without his collaboration.

- My thanks to Prof. Roberto Horowitz for hosting me at the Partners for Advanced Transportation Technology, University of California at Berkeley and for the common work whose result are in Section 2.3 and 2.4.

I met many very nice people who made my stays enjoyable in Crete, Holland and California. I want to thank them all for the unforgettable times.

This thesis has been written in the context of WP5 of the European Network of Excellence HYCON2. I want to thanks the members of the group for the stimulating and friendly working environment. Specially, I want to thank Alberto Nai Oleari, Luis Leon Ojeda, Silvia Siri, Antonella Ferrara, Carlos Canudas and Simona Sacone.

Having had so many colleagues in Seville that cannot all be mentioned but that I nonetheless appreciate for creating a nice and friendly working environment, I would like to thank to all the members of the Dept. de Ingeniería de Sistemas y Automática of the University of Seville.

Last but not least, I have to say that I owe everything that I am to my parents, José Ramón and Isabel. If any time I have been proud of myself, it is because of them. I have no words to express my gratitude but at least I can dedicate them my thesis.

Contents

List of Tables

List of Figures

Chapter 1

Introduction and Background

1.1 Motivation and objectives

Traffic congestion on freeways is a critical problem due to its negative impact on the environment and many other important consequences like higher delays, waste of fuel, a higher accident risk probability, etc. Freeways were originally conceived to provide virtually unlimited mobility to road users. However, the continuous increase in car ownership and demand has led to a steady increase (in space and time) of recurrent and non recurrent freeway congestions, particularly within and around metropolitan areas.

Figure 1.1: Traffic congestion during the 2010 China National Highway 110 Traffic Jam.

In many places, the current capacity of transport networks is not able to meet the demand. In those circumstances, the inevitable result is congestion in urban areas and metropolitan regions (at the entrance of the main cities) and on the key

transit roads, overcrowding on some public transport links and lengthy queues at some airports. When networks are overused, journey times lengthen and reliability suffers.

The building of new infrastructure to reduce congestion and accommodate higher levels of traffic is less and less a practicable solution. The impact of infrastructure on the environment is a growing concern. In addition, the current economic crisis reasserts the importance of putting budget accounts into a long-term sustainable path. This implies reducing public deficit and debt and improving the quality of public finance. More cost-effective solutions would have to be found to tackle congestion than relying on expanding "hard" infrastructure.

The European Commission's "White Paper on Transport: Roadmap to a Single European Transport Area - Towards a competitive and resource efficient transport system" [50] concludes that:

- "The European Union has not succeeded in containing the growth of the economic, environmental and social costs of mobility while simultaneously ensuring that current and future generations have access to safe, secure, reliable and affordable mobility resources to meet their own needs and aspirations. The Commission is therefore of the opinion that the EU transport system today is not sustainable enough."
- "The EU transport system it is not sufficiently resource efficient so as to promote sustainable growth in the meaning of the EU 2020 strategy. Transport is extremely dependent upon oil whereas CO2 emissions from transport-related activities account are still growing."
- "With congestion growing, the EU transport system does not sufficiently keep pace with the mobility needs and aspirations of people and businesses."
- "In particular, congestion would continue to represent a huge burden on the society. Congestion costs are projected to increase by about 50% by 2050, to nearly 200 billion euros annually."

The construction of new freeways are not always viable to implement in the short-term due to technical, political, legal, or economic reasons. Therefore, in the last decades, most research has been focused on making a better use of the available traffic infrastructure.

It has been reported in the literature that dynamic traffic control is a good solution to decrease congestion [85, 38, 72, 53, 42, 11]. In general, dynamic traffic control uses measurements of the traffic conditions over time and computes dynamic control signals to influence the behavior of the drivers and to generate a response in such a way that the performance of the network is improved, by reducing delays, emissions, fuel consumption, etc.

Variable speed limits, ramp metering, and reversible lanes are some of the most often used examples of freeway traffic signals that can be used to dynamically control traffic. These measures have been already successfully implemented in USA, Germany, Spain, the Netherlands, and other countries [99, 82, 41].

Nowadays, most of the dynamic traffic control systems operate according to a linear and local control loop. However, as explained in the thesis, the use of appropriate non-local and multivariable techniques can improve considerably the reduction in the total time spent by the drivers and other traffic performance indexes. Among the available options described in the literature, the methods based on the use of advanced control techniques like Model Predictive Control (MPC) [9] have proved to substantially improve the performance of the controlled traffic system [28, 72, 29, 37] in various simulation studies.

The main problem of nonlinear centralized MPC is that the computational time quickly increases with the size of the network making difficult to apply centralized MPC for large networks. Therefore, completely centralized control of large networks is viewed by most practitioners as impractical and unrealistic. The main objective of this thesis is the proposal on MPC techniques which can be applied in practice to large traffic networks.

Some possible solutions are the use of a distributed MPC (by considering the network as a set of subsystems controlling each subsystem by one independent MPC, i.e. to use a decentralized or distributed control scheme), an hybrid MPC (by splitting the problem in a continuous optimization for the ramp metering signals and in a discrete optimization for speed limits), or genetic algorithms (by finding the fittest individuals within a generation, applying genetic operators for the recombination of those individuals, and generating a good offspring). This thesis proposes and analyses these solutions.

Other open problem in freeway traffic control is the dynamic operation of reversible lanes. Despite the long history and widespread use of reversible lanes worldwide, there have been few quantitative evaluations and research studies conducted on their performance. To address this problem, this thesis proposes a macroscopic model for reversible lanes and on-line controllers for the operation of reversible lanes.

Moreover, a MPC controller for freeway traffic requires a model to make accurate and reliable predictions of the traffic flow. On the other hand, this model is required to be fast enough, so that it can be used for on-line based control applications. Therefore, it is imperative to select or develop appropriate models, i.e., models that are fast and that provide accurate predictions. In this thesis, the METANET model and its extensions have been selected to be used for the prediction of the traffic flow and, based on this model, new advances in freeway traffic modeling for optimal control strategies are proposed.

1.2 Introduction to freeway traffic control

The most commonly used strategy to increase the efficiency of the road infrastructure is traffic control. The general goal of traffic control is to achieve network optimum terms of macroscopic indicators, which are favorable to characterize the network-wide performance level. The Total Time Spent (TTS) measure is a widely used quantity to describe efficiency by summing the time that vehicles spend in a traffic network.

It is a known fact that the outflow of a traffic congestion is significantly lower than the theoretical capacity of the same location. The phenomena, known as capacity drop, is due to the throughput degradation caused by moving jams. Traffic control uses control measures which prevent the occurrence of shock waves and capacity drops.

The performance of the overall controlled system is determined by the relevance and efficiency of the control strategy. Therefore, control strategies may be designed with care by applying powerful and systematic methods of optimization and automatic control, rather than via questionable heuristics.

This section gives a brief introduction of the most common control measures used in freeway traffic control. Two extended overviews of freeway traffic control are given by Hegyi [37] and Papageorgiou et al. [85].

1.2.1 Ramp metering

The most commonly applied traffic control measure is ramp metering (see Figure 1.2). Ramp metering allows to control the flow of vehicles entering the freeway from an on-ramp by a traffic light.

The ramp flow is determined by red, green and amber light timings. There are many implementations to achieve a certain ramp flow. In some countries like USA or The Netherlands, the ramp metering allows one car per green per lane. However, in other countries, there are implementations that allow two or more cars per green.

There are two main modes when using ramp metering:

- **Traffic spreading mode:** This mode is used in order to spread the vehicles which arrive in platoons to the freeway on-ramp. This platoons are caused, for example, when the traffic on the on-ramp arrives from a controlled intersection. This platoons may create undesirable disturbances which can be mitigated by spreading the platoon so the vehicles enter the freeway one-by-one.

- **Traffic restricting mode:** This mode is used to prevent a traffic breakdown on the freeway. This can be achieved by adjusting the metering rate in such a way that the density remains below the critical value. Therefore, travel times can be reduced and congestion blocking the off-ramp upstream can be prevented.

Figure 1.2: Ramp metering implementation in the United States

Previous works have proposed (and, in some cases, also implemented) several ramp metering strategies which can be divided in two main groups:

- **Fixed-time strategies:** These algorithms are determined off-line based on historical demands [80]. Standard optimization methods can solve these kind of problems and the resultant control input is easy to implement since no traffic measurements are necessary. However, these strategies do not take into account the traffic demand variations or other traffic disturbances during a day or from day-to-day.

- **Traffic-responsive strategies:** These techniques control the metering rate by an on-line feedback of the traffic conditions (flows, speeds, densities and/or ramp queues). The most used strategies are the demand-capacity strategy, ALINEA [88], METALINE [82] and FLOW [48].

The effectiveness of ramp metering has been shown on field studies like the ones made in Paris, in Amsterdam, and in Minnesota:

- In Paris metropolitan region, ALINEA and METALINE were applied and compared obtaining both considerable performance improvement [82].
- In The Netherlands, the effects of ramp metering on capacity, flow, speed, and travel time on eight of locations was studied. The different algorithms used and their impact are described in [96].
- In Minnesota, 430 ramp meters were shut down to evaluate their effectiveness reducing the traffic volume (in a 9%), the number of crashes, the emissions and the fuel consumption [76].

Many other studies have simulated ramp metering strategy for different conditions and with different control approaches using microscopic and/or macroscopic traffic flow models.

This thesis focuses on the use of traffic-responsive strategies in the traffic restricting mode in order to reduce the total time spent by the drivers. In Chapters 3, 4 and 5, ramp metering rates are optimally computed using a continuous Model Predictive Controller as in previous works [37, 53]. In Chapter 4, a linear feedback control algorithm (ALINEA) is also simulated and compared with the optimal controllers.

1.2.2 Variable speed limits

Variable Speed Limits (VSL) signs (Figure 1.3), which change the maximum speeds allowed by law for motor vehicles, are used in many recent freeways.

Currently, VSL are usually used to increase safety by decreasing the speed limits upstream of congested areas. A more stable traffic flow increases the safety and decreases the fuel consumption. This can be achieved by using speed limits that are above the critical speed causing a speed and density homogenization. These VSL result in a more stable and safer traffic flow, but no significant improvement of traffic volume is expected.

Nevertheless, VSL can also increase the traffic flow by using more complex algorithms. Preventing too high densities (especially on bottlenecks), higher flow can be achieved so the total time spent by drivers can be reduced. The idea is to allow speed limits that are lower than the critical speed in order to limit the inflow to these areas.

Many studies have proven the benefits of VSL:

- In [99], it is studied the application of variable speed limit on some motorways in The Netherlands concluding that the differences in volume, speed, and occupancy between and within lanes became smaller when variable speed limit was implemented.

Figure 1.3: VSL implementation on Seattle, USA

- In [2], a dynamic macroscopic model is proposed to estimate traffic density in real time and to activate VSL based on the density estimated by the model. They found that speed signaling can avoid congestion and improve the stability of traffic condition with constant flow and higher average speed.
- In [110], different VSL scenarios are tested concluding that the VSL benefits are clear when the traffic volume is equal to or greater than 2800 veh/h. They reached this number by studying demand driven congestion in one direction on a freeway with two lanes. Other benefits are: increase of the served traffic volume, travel time savings and reduction of speed standard deviation. It is also concluded that VSL can reduce queue time, reduce number of stops, and avoid congestion when the traffic volume is equal to or greater than 2000 veh/h for the same freeway configuration.

This thesis focuses on the use of VSL for the reduction of the total time spent by the drivers without an explicit consideration of the safety improvement. Con-

sequently, in Chapters 3 and 4, VSL are optimally computed using a continuous Model Predictive Controller as in previous works [37, 53]. Chapter 5 proposes an MPC framework which considers explicitly the effects of using discrete signals for the VSL panels.

1.2.3 Reversible lanes

A reversible lane is a lane in which traffic may travel in both direction, depending on certain conditions, by having overhead traffic lights and lighted street signs which notify drivers which lanes are open or closed to driving. Some more recent implementations of reversible lanes use a movable barrier to establish a physical separation between allowed and disallowed lanes of travel.

Reversible traffic operations [104] are widely regarded as one of the most cost-effective methods to increase the capacity of an existing freeway. The principle of reversible lanes (also known as "tidal flow") is to match available capacity to the traffic demand taking advantage of the unused capacity in the minor-flow direction lanes to increase the capacity in the major-flow direction.

There are numerous examples of reversible lanes successfully implemented during the last 85 years in many countries (especially in USA, Australia, Canada and UK) [47]. The reversible lanes are generally used in bottlenecks like bridges or tunnels but there are also examples of entire roadways routinely reversed [20].

Surprisingly, despite the long history and widespread use of reversible lanes worldwide, there have been few quantitative evaluations and research studies conducted on their performance [104]. There is also a limited number of published guidelines and standards related to their planning, design, operation, control, management, and enforcement.

Therefore, most reversible lane systems have been developed and managed based primarily on experience, professional judgment, and empirical observation.

In Section 2.1 a modification of the second-order macroscopic model METANET to address reversible lanes in order to allow the design of on-line control techniques for the reversible lane operation is proposed. The design of the corresponding control algorithms are in Section 6 (using dicrete MPC and a logic-based controller).

Figure 1.4: Reversible lane on The Lions Gate Bridge in Vancouver, Canada

1.2.4 Other traffic control measures

This section briefly describes other dynamic traffic control measures that can potentially improve the traffic performance. These traffic measures have not been considered in this thesis but it may be an interesting future work to adapt the proposed methods and controllers to these measures.

Main-stream metering

Main-stream metering [10] limits the flow on the freeway itself. The technical implementation is similar to a ramp metering: the number of vehicles that pass is controlled by choosing the relative green time in the traffic light cycle. Because of the similarity with on-ramp metering the same models are used for main-stream metering as for ramp metering.

Route guidance

Many modern freeways are equipped with route guidance systems which assist drivers in choosing their route to their destination when more alternative routes exist. The route guidance systems display traffic information on variable messages signs such as travel time to the next common point, congestion length, or delay on the alternative routes.

Shoulder lanes

In some freeways is allowed to be used the existing hard shoulder lane as a travel lane during highly congested times. The shoulder serves as a extra lane that will be tolled during high traffic periods. During off peak times, the lane would revert to a shoulder. Due to the extra lane the capacity of the road is increased preventing or reducing congestion. The disadvantage of using the shoulder lane is that safety is reduced. In the UK, usage of the hard shoulder is known as “hard shoulder running”.

Dedicated lanes

Dedicated lanes are traffic lanes set aside for particular types of vehicles. This reduces the hindrance that congestion causes to these vehicles. Moreover, dedicated lanes makes public transport more reliable and thus more attractive. A dedicated freight transport lane increases the stability and homogeneity of the traffic flow. Other common kind of dedicated lanes are carpool lanes which only are allowed to vehicles with a high occupancy (usually, more than 2 or 3 passengers).

The "keep your lane directive"

The disturbances in the freeway traffic flow can be reduced using the "keep your lane" directive, which forces the drivers to keep in the same lane (i.e. the drivers are not allowed to change lanes). This traffic control measure is useful when the traffic flow is close to the critical density and may be an alternative to the homogenizing speed limits.

1.2.5 Necessary conditions for a successful traffic control on freeways

In [37], the conditions that are necessary for a successful traffic control are presented assuming that the TTS is the performance measure to be minimized. It is also assumed that the performance degradation is caused by the capacity drop or the blocking of traffic not traveling over the real bottleneck location.

This study concludes that the main conditions for a effective traffic control are:

- The presence of the capacity drop or blocking in the real traffic network.
- The use of a model that is able to reproduce with a sufficiently accuracy the capacity drop or blocking.
- The possibility to reduce the inflow of the congested area.
- The vehicles which are delayed by traffic control are inside the network boundaries.
- The roads downstream have to be able to accommodate the improved traffic flows.
- The presence of traffic demands for which control is useful.

1.3 Introduction to macroscopic traffic flow models

In order to build the basic understanding of traffic flow models, this chapter begins with first providing a brief overview of macroscopic traffic flow models in Section 1.3 and, more specifically, of the second order model METANET in Section 1.4.

1.3.1 Traffic flow models

Three main approaches have been used for the traffic flow modeling:

- Macroscopic models, which model traffic as a particular fluid with aggregate variables such as density, mean speed and flow. Macroscopic models are low-detailed representation of the process using only aggregated variables based on hydrodynamical analogies. Consequently, the individual vehicle motions and interactions are completely neglected. Therefore, they use a high level of aggregation without distinguishing between individual vehicle actions such as a lane change. These models can further be classified according to the number of independent state variables.

- Mesoscopic models, which do not track individual vehicles, but describe the behavior of individual vehicles in probabilistic terms. These models are medium detailed models where small groups of interacting vehicles are traced in these frameworks besides of individual particles. In addition behavioral information can be incorporated by means of probabilistic terms. The most known representatives of mesoscopic approach are based on gas kinetic consideration.

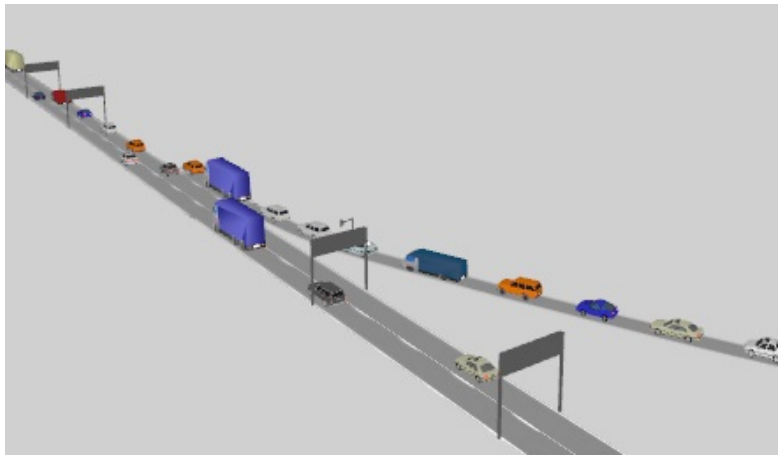

Figure 1.5: Instant of 3D Microscopic freeway traffic simulation with ITS signals (ramp metering and variable speed limits) using Aimsun 6.0

- Microscopic models, which describe the longitudinal and lateral movement of individual vehicles individually. These models incorporate a high-detailed description of each individual vehicle motions and their interactions. Since the vehicles are modeled individually, it is easy to assign different characteristics to each vehicle. These characteristics can be related to the driving style of the driver (aggressive, patient), vehicle type (car, truck), its destination, and chosen route. We refer the interested reader to [90] for an extensive comparison of microscopic simulation models.

One of the main advantages of microscopic models is the ability to study individual vehicle motion perceiving visually the real process using a simulation software like AIMSUN (see Fig. 1.5) or Paramics. However, microscopic models are difficult to use for model based control due to the computational effort needed; therefore, these models are commonly used only for simulations and off-line controller design.

On the other hand, macroscopic models are more suitable for real-time applications because they are relatively fast and have an analytical definition of the aggregate variables. Therefore, this work focuses on macroscopic traffic models.

1.3.2 Common link equations

Since 1955 [57], a number of dynamic macroscopic models have been proposed, mostly based on partial differential equations [44]. This section shows some common characteristics of the different macroscopic freeway traffic models.

The macroscopic models are discrete in both space and time, dividing the freeway into consecutive sections. The network is represented as a graph where the links m correspond to freeway stretches. Each link m is divided into N_m segments of length L_m with λ_m lanes as can be seen on Fig. 1.6.

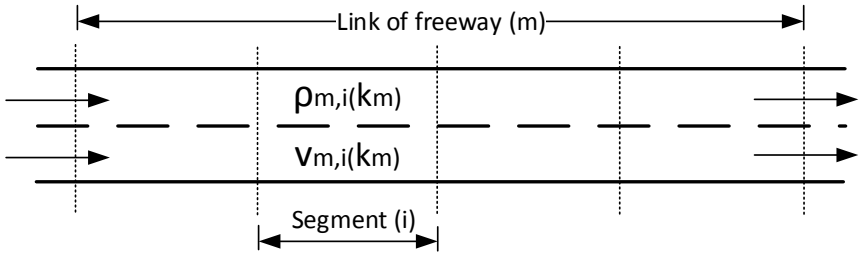

Figure 1.6: Freeway Link divided into segments

For simplicity, in this work all the segments are considered to have different lengths (i.e. $N_m = 1 \ \forall \ m$), making it unnecessary to differentiate between links and segments; thus, hereafter only index i will be used.

Each segment is dynamically characterized by the traffic density $\rho_i(k_m)$ (the number of vehicles occupying a length of freeway per lane) and the mean speed $v_i(k_m)$ (average of the instantaneous speed of vehicles occupying a section of a freeway) where k_m is the time instant $t = k_m T_m$ and T_m is the simulation time step size.

For numerical stability the following constraint (where $v_{free,i}$ is the free flow speed that the cars reach in steady state for segment i) has to be established:

$$T_m \cdot v_{free,i} \leq L_i \tag{1.1}$$

The traffic flow $q_i(k_m)$ (the number of vehicles passing a point in a given time) that leaves each segment can be computed for each time step by:

$$q_i(k_m) = \lambda_i \rho_i(k_m) v_i(k_m) \tag{1.2}$$

All the macroscopic models include the conservation equation (1.3) representing the fact that change in number of cars in a segment is due to the flows at the boundaries:

$$\begin{aligned} &\rho_i(k_m + 1) = \rho_i(k_m) + \\ &+ \frac{T}{\lambda_i L_i}(q_{i-1}(k_m) - q_i(k_m) + q_{ramp,i}(k_m) - \beta_i(k_m) \cdot q_{i-1}(k_m)) \end{aligned} \tag{1.3}$$

where $q_{ramp,i}(k)$ is the traffic flow that enters the freeway from an on-ramp and $\beta_i(k)$ is the split ratio of an off-ramp (i.e. the percentage of vehicles exiting the freeway through an off-ramp in segment i). For segments without an off-ramp or an on-ramp at the end of the segment, $\beta_i(k_m) = 0$ and $q_{ramp,i}(k) = 0$, respectively.

The following equation expresses the dynamics of ramp queues $w_i(k_m)$ and is commonly used by first and second order macroscopic traffic models:

$$w_i(k_m + 1) = w_i(k_m) + T_m(D_i(k_m) - q_{ramp,i}(k_m)) \tag{1.4}$$

where $D_i(k_m)$ is the vehicle flow arriving to the beginning of an on-ramp (i.e. the ramp demands).

In order to complete the model, first order models (like CTM) include a static speed-density relationship. However, second order models (like METANET) address the speed dynamics introducing a new state variable which is capable to capture more dynamics.

In this work, we have selected the traffic model METANET [87]. However, it is important to note that the methods we propose (in Section 2.1, 2.2 and Chapters 3, 4 and 5) are independent of the traffic model used, so they can be equivalently applied using other macroscopic traffic models, if those are capable of including the effect of traffic control measures in their formulation (like some versions of the Cell Transmission Model CTM [15]).

1.4 Introduction to METANET

The METANET model [65, 87] is a macroscopic second order traffic flow model that provides a good trade-off between simulation speed and accuracy for on-line traffic control purposes [26, 94].

METANET is a deterministic model for the simulation of traffic systems in motorway networks of arbitrary topology and characteristics, including motorway stretches, bifurcations, on-ramps, and off-ramps. This model allows all kinds of traffic conditions and events to be simulated with prescribed characteristics. Furthermore, METANET can incorporate the effects of control actions such as ramp metering, route guidance, and variable speed limits.

For the sake of simplicity, merge and join nodes, and other extensions are not considered in this work. For the full description of METANET we refer to the literature [65, 87, 38].

1.4.1 Link equations

The METANET model uses density and speed as state variables with two equations describing the system dynamics. The first one (1.3) expresses the conservation of vehicles. It is the same equation used for other macroscopic traffic flow models like the LN-CTM model (see Section 2.3.1).

The second equation (1.5) expresses the mean speed as a sum of the previous mean speed, a relaxation term (drivers tendency to accelerate or decelerate toward their desired speed), a convection term (influence of the speed of vehicles upstream) and an anticipation term (influence of the speed of vehicles upstream):

$$\begin{aligned} v_i(k_m+1) = v_i(k_m) + \frac{T_m}{\tau_i}(V(\rho_i(k_m)) - v_i(k_m)) + \\ + \frac{T_m}{L_i} v_i(k_m)(v_{i-1}(k_m) - v_i(k_m)) - \frac{\mu_i(k_m) T_m}{\tau_i L_i} \frac{\rho_i(k_m+1) - \rho_i(k_m)}{\rho_i(k_m) + K_i} \end{aligned} \tag{1.5}$$

where K_i, τ_i and μ_i are model parameters that have to be estimated for each segment and $V(\rho_i(k_m))$ is the driver's desired speed (1.9).

As proposed in [37], the model can take different values for μ_i, depending on whether the downstream density is higher or lower than the density in the actual segment:

$$\mu_i(k_m) = \begin{cases} \mu_{i,h} & \text{for } \rho_{i+1}(k_m) \leq \rho_i(k_m) \\ \mu_{i,l} & \text{Otherwise} \end{cases} \tag{1.6}$$

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

The possibility of using traffic flow instead of density as the state variable for traffic control purposes is considered in [106]. The reason for this is that flow measurement using a point sensor, such as an inductive loop detector, while good density estimation is very costly, if not impossible. This equivalent model may be a good option in practical cases but it still needs to be validated. [62] suggests improvements on the model proposing several alternatives for the convection term of the speed equation.

If there is an on-ramp in a segment, the following negative term $\bigtriangledown_{\mathrm{r}} v_i(k_m)$ is added to the right-hand side of equation in order to model the speed drop caused by merging phenomena: (1.5):

$$\bigtriangledown_{\mathrm{r}} v_i(k_m) = -\frac{\delta_i T_m q_{r,i}(k_m) v_i(k_m)}{L_i \lambda_i (\rho_i(k_m) + K_i)} \tag{1.7}$$

where δ_i a is model parameter that is positive if there is a on-ramp at the end of segment i.

Equivalently, in the segments where there is a lane-drop, the following negative term $\bigtriangledown_{\mathrm{d}} v_i(k_m)$ is added:

$$\bigtriangledown_{\mathrm{d}} v_i(k_m) = -\frac{\phi_i T_m \Delta_\lambda \rho_{i-1}(k_m) v_{i-1}^2(k_m)}{L_{i-1} \bar{\lambda} \rho_{crit,i-1}} \tag{1.8}$$

where ϕ_i is a model parameter that is positive if there is a lane-drop at the end of segment i.

As proposed in [37], the driver's desired speed is defined by:

$$V(\rho_i(k_m)) = \min\left(V_{wVSL}(\rho_i(k_m)), (1+\alpha_i) \cdot V_{c,i}(k_m)\right) \tag{1.9}$$

where α_i is a model parameter, $V_{c,i}$ is the speed limit applied in segment i and $V_{wVSL}(\rho_i(k_m))$ is the desired speed without Variable Speed Limits. Different functions can model $V_{wVSL}(\rho_i(k_m))$ (i.e. the fundamental diagram). The following exponential form is used in this work in the absence of variable speed limits:

$$V_{wVSL}(\rho_i(k_m)) = v_{\mathrm{free},i} \exp\left(-\frac{1}{a_i}\left(\frac{\rho_i(k_m)}{\rho_{\mathrm{crit},i}}\right)^{a_i}\right) \tag{1.10}$$

where a_i is a model parameter and $\rho_{crit,i}$ is the critical density (the density corresponding to the maximum flow in the Fundamental Diagram (FD)).

If the desired flow $Q(\rho_i(k_m)) = V(\rho_i(k_m)) \cdot \rho_i(k_m)$ is represented graphically without considering VSL, the Fundamental Diagram of traffic flow is obtained (which can be seen in Fig.1.7). The fundamental diagram gives us the static characteristic of the system (i.e. flow leaving segment i on a homogeneous freeway as a function of the density).

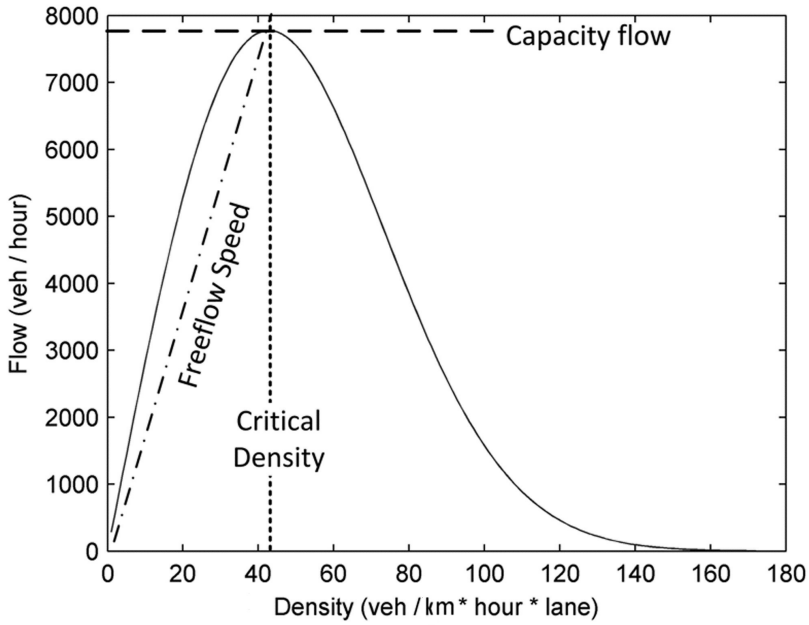

Figure 1.7: Fundamental diagram of traffic

In this work, the VSL are included by a minimum term in the desired speed equation (1.9) [37]. However, in [87], VSL are included in the model rendering the three parameters of the fundamental diagram $\rho_{crit,i}$, $v_{free,i}$ and a_i.

In [86] is studied the effect of VSL on aggregate traffic flow behavior on the basis of traffic data comparing the equation used in this work (1.9) with other options.

Finally, the following equation defines the flow that enters from an on-ramp.

$$q_{\text{ramp},i}(k_m) = \tag{1.11}$$
$$= \min(r_i(k_m) \cdot C_{\text{ramp},i}, D_i(k_m) + \frac{w_i(k_m)}{T}, C_{\text{ramp},i} \cdot \frac{\rho_{J,i} - \rho_i(k_m)}{\rho_{J,i} - \rho_{\text{crit},i}})$$

where $C_{\text{ramp},i}$ is the origin capacity, $\rho_{\text{m},i}$ is the maximum density, and $r_i(k_m)$ is the ramp metering rate.

1.4.2 Boundary equations

Boundary conditions have to be defined because the traffic situation downstream and upstream of a segment influences the traffic in the actual segment. In particular, in the METANET model the states of a segment depend on the upstream speed, the upstream flow, and the downstream density.

Therefore, we need to describe the upstream speed and flow for the main-stream entries of the network and downstream densities for the main-stream exits of the network.

The main-stream origin is suggested to be modeled differently than the on-ramp origins in [37]. It is argued that the inflow of a segment can be limited by the limiting speed $v_{\text{lim}}(k_m)$ of the segment:

$$v_{\text{lim}}(k_m) = \min\left(V_{c,1}(k_m), v_1(k_m)\right) \tag{1.12}$$

[37] assumes that the maximal inflow equals the flow that follows from the speed-flow relationship that can be derived from (1.2) and (1.10) with the speed equal to limiting speed. Therefore, the outflow of the main-stream $q_{\text{o},i}(k_m)$ can be computed by:

$$q_{\text{o},i}(k_m) = \min\left(D_o(k_m) + \frac{w_o(k_m)}{T}, q_{max,o}(k_m)\right) \tag{1.13}$$

where $q_{\text{max},o}(k_m)$ can be computed by:

$$q_{\text{max},o}(k_m) = \tag{1.14}$$
$$= \begin{cases} \lambda_o v_{\text{lim}}(k_m)\rho_{\text{crit},1}(-a_o \ln(\frac{v_{\text{lim}}(k_m)}{v_{\text{free},i}}))^{\frac{1}{a_o}} & \text{if } v_{\text{lim}}(k_m) < V(\rho_{\text{crit},1}) \\ \lambda_o V(\rho_{\text{crit},1})\rho_{\text{crit},1} & \text{Otherwise} \end{cases}$$

The virtual speed of a main-stream origin $v_o(k_m)$ can be user-defined (in Chapters 4, 5 and Sections 3.1) or taken from real data (in Chapter 6 and Sections 2.2, 2.3 and 3.2).

If it is not specified, $v_o(k_m)$ is set to be equal to the speed of the first segment $v_o(k_m) = v_1(k_m)$ (in chapters 3, 4 and 5). The boundary conditions for the upstream flow are described by the origin flow equations (1.15) (1.14).

The only downstream boundary condition required is the virtual downstream density $\rho_{N+1}(k_m)$. In the standard METANET model, it is assumed that the destination is congestion-free, but it is also possible to consider user-defined density scenario (in Sections 2.2 and 2.3).

When the destination is assumed to be congestion free, the virtual downstream density ($\rho_{N+1}(k_m)$) is assumed to be the smallest of the critical density (($\rho_{\text{crit},N}$) and the density of the last segment N ($\rho_N(k_m)$). This can be rewritten as:

$$\rho_{N+1}(k_m) = \min\left(\rho_N(k_m), \rho_{\text{crit},N}\right) \tag{1.15}$$

As previously said, for the sake of simplicity, merge and join nodes, and other extensions are not considered in this work.

1.5 Introduction to Model Predictive Control

Model Predictive Control (MPC) [9], also called receding horizon Predictive Control, originated in the late seventies and has developed considerably since then. The term Model Predictive Control does not designate a specific control strategy but rather an ample range of control methods which make explicit use of a model of the process to obtain the control signal by minimizing an objective function.

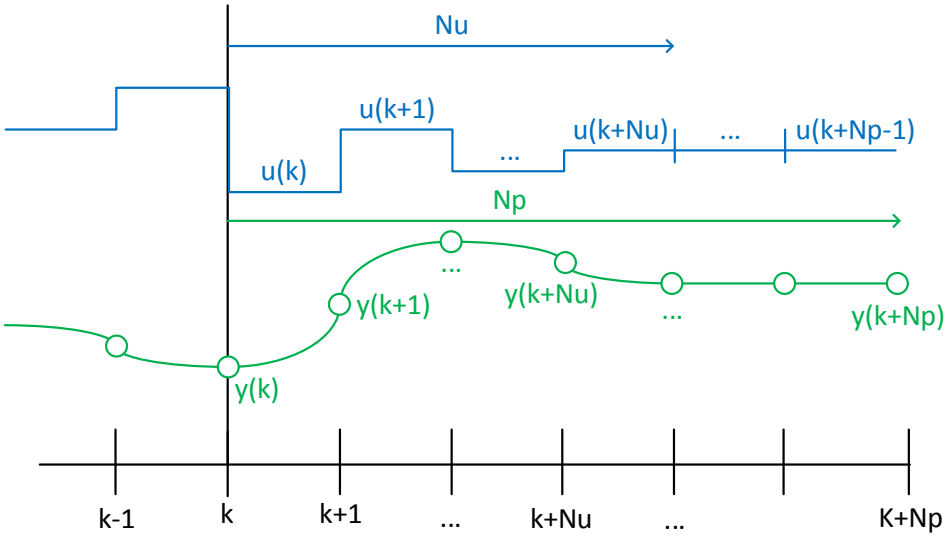

Figure 1.8: Receding horizon strategy

These design methods lead to controllers which have practically the same structure and present adequate degrees of freedom. The main common concepts behind a model-based predictive control (MPC) strategy are:

- The use of a prediction model to obtain the trajectories of relevant variables of the system.
- The optimization of an objective function to determine the best sequence of control actions for the system.
- The application of the rolling horizon procedure: from the best sequence of control actions only the first component is applied to the system and in the next control step the initial conditions are updated and the procedure is repeated again (see Figure 1.8).

The various MPC algorithms only differ amongst themselves in the model used to represent the process, the noises, and the cost function to be minimized. This type

of control is of an open nature, within which many works have been developed and are widely received by the academic world and industry.

There are many applications of predictive control successfully in use at the current time, not only in the process industry but also applications to the control of other processes ranging from robots [79] to clinical anesthesia [59].

1.5.1 Advantages and disadvantages

MPC presents a series of advantages over other methods, among which the following stand out:

- It is particularly attractive to staff with only a limited knowledge of control because the concepts are very intuitive and at the same time the tuning is relatively easy.
- It can be used to control a great variety of processes, from those with relatively simple dynamics to more complex ones, including systems with long delay times or non-minimum phase or unstable ones.
- The multivariable case can easily be dealt with.
- It intrinsically has compensation for dead times.
- It introduces feed forward control in a natural way to compensate for measurable disturbances.
- The treatment of constraints is conceptually simple, and these can be included during the design process.
- It is very useful when future references (robotics or batch processes) are known.
- It is a totally open methodology based on certain basic principles which allows for future extensions.

As is logical, however, it also has its drawbacks. One of these is that the derivation of the control law is more complex than with classical controllers. If the process dynamic does not change, the derivation of the controller can be done beforehand, but in the adaptive control case or in the case of measurable disturbances, all the computation has to be carried out at every sampling time. When constraints are considered, the amount of computation required is even higher.

Even so, the greatest drawback is the need for an appropriate model of the process to be available. The design algorithm is based on prior knowledge of the model and is independent of it, but it is obvious that the benefits obtained will be affected by the discrepancies existing between the real process and the model used.

In practice, MPC has proven to be a reasonable strategy for industrial control, in spite of the original lack of theoretical results at some crucial points such as stability and robustness.

1.5.2 MPC strategy

The methodology of all the controllers belonging to the MPC family is characterized by the receding horizon strategy, represented in Figure 1.8:

1. The future outputs for a determined prediction horizon N_p, called the prediction horizon, are predicted at each instant k using the process model. These predicted outputs depend on the known values up to instant k (past inputs and outputs) and on the future control signals, which are those to be sent to the system and calculated.
2. The set of future control signals is calculated by optimizing a determined criterion. The control effort is included in the objective function in most cases. The future control law is usually considered be constant from a determined control horizon N_u.
3. The control signal is sent to the process whilst the next control signals calculated are rejected, because at the next sampling instant is already known and step 1 is repeated with this new value and all the sequences are brought up to date.

In order to implement this strategy, the basic structure shown in Figure 1.9 is employed. The model is used to predict the future plant outputs, based on past and current values and on the proposed optimal future control actions. These actions are calculated by the optimizer taking into account the cost function (where the future tracking error is considered) as well as the constraints.

The process model plays, in consequence, a decisive role in the controller. The chosen model must be able to capture the process dynamics to precisely predict the future outputs and be simple to implement and understand. As MPC is not a unique technique but rather a set of different methodologies, there are many types of models used in various formulations. The State Space Model is widespread in the academic research community as the derivation of the controller is very simple even for the multivariable linear case. The state space description allows for an easier expression of stability and robustness criteria.

The optimizer is another fundamental part of the strategy since it provides the control actions. If the cost function is quadratic, its minimum can be obtained as an explicit function (linear) of past inputs and outputs and the future reference trajectory. The size of the optimization problems depends on the number of variables and the prediction horizons used and usually turns out to be a relatively modest optimization problem which does not require solving sophisticated computer codes.

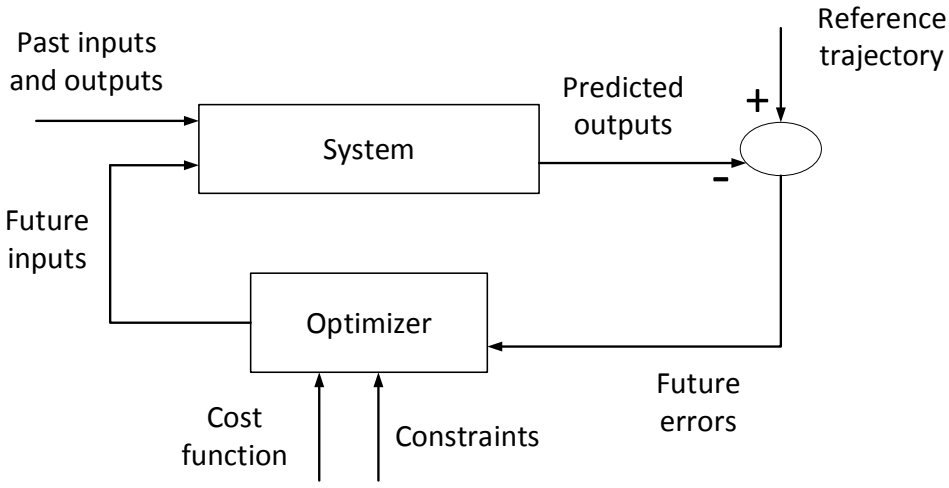

Figure 1.9: Basic structure of MPC

However, using non-linear models, the optimization problem is non-convex so its resolution, which is usually done by using an iterative optimization method, is much more difficult than the Quadratic Programming (QP) problem. Problems relative to local optimum appear, not only influencing control quality but also deriving in stability problems. The difficulty of the optimization problem translates into an important increase in computation time.

Notice that the MPC strategy is very similar to the control strategy used in driving a car. The driver knows the desired reference trajectory for a finite control horizon and by taking into account the car characteristics (mental model of the car) decides which control actions (accelerator, brakes, steering) to take to follow the desired trajectory. Only the first control actions are taken at each instant, and the procedure is repeated for the next control decision in a receding horizon fashion.

It has to be taken into account that when using classical control schemes, the control actions are taken based on past errors. If the car driving analogy is extended, the classical control way of driving a car would be equivalent to driving

the car only using the mirror as shown in Figure 1.10:

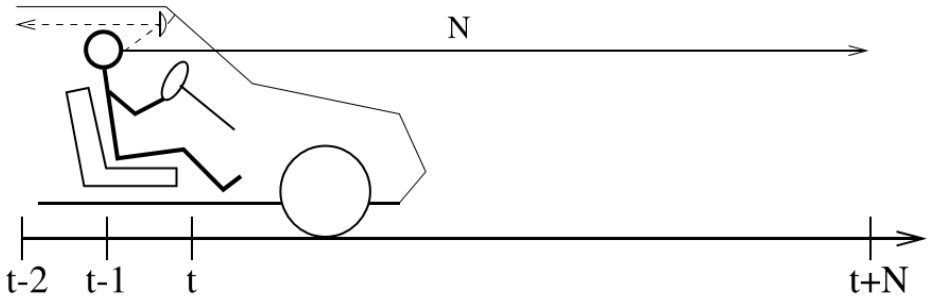

Figure 1.10: MPC Analogy

1.5.3 Problem statement

To formalize all the concepts explained in the previous subsections, consider the discrete-time non-linear system whose dynamic evolution is described by the following state-space model:

$$x(k_{\mathrm{m}}+1) = f(x(k_{\mathrm{m}}), u(k_{\mathrm{m}}), d(k_{\mathrm{m}})) \tag{1.16}$$

with $x(k_{\mathrm{m}})$ the state, $u(k_{\mathrm{m}}) = [(u^{\mathrm{d}}(k_{\mathrm{m}}))^T, (u^{\mathrm{c}}(k_{\mathrm{m}}))^T]^T$ the discrete and continuous input vector, and $d(k_{\mathrm{m}})$ the non-controllable input vector (usually demand profiles and other exogenous variables).

In some circumstances, the simulation sample time T_{m} may be different from the controller sample time T (i.e. $t = k_{\mathrm{m}} \cdot T_{\mathrm{m}} = k \cdot T$). In these cases, the control inputs are usually considered constant during one controller sample resulting in the following state-space model:

$$x(k+1) = f(x(k), u(k), d(k)) \tag{1.17}$$

In a model predictive controller, the core is the optimization of a cost function $J(x_{\mathrm{t}}(k), u_{\mathrm{t}}(k), d_{\mathrm{t}}(k))$, which is used to measure the performance of the system where the vectors $x_{\mathrm{t}}(k)$ and $d_{\mathrm{t}}(k)$ are the state and non-controllable input predictions along the prediction horizon N_{p}; and $u_{\mathrm{t}}(k)$ is the control sequence along the control horizon N_{u}. The control inputs are kept constant after the control horizon N_{u} (i.e. $u(k+N_{\mathrm{u}}-1) = u(k+N_{\mathrm{u}}) = ... = u(k+N_{\mathrm{p}}-1)$).

Since the formulation is based on the solution of an optimization problem, it is possible to explicitly include constraints. In the case of systems with discrete

control actions we can include that characteristic as a constraint, and obtain suitable control laws by using a proper mixed-integer optimization solver.

MPC uses optimization techniques and tools to optimize the control inputs in such a way that the value of the given cost function is minimal. Depending on the system model, the constraints, and the cost function, the solutions obtained can be optimal or suboptimal. In principle MPC uses on-line optimization to design optimal control inputs. But to gain some computational speeds it is also possible (in some cases) to design MPC based on off-line optimization (such MPC is known as explicit MPC).

Assuming that the feasible values of the states and the inputs are given by the following generic constraints $x(k) \in \mathbb{X}$, $u^{\mathrm{c}}(k) \in \mathbb{U}$, and $u^{\mathrm{d}}(k) \in \mathbb{S}$ for all k, representing explicitly physical or operational constraints of the system, the MPC problem can be formulated as the following mixed-integer non-linear optimization problem:

$$\min_{u_{\mathrm{t}}(k)} J(u_{\mathrm{t}}(k), x_{\mathrm{t}}(k), d_{\mathrm{t}}(k)) \tag{1.18}$$

subject to:

$$\begin{aligned}
&x(k+\ell+1) = f(x(k+\ell), u(k+\ell), d(k+\ell)),\\
&x(k) = x_k\\
&x(k+\ell+1) \in \mathbb{X},\\
&u^{\mathrm{c}}(k+\ell) \in \mathbb{U},\\
&u^{\mathrm{d}}(k+\ell) \in \mathbb{S},\\
&h(u_{\mathrm{t}}(k), x_{\mathrm{t}}(k), d_{\mathrm{t}}(k)) \in \mathbb{D},\\
&\text{for } \ell = 0, 1, ..., N_{\mathrm{p}} - 1,
\end{aligned}$$

with $\mathbb{S}$ the set of possible control values $\mathbb{S} = \{u_1, u_2, ..., u_M\}$ for the corresponding feasible discrete input components. The term $h(u_t(k), x_t(k), d_t(k)) \in \mathbb{D}$ corresponds to the rest of the constraints and x_k is last measured state at time step k. Using the rolling horizon procedure, only the first control action $u(k)$ of the optimal sequence is applied to the system, and in the next time step the initial conditions are updated and the procedure is repeated.

1.6 Overview of the thesis

The outline of the thesis is the following (Figure 1.11 clarifies the connections between the chapters and sections):

Chapter 2: Freeway traffic Modeling for Optimal Control. Based on the original METANET model, introduced in in Sections 1.3 and 1.4, this chapter proposes new advances in freeway traffic modeling for optimal control strategies. Firstly, a macroscopic model for reversible lanes on freeways based on METANET is proposed to address reversible lanes in order to allow the design of on-line control techniques for the reversible lane operation. Subsequently, a new identification algorithm for METANET is proposed using a new mathematical definition of the Fundamental Diagram (FD) of traffic flow. Finally, the choice of METANET for the network modeling is justified by comparing the main features and the prediction accuracy of METANET and Cell Transmission Model (LN-CTM).

Chapter 3: Freeway Traffic Control by using Model Predictive Control (MPC). Since traffic systems are highly non-linear and time-variant systems, Model Predictive Control (MPC) is a promising candidate. This chapter provides a brief account of the basic concepts of MPC for traffic systems showing the behavior of a MPC controller in a simple case study. Subsequently, it is analyzed the robustness of MPC for freeway traffic with respect to variations in the mainstream demand and new methods for online demand estimation are proposed.

Chapter 4: Distributed MPC for freeway traffic control. The main problem of nonlinear centralized MPC is that the computational time quickly increases with the size of the network. Thus, centralized MPC could be difficult to apply for large networks. The loss of performance due to the decentralization and the difficulty of implementing a completely centralized control for a large network is analyzed in this chapter by comparing the performance of local and centralized controllers for a relatively large traffic network. Subsequently, distributed MPC algorithms are proposed, which can be implemented in real time for a large enough traffic network minimizing the total time spent by the drivers.

Chapter 5: Discrete and hybrid Model Predictive Control for freeway traffic. The discrete characteristics of the speed limits values and some necessary constraints for the actual operation of VSL are usually underestimated in the literature, so we propose a way to include them by using two hybrid MPC approaches for freeway traffic control considering VSL as discrete variables as in current real world implementations. Since solving such a problem is complex and

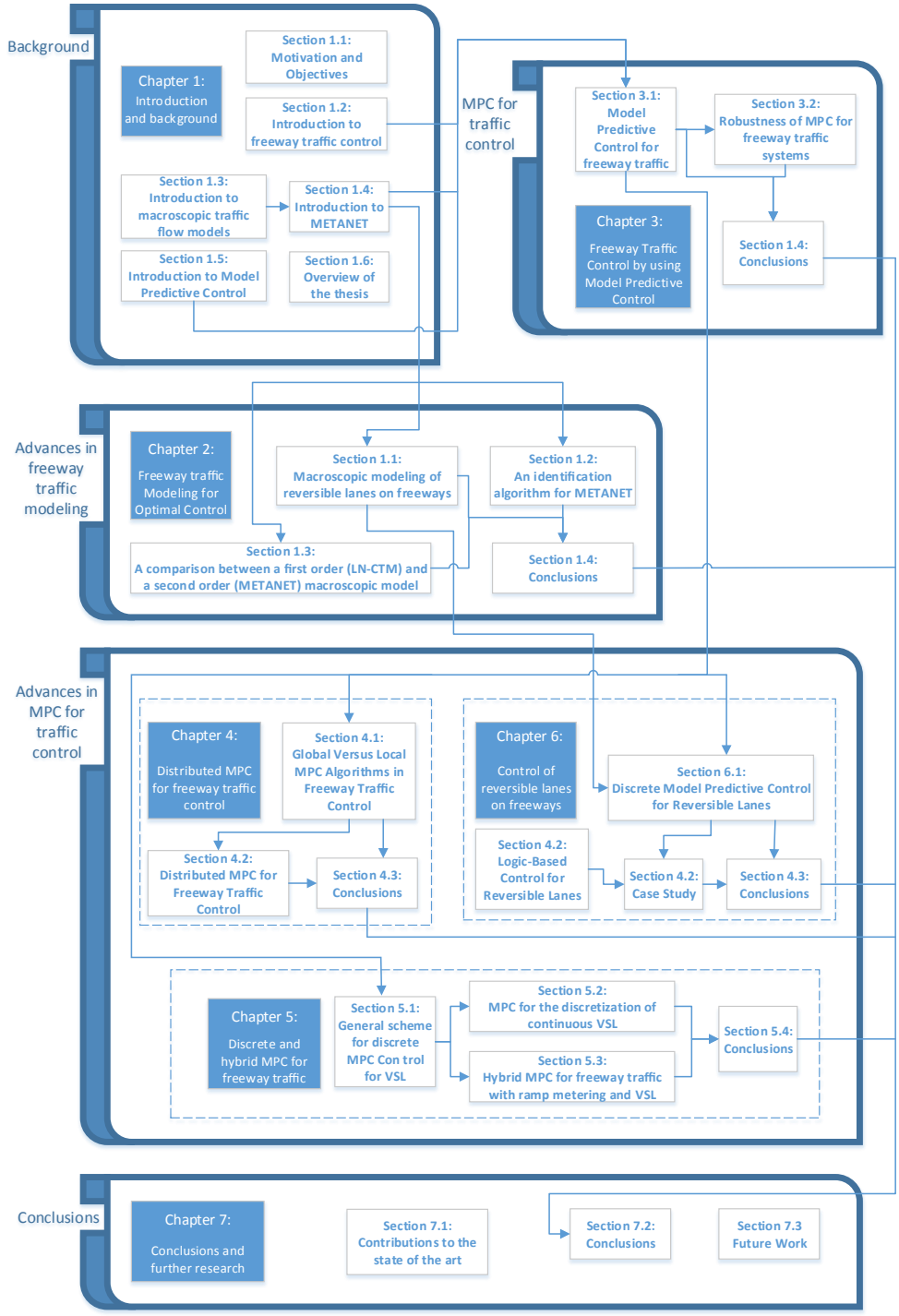

Figure 1.11: Schematic representation of the connections between the sections

difficult to execute in real-time, we propose a set of methods based on the limitation of the number of feasible nodes and the use of genetic algorithms to obtain reasonable control actions in a limited computation time.

Chapter 6: Control of reversible lanes on freeways. This chapter proposes the use of two kinds of controllers for the dynamic operation of reversible lanes on freeways based on the extension of METANET proposed in Section 2.1. The first one is an easy-to-implement logic-based controller which takes into account the congestion lengths generated by the reversible lane bottleneck and uses this information for the dynamic operation of the lanes. The second one is a discrete MPC which minimizes the Total Time Spent of the modeled network within some constraints for the maximum values of the generated bottleneck queues. The discrete optimization is carried out via evaluation of the cost function for all the leafs in a reduced search tree.

Chapter 7: Conclusions and further research The thesis ends with a chapter that analyzes the most relevant contributions and conclusions. Additionally, it is pointed out future research lines in the field of Model Predictive Control for freeway traffic.

Chapter 2

Freeway Traffic Modeling for Optimal Control

Since the main goal of the thesis is to design optimal traffic control strategies, the approach requires a model to make accurate and reliable predictions of the traffic flow. On the other hand, this model is required to be fast enough, so that it can be used for on-line based control applications. Therefore, it is imperative to select or develop appropriate models, i.e., models that are fast and that provide accurate predictions. In this thesis, the METANET model and its extensions have been selected to be used for the prediction of the traffic flow.

In the previous chapter (in Sections 1.3 and 1.4), it was done a brief overview of macroscopic traffic flow models and, more specifically, of the second order model METANET.

Based on this model (the original version of METANET), new advances in freeway traffic modeling for optimal control strategies are proposed in this chapter:

- Firstly, in Section 2.1, a macroscopic model for reversible lanes on freeways is proposed to address reversible lanes in order to allow the design of on-line control techniques for the reversible lane operation.
- Before any traffic model can be used to predict the evolution of the traffic situation, the model needs to be calibrated and validated. Previously proposed identification algorithms for METANET usually falls in local minima, especially if a limited number of sensors are available. Consequently, in Section 2.2, a new identification algorithm for METANET is proposed. Moreover, a new mathematical definition of the Fundamental Diagram (FD) of traffic flow is proposed in this section.
- Finally, the choice of METANET for the network modeling is justified in Section 2.3. This section compares the main features and the prediction

accuracy of METANET and Cell Transmission Model (LN-CTM) over a section of the I-210 West in Southern California.

Parts of this chapter are published in [34, 26, 30]. The research related with the Centenario Bridge benchmark has been published in press in Europa Press [89], El Economista [23], Diario de Sevilla [19], ABC, 20 Minutos and others Spanish newspapers.

2.1 Macroscopic modeling of reversible lanes on freeways

This section proposes a modification of the second-order macroscopic model METANET [65] to address reversible lanes (see subsection 1.2.3) in order to allow the design of on-line control techniques for the reversible lane operation (in Section 6). The proposed modification of the model METANET can be equivalently applied to other macroscopic traffic models as the Cell Transmission Model (CTM) [15]).

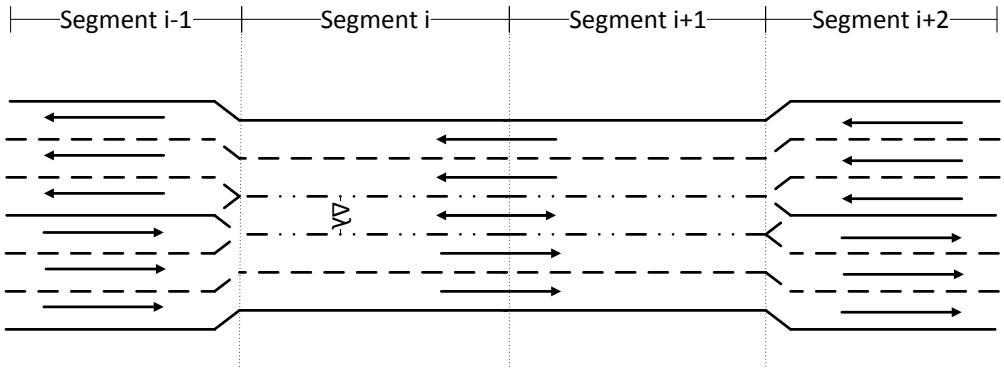

Figure 2.1: Freeway stretch with one reversible lane

Consider the stretch of freeway in Figure 2.1 with $\bar{\lambda}$ lanes in each direction. For a certain number of consecutive segments, there is a bottleneck for which Δ_λ lanes have to be shared between both directions. This is done by Δ_λ reversible lanes in which traffic may travel in either direction depending on the current traffic conditions.

In each direction the reversible lanes may be modeled like variable lanes drop (i.e. lanes drop which could appear or disappear in a certain sample time). Different but equivalent modeling has to be used for the closing and the opening of the reversible lanes. Also different models need to be used depending of the location of the Variable Message Signs (VMS).

2.1.1 Closing of the lanes

VMS at the beginning of the reversible lane

Consider one direction of the stretch of freeway in Figure 2.1 with initially $\bar{\lambda}$ lanes for all the segments. Assume that there are VMS located at the beginning of the reversible lanes to inform drivers about the current status of the reversible lanes (i.e. open or closed for arriving traffic). Assume that, for a certain number of consecutive segments, Δ_λ reversible lanes which were open to the arriving traffic, are closed at time step k_C creating a merging area as can be seen in Figure 2.2.

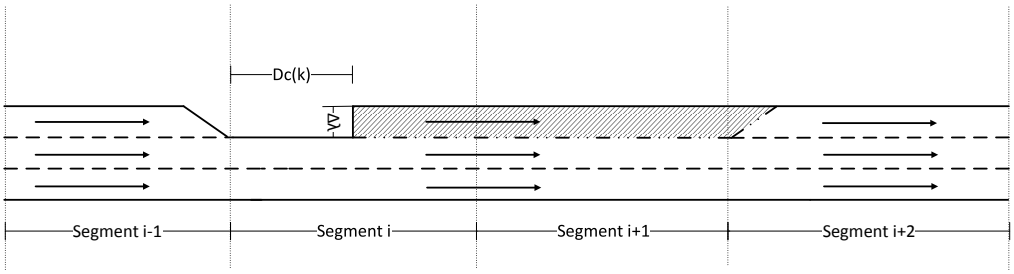

Figure 2.2: One direction of freeway in Figure 2.1 with one lane being closed

For the correct modeling of the reversible lanes, it has to be taken into account that, although the lanes are closed at time step k_C, the remaining cars need a certain time to leave the corresponding segments. This effect is modeled by the definition of $D_c(k)$, which is an estimation of the length of the lane that is already car-free. It has to be pointed out that using the common density equation with an instantaneous change of the number of lanes would entail the violation of the conservation equation.

The distance $D_c(k)$ can be computed according to (2.1) assuming that the speed of the remaining cars in the closed lanes equals the mean speed of all the lanes in the corresponding segment.

$$D_c(k_m + 1) = D_c(k_m) + T_m(v_j(k_m)) \text{ with } D_c(k_C) = 0. \tag{2.1}$$

where $v_j(k_m)v_j(k_m)$ is the speed of the segment which has a reversible lane partially closed at time step k_m.

In the segment upstream of the lane closing ($i-1$ in Fig. 2.2), the lane-drop term $\bigtriangledown_{\mathrm{d}} v_i(k_m)$ (1.8) has to be added in the speed equation (1.5) at sample time k_C. This term remains active as long as the lanes are closed. The density equation (1.3) of the upstream segment $i-1$ is not affected by the lane closing.

In the segments affected by the lane reduction (in Fig. 2.2, segments i and $i+1$), the traffic states per lane are modeled by defining an equivalent number of lanes

$\widehat{\lambda}_i(k_m) \in [\bar{\lambda} - \Delta\lambda, \bar{\lambda}]$ during the period of time that there are still cars leaving the closed lanes. The equivalent number of lanes $\widehat{\lambda}_i(k_m + 1)$ for the first segment (in Fig. 2.2, segment i) can be computed for each time step in terms of $Dc(k_m + 1)$ by:

$$\widehat{\lambda}_i(k_m + 1) = \begin{cases} \bar{\lambda} - \frac{\Delta_\lambda D_c(k_m+1)}{L_i} & \text{for } D_c(k_m + 1) < L_i \\ \bar{\lambda} - \Delta_\lambda & \text{for } D_c(k_m + 1) > L_i \end{cases} \tag{2.2}$$

In order to simplify, equation (2.2) can be also written as:

$$\widehat{\lambda}_i(k_m + 1) = \max(\bar{\lambda} - \Delta_\lambda, \bar{\lambda} - \frac{\Delta_\lambda D_c(k_m + 1)}{L_i}) \tag{2.3}$$

This equation can be generalized for subsequent segments by:

$$\begin{aligned} &\widehat{\lambda}_i(k_m + 1) = \\ &= \min(\bar{\lambda}, \max(\bar{\lambda} - \Delta_\lambda, \bar{\lambda} - \frac{\Delta_\lambda(D_c(k_m + 1) - \sum_{n=I}^{n=i} L_n)}{L_i})) \end{aligned} \tag{2.4}$$

where I is the first segment affected by the reversible lanes.

The term $(\Delta_\lambda(D_c(k_m + 1) - \sum_{n=I}^{n=i} L_n)$ expresses the length of the corresponding lane that is already empty of cars. This term only applies when a part of segment is empty and the rest of the segment is still occupied by vehicles. If the segment is completely empty, the equivalent number of lanes is set to $\bar{\lambda} - \Delta_\lambda$. If the segment is completely occupied, the equivalent number of lanes is set to $\bar{\lambda}$.

The density equations (1.3) are modified as can be seen in (2.5). The first term is multiplied by $\frac{\widehat{\lambda}_i(k_m)}{\widehat{\lambda}_i(k_m+1)}$ in order to adapt the previous density to the current equivalent number of lanes. The second term takes into account the flows entering and leaving the segment with respect to the equivalent number of lanes in the previous time step.

$$\begin{aligned} &\rho_i(k_m + 1) = \frac{\widehat{\lambda}_i(k_m)\rho_i(k_m)}{\widehat{\lambda}_i(k_m + 1)} + \\ &+ \frac{T_m}{\widehat{\lambda}_i(k_m)L_i}(q_{i-1}(k_m) - q_i(k_m) + q_{\mathrm{r},i}(k_m) - \beta_i(k_m)q_{i-1}(k_m)) \end{aligned} \tag{2.5}$$

The speed equation (1.5) does not depend on the number of lanes except in the case of having an on-ramp or a lane drop. In this case, and assuming that the ramps are located at the beginning of a segment, it is necessary to instantaneously change the number of lanes (from $\bar{\lambda}$ to $\bar{\lambda} - \Delta_\lambda$) when $D_c(k_m)$ reaches the corresponding segment. This change only affects the on-ramp penalization term (1.7) and the

lane-drop term as can be seen in (2.6) and (2.7):

$$\nabla_{\mathrm{r}} v_i(k_m) = \tag{2.6}$$
$$= \begin{cases} -\frac{\delta_i T_m q_{r,i}(k_m) v_i(k_m)}{L_i \widehat{\lambda}(\rho_i(k_m)+K_i)} & \text{for } D_c(k_m) < \sum_{n=I}^{n=i}(L_n) \\ -\frac{\delta_I T_m q_{r,i}(k_m) v_i(k_m)}{L_i(\widehat{\lambda}-\Delta_\lambda)(\rho_i(k_m)+K_i)} & \text{for } D_c(k_m) > \sum_{n=I}^{n=i}(L_n) \end{cases}$$

$$\nabla_{\mathrm{d}} v_i(k_m) = \tag{2.7}$$
$$= \begin{cases} 0 & \text{for } D_c(k_m) < \sum_{n=I}^{n=i}(L_n) \\ -\frac{\phi_i T_m \Delta_\lambda \rho_i(k_m) v_i^2(k_m)}{L_i \widehat{\lambda}-\Delta_\lambda \rho_{c,i}} & \text{for } D_c(k_m) > \sum_{n=I}^{n=i}(L_n) \end{cases}$$

The traffic flow leaving the affected segments should be computed by equation (2.8). Equation (2.8) differs from the original equation (1.2) in the use of the equivalent number of lanes $\widehat{\lambda}_i(k_m)$ instead of a constant number of lanes.

$$q_i(k_m) = \widehat{\lambda}_i(k_m)\rho_i(k_m)v_i(k_m) \tag{2.8}$$

The model of the segment downstream the lane opening (in Fig. 2.3, segment $i+2$) is not affected in any way.

The complete model for the closing of the lanes is composed of equations (1.4), (1.5), (1.9), (1.11), (2.1), (2.4), (2.5), (2.6), (2.7) and (2.8).

VMS along the reversible lanes

In the case of having the variable message signs (VMS) distributed along the segments of reversible lanes, it has to be considered that the cars may tend to leave the reversible lanes before they reach their end. Therefore, it would be necessary to consider a new model which takes this effect into account.

However, in real applications it is not recommendable to immediately close the whole length of the reversible lanes for both safety and operational reasons. Instead, the VMS should change progressively in order to avoid unnecessary lane changes.

If the VMS are changed progressively, the traffic behavior of the closing with the VMS along the reversible lanes will be roughly the same as the one with the VMS located at the beginning. Therefore, the authors propose to use the model developed in the previous subsection for both cases.

2.1.2 Opening of the lanes

VMS at the beginning of the reversible lane

Consider again one direction of the stretch of freeway in Figure 2.1 but, in this case, initially Δ_λ lanes are closed and, at time step k_O, these lanes are opened.

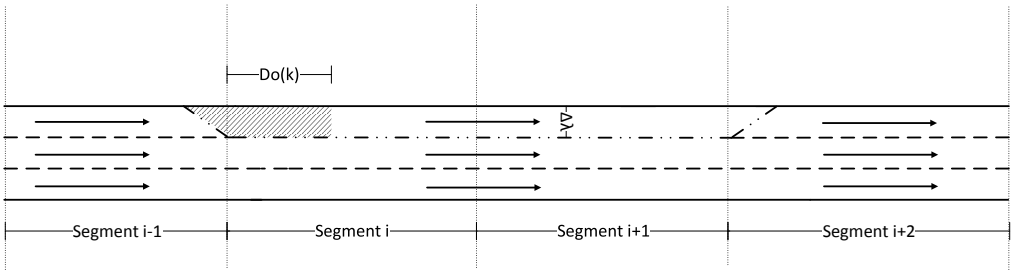

Figure 2.3: Freeway with one lane being opened with the VMS at the beginning of the reversible lanes.

Assuming that the VMS are located at the beginning of the lanes or that the VMS change progressively along the lanes, it has to be taken into account that the cars need a certain time to fill the new opened lanes as can be seen in Fig. 2.3.

In the figure, $D_o(k)$ is the length of the lane that is already occupied by cars and can be computed, as for the lane closing, according to (2.9):

$$Do(k_m + 1) = Do(k_m) + T_m(v_i(k_m)) \text{ with } D_o(k_O) = 0. \tag{2.9}$$

where $v_j(k_m)$ is the speed of the segment which has a reversible lane partially open at time step k_m.

For the segment upstream of the reversible lanes (in Fig, 2.3 segment $i-1$), the lane-drop term $\bigtriangledown_{\mathrm{d}} v_i(k_m)$ (1.8) is removed from the speed equation at sample time k_O so the traffic state equations (1.2)-(1.5) of this segment are the same as in the original model.

The model equations for the segments with the reversible lanes (In Fig. 2.3, segments i and $i+1$) are similar to the ones used for the closing (explained in the previous subsection). These equations are defined in terms of $D_o(k_m)$:

$$\widehat{\lambda}_i(k_m + 1) = \max(\bar{\lambda} - \Delta_\lambda, \min(\bar{\lambda}, \bar{\lambda} - \frac{\Delta_\lambda(D_o(k_m + 1) - \sum_{n=I}^{n=i} L_n)}{L_i})) \tag{2.10}$$

$$\rho_i(k_m + 1) = \frac{\widehat{\lambda}_i(k_m)\rho_i(k_m)}{\widehat{\lambda}_i(k_m + 1)} + \frac{T_m}{\widehat{\lambda}_i(k_m)L_i}(q_{i-1}(k_m) - q_i(k_m)) \tag{2.11}$$

$$q_i(k_m) = \widehat{\lambda}_i(k_m)\rho_i(k_m)v_i(k_m) \tag{2.12}$$

$$\bigtriangledown_{\mathrm{r}} v_i(k_m) = \tag{2.13}$$
$$= \begin{cases} -\frac{\delta_i T_m q_{r,i}(k_m) v_i(k_m)}{L_i(\bar{\lambda}-\Delta_\lambda)(\rho_i(k_m)+K_I)} & \text{for } D_o(k_m) < \sum_{n=I}^{n=i}(L_n) \\ -\frac{\delta_i T_m q_{r,i}(k_m) v_i(k_m)}{L_i\bar{\lambda}(\rho_i(k_m)+K_i)} & \text{for } D_o(k_m) > \sum_{n=I}^{n=i}(L_n) \end{cases}$$

Again, the model of the segment downstream the lane opening (in Fig. 2.3, segment $i+2$) is not affected in any way.

VMS along the reversible lanes

Consider now that the VMS are distributed along the Δ_λ reversible lanes which are opened at time step k_O (see Fig. 2.4). In this case, the cars in the non-reversible lanes are going to change to the reversible lanes as soon as they are able to (especially if the non-reversible lanes are congested before the opening of the lanes). Unlike for the closing, it is not unsafe to instantaneously open all the segments for the reversible lanes. Moreover, this will entail a better performance in terms of congestion.

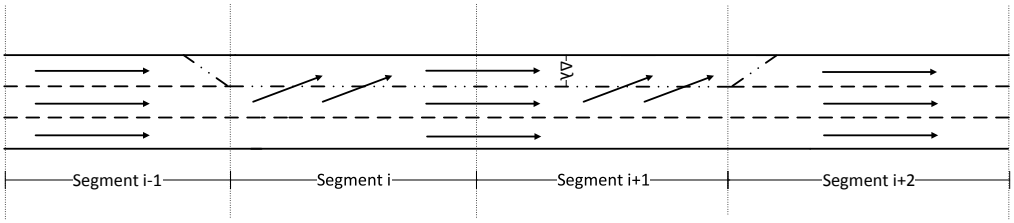

Figure 2.4: Freeway with one lane being opened with the VMS along the reversible lanes.

For the modeling of this case, It is assumed that the number of lanes instantaneously changes from $\bar{\lambda} - \Delta\lambda$ to $\bar{\lambda}$ at time step k_O for all the considered segments. Therefore the densities have to change instantaneously on time step k_O according to (2.14) in order to respect the conservation equation.

$$\rho_i(k_O) = \frac{\bar{\lambda}}{\bar{\lambda} - \Delta_\lambda}\rho_i(k_O) \;\forall\; i \text{ with lanes opened at } k_O \tag{2.14}$$

Equation (2.14) assumes that the cars in the non-reversible lanes will instantaneously occupy the reversible lanes, homogenizing the densities. The common equations of the METANET model (1.2)-(1.5) may be used for all the following time steps, including k_O.

It has to be pointed that the speed is not instantaneously affected except for the upstream segment where the lane-drop term is removed or in the case of having an on-ramp. The speed increase due to the higher number of lanes will appear in the following time steps due to the lower densities.

2.1.3 Simulation example

In order to analyze the performance of the previously proposed model, one direction of the stretch of freeway in Figure 2.1 with one reversible lane ($\Delta\lambda = 1$) has been simulated. Constant mainstream demand and typical and uniform values for model parameters have been used. Each segment has a length of 2 kilometers in order to obtain figures which a considerable portion of time with the lane closing or opening so the effects can be easily analyzed. Using shorter lengths, results are equivalent. The reversible lanes are assumed to be open at the beginning of the simulation. Subsequently, they are closed during a period of time and opened again. The parameter ϕ is chosen to be 0 in order to analyze the behavior of the proposed model in a simplified case without the influence of the speed decrease due to the lane drop.

In the first simulation, a low enough mainstream demand was used to avoid congestion (even with the reversible lane closed). The results can be seen in Figure 2.5. The results show that, as expected, the flow leaving the reversible lanes is almost unaffected by the lane closing if the traffic is totally uncongested. The density is increased during the time that the cars are leaving the reversible lane, because the same number of cars is forced to drive in 2 lanes instead of 3. The speeds are not directly affected by the lane closing but they are slightly modified in the following time steps due to the anticipation term and the change in density.

Equivalent results are obtained for the opening of the lane. The flow and speed almost do not change while the density is decreased in order to adapt to the equivalent number of lanes.

In the second simulation, a mainstream demand was used which does not create congestion with 3 lanes but, in this case, congestion appears when the lane is closed. The results can be seen in Figure 2.6. The results show that, as in the previous situation, the density quickly increases during the transient in which the cars are leaving the reversible lane. When the density starts to approach the critical density, the speed starts to decrease with the corresponding outflow decrease. When the reversible lane is empty, the increase in density due to the closing of the lane finishes. However, the system is already in an unstable congested point so the density keeps increasing (and the speed and flow decreasing) until the lane is opened again. Equivalent results are obtained for the lane opening.

In both simulations, it has been checked that the conservation equation is respected in all the time steps (i.e. there is no loss of vehicles).

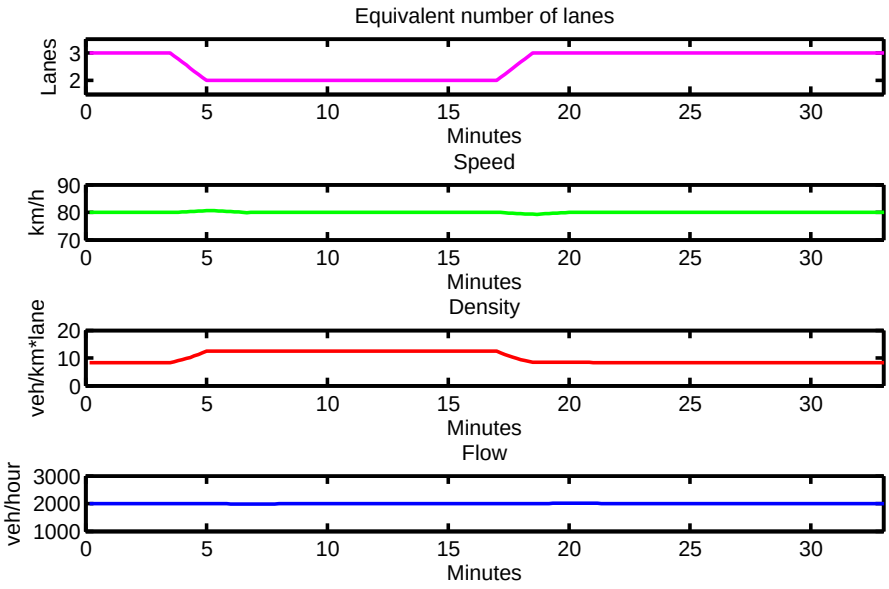

Figure 2.5: Freeflow example of closing and opening a reversible lane

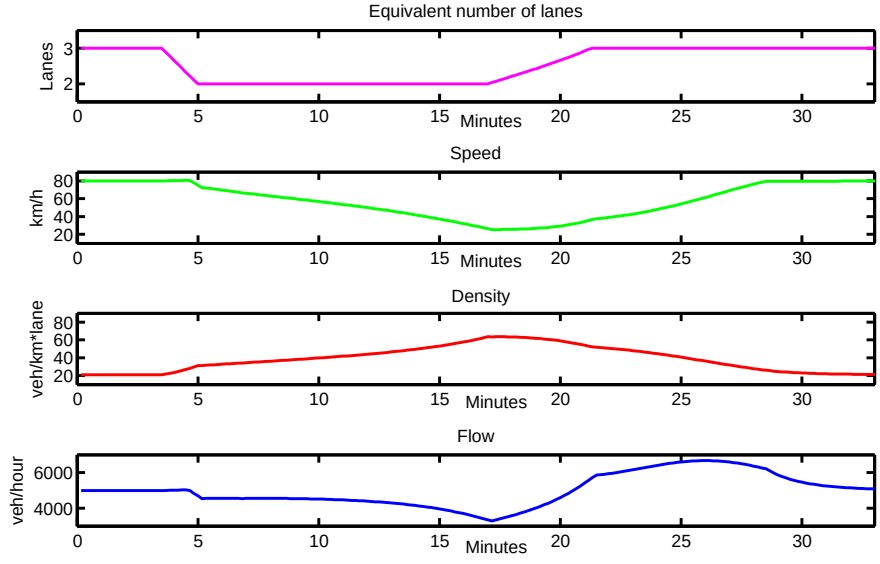

Figure 2.6: Congested example of closing and opening a reversible lane

2.1.4 Model Validation

Case study

Fig. 2.8 shows the freeway stretch that has been considered as a simulation-based testbed. The modeled freeway is a subsection of the ring road SE-30 around Sevilla, Spain. The S-N direction is modeled from marker post 7/5 to marker post 12/0 and the N-S direction from 13/1 to 10/0.

Figure 2.7: Traffic jam on Centenario Bridge, Seville, Spain

The modeled network includes the Centenario bridge [17], which is a bottleneck that creates, during the peek hours in the morning, the biggest recurrent congestion in the region of Seville, Spain [68] (see Figure 2.7).

The bridge has 2 lanes fixed in each direction and one reversible lane as can be seen in Figure 2.8. The reversible lane is currently changed manually by the traffic operators looking at the cameras along the bridge. This real-time manual control is deemed to perform better than fixed control, which would operate with pre-specified switching intervals during pre-specified (peak) periods of the day.

The rest of the modeled freeway has 3 lanes in each direction (except segment 2 in the S-N direction which has two lanes). The morning rush-hour congestion usually occurs between 8 and 9 am. The congestion is created by the bottleneck

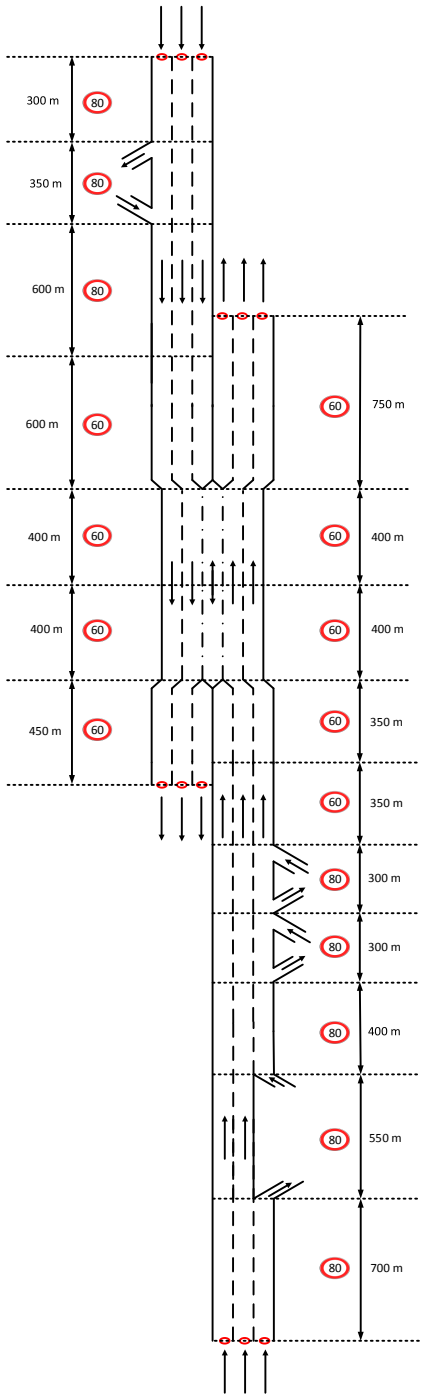

Figure 2.8: Sketch of the modeled network

on the bridge and propagates upstream for both directions. The detector measurements indicate that the traffic downstream the bridge is always uncongested in both directions. There is no other source of recurrent congestion in the modeled network.

This simulation uses loop detector data over the 6AM-11AM time range for ten different weekdays from four loop detectors located in the mainline at the beginning and the end of both directions (marked (red) cycles in Fig. 2.8). Each loop detector provides measurements of aggregated speeds and flows every 15 minutes. There is also data available of the state of the reversible lane indicating the time when the lane is closed or opened in one direction.

For stability, the segment length and the simulation time step should satisfy $L_i > v_{free,i}T$ for every link i. Therefore, the model sample time has to be smaller than the data sample time (15 minutes). Since this paper uses a model sample time T of 10 seconds, a zero-order interpolation was applied to the data.

There are four on-ramps and three off-ramps in the S-N direction and one on-ramp and one off-ramp in the N-S direction. The second and third on-ramps in the S-N direction are modeled as only one ramp in order to avoid stability problems due to their proximity.

The ramp flow data have been directly estimated by taking the difference between the aggregated flow data in the mainline and distributing between the ramps using a-priori knowledge about the ramp flows.

The process of validating the developed model consists of manually calibrating a number of parameters via repeated computer simulations similarly to the validation done in [53]. The results are compared to real data from loop detectors after each simulation, and a manual adjustment of a number of parameters is performed based on the observation of whether or not congestion is predicted accurately enough. A more detailed validation cannot be carried out due to the lack of ramp data, the aggregation of the detector data (15 minutes) and the absence of enough mainline detectors (especially in the segments with the reversible lane).

For a proper identification, the upstream end of the freeway stretch should be congestion-free; otherwise the entering flows are determined by the internal congestion. Due to a lack of measurements at the upstream end of the modeled part of the freeway (in both directions), one additional (virtual) measurement point was produced in the same way as in [65].

The simulations were carried out using MATLAB and an Intel Core i5 CPU. The average time needed is 1.25 seconds for the network simulation (from 6 AM to 11 AM).

Simulation results

Figures 2.9 and 2.10 show the speeds and flows measured upstream of the Centenario bridge for both directions and the corresponding values estimated by the proposed model.

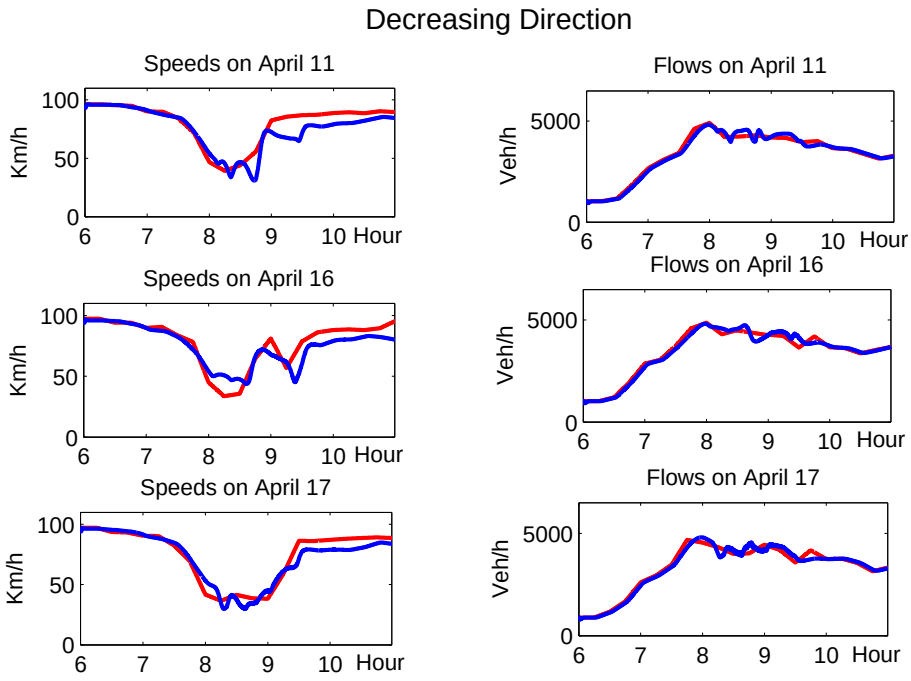

Figure 2.9: Comparison between simulated N-S speeds and flows (blue) and interpolation of 15 minutes aggregated data (red) for 11th, 16th and 17th of April 2012.

It can be seen that the model shows a relatively good speed and flow estimation for three typical days with relatively different congestion profiles.

Table 2.1 shows the mean relative errors for speeds and flows for the detectors upstream the Centenario bridge (marker post 7/5 for the S-N direction and marker post 13/1 for the N-S direction).

The days used for the identification of the model parameters are 11th, 16th and 17th of April 2012. The rest of the days (9th, 10th, 12th, 13th, 18th, 19th and 20th of April 2012) are used to validate the model identification (parameter values).

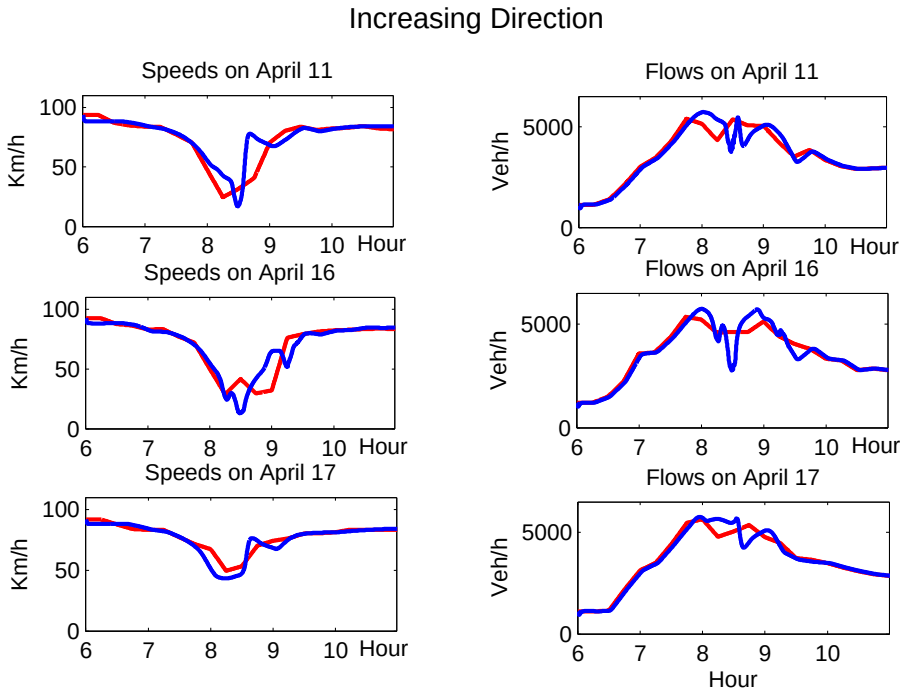

Figure 2.10: Comparison between simulated S-N speeds and flows (blue) and interpolation of 15 minutes aggregated data (red) for 11th, 16th and 17th of April 2012.

Table 2.1: Modeling Errors

Identification days		
	Speed Error (%)	Flow Error (%)
S-N direction	8.47 %	3.46 %
N-S direction	8.79 %	5.21%
Validation days		
	Speed Error (%)	Flow Error (%)
S-N direction	15.68 %	4.10 %
N-S direction	12.90 %	6.29%

2.2 An identification algorithm for the METANET model with a limited number of loop detectors

The estimation of the unknown parameters of the METANET model is not a trivial issue because the dynamic equations are highly nonlinear. The most common way to identify the parameters is the global minimization of the discrepancy between the model calculations and the real process [84, 82]:

$$J = \sum_{k=1}^{N} (x(k_m) - \hat{x}(k_m))^2 \tag{2.15}$$

where $x(k_m)$ is the output state vector of the model, $\hat{x}(k_m)$ is the measured process output and N is the total number of measures. In [53], this parameter estimation is done for a real large-scale motorway network around Amsterdam.

Unfortunately, this optimization usually falls in local minima, especially if a limited number of sensors are available. For example, if we were dealing with a network with 20 segments, 5 on-ramps and 5 off-ramps between two mainline loop detectors it would not be possible to compute so many variables (20*8=160) in a global optimization ensuring that the global minimum is reached. In fact, the research done for this section was motivated by the impossibility of properly identifying the METANET parameters for a specific real network using a global optimization. The network where these problems appeared is the network used for this chapter.

The easier solution for the identification problem could be to consider that the parameters have the same value for all the segments. With this supposition, it is possible to run a previous "evaluation" procedure in order to start the optimization with an initial point close to the global minimum. However, considering that all the segment has the same value of the model parameters causes a considerable loss of accuracy. In previous identifications of the METANET model [53, 82] heuristics are usually used in order to identify properly the model parameters.

An alternative parameter identification algorithm for METANET is proposed in [45]. In that paper the identification is also done by using a global optimization without a known initial point. Therefore, the approach is just well suited for the calibration task at hand.

The main contribution of this chapter is the design of an algorithm which can estimate the parameters of METANET approaching the global minimum of the discrepancy between the prediction of the model and the data trying to avoid suboptimal local minima. With that purpose, a three step identification algorithm is proposed.

In the first step, the fundamental diagram is computed for each segment with data available by a loop detector in the mainline. The fundamental diagram of

the segments without loop detectors is interpolated between the closer segments with data.

In the second step, a minimization of the discrepancy between the model and the data is done by taking as decision variables the parameters of the speed equation. The values of the fundamental diagram are considered as known. In this step, the algorithm uses a pre-evaluation in order to start the optimization as close as possible to the global minimum.

In the third step, a global optimization is run in order to improve the final performance of the system considering all the model parameters as decision variable.

The second contribution of this chapter is the proposal of a new mathematical definition of the Fundamental Diagram (FD) of traffic flow.

2.2.1 Optimization first step

Fundamental Diagram Choice:

The first step of the optimization is the estimation of the parameters of the Fundamental Diagram for each segment with a detector available in the mainline. [22] describes an equivalent procedure for the CTM model using a triangular form of the FD. As explained in 1.4, different forms of the fundamental diagram can be used. This work considers 3 possible choices for the fundamental diagram:
- FD 1. Typical Fundamental Diagram:

$$V(\rho_i(k_m)) = v_{\text{free},i} \exp\left(-\frac{1}{a_i}\left(\frac{\rho_i(k_m)}{\rho_{\text{crit},i}}\right)^{a_i}\right) \tag{2.16}$$

- FD 2. Typical Fundamental Diagram with a different value for a_i for densities higher than the critical density:

$$V(\rho_i(k_m)) = \tag{2.17}$$
$$\begin{cases} v_{\text{free},i} \exp\left(-\frac{1}{a_i}\left(\frac{\rho_i(k_m)}{\rho_{\text{crit},i}}\right)^{a_i}\right) & \text{for } \rho_i(k_m) \leq \rho_{\text{crit},i} \\ v_{\text{free},i} \frac{\exp\left(-\frac{1}{a_i}\right)}{\exp\left(-\frac{1}{b_i}\right)} \exp\left(-\frac{1}{b_i}\left(\frac{\rho_i(k_m)}{\rho_{\text{crit},i}}\right)^{b_i}\right) & \text{for } \rho_i(k_m) > \rho_{\text{crit},i} \end{cases}$$

where b_i is a model parameter equivalent to a_i but only defined for the congested part.
- FD 3. Linear Freeflow and Exponential Congestion:

$$V(\rho_i(k_m)) = \tag{2.18}$$
$$\begin{cases} v_{\text{free},i} & \text{for } \rho_i(k_m) \leq \rho_{\text{crit},i} \\ v_{\text{free},i} \frac{1}{\exp\left(-\frac{1}{a_i}\right)} \exp\left(-\frac{1}{a_i}\left(\frac{\rho_i(k_m)}{\rho_{\text{crit},i}}\right)^{a_i}\right) & \text{for } \rho_i(k_m) > \rho_{\text{crit},i} \end{cases}$$

Capacity Estimation:

Firstly, the capacity (i.e. the maximum flow that can go throw a point of the freeway) of the segments with detectors in the mainline are approximated. The algorithm takes the mean of the five measures with the higher flows among all the dates available for a detector in the segment.
Knowing the capacity (C_i), it is possible to define the critical density as combination of capacity, free-flow speed and variable "a".
- For Fundamental Diagrams 1 and 2:

$$\rho_{\text{crit},i} = C_i/(4v_{\text{free}_i} \exp(-1/(a_i)))$$

- For Fundamental Diagrams 3:

$$\rho_{\text{crit},i} = C_i/4v_{\text{free}_i}$$

Fundamental Diagram Estimation:

Once the capacity is obtained, a SQP optimization algorithm finds the value of the independent parameters of the fundamental diagram minimizing the quadratic error:

$$J_{part1} = \sum_{n=1}^{N} (Q(\hat{\rho}(n)) - \hat{q}(n))^2 \tag{2.19}$$

where $\hat{\rho}(n)$ and $\hat{q}(n)$ are the measured density and flow of the corresponding segment.

Fig. 2.11 shows that the typical fundamental diagram (FD 1) for METANET is not able to model properly the free-flow part accumulating an error of $J_{part1} = 4.68 * 10^8$. Because of that, the authors decided to use a fundamental diagram with different values for free-flow and congested parts (FD 2).

As can be seen on Fig. 2.11, the FD 2 (10) models better the free-flow part. The accumulated error is now $J_{part1} = 2.75 * 10^8$; almost the half than for the typical FD.

However, it can be seen that the free-flow part in this diagram is almost straight with $a_i = 196$. Therefore, it is interesting to eliminate the parameter a_i simplifying the computation. The accumulated error for the FD 3 (11) is almost the same $J_{part1} = 2.79 * 10^8$ than for FD2.

Defining the fundamental diagram in piece-wise form does not increase substantially the computational power for traffic control because, when variable speed limits and ramp metering are considered, the model is inherently piece-wise.

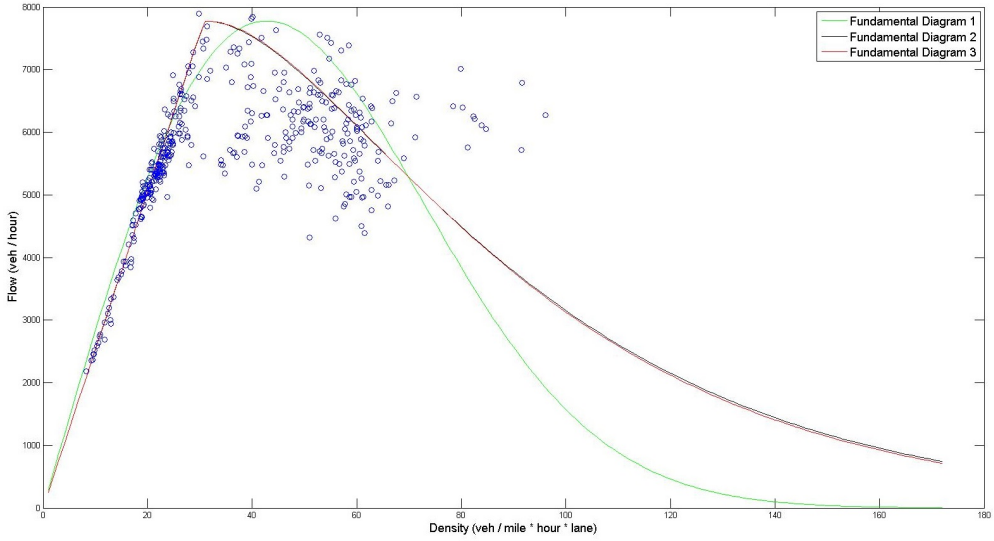

Figure 2.11: Fundamental Diagrams

2.2.2 Optimization second step

Interpolation of the Fundamental Diagram:

After the estimation of the fundamental diagram for the segments with a sensor, it is necessary to set the fundamental diagram of the rest of the network. The values of the FD parameters ($V_{\text{free},i}$, a_i and $\rho_{\text{crit},i}$) of the segments without a detector are computed interpolating between segments with a loop detector according to the following equations:

$$v_{\text{free},i} = \frac{L_d}{L_d + L_u} v_{\text{free},u} + \frac{L_u}{L_d + L_u} v_{\text{free},d} \tag{2.20}$$

$$a_i = \frac{L_d}{L_d + L_u} a_u + \frac{L_u}{L_d + L_u} a_d \tag{2.21}$$

$$\rho_{\text{crit},i} = \frac{L_d}{L_d + L_u} \rho_{\text{crit},u} + \frac{L_u}{L_d + L_u} \rho_{\text{crit},d}. \tag{2.22}$$

Where L_d and L_u are the distances to the closer downstream and upstream detectors and $V_{\text{free},d}$, a_d, $\rho_{\text{crit},d}$, $V_{\text{free},u}$, a_u, $\rho_{\text{crit},u}$ are the values of the fundamental diagram parameters (obtained in the first step) of the closer downstream and upstream detectors.

Pre-evaluation of the speed equation parameters:

With an initial value of the fundamental diagrams defined, the next step is to obtain an estimation of the parameters of the speed equation: τ, K, μ_h, μ_l and δ. In this case, these variable are considered equal for all the segment in the network. Papageorgiou et al. [53] demonstrated that the model is most sensitive with respect to the values of the parameters used in the fundamental diagram equation. This justify to consider equals the speed equation parameters for the full network. These values will be set in order to minimize the following error function:

$$\begin{aligned} J_{part2} = &\sum (\rho_7(k_m) - \rho_h(k_m))^2 + 0.1(\rho_1(k_m) - \rho_m(k_m))^2 + \\ &+0.1(\rho_{10}(k_m) - \rho_{sa}(k_m))^2 + 0.1(v_7(k_m) - v_h(k_m))^2 \\ &+0.1(v_1(k_m) - v_m(k_m))^2 + 0.1(v_{10}(k_m) - v_{sa}(k_m))^2 \end{aligned} \tag{2.23}$$

where $\rho_h(k_m)$, $\rho_m(k_m)$ and $\rho_{sa}(k_m)$ are the measured densities on sensor Huntington, Mountain and Santa Anita respectively. $v_h(k_m)$, $v_m(k_m)$ and $v_sa(k_m)$ are the measured speed forn these detectors. The weights are higher for the segment corresponding to the "missing" density of the detector Huntington.

Because we are using only five decision variables, it is possible to run a "evaluation" before the optimization so this optimization starts as close as possible to the global minimum. Therefore, J_{part2} is evaluated for a "net" of values for the speed equation parameters. The point with lower error cost function is chosen as initial point for the next substep.

Optimization of the speed equation parameters:

Taking the fundamental diagrams defined in the first step and the initial point obtained in the pre-evaluation, an SQP optimization algorithm (minimizing J_{part2}) can be run. The optimization considers the fundamental diagram parameters as known.

2.2.3 Optimization third step

Step 1 and 2 give a estimation of the parameters that is, in theory, close to the global minimum. This set of variables can be used as initial point for a global optimization that considers as decision variables all the parameters of the model. The total number of variables for the network in 2.13 is 14: 5 speed variables + 3*3 FD variables. Step 3 runs a SPQ optimization algorithm in order to minimize $J_{part3} = J_{part2}$ in a global way.

2.2.4 Case study

The used case study is a subsection of the I-210 West freeway located in Pasadena (Los Angeles County, California, United States) (see Figure 2.2.4). The freeway stretch has four mainline lanes and a length of 1.75 kilometers.

Figure 2.12: Traffic jam on I-210, Pasadena, CA, USA.

Fig. 2.13 shows the freeway stretch that has been simulated. The image comes from the tool NetworkEditor of the TOPL Project [75].

Fig. 2.14 shows the freeway section used to identify and test the traffic model schematically. The freeway is partitioned using NetworEditor into 10 segments with lengths $L = [0.118, 0.211, 0.1490, 0.236, 0.124, 0.217, 0.230, 0.099, 0.217, 0.143]$ km.

As can be seen in the figures, the section has three mainline loop detector stations labeled (Mountain ML 34.90, Huntington ML 33.05, Santa Anita ML 32.20), and additional detector stations on each ramp. ML stands for mainline, and the numbers (e.g. 34.05) are the absolute post-mile indexes of the detector stations. The section has three on-ramps and three off-ramps located at the junctions with the Santa Anita, Huntington, Myrtle and Mountain avenues.

This simulation uses loop detector data over the 5AM:12PM time range for ten different days. The morning rush-hour congestion usually occurs at this time.

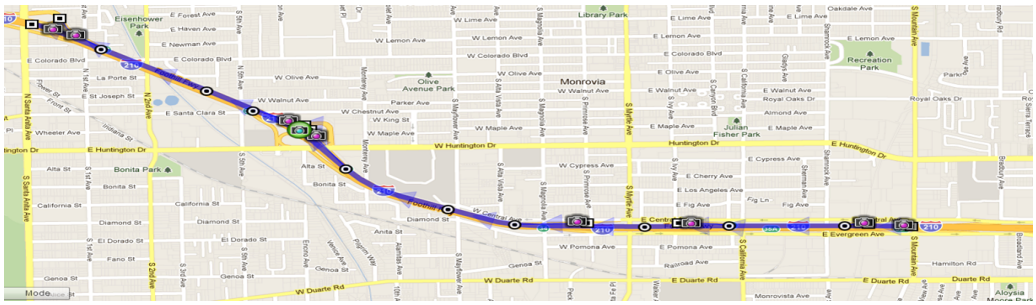

Figure 2.13: Traffic Network Simulated

The data was obtained from the Performance Measurement System (PeMS) [13] using the tool Data ClearingHouse.

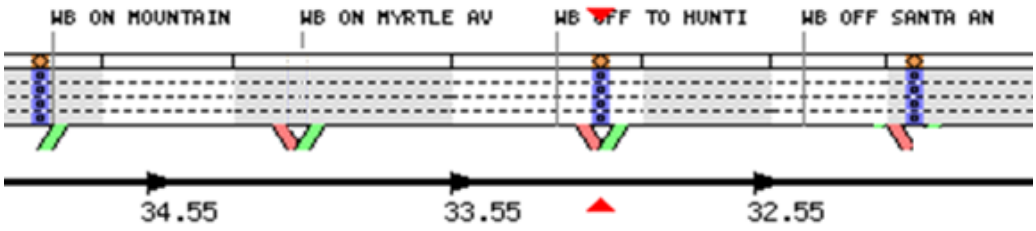

Figure 2.14: Traffic Network Sketch from PeMS

Each loop detector provides measurements of volume (veh/time step) and percent occupancy every 5 minutes. Densities are computed according to the PeMS algorithm [49]. The choice of the network and the data set has been made based on [70], where Munoz et al. estimated the parameters of CTM in a part of this network.

The time range used is also similar but not the same. For stability, the segment length and the simulation time step should satisfy every link i: $L_i > v_{free,i}T$. Therefore, the model sample time has to be smaller than data sample time (5 minutes). As this section uses a model sample time T of 5 seconds, a zero-order interpolation was applied to the PeMS data.

The METANET simulation of the network has been programmed in MATLAB using the "fmincon" function of the Optimization Toolbox.

The ramp flows and split ratios ($q_{ramp,i}(k_m)$ and $\beta_i(k_m)$) are considered as measurable (or estimable) disturbances of the system and are taken from real data.

2.2.5 Numerical results

Optimization First Step

The results given in Table 2.2 were obtained at the middle sensor ("Huntington") when running the algorithm proposed in IV.A for the analyzed network. It can be seen how the accumulated error using the typical fundamental diagram of METANET almost doubles the order two options.

Table 2.2: Step 1 Results

	V_{free}	ρ_{crit}	a	b	error
FD 1	72.07	42.77	2.17	–	$4.68 * 10^8$
FD 2	63.12	31.11	196	0.91	$2.75 * 10^8$
FD 3	63.04	30.83	0.89	–	$2.79 * 10^8$

Optimization Second Step

As explained in IV.B, a pre-evaluation of the cost function for a "net of values" was done before the optimization. The values obtained (i.e. the ones corresponding to the lowest cost function) for the models parameters evaluated are the following:

$$x_{Ev} = [\tau, \mu_l, \mu_h, \delta, K] = [40, 30, 45, 1, 45] \tag{2.24}$$

With an associated cost function of $J_{part2} = 5.8687e + 004$.

Taking x_{Ev} as initial point, the SQP optimization algorithm converges to:

$$x_{part2} = [38.9041, 30.9310, 46.3964, 1, 45.6593] \tag{2.25}$$

With an associated cost function of $J_{part2} = 4.7263e + 004$.

Optimization Third Step

Tables 2.3 and 2.2.5 show the final values obtained after the final step of the optimization. In the table, $Vfree_j, a_j$ and $\rho_{crit,j}$ with $j = m, h, sa$ are the free flow speed, the parameter "a" and the critical density associated to the segment with the loop detectors Mountain, Huntington and Santa Anita, respectively.

Table 2.3: Speed equation parameters

τ	μ_l	μ_h	δ	K
42.11 s	9.60 km^2/h,	38.66 km^2/h	4.19	34.27 veh/km·lane

Table 2.4: Fundamental Diagram parameters

a_m	$V_{free,m}$	$\rho_{crit,m}$
1.56	93.02 km/h	21.70 veh/km·lane.
a_h	$V_{free,h}$	$\rho_{crit,h}$
0.90	101.82 km/h	19.42 veh/km·lane.
a_{sa}	$V_{free,sa}$	$\rho_{crit,sa}$
1.11	93.50km/h	21.22 veh/km·lane

Modelling Results

Table 2.7 shows the mean relative error for the density e^{ρ_h} (2.26) of each day simulated and the mean error (2.26) for all the days.

$$e^{\rho_h} = \sum_{n=1}^{N} \frac{||\rho_7(n) - \rho_h(n)||}{\rho_h(n)} \tag{2.26}$$

where N is the total number of measurements. In this section, $N = 84$ because there measurements are available every 5 minutes during 7 hours ($N_K = 7h/5min = 84$).

Table 2.5: Mean density error of each day simulated

Day	Mean density error (%)
22th January	9.66
29th January	7.49
6th February	11.04
12th February	14.10
26th February	8.2
2th April	8.85
9th April	12.62
20th April	9.07
17th May	16.04
4th June	12.08
Mean	10.95

The mean error of the speed predictions is 11.35% for the 10 days. The identification days were the 22th and 29th of January, the 26th of February and the 2th and 20th of April. The rest of the days are used for the simulation but not for the identification in order to validate the data.

Fig. 2.15 shows the densities and velocities measured in the sensor ML 33.05 and the values estimated by the model for one day.

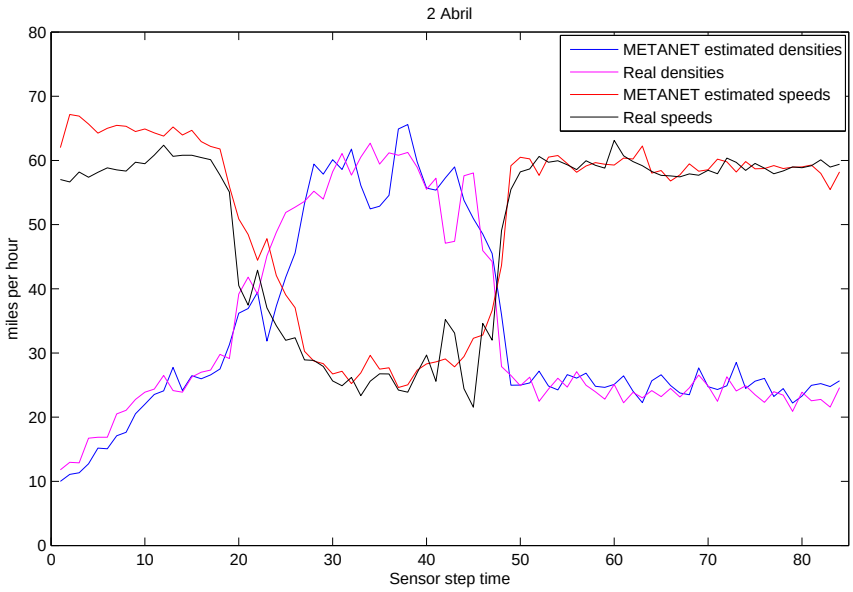

Figure 2.15: PeMS data for the sensor ML 33.05 and model prediction for the corresponding segment.

In the figure, it can be seen that the identified model shows a good estimation in both density and speed validating the identification algorithm.

2.3 A comparison between a first order (CTM) and a second order (METANET) macroscopic model for freeway traffic control

This section compares theoretically and numerically the two most used macroscopic traffic models: Link-Node Cell Transmission Model (LN-CTM) and METANET. Although the theoretical literature on these models is vast, there are not many model validations using real traffic data [87]. Moreover, the different model validations have been made under different conditions and on different networks, making impossible to adequately compare the accuracy of the models. Moreover, there are no papers comparing both models using the same set of validation data.

Firstly, a features comparison is done concluding that the main advantage of METANET is that it is potentially more realistic because this model addresses the mean speed dynamics and includes a greater number of parameters. This idea is supported by two simulations where METANET models the system more consistently to reality in two specific situations. The main advantages of the LN-CTM model are lower computational requirements, easier parameter identification and the possibility of designing linear or piece-wise linear MPC controllers based on its structure.

Subsequently, a numerical comparison was made over a section of the I-210 West in Southern California, using several days of loop detector data collected during the morning rush-hour period. The results show a better density estimation for METANET than for LN-CTM (10.95% versus 20.94%). On the other hand, the simulation time needed is 1.4 s for METANET and 0.5 s for LN-CTM.

2.3.1 Link-Node Cell Transmission Model

The Cell Transmission Model proposed by Daganzo [15] is an analytically simple macroscopic model able to reproduce congestion wave propagation dynamics. It gives relatively accurate predictions of macroscopic traffic flows (in consistency with kinematic wave theory) under all traffic condition. It has been shown that CTM is the first order discrete Godunov approximation of the kinematic wave equation [55].

This section uses the Link-Node Cell Transmission Model (LN-CTM) [72]. This model is an extension of the standard CTM. LN-CTM usually uses occupancies as state variables. In this work, densities are used instead of occupancies in order to include uneven segment lengths, increasing flexibility in partitioning the highway as in [70].

After using the common equations for macroscopic models (1.2), (1.3) and (1.4) explained in section 2, it is necessary to define the relationship between flows and densities in order to complete the model.

This can be done by the following equation which gives the flow that exits in each segment:

$$q_i(k_m) = \min(\lambda_i \rho_i(k) V_{c,i}(k_m), C_i) \cdot \frac{\min(R_i(k_m), S_{i+1}(k_m))}{R_i(k_m)} \tag{2.27}$$

where $V_{c,i}(k_m)$ is the VSL signal, C_i is the capacity of segment i, $R_i(k_m)$ is the demand function and $S_i(k_m)$ is the supply function.

The demand function $R_i(k_m)$ defines the number of vehicles available for moving out of segment i and $S_{i+1}(k_m)$ defines the maximum number of vehicles that can move into segment $i+1$:

$$R_i(k_m) = \min(\lambda_i \rho_i(k_m) V_{c,i}(k_m), C_i) \cdot (1 - \beta_i(k_m)) + d_i(k_m) \tag{2.28}$$

$$S_i(k_m) = \min(W_i \cdot (\rho_{J,i} - \rho_i(k_m)), C_i) \tag{2.29}$$

where W_i is the congestion wave speed, $\rho_{J,i}$ is the jam density and $d_i(k_m)$ is the effective demand for on-ramp i:

$$d_i(k_m) = \min(r_i(k_m) \cdot C_{ramp,i}, \frac{w_i(k_m)}{T_m}) \tag{2.30}$$

where $C_{ramp,i}$ is the ramp capacity and $r_i(k_m)$ is the ramp metering rate.

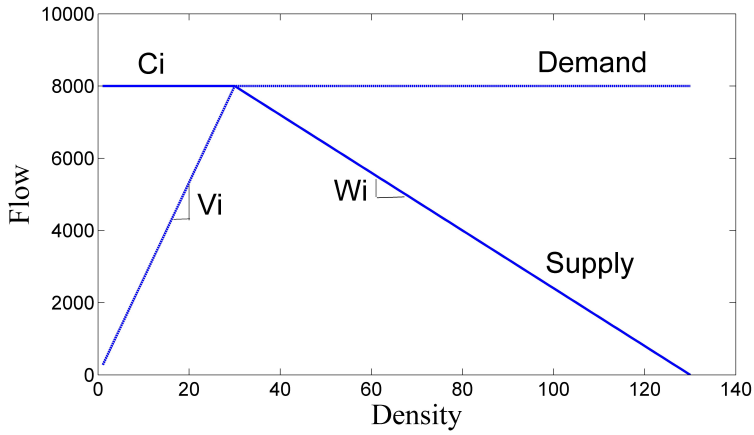

Figure 2.16: Triangular Fundamental Diagram of CTM

The definition of the demand and supply function defines a triangular form of the Fundamental Diagram (FD) as can be seen in Fig. 2.16. If necessary, the speed of a segment can be estimated by using equation (1.2).

Finally, the flow that enters from an on-ramp has to be defined:

$$q_{ramp,i}(k_m) = d_i(k_m) \cdot \frac{min(R_i(k_m), S_{i+1}(k_m))}{R_i(k_m)} \tag{2.31}$$

Finally, in order to complete the model, it is necessary to know the upstream speed of the first segment, the downstream density of the final segment and the flow entering the mainline. In this section, these variables are taken from real data.

2.3.2 Features comparison

Computational power

The computational time needed for model simulation is higher in METANET than in LN-CTM because there are exponential and bi-linear equations and a higher number of state variables. However, both models are fast in their simulation.

More precisely, the average time needed for the simulation of the example proposed in section 2.3.3 (7 hours and 10 segments) is 1.4 seconds for METANET and 0.5 seconds for LN-CTM using MATLAB and an Intel Core i3 CPU.

Estimation

For CTM, identification of the model parameters [70, 71, 69] can be made easily and quickly by minimizing the error in the FD (2.32) for each segment l with available data.

$$e^{\mathrm{FD}} = \sum_{n=1}^{N} \frac{||q_l(\widehat{\rho}_l(n)) - \widehat{q}_l(n)||}{\widehat{q}_l(n)} \tag{2.32}$$

where N is the total number of measurements, $\widehat{q}_l(n)$, and $\widehat{\rho}_l(n)$ are the measured density and flow. Subsequently, it is only necessary to interpolate the results of each FD identified for the intermediate segments.

In the validation done in this section in 2.3.3, a last step has been added refining the CTM parameters by minimizing (2.33) using the solution of (2.32) as starting point. This step has been added in order to improve the final performance of the system and to make the comparison with METANET as fair as possible.

$$e^{\mathrm{model}} = \sum_{n=1}^{N} \frac{||x_{CTM}(n) - \widehat{x_{CTM}}(n)||}{\widehat{x}_{CTM}(n)} \tag{2.33}$$

where $\widehat{x}_{CTM}(n)$ is a vector with all the measured densities and queues and $x_{CTM}(n)$ is the corresponding CTM prediction.

Estimation of the unknown parameters of the METANET model is not a trivial issue because the dynamic equations are highly non-linear. The most common way to identify the parameters is the global minimization of the discrepancy between the model calculations and the real process (2.34) [84] [83]:

$$e^{\text{model}} = \sum_{n=1}^{N} \frac{||x(n) - \widehat{x}(n)||}{\widehat{x}(n)} \tag{2.34}$$

where $\widehat{x}(n)$ is a vector with all the measured variables (speeds, densities and queues) and $x(n)$ is its METANET prediction.

Unfortunately, this optimization usually falls in local minima, especially if a limited number of sensors are available. Therefore, the three steps identification algorithm (which tries to avoid undesirables local minima) proposed in the previous section 2.2 will be used for the validation done in this section in 2.3.3.

Controller design

The macroscopic models allow the use of advanced control techniques which substantially improves the performance of the controlled traffic system in simulations.

The definition of a system with linear dynamics makes the design of an MPC controller much easier. Both LN-CTM and METANET are non-linear models but LN-CTM can be converted into a piece-wise linear model [69, 98] or approximated by a linear model in order to design a controller. For example, in [36], the ACTM (a linear piece-wise version of the CTM model) is defined and used for the definition of a MPC. In [72], a MPC using LN-CTM is proposed by taking the optimal flows as decision variables. Subsequently, the VSL and ramp metering are computed. With optimal flows, the VSL and ramp metering are computed subsequently.

Therefore, the main advantage of the LN-CTM model is the possibility of designing a MPC controller based on a linear or partially linear model because this allows the optimal control input (for the LN-CTM model) to be computed in a reasonable time.

On the other hand, the optimization problems based on second order models such as METANET are non-linear, computationally intensive and the solutions obtained are sometimes only locally optimal. Therefore, the centralized MPC control using METANET [38, 11, 33] does not allow the consideration of large-scale applications. Some papers [28] [29] deal with these problems using distributed and multi-start algorithms. Some recent papers propose relaxed linear versions of the METANET model in order to design linear MPC.

Traffic controllers based on MPC may be difficult to implement in practical cases. In order to propose an easily implementable controller for VSL and ramp metering

some papers [88] [42] [10] use controllers without online model usage and demand predictions. In these cases, the model is only used for simulations, diminishing the importance of the nonlinearities.

Analyzed incongruities

In some cases, METANET and LN-CTM show a conceptually different behavior which is interesting to analyze. In this subsection, two incongruities, which have not been analyzed before, are studied in the network proposed in Fig. 2.17.

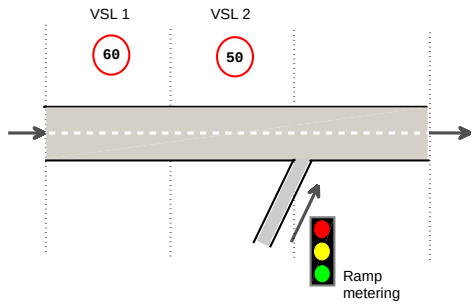

Figure 2.17: Network used for the analyzed incongruities

For the simulation, the downstream density $\rho_4(k_m)$ is assumed to be constant with a value of 20.5 $veh/(lane \cdot km)$ and the upstream density is assumed to be higher than the critical density $\rho_0(k_m)$ >19 $veh/(lane \cdot km)$. In other words, the upstream demand function $R_0(k_m)$ is assumed to be higher than the first segment supply $S_1(k_m)$.

The fundamental diagram is considered to be triangular and equal for all the segments. The FD is defined by the critical density $\rho_{crit} = 18.63\ veh/lane{\cdot}km$, the capacity flow $C_i = 9000\ veh/hour$ and the jam density $\rho_J = 74.53\ veh/(lane{\cdot}km)$, as can be seen in Fig. 2.18. The capacity of the on-ramp is $C_{ramp} = 2000\ veh/hour$ and the sample time is $T = 10\ seconds$.

In steady state, $q_0(k_m) = q_1(k_m) = q_2(k_m)$ and $q_3(k_m) = q_2(k_m) + q_{ramp}(k_m)$. Thus, using equations (2.27), (2.28) and (2.29) and considering $S_3(k_m) < R_2(k_m)$, it can easily be seen that $\rho_1(k_m) = \rho_2(k_m) = S_2(k_m)$ and:

$$\begin{aligned} &S_4(k_m) = \\ &= C_{ramp}\frac{\min(R_2(k_m), S_3(k_m))}{R_2(k_m)} + C_2\frac{min(R_2(k_m), S_3(k_m))}{R_2(k_m)} \end{aligned} \tag{2.35}$$

Therefore, there are infinite possible steady states according to equation:

$$\rho_2(k_m) = \rho_1(k_m) = 13.54 + 0.5081\rho_3(k_m) \tag{2.36}$$

For this simulation, we assume the steady state $\rho_1(k_m) = \rho_2(k_m) = 37.5\ veh/(lane \cdot km)$, $\rho_3(k_m) = 29.16\ veh/(lane \cdot km)$ with the corresponding speed $v_1(k_m) = v_2(k_m) = 40\ kms/hour$, $v_3(k_m) = 63.25\ kms/hour$. These states correspond to points P1=P2 and P3 in Fig. 2.18.

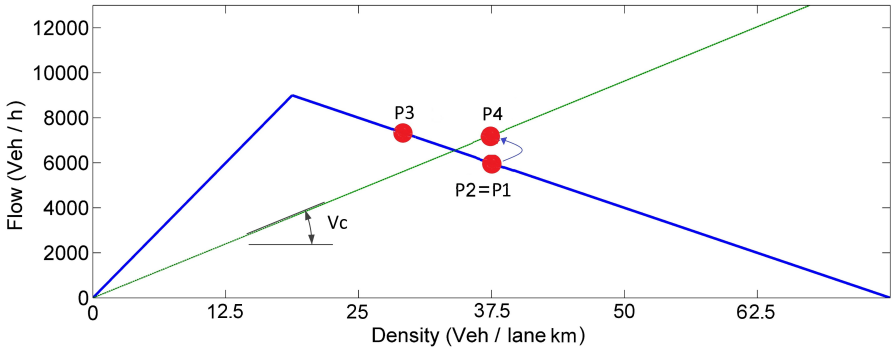

Figure 2.18: CTM Fundamental Diagram for this subsection

VSL incongruity Lets suppose that the speed limits are set as $V_c = 96.6\ kms/hour$, the speed of the first two segments being 40 $kms/hour$, as previously stated. In time step 200, the VSL signals of segments 1 and 2 decrease suddenly from 100 to 48 $kms/hour$.

Should the response of drivers change in the following time steps? Common sense says that if a VSL signal is higher than the current speed of the corresponding segment, the response of the system should not change.

– LN-CTM response: Fig. 2.19 shows the response of the system when the speed limit is changed in sample time 200. It can be seen how the state of the system changes even with a higher value in the VSL signal that the current speeds for the corresponding segments.

In the figure, it can be seen how the speed of the first two segments is decreased (and the density increased) by more than a 5 % due to the change in the VSL signal. In order to explain this effect, taking equation (2.31) of LN-CTM model into account, it can be seen that the flow that enters from an on-ramp is defined by (assuming that $\frac{w(k_m)}{T_m} > C_{ramp}$, $r(k_m) = 1$ and $S_3(k_m) < R_2(k_m)$):

$$q_2(k_m) = \tag{2.37}$$
$$= \frac{\min\left(4 \cdot \rho_2(k_m) \cdot V_{c,2}(k_m), 9000\right)}{\min\left(4 \cdot \rho_2(k_m) \cdot V_{c,2}(k_m), 9000\right) + 2000} S_3(k_m)$$

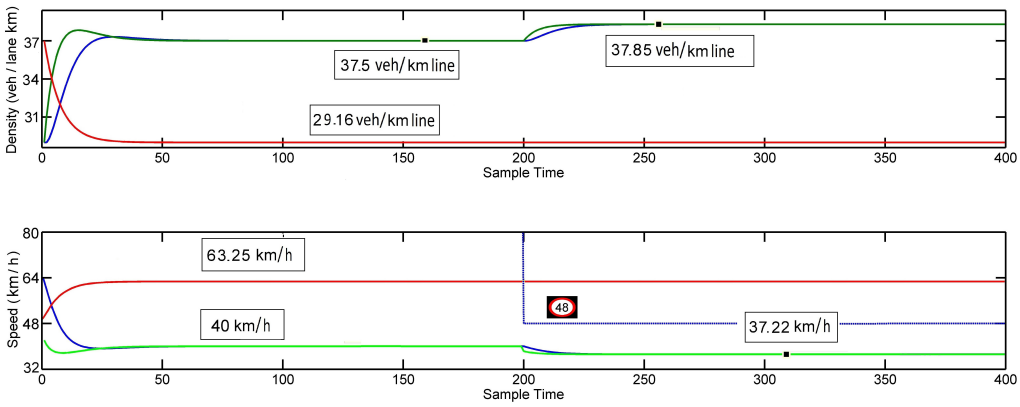

Figure 2.19: LN-CTM response for "VSL incongruity" case. Segments 1, 2 and 3 are plotted in Blue, Green and Red, respectively.

Taking into account that before the change of the VSL in time step 200, $\rho_2(k_m) = 37.5\ veh/(lane \cdot km)$ and $S_3(k_m) = 7333\ veh/hour$ (point P1 in Fig. 2.18):

$$q_2(200) = \frac{\min\left(150 \cdot V_{c,2}(200), 9000\right)}{\min\left(150 \cdot V_{c,2}(200), 9000\right) + 2000} 7333 \tag{2.38}$$

It can be seen that the response of the system changes if $150 \cdot V_{c,2}(k_m) < 9000$. Therefore, when the variable speed limit decreases from 100 to 48 kms/hour (point P4 in Fig. 2.18) the flow q_2 changes from 6000 to 5740 veh/hour and the speed changes v_2 from 40 to 37.5 kms/hour in just one sample time.

However, q_1 and v_1 does not change immediately, being also the new speed limit $V_{c,1}(k_m) = 48$ kms/hour. Therefore, it can be concluded that this incoherent speed reduction only appears when there is an on-ramp in the following segment.

It could be argued that this effect is due to the fact that the current speed is not a real speed (just the average speed of all the drivers in a segment) and, therefore, some drivers would be affected by the decrease in the speed limit. However, this effect persists no matter how small the size of the segment or the sample time are.

– METANET response: Using the version of METANET proposed in [37], the response of the system does not change when the variable speed limit is higher than the desired speed, as can be seen in equation (1.9).

However, using the METANET version proposed in [87] (with the VSL are included in the model changing the parameters of the FD), the change in the VSL signal (even over the current speed) also changes the response of the system.

Incongruity due to queue size For the computation of initial steady state by (2.36), we supposed that the ramp queue is always long enough ($\frac{w(k_m)}{T_m} > C_{ramp}$) to provide all the allowed cars by freeway conditions. With this supposition, the number of vehicles that enter the freeway from the on-ramp is 1333 $veh/hour$, which corresponds to 3.7 $veh/sample$.

Equation (2.30) shows that if $\frac{w(k_m)}{T_m} > C_{ramp}$, the ramp flow stays constant no matter how long the queue is. This corresponds to queues longer than 5.55 vehicles.

However, what happens if $T_m \cdot C_{ramp} > w(k_m) > T_m \cdot q_{ramp}(k_m)$? Lets suppose that the queue is reduced to 4 vehicles and kept constant at this value. Should the response of the system change during the following sample times? Common sense says that if the freeway allows 3.7 vehicles to enter during the next sample time, 3.7 vehicles will actually enter if we have more than 3.7 vehicles waiting. Having 4, 5 or 200 vehicles waiting behind them outside the freeway should not affect the number of vehicles that actually enter.

– LN-CTM response: Fig. 2.20 shows the response of the system with the queue change in sample time 200. It can be seen how the speed of the first two segments increases (and the density decreases) by almost 10%.

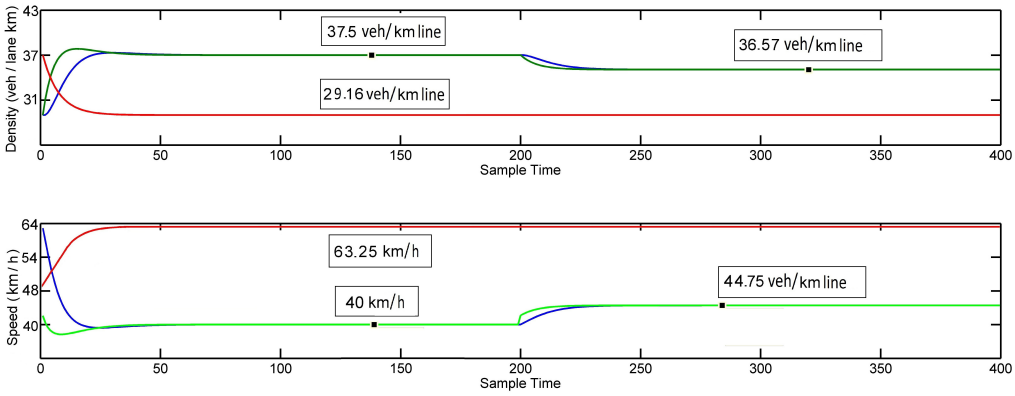

Figure 2.20: LN-CTM response for "Incongruity due to queue size" case. Segments 1, 2 and 3 are plotted in Blue, Green and Red, respectively.

In order to explain this effect, applying equation (2.31) to the ramp in Fig. 2.18 and supposing that $V_{c,2}(k_m) > V_{free}$, $\frac{w(k_m)}{T_m} < C_{ramp}$ and $S_3(k_m) < R_2(k_m)$, it can be seen that:

$$q_{ramp}(k_m) = \frac{w(k_m)}{T_m} \frac{S_3(k_m)}{9000 + w(k_m)/T} \tag{2.39}$$

Taking into account that before the queue changes in time step 200, $S_3(k_m) = 7333$

$veh/hour$ (P1 in Fig. 2.18) and $T_m = 10\ s$ (if $w(200) < 5.55\ vehicles$):

$$q_{ramp}(200) = \frac{7333 \cdot w(200)}{25 + w(200)} \tag{2.40}$$

Therefore, when $w(k_m)$ is reduced to 4 vehicles, q_{ramp} will change from 3.7 to 3.3 $veh/sample$. It is important to note that other versions of the CTM model, which use a different merging model [97], would not experiment this behavior.

– METANET response: The response of the system is not affected by the queue when the number of vehicles in it is higher than the number of cars, allowed to enter by freeway conditions allow to enter, as can be seen on equation (1.11).

Others aspects

– One advantage of second order models like METANET for control purposes is the possibility of including an emission model in order to minimize some environmental criteria such as "total emissions" or "maximum emission dispersion level". [109]. LN-CTM is based on fundamental diagram and do not have a speed update equation. Therefore, CTM describes the speed evolution with less accuracy than second-order models and they will also produce much less accurate estimations of the accelerations. When making abstraction of the fact that we are using space-mean values, accelerations can be considered as derivatives of speeds. Hence, less accurate estimations of speeds will translate into significantly less accurate estimations of accelerations.

– In [16], C. Daganzo sharply criticizes second-order macroscopic traffic flow models explaining some theoretical inconsistencies. These problems could lead to negative speeds or other undesirable effects. Some papers [3, 81, 1] refuse or support these criticisms. This section does not go into these topics because there is great literature regarding the field.

– During the METANET simulations with certain values of the model parameters and ramp flows, the state of the system tends to oscillate, even in steady state. It is necessary to have a proper identification procedure in order to avoid this undesirable effect. Nevertheless, these oscillations may lead to an unreal simulation when an on-line identification is being carried out. LN-CTM is a more stable model in these aspects.

– As explained in [36], second order models have a distinct advantage over first order models in that they can reproduce the capacity drop which is the observed difference between the freeway capacity and the queue discharge rate. Since first order models do not capture this phenomenon, they are incapable of exploiting the benefits of increasing bottleneck flow. They can only reduce travel time by increasing off-ramp flow.

– Many other second-order models (like the phase transition model or PTM [8]) have been proposed to try to model the speed dynamics in a simpler way, reducing the computation power needed. Others papers [106] have proposed to use flow and speed as state variables because flow estimation is easy and reasonably accurate while density estimation is laborious.

2.3.3 Case study

Network analyzed

The scenario used in the previous section (Fig. 2.13 in section 2.2) has been used for the numerical comparison of the models. The upstream and downstream mainline data and the ramp flow data are assumed to be known. $\widehat{\rho_h}$ is the density measurement of the detector loop at the Huntington station. The seventh segment of the model corresponds to this detector and, therefore, ρ_7 is a estimation of $\widehat{\rho_h}$. The density and speed of this sensor is assumed to be missing, hence the need for estimation.

The LN-CTM simulation was carried out using the Aurora Road Network Modeler (RNM) [54] software of the TOPL Project. The results were exported to MATLAB for the comparison and, subsequently, the parameters were refined by minimizing (2.33) as explained previously.

The FD parameters used for the simulation of the LN-CTM model can be seen in Table 2.6. In the table, $W_j, \rho_{J,j}$ and C_j with $j = m, h, sa$ are the congestion wave speed, the jam density and the capacity associated to the segments with a loop detector: Mountain, Huntington and Santa Anita. The LN-CTM parameter identification was made by minimizing the error in the FD for each segment with data available by a loop detector on the mainline (i.e. the Huntington, Myrtle and Santa Anita Stations). The parameters used for the simulation of the METANET model are the ones obtained in the previous section (See Table 2.3).

Table 2.6: LN-CTM parameters

W_m	$\rho_{J,m}$	C_m
3.63 km·lane/h	108.39 veh/km·lane	8125 veh/h
W_h	$\rho_{J,h}$	C_h
2.62 km·lane/h	140.32 veh/km·lane	7905 veh/h
W_{sa}	$\rho_{J,sa}$	C_{sa}
3.26 km·lane/h	118.82 veh/km·lane	8237 veh/h

Results

Fig. 2.21 shows the densities measured in sensor ML 33.05 and the corresponding values estimated by both models for 29th January.

In the figure, it is possible to see that both METANET and LN-CTM models show a relatively good density estimation for a typical day. Equivalently, for the majority of days simulated, the results show that:

- LN-CTM and METANET accurately model the uncongested traffic (i.e. with a density lower than the critical density).

- LN-CTM and METANET roughly model congested traffic (i.e. with a density greater than the critical density). In these cases, METANET shows a lower model error than LN-CTM.

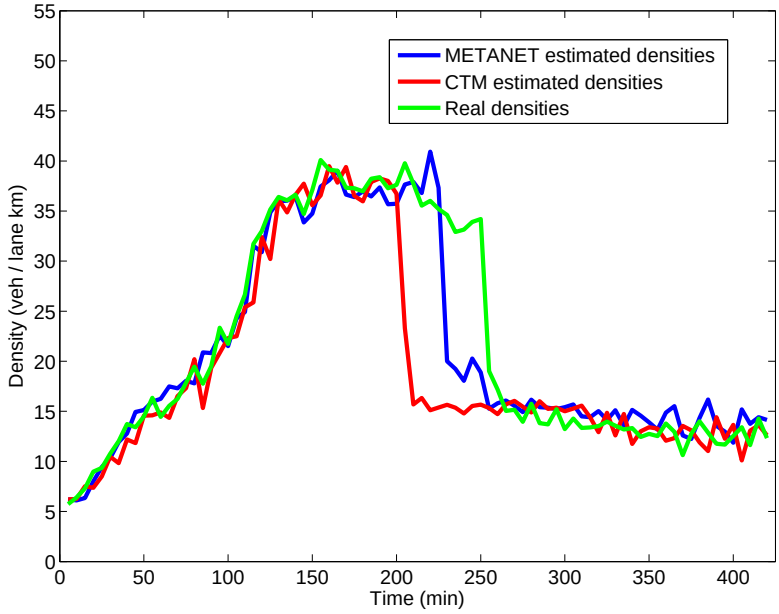

Figure 2.21: 29nd January: PeMS data for sensor ML 33.05 (Green). LN-CTM prediction (Red) and METANET prediction (Blue) for the corresponding segment.

However, on certain days the congested part of the simulation is lost in the LN-CTM prediction(Fig. 2.22). For the day shown (26th February), LN-CTM says that the system does not get to be congested, forecasting the response of the system in an equivalent but unreal and uncongested way (In Fig. 2.22 the LN-CTM uncongested state and the METANET congested state correspond to equivalent flows, as can be seen in Fig. 2.23). If a small disturbance in the prediction of the model changes the prediction of the density from a congested point to an uncongested point, the future prediction of the system can be modeling the flow correctly but in a uncongested equivalent point.

This effect can appear in both models, as can be seen in Fig. 2.21: At the end of the congested behavior, the METANET prediction gets uncongested a few minutes before than the real data. However, the convection term, the anticipation term and the higher number of parameters make the METANET model more robust against these undesirable effects. An online feedback of the model would also cause this effect to disappear.

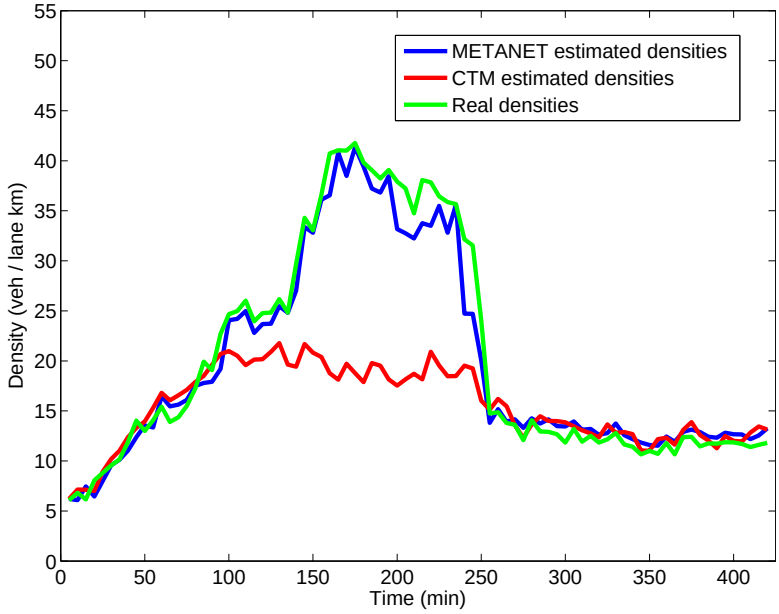

Figure 2.22: 26th February: PeMS data and predictions for sensor ML 33.05.

Table 2.7 shows the mean relative error for the density (2.26) of each day simulated and the mean error for all the days. The identification days were the 22nd and 29th of January, the 26th of February and the 2nd and 20th of April.

Analyzing the numerical results, it can be seen that the METANET model has a smaller prediction error on any day and almost half the prediction mean error (10.95% versus 20.94%). Moreover, it can be seen that the three days when the congested part of the simulation is totally lost in the LN-CTM prediction (as explained previously) show the greatest prediction error. These days are the 12th and 26nd of February and de 20th of April.

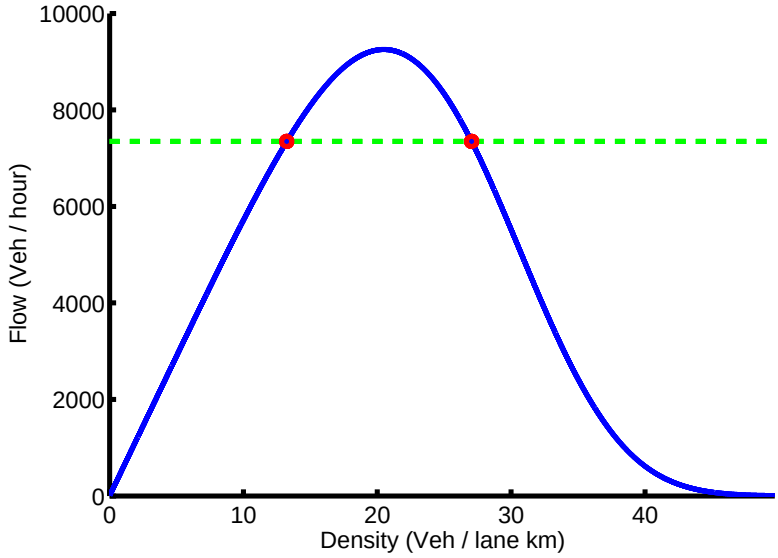

Figure 2.23: Fundamental Diagram with flow equivalent points

Table 2.7: Prediction Errors (%)

Day	METANET	LN-CTM
22nd January	9.66	14.04
29th January	7.49	14.71
6th February	11.04	21.35
12nd February	14.10	32.03
26th February	8.2	26.68
2th April	8.85	16.01
9th April	12.62	12.67
20th April	9.07	29.51
17th May	16.04	23.09
4th June	12.08	19.30
Mean	10.95	20.94

2.4 Conclusions

This chapter has proposed new advances in freeway traffic modeling in order to allow the design and implementation of optimal control strategies.

In Section 2.1, a macroscopic model for reversible lanes on freeways based on METANET has been proposed. The reversible lanes are modeled like variable lane drops (taking into account that the cars in the closed/opened lanes need a certain time to leave/enter the corresponding segments) using the concept of an equivalent number of lanes.

The proposed extension of model METANET has been validated with real data over the Centenario Bridge of the SE-30 freeway in Seville, Spain. The results show that the proposed model is able to reproduce traffic congestion due to the reversible lanes with a mean error on the identification days of 8.63% and 4.33% for speed and flows, respectively. The errors are 14.29% and 5.19% for the days used for validation.

Subsequently, in Section 2.2, an identification algorithm for the macroscopic traffic model METANET that tries to avoid undesirables local minima has been proposed. The algorithm was tested with real data from the I-210 freeway in Pasadena, California. The results show a good estimation of the traffic densities and speeds validating the identification algorithm with a mean error of 10.95% and 11.35%, respectively. The identification algorithm proposed may be especially useful for traffic control using Model Predictive Control, where the accuracy of the model and the possibility of executing the identification in real time are key issues.

Moreover, a new form of the fundamental diagram is proposed. This definition allows to improve the match between the fundamental diagram and the data without an increase in the computational power needed for the simulation.

Finally, in Section 2.3, some general features of the two most used macroscopic models for traffic control, LN-CTM and METANET, have been compared, with the following conclusions:

- Identification of the model parameters of METANET is not a trivial issue. However, it is easier and faster for LN-CTM.

- The computational time needed for the simulation of LN-CTM is smaller, but of the same order of magnitude, than for METANET.

- The design of a MPC controller is easier for LN-CTM. For METANET, computationally intensive optimizations are needed.

- In some cases, METANET simulation can show undesirable oscillations.

- LN-CTM does not produce a realistic simulation when a VSL is set over the current speed and there is an on-ramp in the following segment or when an on-ramp queue is between $T_m \cdot C_r$ and $T_m \cdot q_{ramp}$.

- LN-CTM model, because it does not capture the capacity drop phenomenon, is incapable of exploiting the benefits of increasing bottleneck flow.

- METANET model can describe the speed dynamic increasing the potential improvement of a VSL control and allowing the inclusions of the emissions in the prediction of the model.

From the numerical comparison done over a section of the I-210 West in Southern California using real date, the following conclusions can be drawn:

- METANET simulation shows a lower error than LN-CTM for all the days simulated with almost a half mean error (10.84% versus 20.94%).

- Both models have an accurate prediction for the uncongested part and rough prediction for the congested part.

- In open-loop simulations, small prediction errors can lead to a completely wrong simulation when the system passes from uncongested to congested state, especially for LN-CTM model.

In general, it could be said that a METANET based MPC controller for VSL and ramp metering is expected to have a better closed loop performance. However, the problems that are necessary to face up can compensate, in some cases, the higher potential. The final election of the model for freeway traffic control is open to the researches and to the specific situation and requirements of each case.

Chapter 3

Freeway Traffic Control by using Model Predictive Control

Dynamic traffic control methods continuously measure the state of the traffic network and respond accordingly. Since traffic systems are highly non-linear and time-variant systems, model-based predictive traffic control approaches such as Model Predictive Control (MPC) are promising candidates.

MPC is a model-based control approach that is based on the optimization of control inputs that improve a given cost function over some prediction horizon. The MPC approach can be used for non-linear and time-variant systems including constraints on the inputs, states, and outputs of the system.

The main reasons to use MPC for traffic control are that MPC coordinates the control inputs, it has a feedback structure, it optimizes the desired performance over a prediction horizon and it can handle constraints.

Since the core control strategy used in this thesis is MPC, this chapter provides a brief overview of the basic concepts of MPC for traffic systems in Sections 3.1. The MPC controller is demonstrated in a simple simulation-based case study in subsection 3.1.4. For detailed discussions on MPC for freeway traffic systems, we refer the reader to [37].

Furthermore, Section 3.2 analyzes the robustness of MPC controllers for freeway traffic with respect to variations in the mainstream demand and proposes new methods for online demand estimation.

Parts of this chapter are published in [29, 27, 32].

3.1 Model Predictive Control for freeway traffic

3.1.1 Previous works on MPC for traffic systems

Numerous reports and studies have shown that the steady growth in traffic volume has resulted in societal and environmental critical problems associated with long delays, waste of fuel, higher accident risk, pollution, driver frustration, etc.

It has been reported in the literature under different conditions that dynamic traffic control is a good solution to decrease congestion [53, 38, 10, 41] using ramp metering, Variable Speed Limits (VSL) and other traffic control measures. In general, dynamic traffic control takes the state of the traffic into account and computes control signals that change the response of the traffic system improving its behavior.

NLMPC have been successfully tested in simulations in traffic systems. In [**?**], two simulations using ramp metering with ALINEA or MPC control algorithm are compared obtaining a decrease of 1.3% in the ALINEA case and 6.9% in the MPC case.

In [63], VSL are previously determined without an optimization of a macroscopic model (taking account of factors as maximizing a bottleneck flow, the limits on queues lengths). Subsequently, the ramps metering are computed using MPC.

In [105], Ramp metering rates are computed previously with a given strategy (for example, ALINEA). Following, VSL are calculated using MPC with a simplified METANET model. Using this algorithm, a reduction of the 31,8% in the TTS is obtained in a simulation for a real network.

In [37], it is demonstrate that the use of speed limits in a MPC control framework for traffic systems with ramps metering and VSL can substantially improve the network performance. The improvement in the TTS of the network simulated is a 14.3% being just a 5.3% if only ramps metering are used.

In [91], MPC is used for the control of a urban network.

Even being a generally true conclusion that a decentralized (i.e. local) traffic controller is suboptimal with respect to a centralized one, many recent theoretical papers (e.g. [35], [51]) and almost all the implemented algorithms (e.g. [76], [99]) use local techniques.

It is important to note that the reduction of the TTS strongly depends on the traffic conditions. In order to properly compare two algorithms, they must be simulated in the same network and conditions.

3.1.2 Problem statement

This section explains the general formulation of the freeway traffic MPC controllers used in this thesis. The MPC optimization problem for freeway traffic flow can be formulated according to the general formulation of a non-linear MPC controller shown in equation 1.18:

$$\min_{u_{\mathrm{t}}(k)} J(u_{\mathrm{t}}(k), x_{\mathrm{t}}(k), d_{\mathrm{t}}(k)) \tag{3.1}$$

subject to:

$$x(k+\ell+1) = f(x(k+\ell), u(k+\ell), d(k+\ell)), \tag{3.2}$$

$$x(k) = x_k$$

$$x(k+\ell+1) \in \mathbb{X}, \tag{3.3}$$

$$u^{\mathrm{c}}(k+\ell) \in \mathbb{U}, \tag{3.4}$$

$$u^{\mathrm{d}}(k+\ell) \in \mathbb{S},$$

$$h(u_{\mathrm{t}}(k), x_{\mathrm{t}}(k), d_{\mathrm{t}}(k)) \in \mathbb{D},$$

$$\text{for } \ell = 0, 1, ..., N_{\mathrm{p}} - 1,$$

In the case of freeway control traffic, we can relate the general variables of a MPC formulation with traffic variables:

- The cost function $J(u_{\mathrm{t}}(k), x_{\mathrm{t}}(k), d_{\mathrm{t}}(k))$ models the Total Time Spend (TTS) by the drivers. The cost function also include some penalization terms as can be seen on (6.3) and (3.5).

- The input vector $u(k)$ includes the traffic measures available in each case. These variables can be divided in continuous variables $u^{\mathrm{c}}(k)$ and discrete variables $u^{\mathrm{d}}(k)$.

  In this thesis, ramp metering will be used and treated as a continuous input variable in Chapters 3, 4 and 5.

  Variable speed limits (VSL) will be used in Chapters 3, 4, 5. The first two chapters treat the VSL as a continuous input as in previous references. However, Chapter 5 treats the VSL as a discrete input variables with $\mathbb{S} = \{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$ the set of possible control values.

  Chapter 6 only uses the reversible lane as input variable. This variable is treated as a discrete variable with $\mathbb{S} = \{-1, 0, 1\}$

- The state vector $x(k)$ includes the speed, density and queue of each segment/ramp in the traffic network.

- The non-controllable input vector $d(k)$ includes demand profiles, upstream speeds and downstream densities.
- The continuous constraints set $h(u_t(k), x_t(k), d_t(k)) \in \mathbb{D}$ includes maximum and minimum values for the density and speed of each segment, for the queue in some on-ramps and/or in the mainline and for the continuous input variables (the ramp metering rates and, when they are considered continuous, the VSL).

Once that the optimization problem is defined, it has to be applied the receding horizon strategy (represented in Figure 3.1) to the case of freeway traffic systems:

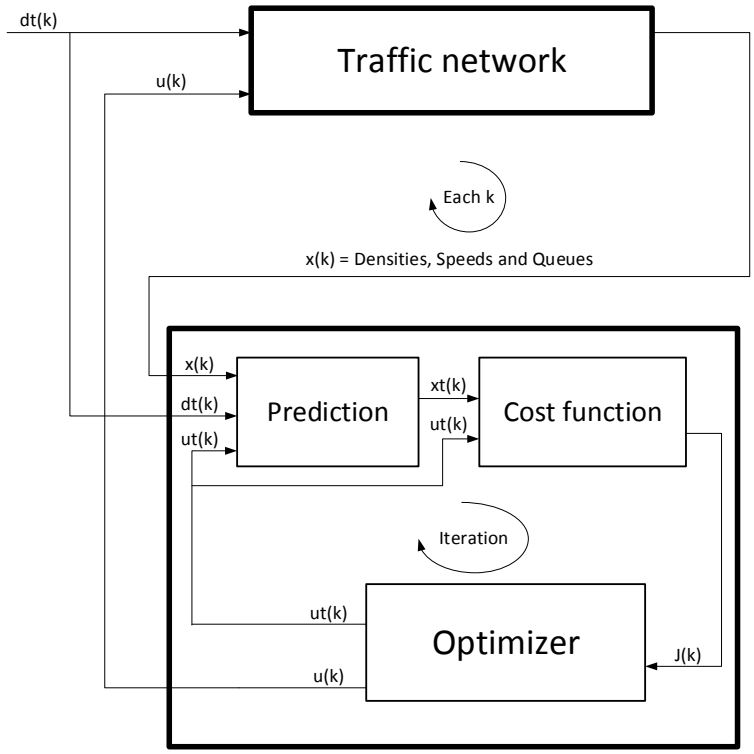

Figure 3.1: Receding horizon strategy. The set of future control signals are computed considering the predicted outputs during the prediction horizon, but only the first control $u(k)$ is applied

1. The future outputs for a determined prediction horizon N_p, called the prediction horizon, are predicted at each instant k using the model METANET. These predicted outputs depend on the known values up to instant k (densities, speeds, queues, last implemented VSL, future mainstream and ramp

demands...) and on the future control signals (VSL, ramp metering rates and reversible lanes), which are those to be sent to the system and calculated.

2. The set of future control signals is calculated by optimizing, mainly, the total time spend by the drivers including the control effort in the objective function. The future control law is considered be constant from a determined control horizon N_u.

3. The control signal is sent to the process whilst the next control signals calculated are rejected, because at the next controller sampling instant (2 minutes) step 1 is repeated with this new measured values and all the sequences are brought up to date.

3.1.3 Particularities of the proposed MPC controllers

This section explains the main particularities of the freeway traffic MPC controllers used in this thesis. All the aspects which are not explained here have the general structure of a NLMPC (see [9] and section 1.5 for further details).

Model

The model is the cornerstone of MPC; a complete design should include the necessary mechanisms for obtaining the best possible model, which should be complete enough to fully capture the process dynamics and allow the predictions to be calculated, and at the same time to be intuitive.

In this thesis the macroscopic traffic flow model METANET (introduced in Sections 1.3 and 1.4) will be used for the design and simulation of all the MPC controllers proposed. This choice has been justified in Section 2.3.

It has to be pointed out that the majority of the proposed controllers can be equivalently applied to other macroscopic traffic models as the Cell Transmission Model (CTM) (introduced in Section 2.3.1).

Non-linear optimization

The continuous optimization in this thesis is calculated using Sequential Quadratic Programming (SQP) optimization techniques. SQP is a powerful method for solving constrained optimization problems with a smooth objective function based on a continuous nonlinear model. The main drawback of SQP algorithms is that it is not guaranteed that it reaches the global optimum.

In [37], SQP algorithms are used to solve the non-linear optimization of freeway traffic with ramp metering and variable speed limits. In [53], a nonlinear optimal control approach it is used to solve similar traffic problems. Both control approaches do not guaranty that the global optimum is reached.

In this thesis, SQP algorithms are solved by using the Matlab function $fmincon$ of the Optimization Toolbox of Matlab with a maximum number of 20000 cost function evaluations.

The right choice of this set of initial values is a key step in the design of a continuous MPC for traffic due to the high non-convexities of the optimization problem. Using the shifted version of that part of the control signal that was not applied to the process or the minimum feasible metering rates and VSL (as proposed in [37]) is not enough for large network and it would cause high loss of performance for the majority of the simulations done in this thesis.

On the other hand, using a multi-start optimization increases the complexity making, in many cases, impossible to compute the control inputs in real time if the optimizations cannot be solved in parallel.

For small networks (like the one simulated in Section 3.4), it is enough to use as initial point the minimum values of the control inputs within the constraints. However, in order to find a right initial point for larger networks, a pre-evaluation algorithm may be run.

For example, in Chapter 4 and 5, in order to try to avoid that the algorithm ends up in a local minimum, the algorithm runs an evaluation procedure before the optimization. During the procedure, the TTS is evaluated for a set of control values (in this case, 252 points). The unfeasible candidates with a small cost function associated are used for the definition of a new feasible candidate increasing the set of control values. The best control profile obtained is taken as initial values for the optimizations.

Morever, SQP algorithms can be combined with a multi-start approach to solve the MPC problem. In Section 3.2, the algorithm runs five parallel optimizations with different initial points and the best control profile obtained is selected as the solution for the controller.

In Chapters 5 and 6, discrete optimizations have to be solved. Since we are dealing with non-convex integer optimization, the only way to obtain the global optimum is to evaluate the cost function for all the feasible points in the reduced search tree. The main problem is the computation time needed for the evaluation of such a large number of possible combinations of discrete VSL. Therefore, this solution is just applicable for relatively small networks and horizons or in cases where an offline solution is useful. In order to be able to solve the problem for large networks within the limited computation time available, a genetic algorithm is proposed (see Section 5.1.3 for further details).

Increasing horizons

The computation of the control signals has to be done during a controller sample time (in this case, 120 s). If it is possible to compute them in a shorter time, could be interesting to increase the horizons and compute the control signals again (Fig.3.2). This technique, called increasing horizons, is very useful for MPC based traffic controllers.

The use of increasing horizons may be quite useful in real implementation of freeway traffic control systems based on an MPC controller. However, in this thesis they are not used in order to have a fair comparison between the different algorithms with a constant value for the prediction and control horizons.

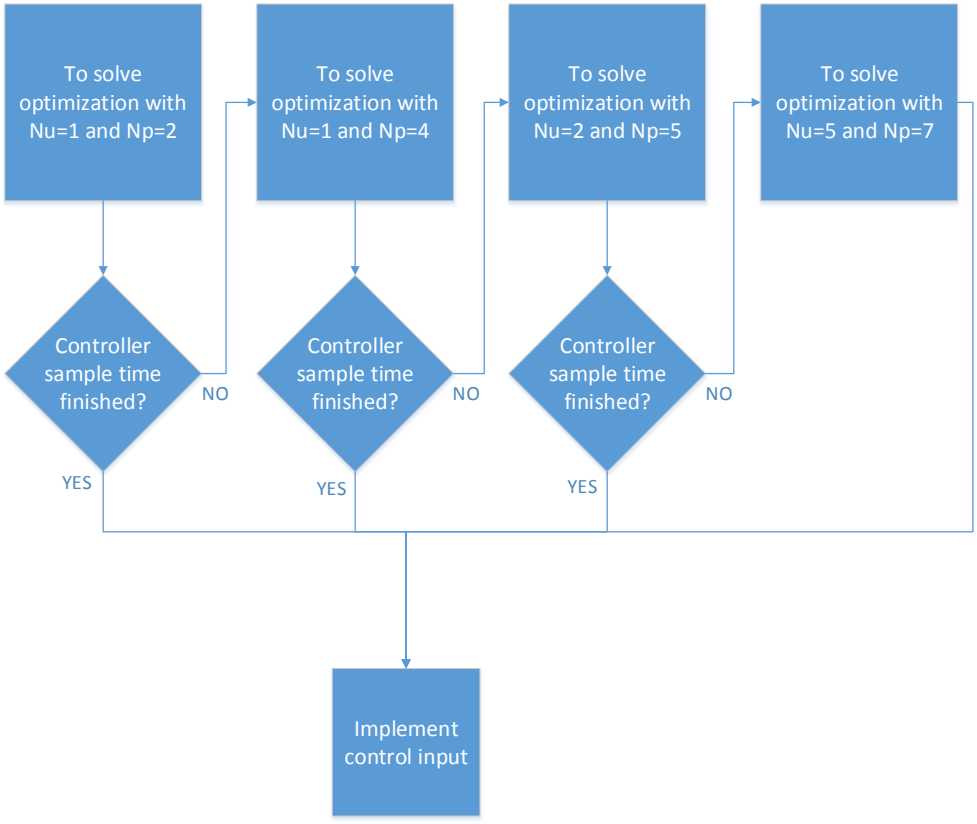

Figure 3.2: Increasing horizons strategy. After each optimization, the algorithm checks if there is more time available before the next controller sample time. In this case, the horizons are increased.

Constraints

The controlled system is subjected to constraints for the maximum and minimum values of density, speed, queue, ramp metering rate and control speed of the VSL.

If state constraints (densities, speeds and queues) are considered explicitly, the optimization cannot be computed in a reasonable time if all the variables are considered with its respective constraints. Therefore, the authors decided to consider explicitly only the control inputs (i.e. ramp metering rates, VSL, reversible lanes...).

The constraints in speed, density and queue are made soft including penalization terms in the cost function [91]:

$$\bar{J}(k) = J(k) + \sum_{\ell=1}^{N_{\mathrm{p}}} \sum_{i=1}^{N_{\mathrm{const}}} \Omega_i(k+\ell) \tag{3.5}$$

where $\Omega_i(k+\ell)$ is a penalization term that is different to zero if the corresponding soft constraint is violated and N_{const} is the total number of soft constraints:

$$\begin{aligned}
&\sum_{\ell=1}^{N_{\mathrm{p}}} \sum_{i=1}^{N_{\mathrm{const}}} \Omega_i(k+\ell) = \\
&= \sum_{l=1}^{N_p} [\sum_{(i)\epsilon I} (\delta_{vel_{min},i} \Delta_{vel} \parallel v_i(k+l) - v_{min} \parallel^2 + \\
&+ \delta_{vel_{max},i}(k+l)\Delta_{vel} \parallel v_{max} - v_i(k+l) \parallel^2 + \\
&+ \delta_{vel_{min},i}(k+l)\Delta_{den} \parallel \rho_i(k+l) - \rho_{min} \parallel^2 + \\
&+ \delta_{vel_{max},i}(k+l)\Delta_{den} \parallel \rho_{max} - \rho_i(k+l) \parallel^2) + \\
&+ \sum_{o\epsilon O} (\delta_{q_{min},i}(k+l)\Delta_q \parallel w_i(k+l) - w_{min} \parallel^2 + \\
&+ \delta_{q_{max},i}(k+l)\Delta_q \parallel w_{max} - w_i(k+l) \parallel^2)]
\end{aligned} \tag{3.6}$$

Where $\Delta_{vel}, \Delta_{den}$ and Δ_q are tuning parameters and the values for δ are defined as:

$$\delta_{vel_{max},i} = \begin{cases} 1 & \text{if } v_i(k+l) > v_{max} \\ 0 & \text{if } v_i(k+l) \leq v_{max} \end{cases} \tag{3.7}$$

$$\delta_{vel_{min},i} = \begin{cases} 1 & \text{if } v_i(k+l) \leq v_{min} \\ 0 & \text{if } v_i(k+l) > v_{min} \end{cases} \tag{3.8}$$

The rest of the paremeters $(\Delta_{vel}, \Delta_{den}, \Delta_q)$ are assigned equivalently.

Cost function

The MPC controllers used in this thesis consider the following cost function (6.3) containing one term for the Total Time Spent (TTS) by all the drivers during the prediction horizon and two terms that penalize abrupt variations in the ramp metering and VSL:

$$J(k) = \sum_{\ell=1}^{N_\mathrm{p}} T[\ \sum_{i\in\mathbb{O}} w_i(k+\ell)\ +\ \sum_{li\in\mathbb{I}} (\rho_i(k+\ell)L_i\lambda_i)\ ] \tag{3.9}$$

$$+\ \sum_{\ell=0}^{N_\mathrm{u}-1} \epsilon_\mathrm{vc} \parallel V_{\mathrm{c},j}(k+\ell) - V_{\mathrm{c},j}(k+\ell-1) \parallel^2 \tag{3.10}$$

$$+\ \sum_{\ell=0}^{N_\mathrm{u}-1} \epsilon_\mathrm{r} \parallel r_{\mathrm{c},j}(k+\ell) - r_{\mathrm{c},j}(k+\ell-1) \parallel^2$$

where ϵ_vc, ϵ_r are weighting parameters, $\mathbb{O}$ is the set of all the segments with an on-ramp, and $\mathbb{I}$ the set of all the segments. If other traffic control measures are used (as reversible lane or route guidance), their variations may be also penalized in the cost function.

Notice that, since MPC can accept any quantifiable cost function, any other traffic performance criteria can be used. The performance criteria could vary depending on the desire of the stakeholders of the network, the time of operation of the network, and the location of the network. For example, environmentalists would choose reduced dispersion of emissions and propagation of sound pollution to a protected target zone, while transport authorities could be interested to improve traffic throughput and safety.

Other performance criteria have been previously used for freeway traffic control as total traveled distance, tracking of set-point values [24], safety, emissions, fuel consumption or dispersion of emissions [109]. If desired, the objective function can be a vector of control objectives with some competing multi-objective criteria.

Controller sample time

In traffic control, it is usually used a controller sample time T longer than the simulation sample time T_m (i.e. $t = k_\mathrm{m} \cdot T_\mathrm{m} = k \cdot T$). For example, using a model sample time (T_m) of 10 seconds and a controller sample time (T) of 120 seconds, $t = 360s$ correspond to $k_\mathrm{m} = 36$ and $k = 3$.

A controller step time of two minutes (or lower) is necessary in order to deal with unexpected congestions due to accidents or other reasons (like unexpected demand increases). On the other hand, a long enough controller sample time is

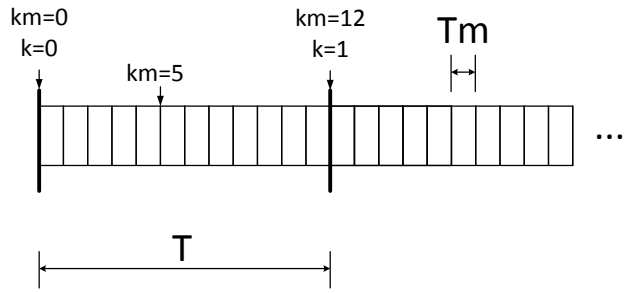

Figure 3.3: Simulation and control sample times

necessary to have enough time between samples in order to solve the optimization problem.

All the controllers proposed in this thesis use a model sample time (T_{m}) of 10 seconds and a controller sample time (T) of 120 seconds. It has to be taken into account that for the majority of the times that the control inputs are computed, the solution is the same as for the previous controller step, but shifted in time avoiding unnecessary changes in the control inputs.

Tuning issues

In this thesis, each controller takes different values for the tuning parameters in order to compare the controllers with its optimal behavior (i.e. each controller uses the values which minimize the Total Time Spent in simulations).

The results are very sensitive with the tuning and, therefore, a meticulous tuning procedure has to be done for each network and controller. Especially important are the set of ϵ_{vc} and ϵ_{r} (i.e. the parameters that multiply the penalization in the changes in the control signals).

As an example of the tuning of a MPC, Table 3.1 shows the results in terms of the TTS reduction for different values of these tuning parameters. The example comes from the simulation done in the following section 3.1.4. The table shows how just changing the parameters from (0.8, 0.2) to (0.5, 0.2), the TTS reduction changes from 25.6 % to 21.8 %. It can be concluded that, in real cases, it could be useful to choose different tuning values for different typical demand profiles or weather conditions in order to have a proper optimization.

In theory, the penalization factors which multiply the soft constraints of density, speed and queues (i.e. δ_{vel} δ_{dem} and δ_q) have to be large. However, in practice, these factors cannot be too large or numerical problem will appear during the optimization.

With respect to the control and prediction horizon, an increase in both horizons will improve the behavior and, at the same time, will increase the computational time needed. In general, the horizons size will depend on the size of the network. For a large network, a greater TTS decrease will be obtained increasing the horizons but the computational time may increase critically. For regular networks, a good trade-off between computational cost and behavior may be to choose the prediction horizon between 3 and 7.

Table 3.1: Tuning of the centralized MPC

ϵ_{vc}	ϵ_{r}	TTS	Reduction (%)
4	4	1357.74	19.3
0.01	0.01	1362.5	19
0.1	0.1	1362.5	20.3
1	0.1	1342	20.3
0.5	0.1	1277.3	24.1
0.5	0.5	1300.2	22.7
0.8	**0.2**	**1252.8**	**25.6**
0.3	0.3	1296.3	23
0.5	0.2	1316	21.8

It is important to note that in order to obtain a good behavior the difference between the control horizon and the prediction horizon has to be either small or zero. It happens because does not make really sense to consider constant the control input during a long final period due to the system does not tend to steady state. If we set an $N_p - N_u$ too large, the system takes too much into account the final values of the control signal causing a suboptimal behavior.

3.1.4 A simple case study

In order to design a first centralized MPC controller for a simple freeway network, the following freeway example (Fig. 3.4) has been used.

The benchmark has been taken from [37]. The freeway has 6 segments and only one link. Each segment has a longitude of $L = 1000m$ with $\lambda = 2$ lanes.

There are three control signals: VSLs on segments 3 and 4 and a ramp metering in segment 5. Thirteen variables are measured at each sample time (mean density and speed of each segment and queue of the ramp metering) and used for the computation of the control signals. It is important to note that we are supposing that the densities can be measured. In real cases, it would be necessary to estimate them.

All the model parameters are considered equals for all the segments. The remaining model parameters can be seen on Table 3.2.

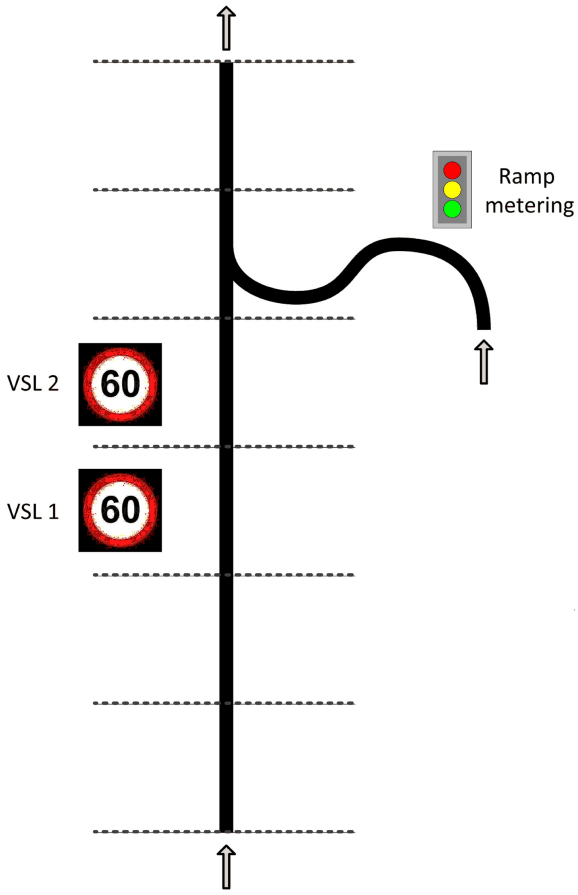

Figure 3.4: Strech used as example.

Table 3.2: Model Parameters

τ	0.05 s	K	40 $\frac{veh}{K_m * Line}$	ρ_{crit}	33.5 $\frac{veh}{K_m * Line}$
α	0.1	a	1.867	v_{free}	102 $\frac{K_m}{hour}$
μ	60 $\frac{K_m * K_m}{hour}$	δ	0.0122	ρ_{max}	180 $\frac{veh}{K_m * Line}$
C_o	4000 $\frac{veh}{hour}$	C_{ramp}	2000 $\frac{veh}{hour}$		

The input flow demands are chosen in order to obtain a simulation with a high density where the traffic control can improve substantially the behavior of the system. The simulation time chosen is two and half hour that corresponds to 75 controller sample time and 900 simulation steps. In the Fig. 3.5, the response of the systems for the no-control case and for the VSL and Ramp Metering case are shown. In the two first graphics, each curve represents the density and speed of one segment respectively. In the third graphic, only the origin inputs flows (main input at segment 1 and ramp metered input) are shown.

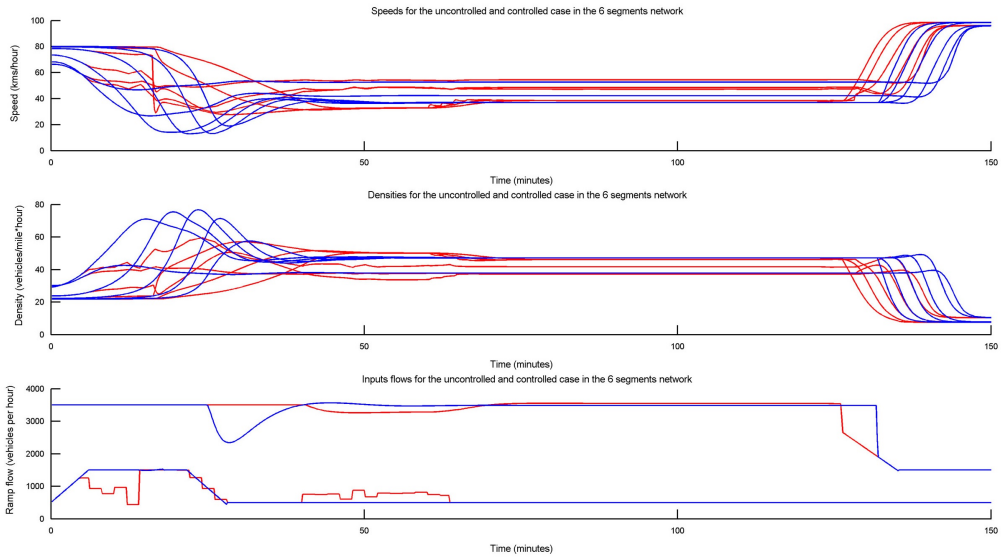

Figure 3.5: Results of local MPC in one stretch. In blue, the no control case response is showed while in red it is possible to see the ramp metering and VSL response.

In the figure, it can be seen how the control system clearly improves the behavior, reducing the shock-wave that appears during the beginning of the simulation and bringing forward the increase in speed that happens during the end of the simulation.

When the freeway is simulated without control the TTS is 1460 $veh * hour$ and there is a violation of the queue constraint during the most of the simulation time (two and half hours).

If just ramp metering are used (computing them by a centralized MPC as explained above), the TTS is reduced to 1411 $veh * hour$, which is a decrease of a 3.4%. Moreover, the constraints are not violated during all the simulation time. When ramps metering and variable speed limits are used, the TTS is 1285 $veh * hour$, which means a 12% reduction and the constraints are satisfied during all the simulation time.

3.2 Robustness of MPC for freeway traffic systems

As previously explained, dynamic control methods for traffic have proved to be a good solution to decrease congestion [41, 10, 39, 99, 82]. However, the number of uncertainties in the traffic itself and in the demand, the unmodeled dynamics in most of the macroscopic traffic models, and the problems related with the communication issues in distributed networks (like delays, missing samples, sensor errors, band with limitations, etc.) make it necessary to re-evaluate the available control strategies, in terms of their robustness capabilities. In other words, it is necessary to know how the controllers will react under different levels of uncertainties and whether they are capable (or not) to guarantee some performance or level of service in a worst case scenario.

Dynamic traffic control takes the state of the traffic into account over time and computes control signals that change the response of the traffic system aiming at improving its behavior. Regarding the methods, the ones based on Model Predictive Control (MPC) are based on the minimization of interpretable performance indexes while considering operational and policies constraints, [28, 72, 38, 109, 10, 36, 5, 33]. However, most of the literature assumes that the real demands are equal to the predicted demands, so the robustness of the proposed methods is unknown. When talking about robustness we refer to the capability of a controller to maintain performance specifications for a range of uncertainties.

In the general literature of MPC, for wide range of applications, there are many ways to describe uncertainty and noise in general. Different techniques have been proposed to achieve a robust performance, robust constraint handling, and robust stability [58]. In general, all those works prove that MPC controllers have inherently some robustness properties. However, for the case of traffic control, to the best of our knowledge, an analysis of the robustness capabilities of MPC for freeway traffic control is not yet available. In this section we will focus on the case of variations in the mainstream demand, in a setup considering ramp metering and variable speed limits (VSL).

The majority of the developments of robust methods for traffic can be found in the literature of urban networks [107, 56]. However, urban traffic control [18] is quite different from the freeway traffic control. In [73], as prediction methods do not capture all the stochasticities of the traffic conditions for every scenario that is faced in a real-life freeway network, it was proposed the use of fuzzy confidence intervals. Those intervals provide information about the variance of the traffic signals, which is quite useful to analyze the effects of those uncertainties and to classify different possible scenarios (normal day, congested, very congested, etc).

This section proposes new methods for online demand estimation and analyzes the robustness of MPC controllers for freeway traffic with respect to variations in the mainstream demand. The scenarios considered correspond to seventeen different

days of available real-life measurements of the freeway A12 of The Netherlands.

3.2.1 Case study

To simulate the MPC controllers, the same network that in the previous section 3.1 has been used (see Figure 3.4), with the same parameters as [37, 29].

The particularities of the MPC controllers used in this paper (cost function, constraints, solver, etc.) are the same as the ones exposed in the previous section 3.1 with two exceptions:

– The control and prediction horizons used are $N_u = 3$ and $N_p = 6$, respectively.

– In order to try to avoid that the algorithm ends up in a local minimum, the algorithm runs five parallel optimizations with different initial points and the best control profile obtained is selected as the solution for the controller.

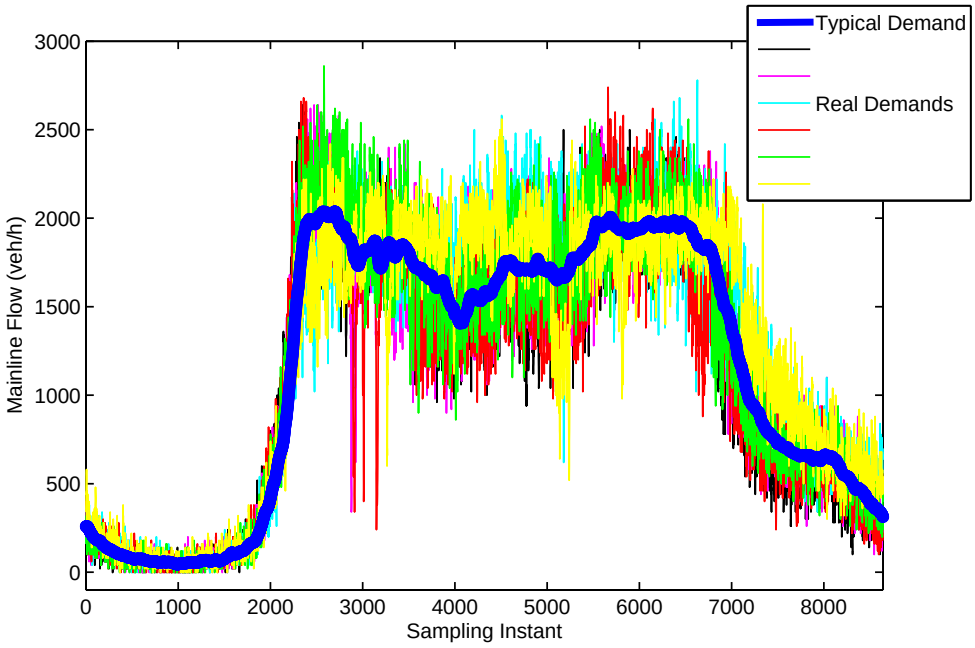

Figure 3.6: Real and typical demands

The mainstream demand considered is based on real data of 17 days from a loop detector on marker post 35.9 (direction east) on the A12 freeway in The Netherlands. Firstly, a typical mainstream demand profile is obtained by averaging and filtering this real data for the 17 days. This has been done by using the Curve Fitting Toolbox of MATLAB. The typical demand and the real demand for many days can be seen in Figure 3.6.

Subsequently, a window of 5 hours (between 6:00 am and 11:00 am) has been chosen corresponding to 150 controller sample steps ($T_c = 120$s) and 1800 simulation steps ($T = 10$s). The analyzed demand from the A12 (Fig. 3.6) is not large enough to create recurrent congestion. As this section focuses on the impact of errors in the demand estimation for MPC controllers solving recurrent congestion on freeways, the real data will be scaled in order to create scenarios with recurrent congestion.

3.2.2 Mainstream demand disturbances estimation

One important issue when using macroscopic traffic models for freeway traffic control is the proper way to include not controllable inputs, called $d(k)$ in Section 1.5. For METANET, these variables are the on-ramps demand, the origins demand, the downstream density for the last segment, and the upstream speed for the first segment.

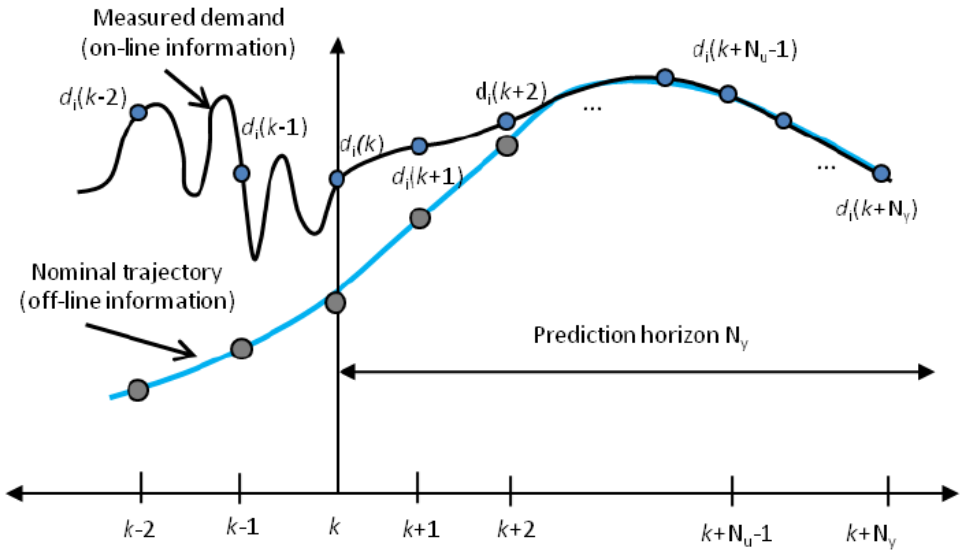

Figure 3.7: Online demand prediction based on a nominal trajectory

In MPC based traffic control an estimation of these variables is necessary to use the model and to predict the future states. This estimation may be done using loop detectors outside the controlled networks. However, in the common cases when there are no valid online data available from sensors outside the considered freeway, this estimation has to be done by combining information coming from historical data and on-line methods to improve the estimations (see Fig. 3.7). For short-term traffic prediction, different methods have been proposed in the

literature (see for example [102, 12] and the references within). In this paper, three simple methods have been used for the online prediction of the demands.

The proposed methods use the variable $\xi(k)$ in order to model the error between the predicted demand $\hat{D}_o(k)$ and the real demand ($D_o(k)$):

$$\xi(k) = \hat{D}_o(k) - D_o(k) \tag{3.11}$$

Diophantine Estimation

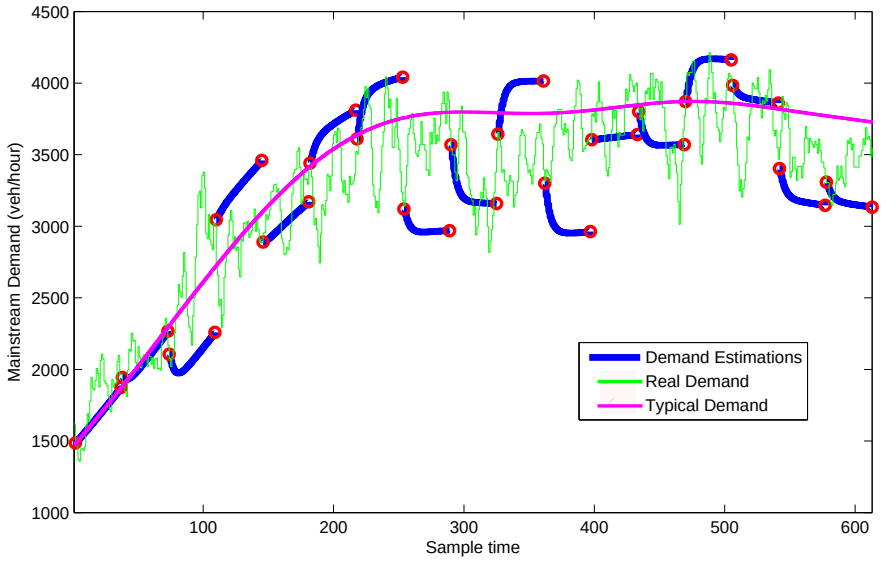

Figure 3.8: Diophantine Demand Estimations.

In the first method used, the difference $\xi(k)$ between the real demand and the typical demand is defined by a CARIMA model with $B(z^{-1}) = 0$ and $A(z^{-1}) = 1$:

$$\xi(k) = \frac{C(z^{-1})}{\triangle} e(t) \tag{3.12}$$

where $e(t)$ is a zero mean white noise with $E[e^2(t)] = 1$. The model can be rewritten as:

$$\begin{aligned} \triangle\xi(k) &= c_0 \cdot e(k) + c_1 \cdot e(k-1) \\ &+ c_2 \cdot e(k-2) + ... + c_n \cdot e(k-n) \end{aligned} \tag{3.13}$$

with $c_0, c_1, ..., c_n$ obtained by solving the following system of equations with n

variables and n equations:

$$E[\triangle\xi(k) \triangle \xi(k-m)] = c_0 \cdot c_m \\ +c_1 \cdot c_{m+1} + ... + c_{n-m} \cdot c_n \\ \text{for } m = 0, 1, ...n \tag{3.14}$$

Finally $E[\xi(k+t)]$ during the prediction horizon is obtained by recursively solving the Diophantine equation $C(z^{-1}) = E_j(z^{-1}) + z^{-j} \cdot F_j(z^{-1})$ as explained in [9].

The results of this estimation approach for some real initial points are shown in Fig. 3.8. In the figure, it can be seen that the estimation does not tend to the typical demand, Moreover, in some cases, for large prediction horizons the estimation can be unstable. This undesirable predictions appears because we are approximating the expected values $E[\triangle\xi(k) \triangle \xi(k-m)]$ and not the prediction of the system.

CARMA Estimation

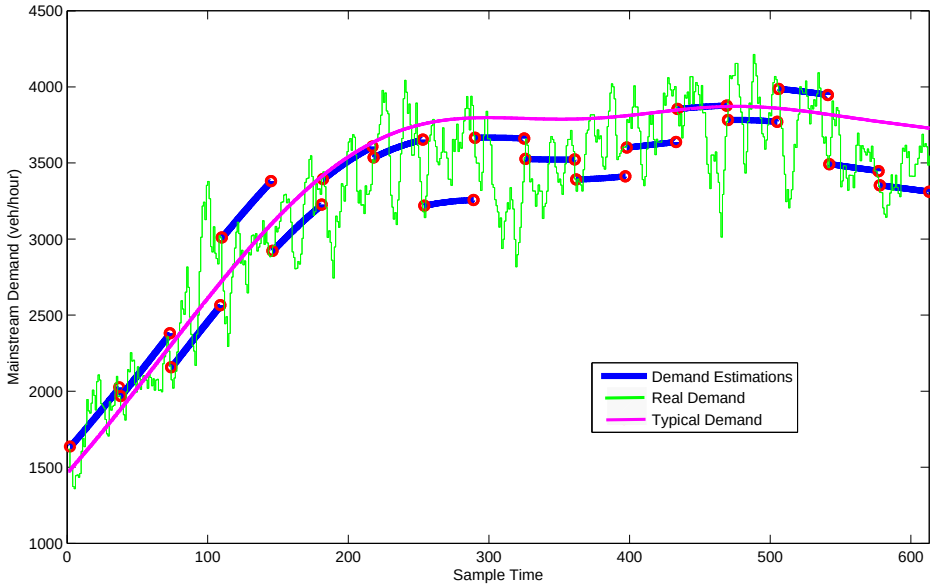

Figure 3.9: CARMA Demand Estimations.

In the second method, $\xi(k)$ is modeled by a CARMA model with $B(z^{-1}) = 0$, $C(z^{-1}) = 1$, and $A(z^{-1}) = 1$ in order to have a more stable demand prediction:

$$\frac{\xi(z^{-1})}{e(z^{-1})} = \frac{1}{1 - a_1 \cdot z^{-1} - ... - a_n \cdot z^{-n}} \tag{3.15}$$

Therefore, $\xi(k)$ can be predicted by:

$$E[\xi(k)] = a_1 \cdot \xi(k-1) + ... + a_n \cdot \xi(k-n)$$

with $a_1, a_2, ..., a_n$ obtained by minimizing the error in the demands $\xi(k+t) = \hat{D}_o(k+t) - D_o(k+t)$ during the prediction horizon for all the simulation sample times with valid data (more than one million points).

The results of this estimation for some real initial points are shown in Fig. 3.9. In the figure, it can be seen that the prediction is almost parallel to the typical demand starting in the last measured demand $\hat{D}_o(k)$.

Exponential Estimation

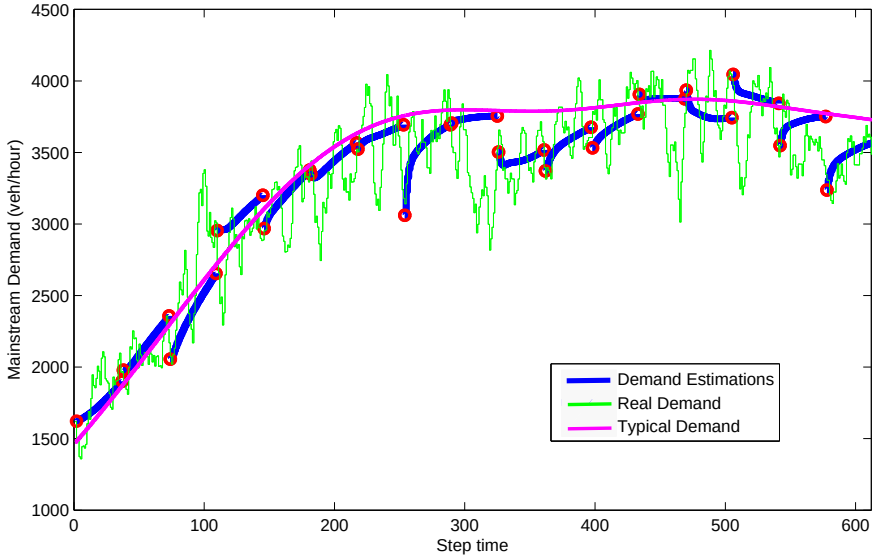

Figure 3.10: Exponential Demand Estimations.

For control purposes, it is desirable that the demand estimation tends to typical demand in the long term. Therefore, a third very heuristic method is proposed in order to force to the estimated error $\xi(k)$ to tend to zero. In this method, $\xi(k)$ is modeled by a series of exponential functions in order to have a demand prediction that explicitly tends to the typical demand:

$$\xi(k+t) = \sum_{n=0}^{N} A_n \cdot \xi(k-n) e^{-(k+t)\cdot B_n} \text{ with } B_n > 0$$

with $A_0, B_0, A_1, B_1, ..., A_n, B_n$ obtained by minimizing the error in the predicted demands in the same way as the CARMA estimations.

The results of the exponential estimation for some real initial points are shown in Fig. 3.9. In the figure, it can be seen that the estimated demands are very close to the real demand, giving the best prediction of three proposed methods.

3.2.3 Simulation results

This subsection shows the simulation results for the analyzed freeway network for the seventeen days simulated. The real demands have been used for the simulation of the traffic network in all the cases but the demand profile used by the model predictive controllers differs for each case.

Using the real demands in the MPC optimization is expected to yield the best performance, but this option is not implementable in real applications as it is not possible to know how the future real demands will evolve. Using the online estimated demands is expected to result in an intermediate performance in between the real demands and the typical one.

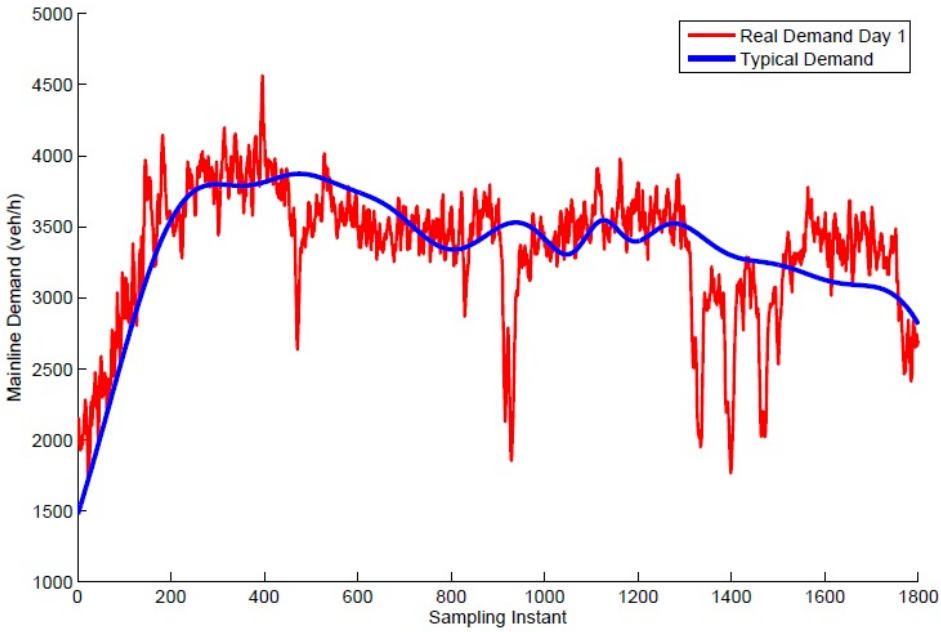

Figure 3.11: First day simulated.

An example of the real mainline demands of one day simulated can be seen in Fig. 3.11. The curves in red show the real demand that enters the freeway which is the one used for simulation. In blue it is possible to see the typical demand. As

previously said, the typical demand is a fitting of all the real demands (seventeen days) used.

Table 3.3: Simulation results for the different demand profiles

	Typical Demand	Real Demand	CARMA Estimation	Exponential Estimation
Day 1	15.65 %	15.70%	15.69%	15.69%
Day 2	11.45 %	11.45%	11.45%	11.45%
Day 3	12.09%	12.10%	12.09%	12.09%
Day 4	0 %	0.01%	0.01%	0.01%
Day 5	24.76%	24.86%	24.82 %	24.86%
Day 6	10.61%	10.62%	10.62%	10.62%
Day 7	22.84%	22.86%	22.85%	22.85%
...	...	...	...	...
Mean	9.77%	9.80%	9.78%	9.79%

In Table 3.3, the numerical results are shown. It can be seen that the reduction is almost the same for the four cases analyzed. As expected, the online estimated demands result in an intermediate performance in between the real demands and the typical one but just a minor improvement (an average of 0.03 % in the TTS reduction) is obtained by having a better estimation of the mainstream demand. Therefore, it can be concluded, at least for this scenario, that the MPC controller is not very sensitive with respect to the demand profiles so the performance of the closed-loop MPC depends much more on the state variables (measured densities, queues, and speeds) than on the not controllable inputs such as the mainline demand.

3.2.4 Expected improvement with respect to the average mainstream demand

Analyzing the results in Table 3.3, it is possible to see that the different days simulated result in a very different TTS reduction. The mean reduction is 9.77 % but in some days there is a large TTS reduction (such as 24.86 % or 22.86 %) while other days the controllers have no repercussions on the performance (a TTS reduction of a 0 %). This is caused by the limited range where the ITS signals can be useful in order to reduce recurrent congestion.

In Fig. 3.12, it is possible to see an estimation of this potential reduction in terms of the average number of vehicles that enter the freeway. The expected value of the

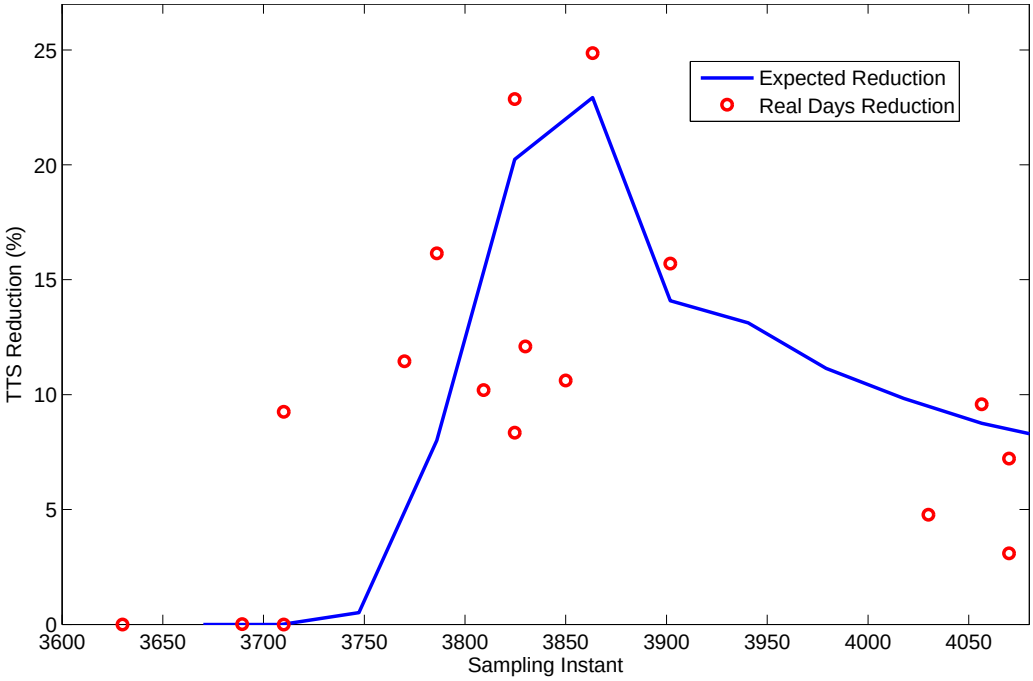

Figure 3.12: TTS reduction with respect to the total number of vehicles.

TTS reduction has been obtained by simulating the networks using scaled version of the typical demand for both simulation and control. It is important to note that this corresponds to a quite rough estimation, due to the lack of information about the distribution of the demand over time. However, this estimation is useful just to get an idea whether or not for one day a considerable TTS decrease can be achieved.

In Fig. 3.12, it can be seen that TTS reduction obtained for the simulation with real data (red circles) is relatively close to the expected potential. For example, the days with a 0 % TTS reduction correspond to days with a low demand which does not create congestion. So, it can be concluded that the range where the VSL signals can be useful to decrease recurrent congestion is relatively limited. However, in the cases where the potential of the TTS reduction is very low or 0%, the MPC computed VSL signals could be useful to reduce shock-waves [41], unexpected bottlenecks [40], homogenize speeds [108], etc. which has not been studied in this paper.

3.3 Conclusions

Firstly, this chapter has provided a brief account of the basic concepts of MPC for traffic systems in Sections 3.1. The MPC controller is demonstrated in a simple simulation-based case study in subsection 3.1.4. For detailed discussions on MPC for freeway traffic systems, we refer the reader to [37].

Moreover, this chapter has proposed new methods for on-line demand estimation and has analyzed the robustness of MPC controllers for freeway traffic with respect to variations in the mainstream demand.

Firstly, it has been proposed three simple estimation procedures for the demand profiles that combine off-line information (historical data) together with on-line information (demand in previous time steps). The first one uses a CARMA model, the second one uses a CARIMA model and third one uses a series of exponential functions. The exponential estimation shows the better close-loop performance of the models considered.

Subsequently, the robustness of MPC controllers for freeway traffic has been analyzed by the simulation of seventeen days with real-life data from the A12 Dutch freeway using different demand profiles for the MPC controllers. The conclusion is that, at least for this scenario, the MPC controller is not very sensitive with respect to the demand profiles so the controller will perform properly even if there are visible errors in the demand estimation.

Finally, the potential reduction of the total time spent with respect to the aggregate demand is studied, showing the range where ITS signals can be used to increase the traffic flows by solving recurrent congestion.

Chapter 4

Distributed MPC for Freeway Traffic Control

Nowadays, most of the dynamic traffic control systems operate according to a linear and local control loop. As explained in Chapter 3, the use of appropriate non-local and multivariable techniques can improve considerably the reduction in the total time spent by the drivers and other traffic performance indexes. Non-linear centralized MPC is probably the best control algorithm choice for a small network as can be seen on [37].

The main problem of nonlinear centralized MPC is that the computational time quickly increases with the size of the network. Thus, centralized MPC could be difficult to apply for large networks. Therefore, completely centralized control of large networks is viewed by most practitioners as impractical and unrealistic.

A possible solution is to consider the network as a set of subsystems controlling each subsystem by one independent MPC (i.e. to use a decentralized control scheme). It is known that such a completely decentralized control strategy may result in unacceptable control performance, especially if the units interact strongly as in control traffic systems.

The loss of performance due to the decentralization and the difficulty of implementing a completely centralized control for large networks is analyzed in Section 4.1. This analysis will be done by the comparison of the performance of local and centralized controllers for a relatively large traffic network .

Section 4.2 proposes distributed MPC algorithms which can be implemented in real time for a large enough traffic network minimizing the total time spent by the drivers.

Parts of this chapter are published in [29, 27, 28].

4.1 Global versus local MPC algorithms in freeway traffic control

This section compares global and local MPC algorithms ("linear", "decentralized MPC", "decentralized MPC with communication after sample", "centralized MPC" and a "roughly optimal solution") in a traffic network controlled by ITS signals (ramp metering and variable speed limits).

It will be shown that the local techniques have a suboptimal behavior and that centralized techniques are difficult to implement in real time. In order to deal with this problem, a local MPC with only one communication cycle at each sampling time is proposed. This controller improves the local controller performance and, although it is suboptimal with regards to the centralized controller behavior, it can be implemented in real time.

4.1.1 Case study

In Section 3.1, the proposed centralized MPC can be computed in less time than the controller sample time since the network was relatively small. In order to analyze a bigger network, three consecutive stretches of the freeway simulated in Section 3.1.4 have been analyzed together.

Each link has the same geometry and traffic control signals that the "one link example" explained in 3.1.4 (see Figure 3.4). Therefore, an 18 kms freeway with three Ramp Metering and 6 Variables Speed Limits will be simulated (Figure 4.1). All the freeway parameters are set equals that in Section 3.1.4.

The network is big enough to make impracticable a centralized controller like the one simulated previously, as explained in the following section.

Analyzing the network, since there is only one destination, the biggest traffic density will appear in the last link. The control actions in links 1 and 2 will have a large effect in the third link that could increase the traffic jam in this link. Therefore, in this network (as happens in real traffic networks), the consideration of the effects of the neighboring controllers will be a critical issue.

4.1.2 Analyzed controllers

Centralized MPC

The centralized MPC is a controller that optimizes the full network (18 kms) for a given prediction and control horizons. It has the same structure that the MPC explained for the one link simulation but increasing the size of the network (i.e. the number of variables and constraints). The behavior of the network must be better or equal than any nor-centralized controller.

The centralized MPC cannot find a solution for the three links case during a sample time (2 minutes) in a Pentium I3 with 3 Mhz using the algorithm previously explained. If more computational power were available (or if a quicker optimization algorithm were used), the centralized controller could be implemented for this network in real time for small horizons.

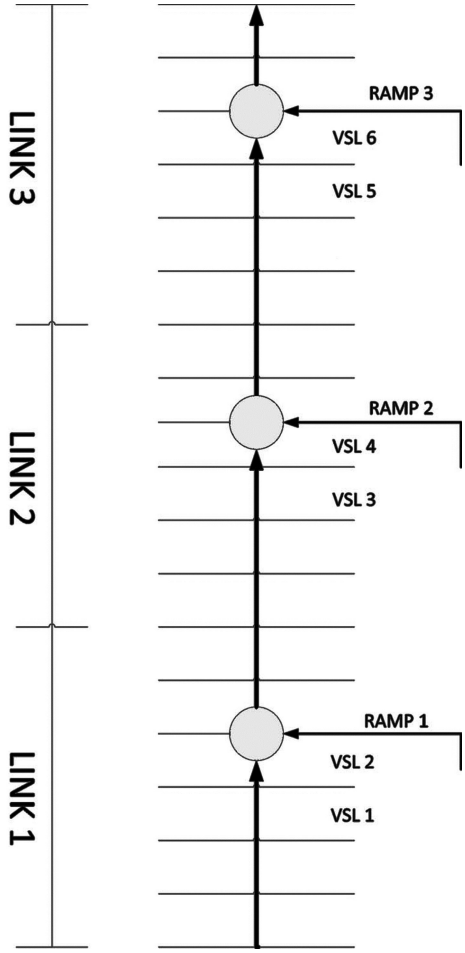

Figure 4.1: Traffic Network simulated. Each link is equal to the stretch showed in Fig 3.4.

However, the computational requirements grow very fast with the control horizons. By extrapolating the results, the computing time required to solve this non-convex optimization problem will be too large for a medium size freeway. Thus, bigger network or bigger horizons would make impracticable the centralized MPC.

Therefore, in this chapter, it will be treated as local a controller that considers the network of Fig.3.4 (6 segments) and as global a controller that considers the full network of Fig.4.1 (18 segments). With a higher computational capacity, different

considerations of local and global MPC may be used but the conclusions would be the same.

Local MPC without Communication

The second controller tested on benchmark is the use of three local MPC, each one controlling one part of the network. Each controller has the same structure that the MPC explained in Section 3.1 for the 1 link example.

There is no communication between controllers. The future disturbances (upstream speed and flow and downstream density) are defined by the simulation of the no-control case for any of the agents. This is a fully decentralized case where there is no communication between controllers at any time.

Local MPC with Communication after Sample

The upstream flow and speed and the downstream density are necessary in order to model a segment. Therefore, each MPC controller will need the current and future values of these variables. These variables can be seen as estimable disturbances.

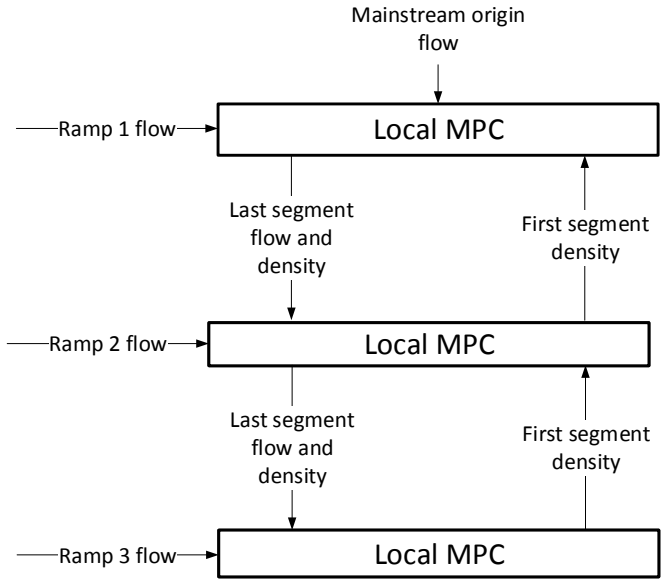

Figure 4.2: Controller interconnection structure for communication. The state variables (and their corresponding estimations along the prediction horizon) are sent between controllers after each optimization.

The communication between controllers after any sample will allow the local MPCs to use an estimation of these disturbances that are defined by the predicted values of the adjacent MPC controllers (Fig. 4.2).

After any sample time, a controller will send the future predicted values of the density of its first segment to the previous controller . The controller will also send to the following controller the future predicted values of the speed of its last segment and of its output flows.

This procedure will allow the others controllers to use a more real prediction of the disturbances (input flow and downstream density). However, the controllers will not take into account the effects of their actions in the others parts of the networks.

Thus, it can be said that this controller communicate but do not cooperate with their neighbors. Two ways in which this controller could be improved are:

–To communicate between controllers many times inside a controller time step reaching the "Nash equilibrium".

–To cooperate with others controllers in order to achieve the common goal using a distributed algorithm such as "Feasible Cooperation Based MPC" [101].

In Section 4.2, these techniques are applied to this benchmark almost reaching the centralized behavior with acceptable computational times.

ALINEA

The forth controller tested on the benchmark is "ALINEA Ramp-Metering Control" [88]. ALINEA is the most implemented ITS control technique. It is a simple, robust and flexible local strategy that does not use VSLs.

The control law is a linear feedback derived by use of classical automatic control methods. The ramp metering rate is computed adding a linear expression of the error between the density downstream and a desired density (usually, the critical density) to the previous metering rate.

ALINEA is just an example of ramp metering control algorithm. There are some other techniques as can be seen on [95] but it will be assumed that all the linear ramp metering algorithms have a relatively equivalent performance that ALINEA.

Roughly Optimal Solution

In order to obtain a solution that optimizes the TTS for the full problem (centralized controller taking into account the complete simulation time of 2.5 hours), a "roughly optimal solution" was computed. It is practically impossible to obtain this result using a MATLAB optimization function like "fmincon". The solution does not converge in a reasonable time.

Therefore, a global random search algorithm is used, which allows to obtain a roughly optimal solution. The algorithm evaluates the TTS for different values of the control signals. Each 100 steps, during the first 50 attempts the algorithm searches new points close to the previous point (11) and evaluate if this points have a shorter TTS.

$$u_i(k, attemptj) = u_i(k, attemptj - 1) + \\ + 0.01 * rand(1) \ - \ 0.005 \tag{4.1}$$

During the next 50 attempts (n), the algorithm searches farther points in order to avoid that the algorithm stays in a local minimum (12).

$$u_i(k, attemptj) = u_i(k, attemptj - 1) + \\ + 0.005 * (n - 50) * rand(1) \ - \ 0.5) \tag{4.2}$$

The computational time needed to converge is huge (around 6 days in this example), which implies that, obviously, this controller cannot be implemented in real time.

The objective of this controller is to have a point of reference of the best control sequence that could be applied to this traffic network.

4.1.3 Results

Numerical analysis

As can be seen on results (Table 4.1), all the controllers reduce the TTS. At the same time, all the controllers keep the variables inside the constraints except in ALINEA control, where a violation of the queue constraint appears during more than half an hour.

In the table, CT_{max} shows the maximum computation time of each controller (in the local case, CT_{max} have been taken from the worst cases, i.e. more restrictive cases, of the three controllers). Red is the reduction of the TTS in percentage with respect to the non-linear case.

Analyzing the results of the local MPC, it can be seen how the communication after sample increase substantially the reduction of the TTS (from 6.5% to 12.88%). It shows how the controllers need to take into account the effects of the acts of other controllers in its part of the network. It can be also seen how an increase in the horizons improves the behavior just a bit increasing critically the computational times needed.

The most important result is that the difference in the TTS reduction between local (6.5%) and centralized (26.4%) control schemes is very large. It shows how

Table 4.1: Results of the implementation of the different controllers

	N_u	N_p	TTS	Red (%)	CT_{max}
Uncontrolled system	-	-	1684	0	-
ALINEA	-	-	1611.1	4.3	≈ 0
Local MPC without communication	5	7	1573.6	6.5	51.6
Local MPC with communication after sample	5	7	1467.1	12.88	98
	8	8	1460.3	13.28	172
	12	12	1437.5	26.4	1103.9
Centralized MPC	5	7	1238.8	26.4	1103.9
Roughly Optimal Solution	-	-	1157.9	31.24	≈ 6 days

a good traffic control system needs to also take into account the effects of the own traffic control system in other parts of the network. Without this consideration, solving a traffic jam in one part of the network could increase the number of vehicles that arrive to a bigger traffic jam, deteriorating the global behavior of the network.

Since the centralized MPC is difficult to implement in real time (the average computational time is 316.47 s), it can be concluded that it is necessary to find an easily implementable control algorithm that has a closer performance to the centralized one. The best solution may be the use of distributed model predictive control algorithms because they try to approximate the centralized behavior in a parallel computation with communication and cooperation [28].

The difference between the centralized MPC and the roughly optimal solution shows that even the centralized MPC can be improved (in a 4.8 % in this example). Some possible solutions in order to achieve the optimal solution could be:

– Computing a roughly optimal solution off line for the typical or previous input profiles and use it as reference for the controller implemented.

– Introducing new terms in the cost function that improve the behavior.

– Increasing the horizons. This option probably will not be implementable in real time.

Graphical analysis

The following figures show a 3D representation of the evolution of the densities for each segment.

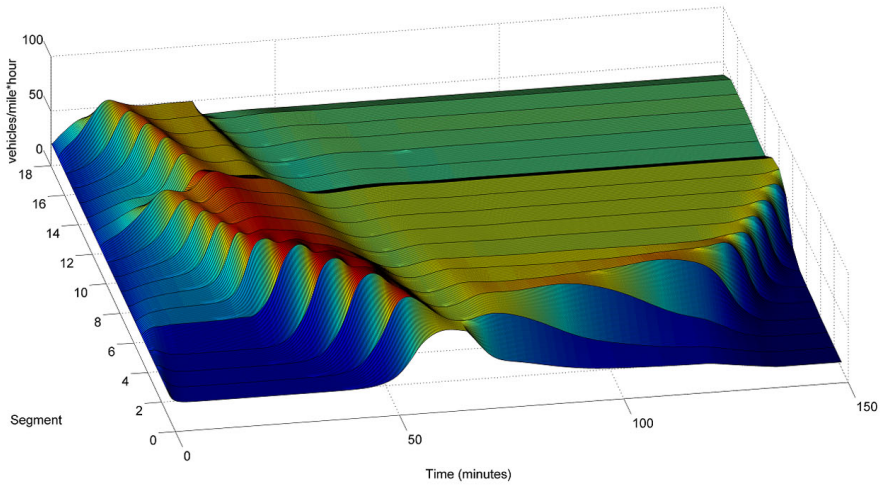

Figure 4.3: Densities for the uncontrolled case.

Fig.4.3 shows the response of the system without control. Two congestions appear at the beginning of the simulation in the second and third on-ramps (segments 12 and 18). These congestions cause two shock waves that move the congestions upstream. Subsequently, both congestions join together in a bigger shock wave causing a bigger traffic jam.

Fig.4.4 shows the response of the system using three local MPC controllers without communication between them. The response shows a large reduction in the size of the shock wave (i.e. in the density of the traffic jam). However, it is possible to see a discontinuity between the zones controlled by each controller, especially on segment 12. Moreover, the density is increased during a new shock wave reaching the critical density during a few minutes.

Fig.4.5 shows the response of the system using three local MPC controllers with communication. The response shows a new reduction in the size of the shock wave reducing the discontinuities between controllers. Furthermore, the largest congestion remains during less time (especially in the last segments of the network). The density is slightly increased during the halfway through the simulation after the shock wave. However, during this part of the simulation, the densities keep under the critical density increasing the TTS.

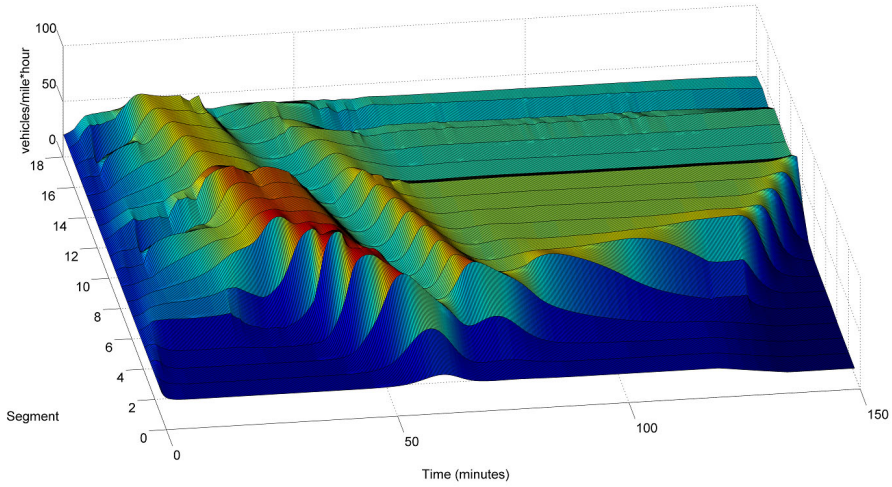

Figure 4.4: Densities using Local MPC without communication.

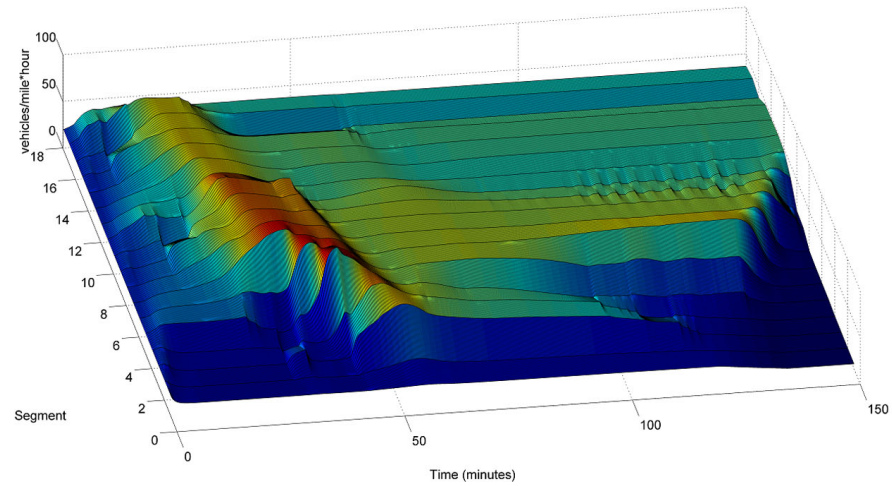

Figure 4.5: Densities using Local MPC with communication after sample

4.2 Distributed MPC for freeway traffic control

The main objective of this section is to design a control algorithm that can be implemented in real time for a large enough traffic network minimizing the total time spent by the drivers. Therefore, this section proposes the use of distributed algorithm for the control of freeway traffic systems.

Distributed MPC algorithms [101] try to solve the problem in a parallel computation using the communication and cooperation between the different MPC controllers in order to achieve the centralized performance.

Using communicative MPC, the interactions between systems are modeled so that each controller takes account of the actions of its neighbors. In each interaction, the predicted trajectories are exchanged between controllers and the optimization is repeated with the new values of the control signal profiles of its neighbors. If the algorithm converges (proved for linear systems), the "Nash Equilibrium" is reached. However, the Nash Equilibrium is suboptimal for many systems such as the traffic system.

In order to improve the behavior, cooperative MPC can be used. This technique modifies the objective functions of the local MPCs including also the objective functions of near agents properly weighted. The iterations and the exchange of information are done in the same way that communicative MPC.

Using feasible cooperation based MPC algorithm (FC-MPC), only the local variables corresponding to each controller are used as decision variables. In [101], it is proven that FC-MPC converges to the optimal centralized MPC control (Pareto-optimum). There are no proven results for non-linear cases. However, in this section is possible to see how for traffic systems the centralized MPC behavior can be roughly reached using FC-MPC.

FC-MPC will be tested and compared with global, local and communicative MPC techniques in a traffic network of 18 segments with ITS (Intelligent Transport Systems) control signals: ramp meters and variable speed limits. It will be shown that local techniques have a suboptimal behavior and that centralized techniques are very difficult, if not impossible, to implement in real time.

Communicative MPC improves the behavior of the controlled system versus the decentralized one. However, the solution is still suboptimal with respect to the centralized performance. On the other hand, FC-MPC is closed to the centralized behavior and has a much lower computational effort that the centralized one.

4.2.1 Proposed controllers

Communicative MPC

Communicative MPC uses the same procedure that the algorithm "Local MPC with Communication after Sample" explained in the previous section but making the process "communication + optimization" many times inside a controller step time. In this case, four iterations were used and it was enough in the majority of the cases to converge to an equilibrium point, which is the Nash Equilibrium. This control technique has the same problem explained previously: the controllers do not cooperate and, therefore, two controllers could be counteracting.

Cooperative MPC (Feasible Cooperation Based MPC)

In the cooperative MPC the local cost function of each controller is replaced by a global cost function (in this case, the TTS of the 18 segments). In order to reduce the computation effort, only the control signals of each part of the network (in this case, two VSLs and one ramp metering) are considered as decision variables. Therefore, the following cost function is considered:

$$
\begin{aligned}
&J(k) = \qquad\qquad (4.3)\\
&= \sum_{k=Mk}^{M(k+N_p)-1} \Big[\sum_{o\epsilon O} w_o(k \mid Mk) + T \sum_{(m,i)\epsilon I} (\rho_{m,i}(k \mid Mk) L_m \lambda_m) \Big] +\\
&+ \sum_{l=1}^{N_c} \epsilon \parallel u(k+l \mid k) - u(k+l-1 \mid k \parallel^2 \; +\\
&\quad + \; \epsilon \parallel u(k \mid k) - u(k-1 \mid k) \parallel^2
\end{aligned}
$$

where I is the full network (in a bigger network, I would be the part of the network corresponding to the controller adding some upstream and downstream parts) and O is the set of all origins.

The first term of the cost function considers the TTS of the full network (18 segments) and the other terms expresses the penalization on the control inputs variations considered in each controller. In Cooperative MPC, all the variables contained in I could be optimized. However, it will increase the computational time since we are solving the centralized MPC in each iteration. FC-MPC tries to solve this problem reducing the decision variables to the local variables of the controller. Therefore, the optimization variables are:

$$
u(l \parallel k) = [v_{c1,m_{cont}}(l \parallel k), v_{c2,m_{cont}}(l \parallel k), r_{m_{cont}}(l \parallel k)] \qquad (4.4)
$$

In this simulation, FC-MPC uses only four iterations as Communicative MPC.

In the majority of the sample times, it is enough to converge. It is important to note that the computational time required for each iteration rapidly decrease. It happens because we start each optimization in the optimum obtained in the previous iteration. As the optimization problem changes just a little bit when the variables are exchanged, the optimization algorithm does not need to move to a very far point.

4.2.2 Controllers summary and results

Summary of the cost functions and decision variables

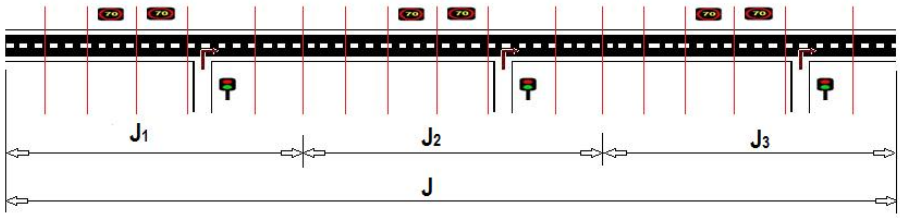

Figure 4.6: Traffic Network Simulated

The proposed controllers use different cost function and decision variables. In Table 4.2 the cost functions and decision variables used by each controller are summarized. J_i is the cost function associated to link i and J is the cost function associated to the full network (see Fig.4.6). The term $u_{i,p}$ expresses that the controllers are using the past values of the control variables of the link i but not the value that will be implemented in the current sample time.

Results

The case study used for the simulation is the same three links benchmark with 18 segments used in the previous Section 4.1 (Fig. 4.1). As can be seen on results, all the controllers reduce the TTS and, at the same time, all the controllers keep the variables inside the constraints. In the table, MCT shows the maximum computation time (in the local case, MCT have been taken from the worst cases, i.e. more restrictive cases, of the three controllers). Red is the reduction of the TTS in percentage with respect to the nonlinear case.

Analyzing the results of the local MPC, it can be seen how the communication after sample increase substantially the reduction of the TTS (from 6.5 % to 11.31 %). If the communication is made inside the sample time the reduction in the TTS increase just from 11.31 % to 13.55 %. However, if cooperation is considered

Table 4.2: Summary of local, distributed and global controllers

	Controller 1	Controller 2	Controller 3
Local MPC	$\min_{u_1} J_1(u_1)$	$\min_{u_2} J_2(u_2)$	$\min_{u_3} J_3(u_3)$
Local MPC with comm. after sample	$\min_{u_1}$ $J_1(u_{1,p}, u_{2,p}, u_{3,p})$	$\min_{u_2}$ $J_2(u_{1,p}, u_{2,p}, u_{3,p})$	$\min_{u_3}$ $J_3(u_{1,p}, u_{2,p}, u_{3,p})$
Communicative MPC	$\min_{u_1}$ $J_1(u_1, u_2, u_3)$	$\min_{u_2}$ $J_2(u_1, u_2, u_3)$	$\min_{u_3}$ $J_3(u_2, u_2, u_3)$
Cooperative MPC	$\min_{u_1}$ $J(u_1, u_2, u_3)$	$\min_{u_2}$ $J(u_1, u_2, u_3)$	$\min_{u_3}$ $J(u_1, u_2, u_3)$
Centralized MPC	$\min_{u_1,u_2,u_3}$ $J(u_1, u_2, u_3)$	$\min_{u_1,u_2,u_3}$ $J(u_1, u_2, u_3)$	$\min_{u_1,u_2,u_3}$ $J(u_1, u_2, u_3)$

Table 4.3: Numerical results

	H_c	H_p	TTS	Red (%)	MCT
Uncontrolled system	-	-	1684	0	-
Local MPC	3	3	1573.6	6.5	51.6
Local MPC with communication after sample	3	3	1493.4	11.3	29.1
Communicative MPC	3	3	1455.8	13.6	45.1
Cooperative MPC (FC-MPC)	3	3	1262	25.1	55.1
	5	7	1247.2	25.9	237.7
Centralized MPC	3	3	1252.8	25.6	231.5
	5	7	1238.8	26.4	1103.9

(using FC-MPC) the reduction achieves the 25.06 % (really close to the 25.6 % of the centralized MPC). It shows that the Nash Equilibrium is quite suboptimal in freeway traffic systems using METANET model. The reason is that a good traffic control system needs to the take into account the effects of the own traffic control system in other parts of the network. Without this consideration, the global behavior of the network may get worse by solving a traffic jam in one part of the network because it could increase the number of vehicles that arrive to a bigger traffic jam.

On the other hand, it can be seen how the FC-MPC approximates the centralized behavior in just a few iterations (four in this case) showing the Pareto-Optimum can be reached. Looking the computational time needed it is possible to see how the centralized controller cannot be implemented in real time for all sample even with a small horizons such as three. However, decentralized computations requires between a third and a fifth of the computational effort making them implementable in real time. For larger horizons (5-7), the reduction of the TTS is increased a small percentage but the computational time is highly increased.

It is important to note that the minimization of the TTS is just a criterion for the operation of the traffic system. Other objectives can be considered as the reduction of emission, the homogenization of the traffic flows or the minimization of fuel consumption. This is one of the main advantages of Model Predictive Control; the policy can be changed just modifying the cost function without changing any other part of the controller.

4.3 Conclusions

In this section, local, decentralized and centralized control techniques have been evaluated on a simulated 18 km stretch of a freeway.

The first conclusion is that a centralized controller allows to obtain much greater reductions than the fully decentralized one but requires higher computational time. In fact, the results show that a centralized traffic control would be very difficult to implement in a real (i.e. large) traffic network and that a fully decentralized controller for a traffic network is quite suboptimal (6.5 % versus 26.4 % in the reduction of the TTS). Even being a generally true conclusion that decentralized controller is suboptimal with respect to the centralized case, this section shows how far are both options in order to motivate the use of distributed solutions since the majority of the controller for ITS signal uses local techniques. An algorithm that properly uses communication and cooperation between different controllers in order to get close to the centralized behavior without increasing critically the computational effort. It has been shown in the study case considered that it is possible to increase the TTS reduction from 6.5% to 12.88% just by a single communication at each sample.

The second conclusion is that distributed MPC algorithms converge, in this case, in just a few iterations and can be computed in a fraction of the time needed by the centralized one (between a third and a fifth).

The third conclusion is that Nash Equilibrium is far away to Pareto Optimal Equilibrium in this traffic system (i.e. cooperation is a key issue). In the simulation, the TTS decreases from 11.31 % to 25.06 % thanks to cooperation.

As a general conclusion, it can be said that the proposed cooperative controller (FC-MPC) practically fulfills the design objectives; the centralized behavior is almost reached requiring between a third and fifth of the computational effort.

Chapter 5

Discrete and Hybrid MPC for Freeway Traffic

In this chapter, two hybrid Model Predictive Control (MPC) approaches for freeway traffic control are proposed considering variable speed limits (VSL) as discrete variables as in current real world implementations.

The discrete characteristics of the speed limits values and some necessary constraints for the actual operation of VSL are usually underestimated in the literature, so we propose a way to include them using a macroscopic traffic model within an MPC framework.

For obtaining discrete signals, the MPC controller has to solve a highly non-linear optimization problem, including mixed-integer variables. Since solving such a problem is complex and difficult to execute in real-time, we propose a set of methods based on the limitation of the number of feasible nodes and the use of Genetic Algorithms (GA) to obtain reasonable control actions in a limited computation time.

Section 5.1 explains the limitation of the number of feasible nodes by using the VSL implementation constraints used by all the proposed methods.

- The first class of methods (θ-exhaustive and θ-genetic discretization in Section 5.2) consists of relaxing the discrete constraints for the VSL inputs; and then, based on this continuous solution and using a genetic or an exhaustive algorithm, finding discrete solutions within a distance θ of the continuous solution that provide a good performance.

- The second class of methods (Alternating optimization in Section 5.3) splits the problem in a continuous optimization for the ramp metering signals and in a discrete optimization for speed limits. The speed limits optimization, which is much more time-consuming than the ramp metering one, is solved

by a genetic or an exhaustive algorithm in communication with a non-linear solver for the ramp metering.

The proposed methods are tested by simulation, showing not only a good performance, but also keeping the computation time reduced. As in previous chapters, the methods we propose are independent of the traffic model used. Therefore, it would be interesting to apply them using other macroscopic traffic models which are capable of including the effect of VSL in their formulation (like some versions of the Cell Transmission Model [15]).

Parts of this chapter are published in [33, 31].

5.1 General scheme for discrete Model Predictive Control for VSL

5.1.1 Introduction

In most of the works about VSL computed with MPC, the VSL signals are assumed to have continuous values, meaning that the real VSL panels implemented in the network should display those values to the driver [72, 28, 37, 109, 36].

However, in the real implementations of VSL panels, the displayed signals are only allowed to take a limited set of discrete values. For example, in the Dutch freeway A12 the signals of the panels are only allowed to take values in the set $\{60, 80, 100\}$km/h [41]. Moreover, for safety reasons, some extra constraints should be considered like a limited variation over time (for each panel) and a limited variation over space (consecutive panels) so to avoid drastic changes in speed [43, 10].

The following works have proposed ways to deal explicitly with discrete VSL. In [42], it is proposed a discrete VSL controller based on shock wave theory. In [21], a traffic model with variable length segments is used to compute a simple best-effort controller that reduces congestion considering VSL signals that can only be decreased or increased by steps of 10 km/h. Both controllers, [42] and [21], use simple control laws that are not explicitly designed to optimize a performance index of the network.

A few studies consider the discrete characteristics of the speed limits values within an MPC framework. In [43, 10] the VSL are discretized (by rounding, ceiling, or flooring) after computing them in a continuous way. These papers conclude that the performance of the discretized speed limits was comparable with the continuous case. However, those results depend on the network configuration and the demand conditions. As a matter of fact, in our case study we found some important loss of performance due to the discretization.

Therefore, in this work, we consider explicitly the effects of using discrete signals for the VSL panels in an MPC framework. Subsequently, we propose new efficient algorithms for the computation of discrete VSL signals together with continuous ramp metering rates.

5.1.2 VSL implementation constraints

In the majority of the references about VSL computed by MPC, the variable speed limits are considered to be implemented in the network without any operational constraints. Hereafter, we will assume that the MPC controllers will have to consider the following safety and operational constraints for the VSL:

Temporal constrained case:

In real implementations, the VSL cannot change abruptly due to driver safety and comfort. Therefore, we just allow the VSL to change γ_i km/h as maximum in a controller sample step including the following constraint as in [43, 10]:

$$|V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{c},i}(k+\ell-1)| \leq \gamma_i \tag{5.1}$$
for each segment i with a VSL and for $\ell = 0, 1, ...N_{\mathrm{u}} - 1$.

Hereafter, the MPC proposed with temporally constrained VSL will be denoted by T-MPC. Unlike the state contraints, the constraint (5.1) is treated as hard constraint.

Spatial and temporal constrained case:

In addition, in real implementations it is necessary to limit the difference between the VSL values of two adjacent segments (ζ_i). This can be done by including the following hard constraint as in [43, 10]:

$$|V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{c},i}(k+\ell-1)| \leq \gamma_i \tag{5.2}$$
for each segment i with a VSL and for $\ell = 0, 1, ...N_{\mathrm{u}} - 1$.
$|V_{\mathrm{c},j+1}(k+\ell) - V_{\mathrm{c},j}(k+\ell)| \leq \zeta_j$
for j and $j+1$ corresponding to adjacent VSL
and for $\ell = 0, 1, ...N_{\mathrm{u}} - 1$.

Hereafter, the MPC proposed with spatially and temporally constrained VSL will be denoted by ST-MPC.

Discrete case:

Even with the previously proposed ST-MPC, the solution obtained cannot be implemented in practice since this solution is continuous and the freeway signs are only allowed to show a limited set $\mathbb{S}$ of discrete speed limits.

An easy but suboptimal method is to approximate the continuous optimal control inputs by allowed VSL values (i.e. to apply a discretization process). The most used way to do this discretization is by rounding the available continuous solution to the closest discrete value of the VSL [43, 10]:

$$V_{\mathrm{c},i}(k) = \arg\min_{s\in\mathbb{S}}(|V_{\mathrm{cST},i}(k) - s|) \tag{5.3}$$

where $V_{\mathrm{cST},i}(k)$ is the VSL of segment i computed by ST-MPC controller explained in the previously.

Other easy ways to do the discretization are by ceiling (5.4), and by flooring (5.5):

$$V_{\mathrm{c},i}(k) = \arg\min_{s\in\mathbb{S}}(|V_{\mathrm{cST},i}(k) - s| \text{ subject to: } s \geq V_{\mathrm{cST},i}(k)) \tag{5.4}$$

$$V_{\mathrm{c},i}(k) = \arg\min_{s\in\mathbb{S}}(|V_{\mathrm{cST},i}(k) - s| \text{ subject to: } s \leq V_{\mathrm{cST},i}(k)) \tag{5.5}$$

In [37], it is reported that the discretization made by ceiling yields a slightly better performance than the rounding and the flooring case.

In the following sections alternative methods for the discretization step will be proposed in order to decrease the loss of performance due to the quantization.

5.1.3 Discrete MPC for Variable Speed Limits

Problem Definition

The implementation issues explained in the previous section substantially decrease the performance of the controlled system as will be shown in the simulation results.

In order to reduce this loss of performance, the discrete characteristic of the VSL values should be directly considered during the optimization by solving or approximating the following hybrid optimization problem:

$$\begin{aligned} \min_{r_{\mathrm{t}}(k), V_{\mathrm{c,t}}(k)} \quad & \bar{J}(k) \\ \text{s.t:} \quad & 0 \leq r(k+\ell) \leq 1 \\ & \mathrm{VSL}_{\min} \leq V_{\mathrm{c}}(k+\ell) \leq \mathrm{VSL}_{\max} \\ & \text{for } \ell = 0, 1, ..., N_{\mathrm{u}} - 1 \\ & \text{with } r_{\mathrm{t}}(k) \in \mathbb{R}^{\mathbb{N}_{\mathrm{u}}\cdot\mathbb{N}_{\mathrm{r}}} \text{ and } V_{\mathrm{c},t}(k) \in \mathbb{S}^{\mathbb{N}_{\mathrm{u}}\cdot\mathbb{N}_{\mathrm{VSL}}} \end{aligned} \tag{5.6}$$

where $r_{\mathrm{t}}(k) = [r(k)^T, r(k+1)^T ..., r(k+N_{\mathrm{u}}-1)^T]^T$, and $r(k) = [r_{j_1}(k), ..., r_{j_{N_{\mathrm{r}}}(k)}]^T$ with $\mathbb{I}_{\mathrm{r}} = \{j_1, j_2, ..., j_{N_{\mathrm{r}}}\}$ the set of segments with a metered on-ramps and N_{r} the number of metered on-ramps.

Equivalently, $V_{\mathrm{c},t}(k)$ is the discrete control input profile: $V_{\mathrm{c,t}}(k) = [V_{\mathrm{c}}(k)^T, ..., V_{\mathrm{c}}(k+N_{\mathrm{u}}-1)^T]^T$, $V_{\mathrm{c}}(k) = [V_{\mathrm{c},m_1}(k), ..., V_{\mathrm{c},m_{N_{\mathrm{VSL}}}(k)}]^T$, and $\mathbb{I}_{\mathrm{VSL}} = \{m_1, m_2, ..., m_{N_{\mathrm{r}}}\}$ the set of segments with a VSL and N_{VSL} the number of VSL.

The proposed optimization problem (5.6) is a complex to solve mixed-integer optimization problem with discrete and continuous decision variables so it is difficult to execute in real time. Most of the MPC approaches for hybrid systems do not have a standard strategy to relax a non-convex problem in order to obtain a good solution in a reasonable amount of computation time.

To properly consider processes with discrete and continuous variables (hybrid systems) in an MPC formulation, hybrid predictive control techniques have been developed [93, 7, 6]. For this formulation, the main difficulty is the computation time needed to solve the optimization problem because we are dealing with mixed integer nonlinear programming (MINLP).

In cases when the problem can be recast into a MILP, well-known optimization methods are efficient and many software/toolboxes are available to solve them [25].

However, in the case of traffic systems, the problem is intrinsically non-linear and any simplification of the model may lead to non-acceptable predictions, which are incapable of incorporating the real behavior of traffic. As a way to deal with complexity, and to include the discrete characteristic explicitly in the solution, we propose a set of methods based on the limitation of the number of feasible nodes and the use of genetic algorithms.

Problem Relaxation

Limiting the number of feasible nodes in the mixed integer optimization problem is a strategy that has been used before in the context of MPC for sewer networks [46], as a way to reduce the combinatorial explosion related to the search tree of binary solutions at the expense of a suboptimal solution.

In this work, a limited number of feasible nodes limitation is obtained by using the spatial and temporal implementation constraints (5.2) previously explained.

The feasible VSL set can be represented by a search tree where the unfeasible VSL profiles have already been removed. Fig. 5.1 shows an example of a search tree with two VSL, $V_{\mathrm{c}}(k) = [40, 50]^T$, $\mathbb{S} = \{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$, $N_{\mathrm{u}} = 2$, $\zeta_i = 10$, and $\gamma_i = 10$.

In this case, the temporal constraints reduce the number of possible combinations of discrete VSL from 1331 to only 81. The spatial constraint reduces again this

number from 81 to 38 profiles of discrete VSL.

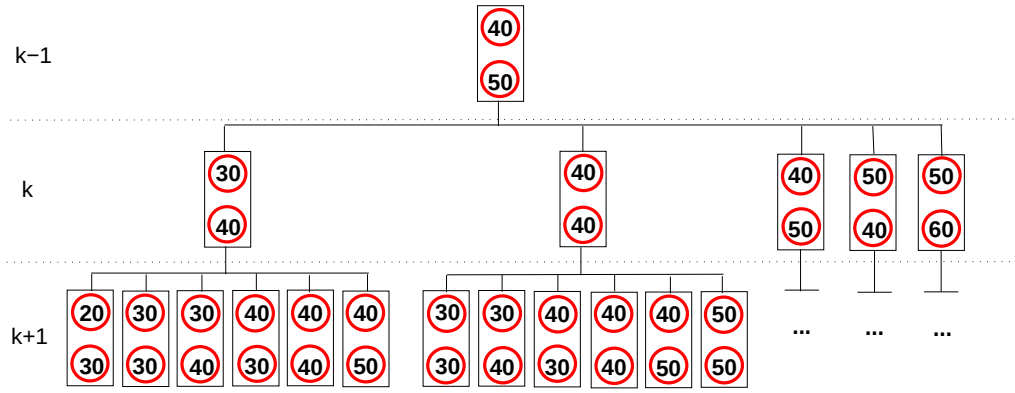

Figure 5.1: Search tree example with $N_u = 2$ reduced by the spatial and temporal constraints.

For larger horizons, the number of feasible VSL profiles should increase exponentially. For example, in the simulation made in this section (with $N_\mathrm{u} = 4$ and two VSL) the search tree is initially composed of $2.14 \cdot 10^8$ possible VSL profiles. However, this number is reduced from $2.14 \cdot 10^8$ to 6561 VSL profiles by the temporal constraint and from 6561 to an average number of 1998 VSL profiles by the spatial constraint.

In the following sections 5.2 and 5.3, some methods will be proposed in order to solve optimization (5.6) with constraints (5.2) using the search tree of Fig. 5.1.

5.2 MPC for the discretization of continuous VSL

5.2.1 θ-Genetic and θ-Exhaustive optimizations

The first proposed methods makes use of the continuous solution of the ST-MPC problem in order to relax problem (5.6) with constraints (5.2). The idea is to run a discrete optimization after the continuous optimization assuming that the optimal discrete solution is close to the continuous one. This is not generally true and so, in some cases, could cause loss of optimality.

Firstly, the VSL search tree explained in the previous section is reduced again by including a new constraint in the optimization problem making use of the ST-MPC solution for the VSL ($V_{\mathrm{cST},i}(k)$):

$$|V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{cST},i}(k+\ell)| \leq \theta \quad \text{for } \ell = 0, 1, \ldots N_{\mathrm{u}} - 1 \text{ and for } i \in \mathbb{I}_{\mathrm{VSL}} \tag{5.7}$$

where θ is a tuning parameter.

Therefore, the feasible VSL set can be represented by a smaller search tree than in the previous subsection (Fig. 5.1). In Fig. 5.2 an example of this search tree with $V_{\mathrm{cST}}(k+1) = [43, 53]^T$, $V_{\mathrm{cST}}(k+1) = [52, 61]^T$ and $\theta = 10$ is shown. In this case, the constraints reduce the number possible combinations of discrete VSL from 1331 to only 6 (versus 38 without using constraint (5.7)).

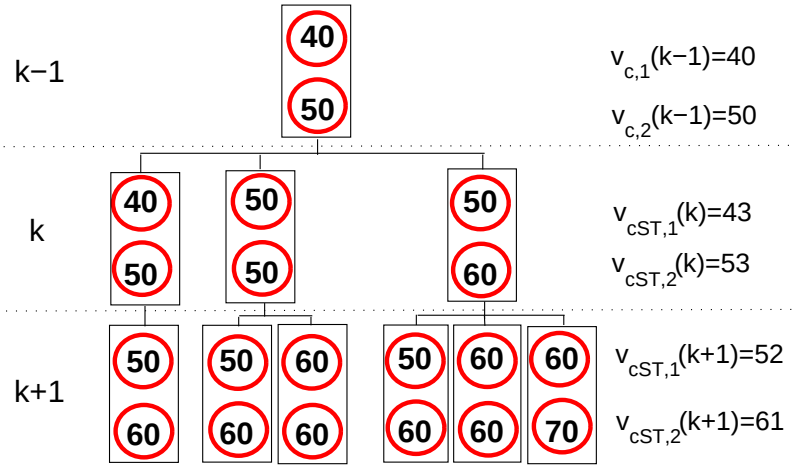

Figure 5.2: Search tree of Fig. 5.1 using the ST-MPC solution

For the simulation done in this section (with $N_{\mathrm{u}} = 4$ and $\theta = 10$) the search tree is reduced from $2.14 \cdot 10^8$ points to an average number of 114 points (versus 1998 without constraint (5.7)). If θ is chosen to be 14, the average number of points in

the reduced tree is 971 because the larger the θ, the more feasible solutions there are to explore.

Moreover, making use of the ST-MPC solution for the ramp metering rates $r_{\mathrm{cST}}(k)$, the optimization can be reduced to an optimization with only discrete variables. Therefore, in order to obtain the discrete speed limits using these assumptions (given ramp metering rate and constraint (5.7)), the following optimization problem with only discrete variables has to be solved:

$$\begin{aligned} \min_{V_{\mathrm{c,t}}(k)} \quad & \bar{J}(k) \\ \text{s.t:} \quad & |V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{c},i}(k+\ell-1)| \leq \gamma_i, \\ & |V_{\mathrm{c},j+1}(k+\ell) - V_{\mathrm{c},j}(k+\ell)| \leq \zeta_j, \\ & |V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{cST},i}(k+\ell)| \leq \theta \quad \text{and} \\ & \mathrm{VSL}_{\min} \leq V_{\mathrm{c},i}(k+\ell) \leq \mathrm{VSL}_{\max} \\ & \text{for } \ell = 0, 1, ...N_{\mathrm{u}} - 1, \text{ for } i \in \mathbb{I}_{\mathrm{VSL}}, \\ & \text{and for } j \text{ and } j+1 \in \mathbb{I}_{\mathrm{VSL}} \text{ with } V_{\mathrm{c},i}(k+\ell) \in \mathbb{S} \end{aligned} \tag{5.8}$$

θ-Exhaustive optimization

As we are dealing with non-convex integer optimization, the only way to obtain the global optimum of (5.8) is to evaluate the cost function for all the feasible points in the reduced search tree. The main problem is the computation time needed for the evaluation of such a large number of possible combinations of discrete VSL. Therefore, this solution is just applicable for relatively small networks and horizons or in cases where an off-line solution is useful. In order to be able to solve the problem for large networks within the limited computation time available, a genetic algorithm is proposed in the following section.

θ-Genetic optimization

To optimize the discrete control action sequences $V_{\mathrm{c},i}(k)$ (i.e. to solve problem (5.8)) within the limited computation time, the use of Genetic Algorithms (GA) [64] is proposed. In cases where the optimization problem is highly nonlinear, it has been reported that GA can efficiently cope with it, particularly for mixed-integer nonlinear problems [64]. In the context of MPC, in cases when the model or the objective function are nonlinear, the use of GA is fully justified as the computation times can be easily controlled and good solutions can be obtained within a fixed sampling time [92, 100, 74]. One of the main characteristics of this method is that it is gradient-free and that it is possible to limit the number of objective function evaluations. So, in order to find a good discrete VSL within the limited time available, a GA will be used to solve optimization problem (5.8).

A candidate control sequence solution in the genetic algorithm is called an individual, and each individual has a fixed number of genes (emulating a chromosome). Each gene, in the context of systems with discrete inputs, will represent a possible input during the control horizon. For MPC, the individual can be seen then as a candidate control action sequence:

$$\begin{aligned}\text{Individual}_1 = [&V_{\text{c},m_1}(k), ..., V_{\text{c},,m_1}(k+N_\text{u}-1), ..., \\ &V_{\text{c},m_{N_\text{VSL}}}(k), ..., V_{\text{c},m_{N_\text{VSL}}}(k+N_\text{u}-1)]\end{aligned} \tag{5.9}$$

where $\{m_1, m_2, ..., m_{N_\text{r}}\}$ is the set of segments with a VSL as explained previously.

In GA the idea is to find the fittest individuals (solutions with best objective function values) within a generation, to apply genetic operators for the recombination of those individuals, and to generate a good offspring [64]. In this work, for the selection, a roulette method is applied, giving the best individuals more chances to be selected for recombination. For the recombination, two fundamental operators are used: crossover and mutation. For the crossover, portions of the chromosomes of two individuals are exchanged with a given probability p_c; and the mutation operator modifies each gene randomly with a given probability p_m. This is just an specific implementation and alternative genetic methods could be used.

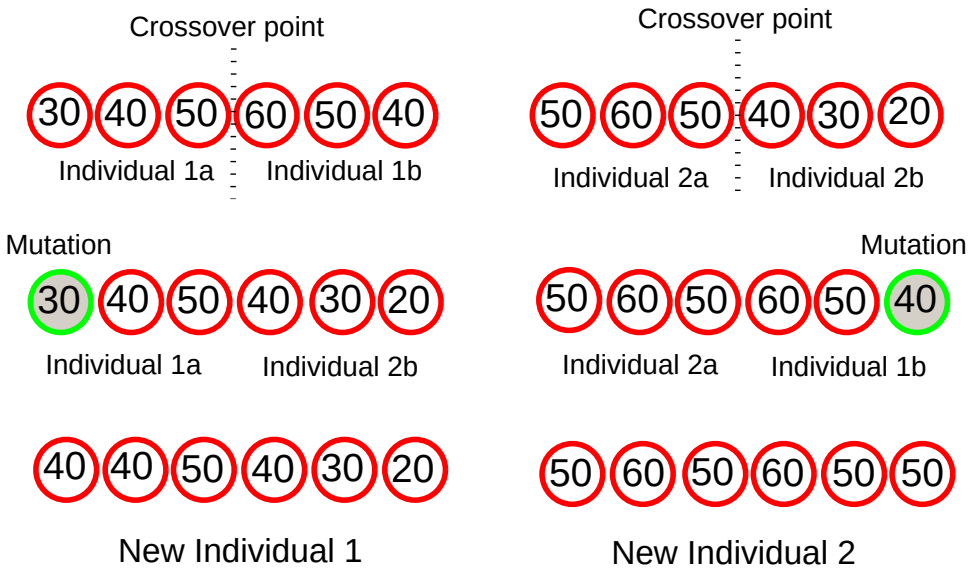

Figure 5.3: The basic operators in a GA-based control strategy for VSL panels.

Fig. 5.3 shows an example of the recombination steps for two individuals assuming that we have two VSLs and the control horizon is 3. First two individuals are

selected for recombination with a higher probability if their objective function is lower. Then, a random crossover point is selected, and two new individuals are generated. Then the mutation operator selects randomly a control to modify and its values are changed. Finally two new individuals for a new generation are obtained.

We have to point out that due to the limited time reaching the global optimum is not guaranteed. Since for traffic control the optimization is a complex mixed-integer and nonlinear problem, using the GA optimization is justified. Many different approaches and adaptations of genetic algorithms have been proposed in the literature in order to deal with many issues like constraint handling, diversity of the solutions, combination with classical optimization methods to assure a local convergence, etc [66, 4, 14, 52]. For constrained optimization, one of the most used is the GENOCOP algorithm proposed by Michakewicz [67]. The algorithm we used for the traffic application is the simplest one [64] and in the case a solution does not satisfy a constraint, we penalize with a high objective function value. Part of the further research is to implement different adaptations of GA, and other algorithms. In the simulation results we will compare GA with exhaustive enumeration, so we can see how far the solutions of GA are from the optimal solution.

5.2.2 Case study

In order to simulate the analyzed controllers, the 1 link benchmark network used in previous chapter 3.1 (Fig. 3.4) has been used. The set of allowed VSL is supposed to be $\mathbb{S} = \{20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120\}$ and the implementation constraints parameters are $\zeta_i = 10$, and $\gamma_i = 10$.

Continuous MPC controllers results

The graphical results of the closed-loop simulations for the VSL computed by each continuous MPC controller and the rounding discretization of ST-MPC can be seen in Fig. 5.4.

It can be seen in the figure that VSL_1 of unconstrained MPC suddenly changes from 76 km/h to 32 km/h at minute 12. However, this change is done at a rate of 10 km/h for T-MPC. Moreover, the absolute difference between VSL_1 and VSL_2 (i.e. $|\text{VSL}_1 - \text{VSL}_2|$) keeps between 40 km/h and 60 km/h for unconstrained MPC and T-MPC during a long period of time. When this difference is constrained in ST-MPC, the value of VSL_2 is strongly decreased in order to obtain a difference of 10 km/h with VSL_1 (which is also slightly increased).

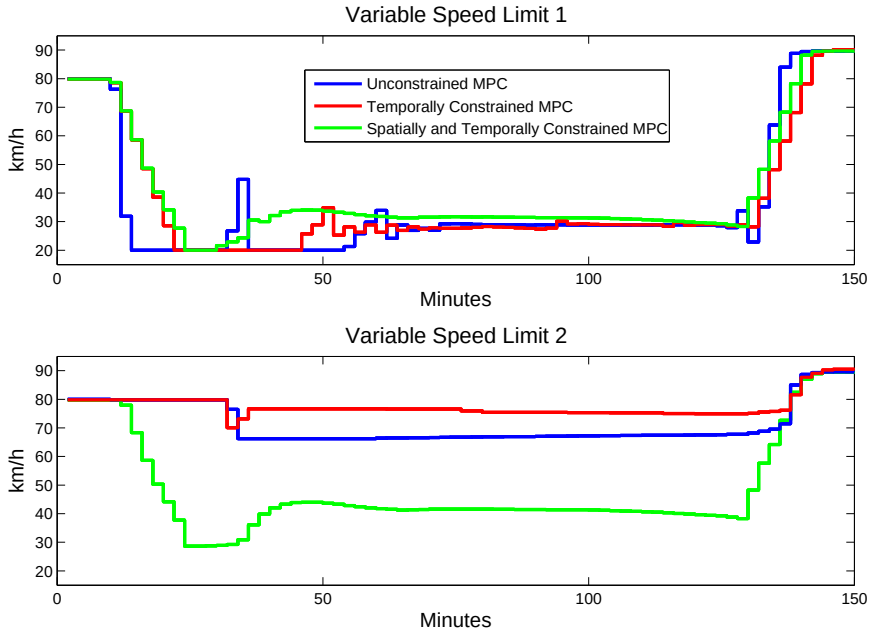

Figure 5.4: VSL of segment 3 and 4 for continuous MPC

In the numerical results on Table 5.1 can be seen how ST-MPC (the most realistic one) results on a TTS reduction that is the 36.7% lower than the TTS reduction corresponding to unconstrained MPC (12.47% versus 8.14%) concluding than the spatial and temporal constraints substantially decrease the potential controller performance. However, in real implementations, the lack of safety constraints may increase the actual TTS reduction due to sudden braking and accidents.

Table 5.1: Continuous MPC Performances

	TTS Reduction (%)
Uncontrolled System	0 %
Unconstrained MPC	12.87 %
T-MPC	10.48 %
ST-MPC	8.14 %

Rounding, Flooring and Ceiling

The closed loop VSL discretization by rounding, flooring, and ceiling, and the continuous ST-MPC solution are shown in Fig. 5.5. It can be seen than different VSL choices during a few sample step during the beginning of the congestion can cause a different simulation during a long period of time (the ceiling VSL are 20 km/h and 30 km/h over the rounding and ceiling VSL during more than 90 minutes).

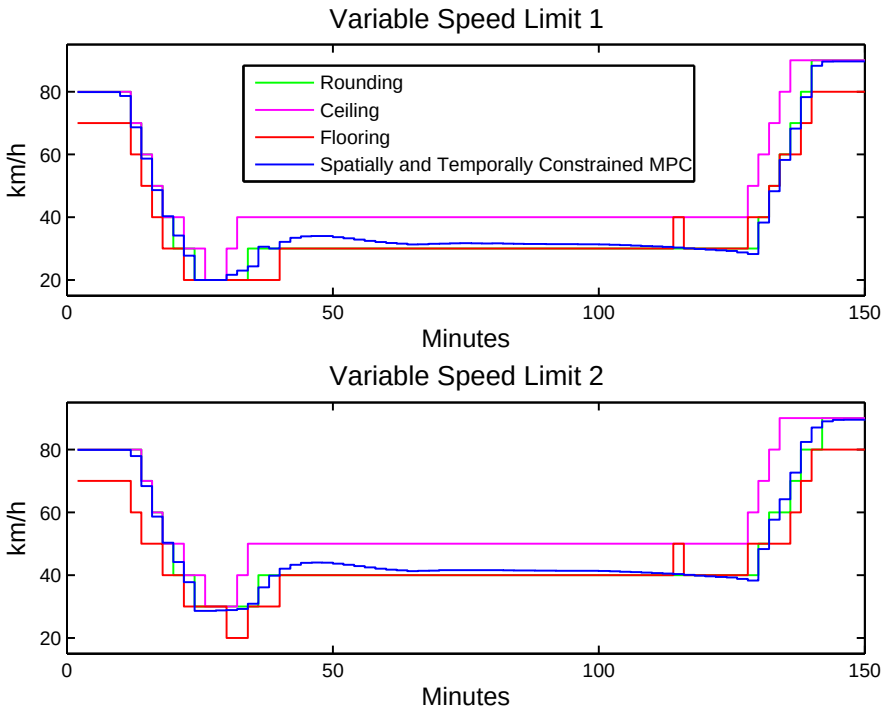

Figure 5.5: Rounding, Ceiling, Flooring and continuous VSL (ST-MPC) of segment 3 and 4

The numerical results of the rounding, ceiling, and flooring cases can be seen in Table 5.2.

In this simulation, the three options give a similar performance with a slight advantage for the rounding case. However, the TTS reduction for the rounding case is 46% lower than the TTS reduction corresponding to ST-MPC (8.14% versus 4.40%). This shows that the supposition done in many previous papers about the continuous implementation of the VSL can entail a large loss of performance for the controlled system for some networks and some traffic conditions.

Table 5.2: Discretized MPC Performances

	TTS Reduction (%)
ST-MPC	8.14 %
Rounding discretization	4.40 %
Ceiling discretization	4.37 %
Flooring discretization	4.22 %

θ-Genetic and θ-Exhaustive Optimizations

The graphical results are shown in Fig.5.6. It can be seen than the solutions of both the genetic and the exhaustive optimizations keep around the continuous solution during the whole simulation. During the time that the continuous solution stays relatively constant, the discrete solutions approximate the ST-MPC switching between two discrete values in the same way as in a pulse-width modulation.

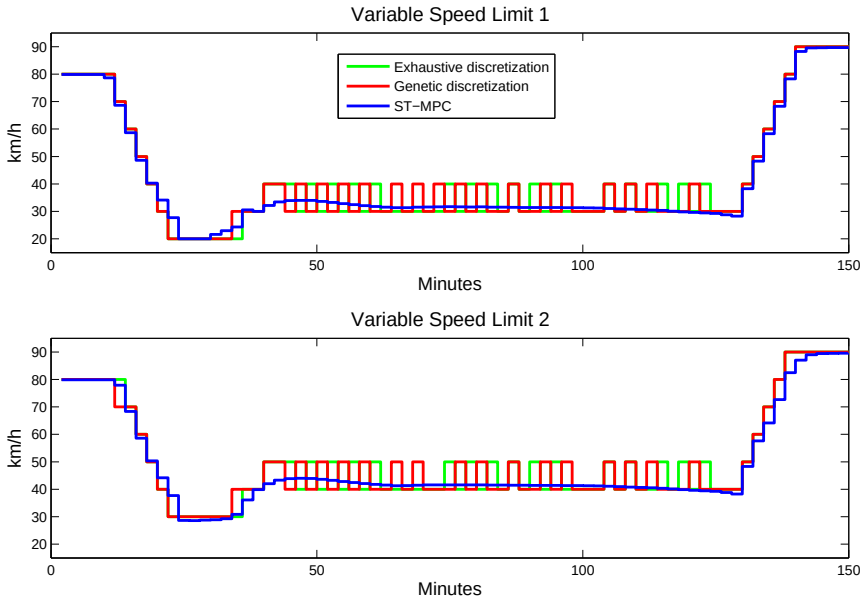

Figure 5.6: Exhaustive and genetic optimizations

The numerical results of the VSL discretization of ST-MPC by θ-Genetic and θ-Exhaustive optimization can be seen in Table 5.3. It can be seen that values of θ higher than 10 do not bring much increase in the performance of the controlled system as can be seen in TTS reduction of the case with $\theta = 18$ and $\theta = 25$. More

concretely, the θ-exhaustive optimization with $\theta = 10$ is only 1.6% worse than for $\theta = 25$. However, the computation time is increased from $1.85s$ to $26.70s$. Therefore, we decided to use a value of $\theta = 10$ for the θ-Genetic optimization algorithm.

Table 5.3: θ-Genetic and θ-Exhaustive performances

	θ	TTS Reduction	Optimization time
θ-Exhaustive opt.	10	7.22%	1.85 s
θ-Exhaustive opt.	14	7.24%	14.12 s
θ-Exhaustive opt.	18	7.25%	16.26 s
θ-Exhaustive opt.	25	7.34%	26.70 s
θ-Genetic opt.	10	6.98%	0.4 s

After a tuning process (equivalent to the one shown in the following subsection for the alternating genetic optimization), 20 individuals with 100 genes were chosen for the GA. A crossover parameter of 0.8 and a mutation parameter of 0.01 are used. It can be seen than the θ-Genetic optimization gives a solution that is close to the θ-Exhaustive one (6.98% versus 7.22%) but with a lower computation times (0.4 s versus 1.85 s).

5.3 Hybrid Model Predictive Control for freeway traffic with ramp metering and variable speed limits

5.3.1 Alternating optimization

In the previous section, the continuous VSL given by a MPC controller have been used for the computation of the discrete ones. However, these discrete VSL could be computed directly (i.e. without using any continuous solution) saving the computation time needed for the continuous optimization (i.e. to solve optimization (6.2) without using the ST-MPC solution).

To solve the mixed-integer optimization problem with discrete and continuous decision variables, we decided to compute the ramp metering and the VSL iteratively in order to decompose the problem in one continuous optimization problem (5.10) and one discrete optimization problem (5.11).

For large networks, where a distributed algorithm is required [29], the iterations between the continuous ramp metering problem and the discrete VSL problem may use information from neighbors controllers (corresponding to nearby regions of the freeway) in order to run a distributed optimization which tries to approach the global minimum using a equivalent procedure that the one used in chapter 4.2 for continuous controllers.

Ramp metering optimization:

The ramp metering optimization is equivalent to the continuous optimization explained in 3.1 but only considering ramp metering rates as decision variables.

$$\begin{aligned} \min_{r_{\mathrm{t}}(k)} \quad & \bar{J}(k) \\ \text{s.t:} \quad & 0 \leq r_i(k+\ell) \leq 1 \\ & \text{for } i \in \mathbb{I}_{\mathrm{r}} \text{ and for } \ell = 0, 1, ...N_{\mathrm{u}} - 1 \\ & \text{with } r_t(k) \in \mathbb{R}^{\mathrm{N_u} \cdot \mathrm{N_r}} \end{aligned} \tag{5.10}$$

where the VSL profile used $V^*_{\mathrm{c},t}(k)$ for the ramp metering optimization is the profile proposed in the previous iteration or, for the first iteration, the profile proposed in the previous sample time shifted by one sample.

This continuous ramp metering optimization problem will be solved using an SQP algorithm. The computation time needed is much lower than the computation time needed for the general continuous optimization problem because it does not include the VSL signals as decision variables.

VSL optimization:

The VSL optimization (5.11) is equivalent to the optimization done for the previous section (5.8) but, in this case, we do not have any continuous MPC solution for the VSL that can be used for the reduction of the search space of the problem.

$$\begin{aligned} \min_{V_{\mathrm{c,t}}(k)} \quad & \bar{J}(k) \\ \text{s.t:} \quad & |V_{\mathrm{c},i}(k+\ell) - V_{\mathrm{c},i}(k+\ell-1)| \leq \gamma_i \\ & |V_{\mathrm{c},j+1}(k+\ell) - V_{\mathrm{c},j}(k+\ell)| \leq \zeta_j \\ & \mathrm{VSL}_{\min} \leq V_{\mathrm{c},i}(k+\ell) \leq \mathrm{VSL}_{\max} \\ & \text{for } \ell = 0, 1, ...N_{\mathrm{u}} - 1, \text{ for } i \in \mathbb{I}_{\mathrm{VSL}}, \\ & \text{and for } j \text{ and } j+1 \in \mathbb{I}_{\mathrm{VSL}} \text{ with } V_{\mathrm{c},i}(k+\ell) \in \mathbb{S} \end{aligned} \tag{5.11}$$

where the ramp profile used is the one proposed in the previous iteration $r_t^*(k)$.

Subsequently, exhaustive and genetic algorithms are proposed to solve the resulting discrete optimization problem (5.11), as previously done for the problem (5.8).

Exhaustive optimization: In the same way that it was explained in previous subsections, the easiest way to solve problem (5.11) is to evaluate the cost function for all the feasible points. Again, the problem is the computation time needed for the evaluation of such a large number of possible combinations of discrete VSL. Therefore, this solution is just applicable for relatively small networks and horizons. For larger networks and horizons, genetic algorithms may be used.

Genetic optimization: In the same way than in subsection 5.2.1, a genetic algorithm is proposed to solve the discrete optimization. The formulation is the same as in the previous subsection but without using the constraint related with a solution within a distance θ from the continuous one. Moreover, it will be convenient to use a different GA parameters tuning due the greater number of feasible solutions and the larger computation times available.

5.3.2 Case study

In order to simulate the analyzed controllers, the same network and traffic conditions than in the previous subsection 5.2 have been used.

The numerical results of the alternating exhaustive optimization for different numbers of iterations are shown in Table 5.4.

It can be seen that the iterations almost do not increase the TTS reduction (7.4229% without iterations versus 7.4252% with three iterations for the ramp metering and the VSL). In fact, in this simulation the algorithm converges in one

Table 5.4: Tuning of Alternating Exhaustive Optimization

Ramp Iterations	VSL Iterations	TTS Reduction (%)
1	1	7.4229
2	1	7.4252
3	3	7.4252

iteration of the ramp metering for 72 out the 75 sample times. Therefore, we decided to use only one iteration for the exhaustive and genetic optimizations (i.e. ramp metering optimization and VSL optimization will be run just one time each sample time).

As the algorithm converges within three iterations for all the sample times and assuming that the ramp metering optimization is using enough initial points to avoid local minima, it can be said that 7.4252% is likely to be the maximum TTS reduction that can be achieved with the used horizons and cost function parameters.

It will be difficult to implement the exhaustive algorithm in real time for large networks due the computation times required. However, results show that, in this case, it is possible to compute the control inputs in real time. For the exhaustive optimization with just one iteration, the mean computation time for one sample step is 31.04 s with a maximum of 40.64 s which is less than the 2 minutes controller sample time. Using 3 iterations for the ramp metering and for the VSL, the mean computation time is 95.43 s with a maximum of 125.78 s.

Some of the numerical results obtained during the tuning of the Alternating Genetic Optimization are shown on Table 5.5:

Table 5.5: Tunning of Alternating Genetic Optimization

Ind	Gen	Muta	Cros	TR_{mean}	TR_{sd}	CT_{mean}	CT_{sd}
20	100	0.01	0.8	6.99 %	0.29	6.14 s	1.71
20	100	0.001	0.8	5.81 %	1.30	7.22 s	1.81
40	200	0.01	0.8	7.37 %	0.14	12.55 s	2.65
40	200	0.001	0.8	7.13 %	0.38	9.85 s	1.93
60	100	0.01	0.6	7.27 %	0.178	8.45 s	1.01
30	100	0.01	0.6	7.11 %	0.36	6.26 s	1.91

In the Table, Ind, Gen are the number of individuals and genes in the GA, Muta and Cros are the Mutation and Crossover parameter of the GA. CT_{mean} is the Mean Computation Time, and CT_{sd} is the Standard Deviation of the Computational Time. In order to evaluate the performance of the genetic algorithm, the simulations have been run 10 times and the mean TTS reduction (TR_{mean}) and the standard deviation of this reduction (TR_{sd}) have been used as performance measures.

The controllers show a good behavior for the majority of the parameters sets evaluated, usually with more than a 7% TTS reduction (close to the maximum of 7.43 %). Moreover, the standard deviation of the TTS reduction is relatively small showing a good performance even in the worst cases.

It can also be seen that the trade-off between computation time and performance can be easily done by changing the number of individuals and genes.

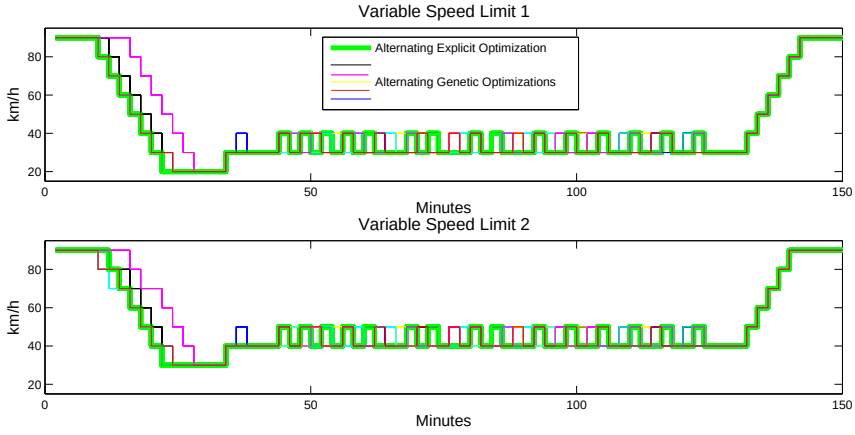

Figure 5.7: Exhaustive optimization versus genetic optimizations

The VSL obtained by Alternating Exhaustive Optimization and six examples of the VSL obtained by using Alternating Genetic Optimization are shown on Fig.5.7. It can be seen than for the majority of the profiles obtained by the GA, the VSL stay very close to the exhaustive solutions.

5.4 Summary of results and conclusions

5.4.1 Summary of results

A brief summary of the numerical results and the computation times can be seen in Table 5.6.

Table 5.6: Continuous, discrete and hybrid controllers summary

	Imp	CT_{mean}	CT_{sd}	CT_{max}	TR_{mean}	TR_{sd}
U-MPC	N	20.74 s	13.06	79.67 s	2.87%	0
T-MPC	N	22.39 s	10.09	49.34 s	0.48%	0
ST-MPC	N	30.02 s	13.67	88.76 s	8.14%	0
Rounding Discretization	Y	31.39 s	16.34	78.90 s	4.4%	0
θ-Exhaustive Optimization	Y	32.62 s	18.23	73.33 s	7.22%	0
θ-Genetic Optimization	Y	31.91 s	17.11	64.21 s	6.98%	0.03
Alternating Exhaustive Optimization	Y	31.04 s	4.13	40.64 s	7.43%	0
Alternating Genetic Optimization (40/200/0.01/0.8)	Y	12.55 s	2.65	27.55 s	7.37%	0.14
Alternating Genetic Optimization (30/100/0.01/0.6)	Y	6.26 s	1.91	13.61 s	7.11%	0.36

In the table, the row corresponding to Imp shows whether the corresponding controller is implementable in terms of respect of the safety constraints and belonging to the discrete set of VSL. CT_{max} is the maximum computation time for all the sample times. It can be seen that the computation times for θ-Exhaustive and θ-Genetic optimizations are of the same order of magnitude as those of the continuous cases because the discrete optimization is faster than the continuous optimization. But the TTS associated to both θ-Exhaustive and θ-Genetic optimizations is significantly larger than for the rounding case (7.24 % and 7.22 % versus 4.40 %). Therefore, it can be concluded that if the algorithm is spending

much computational load in order to computing a good solution for the continuous optimization, it is necessary to run a proper algorithm to obtain a good VSL signals. Otherwise, if we just use a rounding (or ceiling or flooring), we would be wasting the computational effort spent in the continuous optimization.

From Table 5.6 can be also concluded that both Alternating Optimizations have quite lower computation times than the other algorithms and a slightly greater TTS reduction. Especially important is the difference in the Max CT and in the standard deviation of these computation times (SDCT). This means that the computation times for the genetic and exhaustive optimization are more stable and, therefore, the inputs can be more likely computed during the available time.

Comparing the previously proposed solutions for real implementations (rounding, ceiling, flooring) with the genetic optimization it can be seen that the loss of performance corresponding to the rounding is much higher that the corresponding to the suboptimality of the genetic optimization.

5.4.2 Observed advantages of genetic optimization for freeway traffic control

Advantages

- In many cases in freeway traffic control with METANET, continuous MPC based on an SQP optimization algorithm shows a poor performance if a wrong set of initial points is used for the optimization. For example, in this section it was used an evaluation procedure (with 252 points conveniently chosen) for the MPC constrained cases in order to choose a proper initial point for the optimization in a reasonable time. In general, it would be necessary to run many optimizations with different initial points. To have to run multiple optimizations makes it impossible to compute the control inputs in real time. The genetic optimization is able to compute a reasonably good solution faster than the continuous optimization, allowing obtaining in real time a suboptimal solution in cases where the continuous optimization is unable due to the high number of initial points required.

- For the Alternating Optimizations the computation times for the different sample times are lower and with less variation than for typical continuous MPC approaches as can be seen in Table 5.6 (the standard deviation and the mean of the computational times are much lower than for the continuous cases). This allows making the most of the time available ensuring that there will be a suboptimal implementable solution at the end of the time step.

- The genetic and exhaustive optimization substantially improve the performance of the rounding discretization (7.24 % and 6.98 % versus 4.20 %), which is the best implementable solution without running a discrete optimization. Moreover, these algorithms can be run after the continuous optimization in only 1.86 s and

0.4 s, respectively. Both solutions are close to the best implementable solution (7.43%).

Disadvantages

- Using genetic algorithms it is impossible to ensure reaching the global optimum within a limited sampling time. Moreover, stability or other robust properties can be analyzed just numerically. However, using continuous optimization it is also impossible in practice to prove stability or to ensure global optimality when solving highly non-linear non-convex optimization problems.

- GA perform in a different way any time they are executed so the proposed VSL profiles will randomly show slight changes for one simulation to another. However, in this benchmark the observed standard deviation of the TTS reduction is low.

5.4.3 Conclusions

This chapter has proposed hybrid Model Predictive Control approaches for freeway traffic control considering Variable Speed Limits as discrete variables as in current real world implementations.

Firstly, this chapter has analyzed the effect of converting the continuous unconstrained VSL signal to an implementable VSL signal (i.e. a discrete value respecting the safety constraints). In the simulated network, the discrete constrained performance is substantially reduced with respect to the ideal case. Therefore, it can be concluded that the assumption about the continuous implementation of the VSL entails in some cases a large loss of performance for the controlled system. Since the majority of the literature makes this supposition, we have proposed two kind of methods to reduce part of this loss of performance.

The first class of proposed methods discretize the VSL signals solving a discrete optimization problem with a search space limited to a θ distance from the continuous MPC solution. Using an exhaustive evaluation of all the feasible discrete solutions, it is possible to substantially improve the solution. In the network simulated in this chapter, the computation can be done in a short time compared to the continuous MPC. In cases where this method entails unacceptable computation times, it is proposed to relax the exhaustive search using a genetic algorithm. In the analyzed simulation, the θ-Genetic optimization is able to closely approximate the behavior of the θ-Exhaustive optimization but with a reduced computation time.

Finally, we have proposed an algorithm that computes the discrete VSL directly (i.e. without using the solution provided by the continuous MPC optimization). The mixed-integer optimization is solved by computing the ramp metering and the VSL iteratively and separately in order to decompose the problem in one

continuous optimization problem and one discrete optimization problem. The continuous optimization can be solved by using, for instance, common SQP algorithm. For the discrete optimization, an exhaustive algorithm can be used and, again, a GA is proposed in order to solve the optimization in a reasonable time. For large networks, the iterations between the continuous ramp metering problem and the discrete VSL problem may use information from neighbors controllers (corresponding to nearby regions of the freeway) in order to run a distributed optimization as in chapter 4.2.

The results show that, for the given case study, the exhaustive optimization can be executed in real time substantially improving the performance of the rounding discretization, which is the best implementable solution without running a discrete optimization. In case where the exhaustive solution cannot be found in real time, the alternating genetic optimization almost reaches the behavior of the exhaustive one.

The alternating genetic optimization does not use the solution of the continuous MPC problem and so reduces the computation times. Moreover, the trade-off between simulation speed and accuracy can be easily adapted to the available time making the algorithm very useful for the application in practical cases, especially if it is necessary to use a distributed scheme due to the size of the network.

Chapter 6

Control of Reversible Lanes on Freeways

Reversible traffic operations [104] are widely regarded as one of the most cost-effective methods to increase the capacity of an existing freeway. The principle of reversible lanes is to match available capacity to the traffic demand taking advantage of the unused capacity in the minor-flow direction lanes to increase the capacity in the major-flow direction. There are numerous examples of reversible lanes successfully implemented during the last 85 years in many countries (especially in USA, Australia, Canada and UK) [47]. The reversible lanes are generally used in bottlenecks like bridges or tunnels but there are also examples of entire roadways routinely reversed [20].

Surprisingly, despite the long history and widespread use of reversible lanes worldwide, there have been few quantitative evaluations and research studies conducted on their performance [104]. There is also a limited number of published guidelines and standards related to their planning, design, operation, control, management, and enforcement. Therefore, most reversible lane systems have been developed and managed based primarily on experience, professional judgment, and empirical observation.

This chapter proposes the use of on-line control techniques for the dynamic operation of reversible lanes on freeways based on the extension of the second-order traffic flow model METANET proposed in Section 2.1.

Based on this model, two kinds of dynamic controllers have been developed. The first one is an easy-to-implement logic-based controller which takes into account the congestion lengths generated by the reversible lane bottleneck and uses this information for the dynamic operation of the lanes. The second one is a discrete Model Predictive Control (MPC) which minimizes the Total Time Spent (TTS) of the modeled network within some constraints for the maximum values of the gen-

erated bottleneck queues. The discrete optimization is carried out via evaluation of the cost function for all the leafs in a reduced search tree.

The proposed control algorithms are simulated and tested using loop detector data collected over a section of the SE-30 freeway in Seville, Spain. The modeled network includes the Centenario Bridge, which is a bottleneck with a reversible lane that creates recurrent congestion during the morning rush-hour period. The results show that all the proposed controllers (which can be computed in a short time) substantially reduce this congestion.

It has to be pointed out that the model and control algorithms proposed in this paper can be equivalently applied to other macroscopic traffic models as the Cell Transmission Model (CTM) [15]).

Parts of this chapter are published in [34]. The research related with the Centenario Bridge benchmark has been published in press in Europa Press [89], El Economista [23], Diario de Sevilla [19], ABC, 20 Minutos and others Spanish newspapers.

6.1 Discrete Model Predictive Control for reversible Lanes

The first controller proposed is a discrete Model Predictive Control which minimizes the Total Time Spent of the modeled network within some constraints for the maximum values of the generated bottleneck queues. The control algorithm is similar to the discrete MPC used in section 5.1.3.

To formalize these concepts, consider the freeway traffic system with reversible lanes whose dynamic evolution is described in Sections 1.4 and 2.1. The control inputs will be considered constant during one controller sample resulting in the following state-space model:

$$x(k+1) = f(x(k), u(k), d(k)) \tag{6.1}$$

with $x(k)$ the state vector that includes speeds, densities and queues and $d(k)$ the non-controllable input vector that includes demand profiles, upstream speeds and downstream densities. $u(k)$ is the discrete input vector containing a discrete variable $R_j(k)$ for each reversible lane j.

In an MPC controller, the core is the optimization (6.2) of a cost function $J(k)$, which is used to measure the performance of the system along the prediction horizon N_p with respect to the input control sequence along the control horizon N_u:

$$\min_{R_\mathrm{t}(k)} \; J(k) \qquad \text{with } R_t(k) \in \{-1, 0, 1\} \tag{6.2}$$

where $R_t(k) = [R_{j_1}(k),\ R_{j_1}(k+1),\ ...,\ R_{j_1}(k+N_u-1),\ R_{j_2}(k),\ R_{j_2}(k+1),\ ...,\ R_{j_{N_R}}(k+N_u-1)]$ is the set of reversible lanes and N_R the number of reversible lanes. As usual, the control inputs are kept constant after the control horizon N_u.

In this chapter, as in previous ones, the MPC controller uses an cost function (6.3) containing one term for the TTS and another term that limits (using a soft constraint) the maximum values of the queues:

$$J(k) = \sum_{\ell=1}^{N_p}[T\sum_{i\in\mathbb{O}} w_i(k+\ell) + T\sum_{i\in\mathbb{I}_1}(\rho_i(k+\ell)L_i\lambda_i) + \\ +T\sum_{i\in\mathbb{I}_2}(\rho_i(k+\ell)L_i\widehat{\lambda}_i(k+\ell)) + \sum_{i\in\mathbb{O}}\Omega_i(k+\ell)] \tag{6.3}$$

where $\mathbb{O}$ is the set of all the segments with an on-ramp, $\mathbb{I}_1$ the set of all the segments without reversible lanes, $\mathbb{I}_2$ the set of all the segments with reversible lanes and $\Omega_i(k+\ell)$ is a penalization term that is different to zero if the corresponding queue constraint is violated. If it is desired to limit or reduce the frequency of reversible lane switching, an additional term penalizing each switching may be readily included in the cost function.

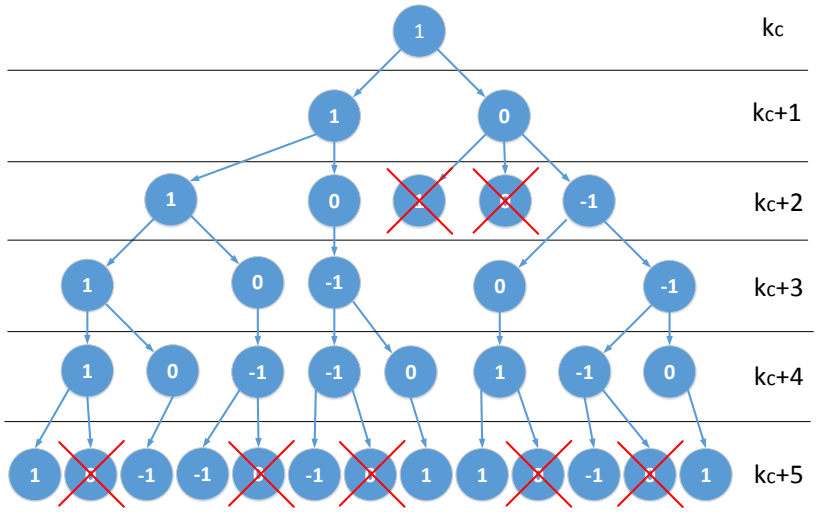

Figure 6.1: Search tree for a reversible lane with $R(k) = 1$ and $N_u = 5$

In order to solve optimization (6.2), we use a search tree including all the feasible profiles for the opening of the reversible lanes $R_t(k)$. As we are dealing with non-convex integer optimization, the only way to obtain the global optimum of (6.2) is to evaluate the cost function for all the feasible points in the search tree. Limiting the number of feasible nodes in the mixed integer optimization problem is a technique that has been successfully used in Section 5.1.3 in the context of

freeway traffic MPC for Variable Speed Limits (VSL) [31, 33].

The feasible nodes limitation can be obtained by using the following constraints:

– A reversible lane has to be closed in both directions during one controller sample time for all direction switches, in order to allow the cars to leave the corresponding lane.

– It is always suboptimal to keep lane closed in both direction during more than one controller sample time.

– If the lane is closed in both directions, it is always suboptimal to open the lane in the same direction in which it was previously open.

– For the last prediction step, it is not allowed to set the last value of the input along the control horizon as 0, because it is suboptimal to leave closed the reversible lane if it is not going to be opened during the next steps.

Fig. 6.1 shows an example of the search tree if only one reversible lane is used and initially the lane is opened in the increasing direction.

The number of leafs $N_l(N_\mathrm{u})$ increases with respect to the control horizon according to the one-step delayed Fibonacci series (1,2,3,5,8,13,21,34,55,89,...):

$$N_l(N_\mathrm{u}) = \frac{\phi^{N_\mathrm{u}+1} - (1-\phi)^{N_\mathrm{u}+1}}{\sqrt{5}} \text{ with } \phi = \frac{1+\sqrt{5}}{2} \tag{6.4}$$

If the reversible lane is closed for the current controller step ($R_\mathrm{t}(k) = 0$), it has to be allowed to open the lane in both directions for $k+1$ due to unpredicted changes during the last controller time step T_c. Therefore, the number of leafs increases according to:

$$N_l(N_\mathrm{u}) = 2 \cdot \frac{\phi^{N_\mathrm{u}} - (1-\phi)^{N_\mathrm{u}}}{\sqrt{5}} \tag{6.5}$$

The a-priori knowledge of the number of leafs allows to know the maximum horizons for which the optimal solution can be computed within a controller time step. The computation time needed for obtaining each MPC solution is the number of leafs multiplied by the time needed for the simulation of the network during the prediction horizon.

For a larger freeway network with many reversible lanes, the main problem is the computation time needed for the evaluation of such a large number of possible combinations of the discrete variables. In these cases, additional constraints connecting the reversible lanes may be used. For instance, the lanes closed in one direction should be on the left hand side of the opened lanes (in the countries with left-hand drive). When even using the new constraints, it is not possible to find the optimal solution within a limited computation time available, thus genetic or distributed algorithms could be used as in [31, 28].

6.2 Logic-based control for reversible lanes

The second controller proposed is a logic-based controller that changes the state of the reversible lane by a simple feedback of the system state. Such a controller can be easily implemented in real applications without the need of an on-line optimization.

The proposed control algorithm is designed for networks with only one reversible lane, but similar controllers could be proposed for cases of more than one reversible lanes.

The proposed controller uses a controller sample time T which is a multiple of the model sample time T_{m} (i.e. $t = k \cdot T = k_{\mathrm{m}} \cdot T_{\mathrm{m}}$). This work uses a controller sample time of 2 minutes. The discrete input variable $R(k)$, which has to be computed for each controller time step, indicates if the reversible lane is open in the increasing direction (1), or in the decreasing direction (-1) or close for both directions (0).

For safety reasons, the reversible lane has to be closed during a certain period of time whenever the opened direction of the lane changes. In this work, it is assumed that the reversible lane has to be closed during one sample time during any direction change (i.e. if $R(k-1) = 1 \implies R(k) \neq -1$).

The feedback of the controller will be done with respect to the congestion lengths $L_c(k)$ generated by the bottleneck with the reversible lane in both directions. The congestion lengths may be estimated for each direction by using the speed measured in the segments upstream the bottleneck:

$$L_c(k) = \sum_{n=Z+1}^{n=I-1} L_n \qquad (6.6)$$
$$\text{with } v_Z(k) > V_{cg} \text{ and } v_i(k) < V_{cg} \ \forall i \in (Z, I)$$

where I is the first segment affected by the reversible lanes and Z is the first uncongested segment. V_{cg} is a speed threshold below which the system can be considered congested. This speed may differ from one network to another but it will usually be around 60 km/h.

Once we have an estimation of the congestion lengths, it is necessary to differentiate between three cases for the operation of the reversible lane:

- **Congestion lengths are 0 for both directions:** In this case, the assignment of the reversible lane is not really critical. A reasonable strategy is to refrain from any change, that is, unless the flow in the direction currently using the reversible lane becomes much lower than in the opposite direction.

This can be implemented via:

$$R(k+1) = \tag{6.7}$$
$$= \begin{cases} 1 & \text{if } R(k-1) = 1 \text{ and } \chi \cdot q_{b,I}(k) > q_{b,D}(k) \\ -1 & \text{if } R(k-1) = 1 \text{ and } \chi \cdot q_{b,I}(k) < q_{b,D}(k) \\ 1 & \text{if } R(k-1) = -1 \text{ and } q_{b,I}(k) > \chi \cdot q_{b,D}(k) \\ -1 & \text{if } R(k-1) = -1 \text{ and } q_{b,I}(k) < \chi \cdot q_{b,D}(k) \end{cases}$$

where $q_{b,I}(k), q_{b,D}(k)$ are the flows through the bottleneck in the increasing and decreasing directions respectively, and χ is a parameter bigger than 1 (in this paper, $\chi = 1.3$).

- **Any congestion length (or both) is (are) bigger than 0 and smaller than the maximum congestion length:** In this case, it is reasonable and equitable to attempt a balance of the congestion lengths on both directions via appropriate switchings. This may be achieved via the following switching regulator:

$$R(k+1) = \tag{6.8}$$
$$= \begin{cases} 1 & \text{if } R(k-1) = 1 \text{ and } \Lambda \cdot L_{c,I}(k) > L_{c,D}(k) \\ -1 & \text{if } R(k-1) = 1 \text{ and } \Lambda \cdot L_{c,I}(k) < L_{c,D}(k) \\ 1 & \text{if } R(k-1) = -1 \text{ and } L_{c,I}(k) > \Lambda \cdot L_{c,D}(k) \\ -1 & \text{if } R(k-1) = -1 \text{ and } L_{c,I}(k) < \Lambda \cdot L_{c,D}(k) \end{cases}$$

where $L_{c,I}(k)$ and $L_{c,D}(k)$ are the congestion lengths in the increasing and the decreasing direction, respectively) and Λ is a parameter bigger than 1 (in this paper, $\Lambda = 1.3$).

It should be noted that each switching decision has a cost due to the mandatory intermediate state $R(k) = 0$ whereby no direction can profit from the capacity of the reversible lane. For $\Lambda = 1$, we would have the highest frequency of switching, but also the lowest differences in congestions lengths in both directions. Thus, the selected value of Λ reflects a trade-off between overall efficiency versus equity of the control system.

- **Both congestion lengths are at or beyond their respective maximum values:** The situation is overcritical and the control strategy to be pursued should be based on a policy decision by the responsible authority. In the present study, the reversible lane will be opened in each direction after a fixed time T_R (in this paper, 15 minutes).

6.3 Case study

In order to simulate the analyzed controllers, the same network and traffic conditions (the Centenario bridge in Seville, Spain) than in the previous subsection 2.1.4 have been used (see Figure 2.8). The model parameters obtained in subsection 2.1.4 are used for the network simulation.

Figure 6.2: Simulated bridge with 2 lanes fixed in each direction and one reversible lane

The bridge has 2 lanes fixed in each direction and one reversible lane as can be seen in Figure 6.2. The reversible lane is currently changed manually by the traffic operators looking at the cameras along the bridge. This real-time manual control is deemed to perform better than fixed control, which would operate with pre-specified switching intervals during pre-specified (peak) periods of the day.

This simulation uses loop detector data over the 6AM-11AM time range for three different weekdays to define mainstream and ramp demands and split ratios. There is also data available of the state of the reversible lane $R_{\mathrm{t}}(k)$ indicating the time when the lane is closed or opened in one direction.

The morning rush-hour congestion usually occurs between 8 and 9 am. The congestion is created by the bottleneck on the bridge and propagates upstream for both directions. The traffic downstream the bridge is always uncongested in both directions. There is no other source of recurrent congestion in the modeled network.

For the controller simulations, it is assumed that there is a sensor in each segment which provides density and speed measurements for each controller step; alternatively, an appropriate estimation scheme (e.g. [103] may be employed in practice). The control parameters were manually selected by performing some simulations with different values of the parameters.

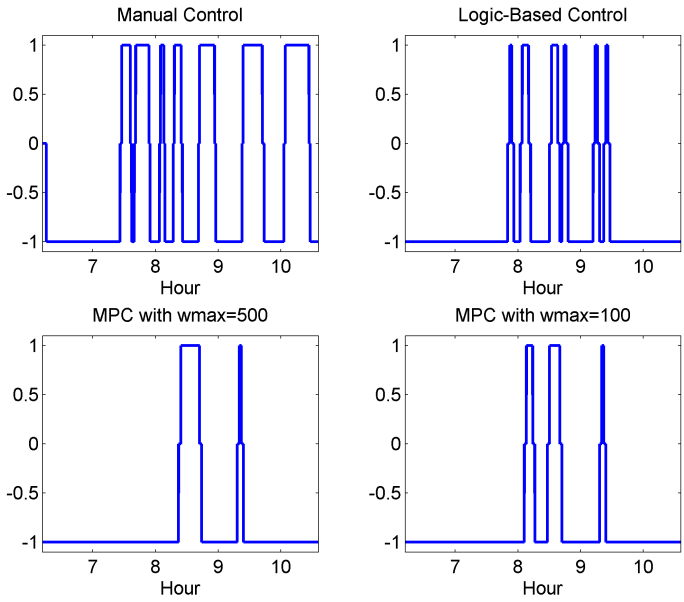

Figure 6.3: Reversible lane operations on April 11, 2012

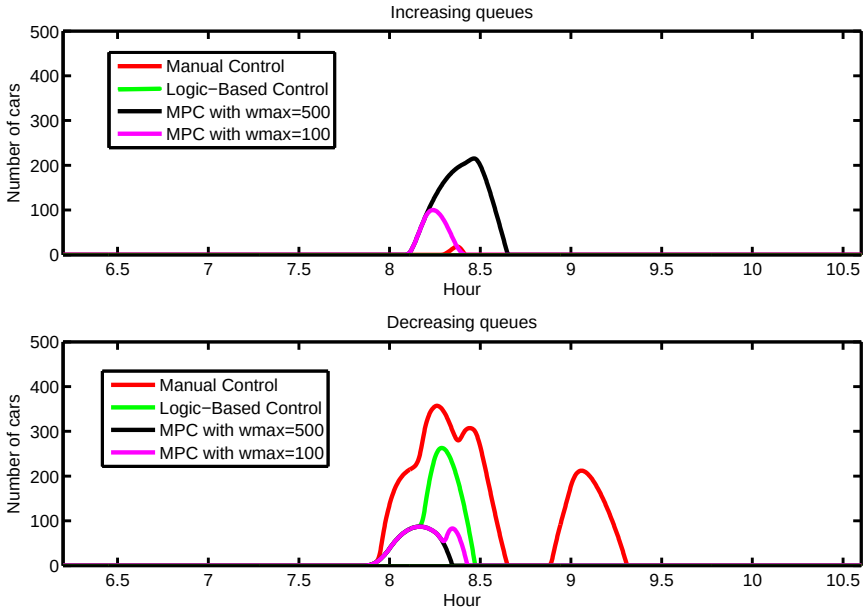

Figure 6.4: Mainstream queues on April 11, 2012

The simulations were carried out using MATLAB and an Intel Core i5 CPU. The average time needed is 1.25 seconds for the network simulation (from 6 AM to 11 AM), 0.0003 seconds (any controller step) for the logic-based controller and 0.65 seconds (any controller step) for the MPC (with $N_{\mathrm{u}} = 5$ and $N_{\mathrm{p}} = 5$). It can be seen that the logic-based control can be computed almost instantaneously so to be implemented at the same time when the measures are taken and the MPC can be computed within a controller time step (i.e. implemented at $k + 1$ for the measures taken at k).

Two different MPC controllers have been simulated. The first one has a constraint of a maximum of 100 vehicles waiting in the queues. The second one has a constraint of 500 vehicles which means to have in practice unconstrained queues.

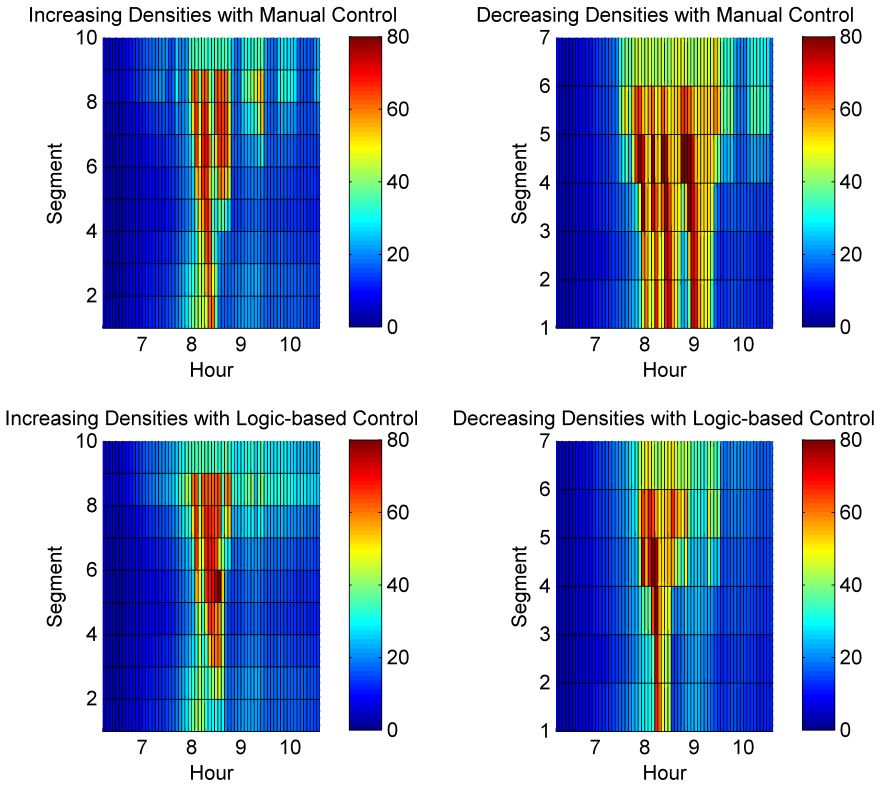

Figure 6.5: Densities for manual control and logic-based control applied on April 11, 2012

The results of the reversible lane operations computed by each controller can be seen in Fig. 6.3. The figure shows that the MPC controllers tend to reduce the time that the reversible lane is closed by decreasing the number of direction

changes in the reversible lane. For the constrained MPC, the queue constraint causes that the reversible lane has to change the direction one more time than for the unconstrained MPC in order to keep the queues under 100 vehicles.

This can be seen in Fig. 6.4 where the queues at the mainstream origins obtained for the different reversible lane operations are shown. None of the simulations carried out create on-ramp queues. Fig. 6.4 shows that the constrained MPC is the only controller simulated that keeps both queues under the constraint. The logic-based controller provides an intermediate solution between the real implemented manual controller and the MPC controller proposed.

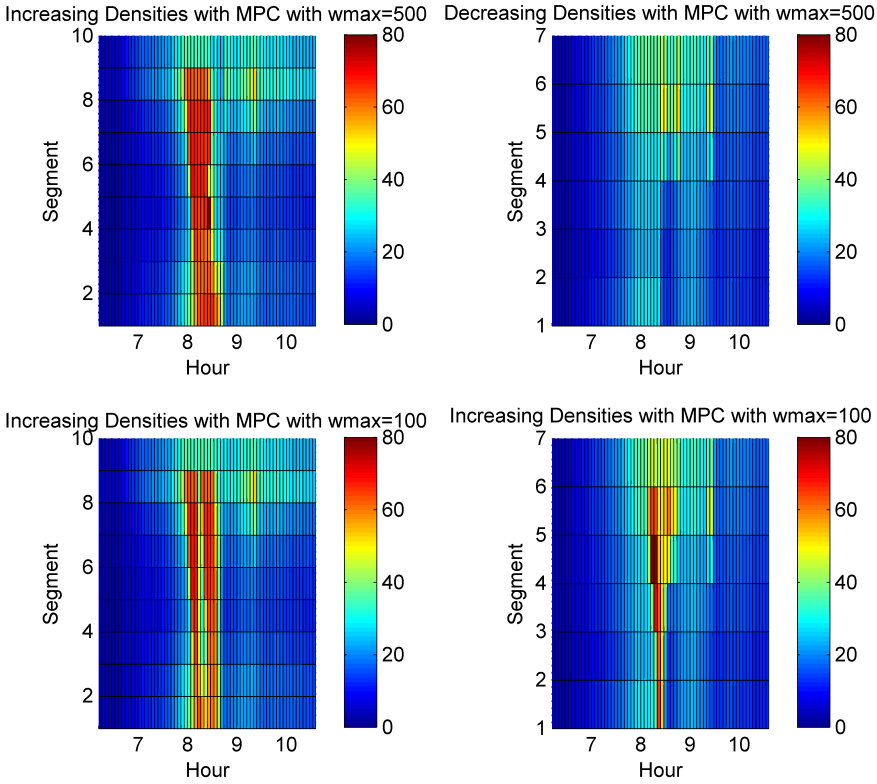

Figure 6.6: Densities for the MPC controllers applied on April 11, 2012

The density contour plots obtained with the different controllers are shown in Fig. 6.5 and 6.6. It can be seen that the unconstrained MPC removes the congestion completely in the N-S direction by increasing the congestion substantially in the S-N direction. This solution may be optimal in terms of cost function performance but it may be difficult to implement due to equity reasons (one direction is seen to be clearly benefited versus the other). On the other hand, the constrained MPC

and the logic-based controller provide control inputs that tend to equalize the congestion in both directions. Therefore, these controllers may be good solutions for real implementations.

In the numerical results on Table 6.1, it can be seen how the three proposed controllers substantially reduce the TTS with respect to the manually controlled case. The TTS reductions obtained for the three controllers are very close to each other. However, it can be seen that unconstrained MPC results in a higher TTS reduction than the other proposed controllers.

Table 6.1: Controller Performances in terms of TTS reduction (%)

	April 11	April 16	April 17
Manually controlled system	0 %	0 %	0 %
Logic-based controller	8.84 %	19.97 %	22.68 %
MPC with $w_{max} = 500$	11.94 %	22.24 %	26.97 %
MPC with $w_{max} = 100$	9.43 %	19.82 %	25.22 %

The TTS reduction shown covers both directions. This reduction is obtained by decreasing the TTS in one direction while increasing the TTS in the opposite direction or by decreasing the TTS for both directions. For example, for April 17 the TTS is decreased for the N-S direction with all the used controllers (38.95 % with the logic-based controller, 40.58 % with the unconstrained MPC and 46.61 % with the constrained MPC). However, for the N-S direction the TTS is increased with two controllers (1.35 % with the logic-based controller and 6.39 % with constrained MPC) and decreased with one controller (6.85 % with the unconstrained MPC).

6.4 Conclusion

This chapter has proposed two control algorithms for the dynamic operation of reversible lanes on freeways using the extension of model METANET proposed in Section 2.1.

Based on this model, two kinds of dynamic controllers have been developed. The first one is an easy-to-implement logic-based controller, which takes into account the congestion lengths generated by the reversible lane bottleneck; and which uses this information for the dynamic operation of the lanes.

The second one is a discrete Model Predictive Control (MPC) where the discrete optimization carried out is via evaluation of the cost function for all the leafs in a reduced search tree.

The main advantages of the proposed MPC are larger TTS reductions, reduction of the number reversible lane switching and possibility of using constraints. The main advantages of the logic-based controller are ease of real implementation, intuitive tuning and equity for opposite directions.

All the proposed controllers show a substantial reduction of the TTS (between 8.84% and 26.97%) and can be computed in a short time (0.65 seconds for the MPC and almost instantaneously for the logic-based controller).

Chapter 7

Conclusions and Further Research

7.1 Contributions to the state of the art

The sections enumerates the main contributions of this thesis to the state of the art including the reference to the papers where the corresponding results were published:

- In Section 2.1, it is extended the METANET model with the modeling of reversible lanes [34].
- In Section 2.2, it is proposed an identification method for the METANET model parameters for cases where there is a limited number of loop detectors [30].
- In Section 2.2, a new mathematical definition of the fundamental diagram is proposed [30].
- In Section 2.3, it is made the first theoretical and numerical direct comparison between the two most used macroscopic traffic models: CTM and METANET [26].
- In Section 3.4, it is analyzed the robustness of model-based predictive controllers for freeway traffic control, with respect to variations in the mainstream demand [32].
- In Section 4.1, it is motivated the use of global or distributed algorithms for freeway traffic control systems instead of local algorithms [29].
- In Section 4.2, two distributed algorithm (FC-MPC and Communicative MPC) for the control of freeway traffic systems are proposed [28, 27].

- In Section 5.1, it is proposed a way to obtain practical VSL signals considering the discrete characteristics of the panels and other implementation constraints [33, 31].
- In Section 5.2, it is proposed an hybrid MPC algorithm for the control of discrete VSL together with other continuous traffic measures like ramp metering [31].
- In Section 6.1, it is proposed a discrete MPC for the optimal operation of reversible lanes [34].
- In Section 6.2, it is proposed an easy-to-implement logic-based controller for the operation of reversible lanes [34].

7.2 Final conclusions

Chapter 2 proposes new advances in freeway traffic modeling in order to allow the design and implementation of optimal control strategies.

Firstly, a macroscopic model for reversible lanes on freeways is proposed. The reversible lanes are modeled like variable lane drops using the concept of an equivalent number of lanes. The proposed model has been validated with real data over the Centenario Bridge of the SE-30 freeway in Seville, Spain. The results show that the proposed model is able to reproduce traffic congestion due to the reversible lanes with a mean error on the identification days of 8.63% and 4.33% for speed and flows, respectively. The errors are 14.29% and 5.19% for the days used for validation.

Afterwards, a three steps identification algorithm for the macroscopic traffic model METANET that tries to avoid undesirables local minima has been proposed. The results show a good estimation of the traffic densities and speeds (with a mean error of 10.95% and 11.35%, respectively) over a section of the I-210 West in Southern California validating the identification algorithm. It can be concluded that the identification algorithm proposed may be especially useful for traffic control using Model Predictive Control, where the accuracy of the model and the possibility of executing the identification in real time are key issues.

Moreover, it is proposed a new form of the fundamental diagram which allows to improve the matching between the fundamental diagram and the data without an increase in the computational power needed for the simulation.

Finally, the two most used macroscopic models for traffic control, LN-CTM and METANET, have been compared, with the following conclusions:

- Identification of the model parameters of METANET is not a trivial issue. However, it is easier and faster for LN-CTM.

- The computational time needed for the simulation of LN-CTM is smaller, but of the same order of magnitude than for METANET.

- The design of a MPC controller is easier for LN-CTM. For METANET, computationally intensive optimizations are needed.

- In some cases, METANET simulation can show undesirable oscillations.

- LN-CTM does not produce a realistic simulation when a VSL is set over the current speed and there is an on-ramp in the following segment or when an on-ramp queue is between $T \cdot C_r$ and $T \cdot q_{ramp}$.

- LN-CTM model, because it does not capture the capacity drop phenomenon, is incapable of exploiting the benefits of increasing bottleneck flow. However, METANET model can describe the speed dynamic increasing the potential improvement of a VSL control and allowing the inclusions of the emissions in the prediction of the model.

From a numerical comparison which has been done over a section of the I-210 West in Southern California using real date, the following conclusions can be drawn:

- METANET simulation shows a smaller error than LN-CTM for all the days simulated with almost half the mean error (10.84% versus 20.94%).

- Both models have an accurate prediction for the uncongested part and rough prediction for the congested part.

- In open-loop simulations, small prediction errors can lead to a completely wrong simulation when the system passes from uncongested to congested state, especially for LN-CTM model.

Chapter 3 has provided a brief account of the basic concepts of MPC for traffic systems in Sections 3.1. The MPC controller is demonstrated in a simple simulation-based case study in Subsection 3.1.4. For detailed discussions on MPC for freeway traffic systems, we refer the reader to [37].

Moreover, this chapter has proposed new methods for online demand estimation and has analyzed the robustness of MPC controllers for freeway traffic with respect to variations in the mainstream demand. Firstly, it has been proposed three simple estimation procedures for the demand profiles that combine off-line information (historical data) together with on-line information (demand in previous time steps). The first one uses a CARMA model, the second one uses a CARIMA model and third one uses a series of exponential functions. The exponential estimation shows the better close-loop performance of the models considered.

Subsequently, the robustness of MPC controllers for freeway traffic has been analyzed by the simulation of seventeen days with real-life data from the A12 Dutch freeway using different demand profiles for the MPC controllers. The conclusion is that, at least for this scenario, the MPC controller is not very sensitive with respect the demand profiles so the controller will perform properly even if there are visible errors in the demand estimation.

Finally, the potential reduction of the total time spent with respect to the aggregate demand is studied, showing the range where ITS signals can be used to increase the traffic flows by solving recurrent congestion.

Chapter 4 analyzes the loss of performance due to the decentralization and the difficulty of implementing a completely centralized control for large networks and proposes distributed MPC algorithms which can be implemented in real time for a large enough traffic network, minimizing the total time spent by the drivers.

Even being a generally true conclusion that decentralized controller is suboptimal with respect to the centralized case, this chapter shows how far are both options, in order to motivate the development of distributed solutions.

Firstly, local, decentralized and centralized control techniques have been evaluated on a simulated 18 km stretch of a freeway, concluding that a centralized controller

allows to obtain much greater reductions than the fully decentralized one (26.4 % against 6.5 % in the reduction of the TTS) but requires much higher computational effort. Therefore, a centralized traffic control would be very difficult to implement in a real (i.e. large) traffic network. A local MPC with communication with the neighbors allows to obtain greater reduction than the local controllers (12.88 % versus 6.5 %) and does not require much more computational effort. This result shows that an algorithm that properly uses communication and cooperation between different controllers in order to get close to the centralized behavior without increase critically the computational effort, as the one presented in the section, improves the performance of the traffic system without requiring excessive computational time.

Subsequently, distributed MPC for freeway networks are proposed, which can be implemented in real time for a large enough traffic network, minimizing the total time spent by the drivers. The proposed algorithms converge, in the simulation carried out, in just a few iterations and can be computed in a fraction of the time needed by the centralized one (between a third and a fifth).

The last conclusion is that Nash Equilibrium is far away to Pareto Optimal Equilibrium in this traffic system (i.e. cooperation is a key issue). In the simulation, the TTS reductions increases from 11.31 % to 25.06 % thanks to cooperation. On the other hand, Cooperative MPC almost reaches the centralized behavior fulfilling the design objectives.

Chapter 5 has proposed hybrid Model Predictive Control (MPC) approaches for freeway traffic control considering Variable Speed Limits (VSL) as discrete variables as in current real world implementations.

Firstly, this chapter has analyzed the effect of converting the continuous unconstrained VSL signal to an implementable VSL signal (i.e. a discrete value respecting the safety constraints). In the simulated network, the discrete constrained performance is substantially reduced with respect to the ideal case. Therefore, it can be concluded that the assumption about the continuous implementation of the VSL entails in some cases a large loss of performance for the controlled system. Since the majority of the literature makes this supposition, we have proposed two kind of methods to reduce part of this loss of performance.

The first class of proposed methods discretizes the VSL signals solving a discrete optimization problem with a search space limited to a θ distance from the continuous MPC solution. Using an exhaustive evaluation of all the feasible discrete solutions, it is possible to substantially improve the solution. In the network simulated in this section, the computation can be made in a short time compared to the continuous MPC.

In cases where this method entails unacceptable computation times, it is proposed to relax the exhaustive search using a genetic algorithm. In the analyzed simulation, the θ-Genetic optimization is able to closely approximate the behavior of

the θ-Exhaustive optimization but with a reduced computation time.

Finally, we have proposed an algorithm that computes the discrete VSL directly (i.e. without using the solution provided by the continuous MPC optimization). The mixed-integer optimization is solved by computing the ramp metering and the VSL iteratively and separately in order to decompose the problem in one continuous optimization problem and one discrete optimization problem. The continuous optimization can be solved by using, for instance, common SQP algorithm. For the discrete optimization, an exhaustive algorithm can be used and, again, a GA is proposed in order to solve the optimization in a reasonable time.

The results show that, for the given case study, the exhaustive optimization can be executed in real time substantially improving the performance of the rounding discretization, which is the best implementable solution without running a discrete optimization. In cases where the exhaustive solution cannot be found in real time, the alternating genetic optimization almost reaches the behavior of the exhaustive one. The alternating genetic optimization does not use the solution of the continuous MPC problem and so reduces the computation times. Moreover, the trade-off between simulation speed and accuracy can be easily adapted to the available time, making the algorithm very useful for the application in practical cases, especially if it is necessary to use a distributed scheme due to the size of the network.

Chapter 6 has proposed two control algorithms for the dynamic operation of reversible lanes on freeways using the extension of model METANET proposed in Section 2.1.

Based on this model, two kinds of dynamic controllers have been developed. The first one is an easy-to-implement logic-based controller, which takes into account the congestion lengths generated by the reversible lane bottleneck; and which uses this information for the dynamic operation of the lanes. The second one is a discrete Model Predictive Control (MPC) where the discrete optimization carried out is via evaluation of the cost function for all the leafs in a reduced search tree.

The main advantages of the proposed MPC are larger TTS reductions, reduction of the number reversible lane switching and possibility of using constraints. The main advantages of the logic-based controller are: ease of real implementation, intuitive tuning and equity for opposite directions.

All the proposed controllers show a substantial reduction of the TTS (between 8.84% and 26.97%) and can be computed in a short time (0.65 seconds for the MPC and almost instantaneously for the logic-based controller).

7.3 Future Work

- **Distributed and hybrid MPC for freeway traffic:** For large networks, where a distributed algorithm is required [29], the iterations between the continuous ramp metering problem and the discrete VSL problem may use information from neighbors controllers (corresponding to nearby regions of the freeway) in order to run a distributed optimization which tries to approach the global minimum using a equivalent procedure that the one used in [28] for continuous controllers.

- **Application of the proposed algorithms to other macroscopic traffic models:** Since the methods we propose are independent of the traffic model used, it would be interesting to apply them using other macroscopic traffic models which are capable of including the effect of VSL in their formulation (like some versions of the Cell Transmission Model [15]).

- **Integrating reversible lanes with VSL and ramp metering:** In Section 6 a discrete controller for reversible lanes is proposed. It may be interesting in a future work to analyze the traffic performance improvement obtained by combining reversible lanes and other traffic control measure (especially, VSL).

- **Parametrized MPC for freeway traffic:** One of the main drawbacks of MPC is that it is computationally demanding, especially when dealing with highly complex non-linear systems. In future works, in order to overcome this limitation, one possible solution is the use of Parametrized MPC, where a explicit function is able to reproduce the controllers behavior. This work is currently being developed in a common work between the University of Sevilla and the universities of Pavia and Genova [78] [77].

- **Analysis of the robustness with respect to model uncertainties:** In Section 3.2, the robustness of MPC controllers for freeway traffic is analyzed with respect to changes in the predicted mainstream demand. In future works, it will be interesting to analyze the robustness with respect to model uncertainties.

- **Real-world implementation of the MPC controllers for freeway traffic:** Although the theoretical results of MPC applied to freeway traffic systems are very promising, the ultimate proof is the testing in a real-world situation. Further investigations regarding the real-world applicability are necessary.

- **The use of other traffic control measures:** This thesis has proposed controllers using ramp metering, VSL and reversible lanes. Future works may extend the proposed controller to other traffic control measures like route guidance, dedicated lanes...

- **Precomputed MPC for freeway traffic:** In order to overcome the computational demand of non-linear MPC one possible solution is the use of precomputed MPC. In this case, an off-line computed solution is combined with a linear feedback controller in order to obtain an easy-to-implement controller which approximates the optimal behavior.

- **Robust MPC for freeway traffic:** In real traffic networks there are various types of uncertainties or disturbances, such as demand uncertainties, model uncertainties, missing samples, sensor errors, and delays. Including these uncertainties when determining control strategies offers a significant potential for obtaining a better control performance. Hence, it is important to develop robust MPC approaches for traffic networks that maintain performance specifications for a given range of uncertainties. This work is currently being developed in a common work between the University of Sevilla and TU Delft [60] [61].

- **A logic-based controller for VSL:** In Section 6.2, a logic-based controller for reversible lanes that changes the state of the reversible lane by a simple feedback of the system state is proposed. This work may be extended to VSL upstream the reversible lanes, improving the benefits of the control system.

Appendix A

Resumen de la Tesis en Castellano

A.1 Introducción

Actualmente, el ahorro de combustible, la mejora de la movilidad de los ciudadanos, la reducción de las emisiones atmosféricas y de los accidentes de tráfico son algunos de los aspectos claves en las políticas gubernamentales en el primer mundo. Durante los últimos años, un gran esfuerzo investigador se ha centrado en resolver, o mitigar, estos problemas. Debido a que la construcción de nuevos ramales viarios (o la ampliación de las ya presentes) no es siempre una opción viable (por razones económicas o técnicas), es necesaria la búsqueda de otras alternativas.

En concreto, el "control dinámico de tráfico" (es decir, la aplicación de señales de control inteligentes o ITS) podría ser una solución viable. El "control dinámico de tráfico" mide o estima el estado de la circulacin (densidad media, velocidad media y cola de coches en cada tramo de interés) en cada instante y calcula la señal de control que cambia la respuesta del sistema mejorando su funcionamiento.

Las señales de control de tráfico más útiles son los "ramps metering" (o rampas de acceso controlado) y los "límites dinámicos de velocidad" (VSL) porque son fáciles de implementar, relativamente baratos y suponen una mejora sustancial en el tiempo total de conducción empleado por los conductores (TTS). Sistemas de "ramp metering" y/o VSL han sido implementados con éxito en USA, Alemania, España, Holanda y otros países.

En la actualidad, la mayoría de los sistemas de control de tráfico operan usando un control clásico por realimentación, lineal y local. Sin embargo, el uso apropiado de técnicas multivariables y no locales mejorará substancialmente el comportamiento del sistema controlado. El uso de un controlador predictivo basado

en modelo (MPC) centralizado es posiblemente la mejor elección para una red de tráfico pequeña como puede observarse en [37]. El problema fundamental del MPC centralizado es que el tiempo de computación crece exponencialmente con el tamaño de la red. Por tanto, este tipo de controladores son imposibles de implementar en tiempo real en redes suficientemente grandes. Una posible solución es considerar la red vial a controlar como un conjunto de subsistemas más pequeños y controlar cada uno con un MPC independiente (es decir, usar una estructura de control descentralizada).

El principal objetivo de la tesis que se presenta es diseñar un algoritmo de control que pueda ser calculado en tiempo real en una red viaria de gran escala minimizando, al mismo tiempo, el tiempo total de conducción empleado. Para evaluar el funcionamiento de los controladores propuestos, se simulan distintos tramos de utovías y los resultados numéricos son comparados con los resultantes de simular otros esquemas de control (descentralizado, centralizado y comunicativo) en la misma red.

A.2 Resumen de la tesis

A.2.1 Capítulo 1: Introducción y trasfondo

El modelo de tráfico macroscópico METANET ha sido elegido en esta tesis para el diseño de los controladores y las simulaciones de redes viarias. METANET es un modelo de segundo orden discretizado espacial y temporalmente, en el cual la red viaria es representada como un grafo donde los links (m) corresponden a tramos de autovía. Cada link m es dividido en N_m segmentos (i) de longitud L_m. Cada segmento está caracterizado dinámicamente por su densidad media de tráfico y su velocidad media.

El Control Predictivo Basado en Modelo o "Model Predictive Control" es una solución flexible al problema del control dinámico del tráfico que calcula la señal de control optimizando una función de coste en un horizonte deslizante. La función de coste, es decir, la función a minimizar en sistemas de tráfico suele ser el tiempo total empleado por los vehículos (TTS) u otro criterio de seguridad o ambiental. Cambiando la función de coste, la política de control del tráfico puede ser rápidamente modificada. Además, el control predictivo puede tomar en consideración restricciones (en velocidad o densidad por ejemplo) y tomar en consideración los futuros valores de los flujos entrantes.

El control predictivo basado en modelo (MPC) nació en la década de los 70 y se ha desarrollado considerablemente desde entonces. Las principales ideas que lo componen son:

–Uso explícito de un modelo de predicción de las salidas del sistema en instantes futuros (horizonte de predicción).

–Cálculo de una secuencia de control que minimiza una función objetivo.

–Estrategia de horizonte deslizante, de tal forma que en cada instante de muestreo el horizonte es desplazado un instante temporal implementando la primera señal de control de la secuencia calculada previamente.

Los diferentes algoritmos MPC solo diferentes entre ellos en el modelo usado para representar el proceso y en la función de coste minimizada.

Algunas de las principales ventajas del MPC son que es un control muy intuitivo (durante el diseño y el ajuste), que puede tratar problemas complejos (como, por ejemplo, modelos multivariables y/o no lineales), que tiene una compensación intrínseca de los tiempos muertos, que puede usar referencias futuras

La principal desventaja del control predictivo es el tiempo de computación necesario, especialmente para casos multivariables y no lineales.

A.2.2 Capítulo 2: Modelado de tráfico en autovías orientado al diseño de controladores óptimo

Tomando como base el modelo METANET original introducido en el capítulo anterior, este capítulo propone nuevos avances en el modelado de tráfico en autovías orientado al diseño de controladores óptimos o predictivos.

Al principio del capítulo, se propone una modificacin de METANET para carriles reversibles en autovías con el objetivo de permitir el diseño de controladores de tráfico en tiempo real que operen la apertura de los carriles reversible de forma óptima. Más concretamente, los carriles reversibles han sido modelados como una disminución o aumento variable en el tiempo del número de carriles en cada dirección (basándose en el concepto del número equivalente de carriles).

El modelo propuesto ha sido validado usando datos reales del Puente del Centenario, situado en la autovía de circunvalación SE-30 en la ciudad de Sevilla. Los resultados muestran que el modelo propuesto es capaz de reproducir la congestión de tráfico formada por el carril reversible. Los errores medios en la estimación de velocidades y densidades en los días usado para identificar el modelo son de un 8.63% and 4.33%, respectivamente. Los días usado para la validación del modelo presentan un error del 14.28% en velocidad y 5.19% en densidad.

A continuación, en este capítulo se propone un algoritmo de identificación en tres pasos para el modelo METANET, el cual intenta evitar mínimos locales subóptimos. El algoritmo propuesto ha sido validado en una sección de la autovía I-210 en Pasasadena, California. Los resultados obtenidos muestran un buena estimación tanto de densidades como de velocidades (con un error medio del 10.95% y del 11.35 %, respectivamente).

Se puede concluir que el algoritmo de identificación propuesto es especialmente útil para sistemas de control de tráfico que usen control predictivo, donde la exactitud del modelo y la posibilidad de ejecutar dicho modelo en tiempo real son factores clave. Además, este algoritmo de identificación propone el uso de una nueva definición matemática del diagrama fundamental de tráfico que permite mejorar la relación entre la curva teórica y los datos experimentales sin incrementar la carga computacional.

Posteriormente, hemos comparado los modelos más usados en control de tráfico en autovías: METANET y CTM (Cell Transmission Model).

Por un lado, se comparan características teóricas de los modelos con las siguientes conclusiones:

- La identificación de los parámetros del modelo METANET no es una tarea trivial. Sin embargo, dicha identificación es más fácil y rápida para CTM.

- La carga computacional necesaria para simular el modelo CTM es menor, pero en el mismo orden de magnitud que para METANET.

- El diseño de un controlador predictivo es más fácil usando CTM. Usando METANET son necesarias técnicas computacionalmente intensas.

- CTM no produce predicciones realistas cuando un límite dinámico de velocidad (VSL) está por debajo de la velocidad media en ese instante y hay una incorporación en el segmento siguiente o cuando la cola de vehículos de una incorporación se encuentra entre los valores $T \cdot C_r$ y $T \cdot q_{ramp}$.

- El modelo CTM, al no ser capaz de capturar en su modelado el efecto de la "Caída de Capacidad" (Capacity Drop), es incapaz de explotar los beneficios de incrementar el flujo de tráfico en los cuellos de botella. Sin embargo, METANET modela la dinámica de las velocidades medias incrementando el potencial de un controlador para límites dinámicos de velocidad y permitiendo la inclusión de predicciones de emisiones o gastos de combustible asociadas al modelo.

Finalmente, se comparan ambos modelos numéricamente en una sección de la I-210 en California con las siguientes conclusiones:

- El modelo METANET muestra una mejor predicción que CTM para todos los días simulados con la casi la mitad de error medio (10.84 % frente a 20.94 %).

- Ambos modelos tienes predicciones bastante buenas cuando el trafico está descongestionado y aproximaciones mas parciales cuando el tráfico está congestionado.

- En simulaciones en bucle abierto de sistemas de tráfico en autovías, pequeños errores pueden conllevar predicciones totalmente erróneas cuando el tráfico pasa de estar descongestionado a estarlo, especialmente usando el modelo CTM.

A.2.3 Capítulo 3: Control de tráfico en autovías usando control predictivo basado en modelo (MPC).

Los controladores predictivos han demostrado ser prometedores candidatos para la gestión de sistemas de tráfico en autovías ya que dichos sistemas presentan una estructura altamente no lineal y variante en el tiempo.

Inicialmente, este capítulo desarrolla los conceptos básicos de los controladores MPC aplicados a sistema de tráfico en autovías. Las principales particularidades de los controladores diseñados son explicadas a continuacin:

– El sistema controlado está sujeto a restricciones en los valores máximos y mínimos de densidad, velocidad, colas, velocidades variables de control y tasa de apertura del ramp metering.

– La función de coste usada expresa el tiempo total empleado (TTS) por todos los conductores durante el horizonte de predicción más términos que penalizan las variaciones de las señales de control.

– La función "fmincon" del Optimization Toolbox de Matlab ha sido usada para el cálculo de la optimización usando técnicas SQP. Si todas las variables con sus respectivas restricciones fueran consideradas, la optimización no podría ser realizada en un tiempo razonable; es por ello que solo las variables de control son introducidas en la optimización explícitamente. Las restricciones en velocidad, densidad y cola son suavizadas introduciendo términos de penalización en la función de coste.

– Para evitar que el algoritmo caiga en un mínimo local, un "procedimiento de evaluación" es ejecutado antes de la optimización. Durante el mismo, el TTS es evaluado para un mallado de valores de control. El valor mínimo obtenido en esta evaluación es tomado como valor inicial para la optimización.

– Durante el ajuste del controlador, los parámetros de la función de coste han sido elegidos para obtener el mínimo TTS. Los resultados son muy sensibles a los parámetros de la función de coste y, por tanto, un meticuloso proceso de ajuste debe ser realizado para cada red viaria controlada. Especialmente importante es el ajuste de los parámetros que multiplican a los cambios en las variables de control. Teóricamente, los factores de penalización que multiplican a las restricciones suavizadas de densidad, velocidad y colas deberían ser muy altos. Sin embargo, en la práctica, estos parámetros no pueden ser excesivamente altos o se crean problemas numéricos en la optimización.

– Un incremento en los horizontes mejorará la respuesta del sistema pero también aumentará el tiempo de cálculo. Es importante resaltar que la diferencia entre el horizonte de predicción y de control debe ser pequeña o 0 para obtener una respuesta adecuada. Esto se debe a que no es adecuado considerar constante las señales de control durante un periodo largo al final de la predicción ya que el sistema no tiende a un equilibrio estable. Si se selecciona (Np Nc) demasiado grande, el sistema toma demasiado en consideración los valores finales de la señal de control causando un comportamiento subóptimo.

Asimismo, se muestra el comportamiento de un controlador predictivo aplicado a una pequeña red de 6 kilómetros ya simulada en referencias anteriores.

Como contribución al estado del arte, este capítulo propone nuevos métodos para la estimación en línea de la demanda de tráfico y, usando estos métodos, analiza la robustez de un controlador MPC con respecto a variaciones en la demanda de tráfico.

Inicialmente, se proponen tres métodos simples de estimación para los perfiles de demanda combinando información fuera de línea (series históricas) con información en línea (demandas medidas en instantes anteriores). El primer método de estimación usa un modelo CARMA, el segundo un modelo CARIMA y el tercero una serie de funciones exponenciales. La identificación exponencial muestra el mejor funcionamiento para estimaciones en bucle cerrado de los modelos considerados.

A continuación se analiza la robustez de controladores predictivos en autovías mediante la simulación, a partir de datos reales, de 17 días en la autovía holandesa A12. En estas simulaciones se compara el funcionamiento de los controladores usando diferentes estimaciones en línea para los perfiles de demanda. La conclusión obtenida es que, al menos para este estudio, los controladores MPC no son muy sensitivos con respecto a cambios en los perfiles de demanda estimados. Por tanto, los controladores funcionaran correctamente incluso si hay errores considerables en la demanda estimada.

Finalmente, este capítulo analiza la reducción potencial del tiempo de conducción empleado por todos los vehículos con respecto a la demanda agregada mostrando el rango en el cual las señales de tráfico inteligentes (ITS) pueden ser aplicadas para aumentar el flujo de tráfico resolviendo, o mitigando, congestiones recurrentes.

A.2.4 Capítulo 4: MPC distribuido para sistemas de control de tráfico en autovías

La mayoría de las implementaciones de control predictivo a sistemas en red dividen el sistema en pequeños subsistemas y aplican MPC individualmente a cada parte. Está demostrado que una estrategia de control totalmente descentralizada conlleva un comportamiento subóptimo del sistemas controlado, especialmente si los distintos subsistemas interaccionan fuertemente como en los sistemas de control de tráfico. En el lado opuesto, un control completamente centralizado de redes de gran escala es visto poco práctico o utópico para su aplicación a casos reales.

Los algoritmos MPC distribuidos intentan resolver el problema usando una computación en paralelo en la cual los diferentes controladores predictivos que controlan cada subsistema se comunican entre sí para alcanzar la respuesta que tendría un controlador centralizado.

En el MPC comunicativo, las interacciones entre los distintos subsistemas son modeladas y tenidas en cuenta por los controladores vecinos. Es decir, en cada iteración, las trayectorias predichas son intercambiadas entre los controladores locales y el proceso de optimización es repetido. Si el algoritmo converge (probado para sistemas lineales, pero no para no lineales), el sistema alcanzará el equilibrio de Nash. Sin embargo, este equilibrio es subóptimo para muchos sistemas en red, como los sistemas de tráfico. Esto es debido a los controladores están teniendo en cuenta los efectos de otros controladores en el subsistema que les corresponde, pero no tienen en cuenta sus efectos en otros subsistemas. En otras palabras, los controladores se comunican pero no colaboran.

Con el objetivo de mejorar este funcionamiento, el MPC cooperativo puede ser usado para tener en cuenta la importante interrelación entre los distintos agentes locales. Estas técnicas modifican la función objetivo de los MPC locales incluyendo

también las funciones de coste de ciertos controladores vecinos apropiadamente ponderadas. Si se usa el algoritmo feasible cooperation based MPC (FC-MPC), solo las variables locales correspondiente a cada subsistema son tomadas como variables de decisión. En [10], se prueba que el FC-MPC converge hacia el óptimo de controlador centralizado (Pareto-óptimo) en sistemas lineales. No hay resultados probados para sistemas no lineales. Sin embargo, en esta tesis es posible observar como para el control de sistemas de tráfico usando el modelo METANET, el comportamiento centralizado es aproximado por el FC-MPC en pocas iteraciones.

En este capítulo, una autovía de 18 kilómetros ha sido simulada usando diferentes técnicas de control de tráfico, algunas previamente usadas y otras diseñadas para la ocasión. La primera conclusión es que un controlador totalmente descentralizado es claramente subóptimo con respecto al centralizado (6,5 % frente a 26,4 % en la reducción de la TTS). Asimismo, el MPC centralizado no será normalmente implementable en una red real de tráfico debido al esfuerzo computacional necesario. La segunda gran conclusión es que un algoritmo predictivo distribuido converge, al menos en este caso, en unas pocas iteraciones siendo, por tanto, necesario un esfuerzo computacional mucho menor. La tercera conclusión es que el equilibrio de Nash está muy alejado del Pareto-óptimo en este tipo de sistemas. Dicho en otras palabras, la comunicación es un aspecto clave pues permite pasar, como en este caso, de reducciones del 11,31 % al 25,06 %. Como conclusión general, el MPC cooperativo (FC-MPC) prácticamente alcanza el comportamiento centralizado satisfaciendo los objetivos de diseño del controlador. Por último, y como muestran muchos artículos y estudios anteriores, se puede observar como el uso de señales ITS mejora substancialmente la funcionamiento del sistema de tráfico, especialmente si estas han sido calculadas usando Control Predictivo basado en Modelo.

A.2.5 Capítulo 5: MPC discreto o híbrido para sistemas de control de tráfico en autovías

La característica discreta de los límites dinámicos de velocidad (así como algunas restricciones operacionales necesarias) usualmente no son tenidas en cuentas al diseñar sistema de control de tráfico (como se puede observar en referencias previas). Para enfrentarnos a estos problemas usualmente infravalorados, este capítulo propone un método para incluir la característica discreta de los VSL de forma explícita en el cálculo de la señal de control. Para ello, se propone el uso de controladores predictivos híbridos. El problema de este tipo de controladores es que necesitan resolver una optimización compleja (al tener tanto variables continuas como discretas), lo cual incrementa la carga computacional necesaria para el cálculo en tiempo real. Con el objetivo de aliviar este problema, este capítulo propone una serie de métodos basados en la limitación del número de ramas (perfiles de control) admisibles así como el uso de algoritmos genéticos. La combinación

de ambos métodos permite obtener las señales de control en un tiempo razonable. El capítulo se divide en tres partes fundamentales:

- Inicialmente, se analiza el efecto de convertir los VSL calculados de forma continua en VSL implementables (discretos y cumpliendo una serie de restricciones operacionales). En la red simulada, la respuesta del sistema controlado se ve substancialmente empeorada con respecto al caso ideal (continuo). Esto demuestra que asumir que los límites de velocidad son continuos puede conllevar un gran empeoramiento de la respuesta del sistema controlado en ciertas redes de tráfico contradiciendo referencias previas.

- A continuación, se proponen una serie de métodos que discretizan el límite de velocidad obtenido por el MPC continuo imponiendo que la solución discreta debe estar a una distancia máxima θ de la continua. Usando un método exhaustivo ("fuerza bruta"), es posible evaluar todos las ramas del árbol de búsqueda para el ejemplo simulado. En casos donde esto no sea posible, se propone relajar el problema mediante técnicas genéticas. En la simulación realizada, el uso de dichas técnicas da como resultado un funcionamiento muy cercano a la solución óptima (o exhaustiva) al mismo tiempo que reduce considerablemente la carga computacional.

- Finalmente, este capítulo propone el uso de un algoritmo de controlador que calcule los VSL discretos directamente (sin usar ninguna solución continua obtenida con un controlador predictivo). La optimización mixta es resuelta obteniendo los valores para los "ramp metering" y los VSL de forma separada e iterativa (descomponiendo el problema en una optimización continua para los ramp metering" y una discreta para los VSL). La optimización continua se resuelve, como de costumbre, usando algoritmos SQP y la discreta mediante técnicas exhaustivas o genéticas.

Los resultados muestran que, para la red simulada, la optimización exhaustiva puede ser calculada en tiempo real al mismo tiempo que mejora el resultado de los controladores continuos discretizados. Usando los algoritmos genéticos, se obtiene una gran aproximación del comportamiento ideal. El ajuste entre velocidad de cálculo y rendimiento es fácilmente realizable pudiendo, además combinar las iteraciones del MPC híbrido con las de un controlador distribuido como el del capítulo anterior.

A.2.6 Capítulo 6: Control de carriles reversibles en autovías.

En este capítulo se proponen dos algoritmos para el control dinámico de carriles reversible en autovías. Estos algoritmos se basan en la extensión del modelo METANET para carriles reversibles realizada en la sección 2.1.

El primer algoritmo propuesto es un controlador lógico fácil de implementar en aplicaciones reales. La operación del carril reversible es realizada teniendo en

cuenta la longitud de la congestión (en cada dirección) generada por el cuello de botella con los carriles reversible. A partir de esta información, se decide en qué dirección debe abrirse o cerrarse el carril.

El segundo controlador propuesto es un controlador predictivo discreto (parecido al usado en el capítulo anterior para los límites de velocidad) que minimiza el tiempo total empleado por todos los conductores (TTS) en la red modelada. Además, se tienen en cuenta unas restricciones sobre el valor máximo de la cola de congestión provocada en cada una de las entradas a la autovía. La optimización discreta es realizada de forma exhaustiva evaluando todas las ramas de un árbol de búsqueda reducido.

Los controladores propuestos han sido evaluados con datos de una sección de la SE-30 sevillana incluyendo el Puente del Centenario (el cual es un cuello de botella con un carril reversible que crea congestión recurrente durante las horas puntas). Los resultados muestras que los controladores propuestos reducen substancialmente la congestión y pueden ser calculados en un tiempo razonable.

A.2.7 Capítulo 7: Conclusiones y Trabajo futuro.

La tesis termina con un capítulo que analiza las principales conclusiones y contribuciones al estado del arte, así como líneas futuras de investigación relacionadas.

A.3 Contribuciones al estado del arte

En este apartado se enumeran las principales contribuciones al estado del arte de la tesis. Igualmente, se incluye la referencia al artículo donde han sido publicados los respectivos resultados

- En la Sección 2.1, el modelo de tráfico en autovías METANET es extendido para permitir el modelado de carriles reversibles [34].
- En la Sección 2.2 se propone un algoritmo de identificación para los parámetros del modelo METANET, especialmente pensado para casos donde solo hay disponible un número limitado de sensores [30].
- En la Sección 2.2 se propone el uso de una nueva definición matemática del diagrama fundamental de tráfico [30].
- En la Sección 2.3 se realiza la primera comparación directa entre los dos modelos macroscópicos de tráfico más comúnmente usados [26].
- En la Sección 3.4, la robustez de los controladores predictivos aplicados a sistemas de tráfico (con respecto a variaciones de la demanda de tráfico) es analizada [32].
- En la Sección 4.1 se justifica la necesidad de usar algoritmos de control globales o distribuidos (y no algoritmos locales) en sistemas de control de tráfico [29].
- En la Sección 4.2 se proponen dos algoritmos predictivos distribuidos (FC-MPC and Communicative MPC) para el control de tráfico en autovías [28, 27].
- En la Sección 5.1 se explica un método para obtener los valores óptimos de los límites de velocidad considerando la característica discreta de los mismos y otras restricciones prácticas [31].
- En la Sección 6.1 es propuesto un controlador MPC discreto para la operación de carriles reversibles [34].
- En la Sección 6.2 se propone un controlador lógico fácilmente implementable para la operación de carriles reversibles [34].

Bibliography

[1] C. Alecsandru and Md. A. Quddus. An assessment of the cell-transmission traffic flow paradigm: Development and applications. 215(4):283303, 2001.

[2] A. Alessandri, A. Di Febbraro, A. Ferrara, and E. Punta. Nonlinear optimization for freeway control using variable-speed signaling. *IEEE Transactions on Vehicular Technology*, 48(6):2042–2052, Nov 1999.

[3] A. Aw and M. Rascle. Resurrection of "second order" models of traffic flow? *SIAM Journal on Applied Mathematics*, 60:916–938, 1999.

[4] T. Back, D. Fogel, and Z. Michalewics, editors. *Advanced Algorithms and Operators*. Institute of Physics Publishing, Bristol, UK, 2000.

[5] T. Bellemans, B. De Schutter, and B. De Moor. Model predictive control for ramp metering of motorway traffic: A case study. *Control Engineering Practice*, 14(7):757–767, July 2006.

[6] A. Bemporad, W. Heemels, and B. De Schutter. On hybrid systems and closed-loop MPC systems. *IEEE Transactions on Automatic Control*, 47(5):863–869, 2002.

[7] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35(3):407–427, 1999.

[8] S. Blandin, D. Work, P. Goatin, B. Piccoli, and A. Bayen. A general phase transition model for vehicular traffic. *SIAM Journal on Applied Mathematics*, 71(1):107–127, 2011.

[9] E. F. Camacho and C. A. Bordons. *Model Predictive Control*. Springer-Verlag, London, 2004.

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

[11] R. C. Carlson, I. Papamichail, M. Papageorgiou, and A. Messmer. Optimal motorway traffic flow control involving variable speed limits and ramp metering. *Transportation Science*, 44(2):238–253, May 2010.

[12] M. Castro-Neto, Y. Jeong, M. Jeong, and L.D. Han. Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions. *Expert Systems with Applications*, 36(3, Part 2):6164 – 6173, 2009.

[13] C. Chen. Freeway performance measurement system (PeMS). Institute of transportation studies, research reports, working papers, proceedings, Institute of Transportation Studies, UC Berkeley, 2003.

[14] C. Coello. Theoretical and numerical constraint handling techniques used with evolutionary algorithms: A survey of the state of the art. *Computer Methods in Applied Mechanics and Engineering*, 191(11-12):1245–1287, 2002.

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

[16] C. F. Daganzo. Requiem for second-order fluid approximations of traffic flow. *Transportation Research Part B: Methodological*, 29(4):277–286, 1995.

[17] Revista de Obras Públicas. Puente del Centenario. Sevilla (España). 139(3.316):126–127, December 1992.

[18] B. De Schutter and B. De Moor. Optimal Traffic Light Control for a Single Intersection. *European Journal of Control*, 4(3):260–276, 1998.

[19] Diario de Sevilla. Las matemáticas como método para evitar atascos en el puente del centenario. 18th February 2015.

[20] E. Alcantara de Vasconcelos. Traffic optimization on congested highway systems: The anchieta-lmigrantes case. In *lTE1988 Compendium of Technical Papers*, page 344, 1988.

[21] C.C. de Wit. Best-effort highway traffic congestion control via variable speed limits. In *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*, pages 5959 –5964, dec. 2011.

[22] G. Dervisoglu, G. Gomes, J. Kwon, R. Horowitz, and P. Varaiya. Automatic calibration of the fundamental diagram and empirical observations on capacity. In *88th Annual Meeting Transportation Research Board*, Washington, DC, USA, January 2009.

[23] El Economista. Estudian cómo mejorar la fluidez del tráfico en el puente del v centenario con modelos matemáticos. 18th February 2015.

[24] A. Ferrara, A. Nai Oleari, S. Sacone, and S. Siri. Case-study based performance assessment of an event-triggered MPC scheme for freeway systems. In *Proceedings of the 2013 European Control Conference (ECC)*, pages 4027–4032. IEEE, July 2013.

[25] C.A. Floudas. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Topics in Chemical Engineering. Oxford University Press, USA, 1995.

[26] J.R.D. Frejo and E.F. Camacho. Analyzing the trade-off between a first order (LN-CTM) and a second order (METANET) macroscopic model for freeway traffic control. *Submitted to Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*.

[27] J.R.D. Frejo and E.F. Camacho. Control predictivo cooperativo para el control de redes de tráfico mediante el uso de seales ITS. In *XXXII Jornadas de Automática*, Sevilla, Spain, September 2011.

[28] J.R.D. Frejo and E.F. Camacho. Feasible Cooperation based Model Predictive Control for freeway traffic systems. In *50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pages 5965 –5970, December 2011.

[29] J.R.D. Frejo and E.F. Camacho. Global versus Local MPC algorithms in freeway traffic control with ramp metering and variable speed limits. *IEEE Transactions on Intelligent Transportation Systems*, 13(4):1556–1565, Dec 2012.

[30] J.R.D. Frejo, E.F. Camacho, and R.Horowitz. A parameter identification algorithm for the METANET model with a limited number of loop detectors. In *51th IEEE Conference on Decision and Control(CDC)*, pages 6983 – 6988, Maui, Hawai, USA, December 2012.

[31] J.R.D. Frejo, A. Núñez, B. De Schutter, and E.F. Camacho. Hybrid model predictive control for freeway traffic using discrete speed limit signals. *Transportation Research Part C: Emerging Technologies*, 46(0):309 – 325, 2014.

[32] J.R.D. Frejo, A. Núñez, B. De Schutter, and E.F. Camacho. Analyzing the robustness of model predictive controllers for freeway traffic. To be submitted.

[33] J.R.D. Frejo, A. Núñez, B. De Schutter, and E.F. Camacho. Model predictive control for freeway traffic using discrete speed limit signals. In *12th IEEE European Control Conference (ECC)*, pages 4033 – 4038, Zurich, Switzerland, July 2013.

[34] J.R.D. Frejo, I. Papamichail, M. Papageorgiou, and E.F. Camacho. Macroscopic modeling and control of reversible lanes on freeways. *Submitted to IEEE Transactions on Intelligent Transportation Systems.*

[35] A.H. Ghods, A.R. Kian, and M. Tabibi. A genetic-fuzzy control application to ramp metering and variable speed limit control. In *Conference on Systems, Man and Cybernetics*, January 2008.

[36] G. Gomes and R. Horowitz. Optimal freeway ramp metering using the asymmetric cell transmission model. *Transportation Research Part C: Emerging Technologies*, 14(4):244 – 262, 2006.

[37] A. Hegyi. *Model Predictive Control for Integrating Traffic Control Measures.* PhD thesis, Delf Center of Systems and Control, Delft University of Technology, February 2004.

[38] A. Hegyi, B. De Schutter, and H. Hellendoorn. Model predictive control for optimal coordination of ramp metering and variable speed limits. *Transportation Research Part C*, 13(3):185–209, June 2005.

[39] A. Hegyi, B. De Schutter, H. Hellendoorn, and T. van den Boom. Optimal coordination of ramp metering and variable speed control - an MPC approach. In *Proceedings of the 20th American Control Conference, Vols 1-6*, pages 3600–3605, Anchorage, AK, USA, May 2002.

[40] A. Hegyi, B. De Schutter, and J. Hellendoorn. Optimal traffic control in freeway networks with bottlenecks. In *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, July 2005. Paper 3750 / Mo-M18-TO/5.

[41] A. Hegyi and S.P. Hoogendoorn. Dynamic speed limit control to resolve shock waves on freeways - field test results of the specialist algorithm. In *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, pages 519 –524, sept. 2010.

[42] A. Hegyi, S.P. Hoogendoorn, M. Schreuder, H. Stoelhorst, and F. Viti. Specialist: A dynamic speed limit control algorithm based on shock wave theory. In *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*, pages 827 –832, oct. 2008.

[43] A. Hegyi, B. De Schutter, and J. Hellendoorn. Optimal coordination of variable speed limits to suppress shock waves. *IEEE Transactions on Intelligent Transportation Systems*, 6(1):102–112, 2005.

[44] S. P. Hoogendoorn and P. Bovy. State-of-the-art of vehicular traffic flow modelling. *Special Issue on Road Traffic Modelling and Control of the Journal of Systems and Control Engineering Proceedings of the IME*, 215(4):283303, 2001.

[45] S.P. Hoogendoorn, M. Schreuder, and S. Hoogendoorn-Lanser. Calibrating dynamic network models for traffic operator decision support. *Transportation Research Record: Journal of the Transportation Research Board*, pages 58–66, 2007.

[46] A. Ingimundarson, C. Ocampo-Martinez, and A. Bemporad. Model predictive control of hybrid systems based on mode-switching constraints. New Orleans, LA, 2007.

[47] Texas Transportation Institute. *Managed Lanes Website http://managed-lanes.tamu.edu/. (accessed: 02/11/2013).* 2013.

[48] L.N. Jacobson, K.C. Henry, and O. Mehyar. *Real-time Metering Algorithm for Centralized Control*. Washington (State) Department of Transportation and Washington State Transportation Center and Washington State Transportation Commission and United States. Federal Highway Administration. Washington State Transportation Center, 1989.

[49] Z. Jia, C. Chen, B. Coifman, and P. Varaiya. The pems algorithms for accurate, real-time estimates of g-factors and speeds from single-loop detectors. *IEEE 4th International ITS Conference*, February 2001.

[50] S. Kallas, European Commission. 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.

[51] B. Kamel, A. Benasser, and D. Jolly. Ramp metering by limitation of density and queue length. In *2nd Mediterranean Conference on Intelligent Systems and Automation*, March 2009.

[52] S. Kimbrough, G. Koehler, M. Lu, and D. Wood. On a feasible-infeasible two-population (fi-2pop) genetic algorithm for constrained optimization: Distance tracing and no free lunch. *European Journal of Operational Research*, 190(2-16):310–327, 2008.

[53] A. Kotsialos, M. Papageorgiou, C. Diakaki, Y. Pavlis, and F. Middelham. Traffic flow modeling of large-scale motorway networks using the macroscopic modeling tool METANET. *IEEE Transactions on Intelligent Transportation Systems*, 3(4):282–292, December 2002.

[54] A. Kurzhanskiy and P. Varaiya. Using Aurora Road Network Modeler for Active Traffic Management. pages 2260–2265, 2010.

[55] J. P. Lebacque. *The godunov scheme and what it means for first order traffic flow models*. J. B. Lesort, editor, 13th ISTTT Symposium, pages 647678, Elsevier, New York, USA, 1996.

[56] J.Q. Li. Discretization modeling, integer programming formulations and dynamic programming algorithms for robust traffic signal timing. *Transportation Research Part C: Emerging Technologies*, (4):708–719.

[57] M. Lighthill and G. Whitham. On kinematic waves. II. A theory of traffic flow on long crowded roads. In *Royal Society of London. Series A, Mathematical and Physical Sciences*, volume 229, pages 317–345, may 1955.

[58] D. Limon. *Control predictivo de sistemas no lineales con restricciones: estabilidad y robustez*. PhD thesis, Universidad de Sevilla, September.

[59] D.A. Linkers and M. Mahfonf. *Advances in Model-Based Predictive Control*. Chapter Generalized Predictive Control in Clinical Anaesthesia. Oxforf University Press, 1994.

[60] S. Liu, J.R.D. Frejo, A. Núñez, B. De Schutter, A. Sadowska, H. Hellendoorn, and E.F. Camacho. Tractable robust predictive control approaches for freeway network. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, Qingdao, China, October 2014.

[61] S. Liu, A. Sadowska, J.R.D. Frejo, A. Núñez, E.F. Camacho, B. De Schutter, and H. Hellendoorn. Tractable scenario-based min-max receding horizon parameterized control for freeway networks based on multi-class traffic models. *Submitted to International Journal of Robust and Nonlinear Control*, 2015.

[62] X. Lu, T. Z. Qiu, R. Horowitz, A. Chow, and S. Shladover. METANET: Model improvement for traffic control. In *14th International IEEE Conference on Intelligent Transportation Systems, 2011*, page 536541, October 2011.

[63] X. Lu, P. Varaiya, R. Horowitz, D. Su, and S.E .Shladover. A new approach for combined freeway variable speed limits and coordinated ramp metering. In *13th Annual Conference on Intelligent Transportation Systems*, September 2010.

[64] K. F. Man, K. S. Tang, and S. Kwong. *Genetic Algorithms: Concepts and Designs with Disk*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2nd edition, 1999.

[65] A. Messmer and M. Papageorgiou. METANET: A macroscopic simulation program for motorway networks. *Traffic Engineering & Control*, 31(8-9):466–470, 1990.

[66] Z. Michalewicz. Do not kill unfeasible individuals. In Dabrowski, Michalewicz, and Ras, editors, *Proceedings of the Fourth Intelligent Information Systems Workshop (IIS'95)*, pages 110–123, Augustow, Poland, 5–9 June 1995.

[67] Z. Michalewicz and G. Nazhiyath. Genocop iii: a co-evolutionary algorithm for numerical optimization problems with nonlinear constraints. In *IEEE International Conference on Evolutionary Computation*, volume 2, pages 647–651, 1995.

[68] A. Morente. El V Centenario soporta el doble de tráfico que cuando se abrió. *El Correo de Andalucía*, 26th December 2010.

[69] L. Múñoz, X. Sun, R. Horowitz, and L. Alvarez. A piecewise-linearized cell 3 transmission model and parameter calibration methodology. *Journal of the Transportation Research Board*.

[70] L. Múñoz, Xiaotian Sun, R. Horowitz, and L. Alvarez. Traffic density estimation with the cell transmission model. In *American Control Conference, 2003. Proceedings of the 2003*, volume 5, pages 3750 – 3755 vol.5, june 2003.

[71] L. Múñoz, S. Xiaotian, G. Gomes, and R. Horowitz. Methodological calibration of the cell transmission model. In *Proceedings of the American Control Conference*, Boston, USA, July 2004.

[72] A. Muralidharan and R. Horowitz. Optimal control of freeway networks based on the link node cell transmission model. In *American Control Conference (ACC), 2012*, pages 5769 –5774, june 2012.

[73] A. Núñez and B. De Schutter. Distributed fuzzy confidence interval for traffic measurements. In *Proceedings of the 51th IEEE Control and Decision Conference (CDC)*, pages 1021–1026, Maui, Hawai, USA, 2012.

[74] A. Núñez, D. Sáez, S. Oblak, and I. Akrjanc. Fuzzy-model-based hybrid predictive control. *ISA Transactions*, 48(1):24 – 31, 2009.

[75] University of California at Berkeley. Tools for operational planning. http://gateway.path.berkeley.edu/topl/index.html.

[76] Minnesota Department of Transportation. Ramp meter study. 2001.

[77] A. Nai Oleari, J.R.D. Frejo, A. Ferrara, and E.F. Camacho. A parametrized model predictive control scheme for freeway traffic control. *To be submitted*.

[78] A. Nai Oleari, J.R.D. Frejo, A. Ferrara, and E.F. Camacho. A model predictive control scheme for freeway traffic systems based on the classification and regression trees methodology. In *Accepted for the 12th IEEE European Control Conference (ECC)*, Linz, Austria, July 2015.

[79] J. Gómez Ortega and E.F. Camacho. Mobile robot navigation in a partially structured static environment, using neural predictive control. *Control Engineering Practice*, 4(12):1669 – 1679, 1996.

[80] M. Papageorgiou. A new approach to time-of-day control based on a dynamic freeway traffic model. *Transportation Research Part B: Methodological*, 14(4):349 – 360, 1980.

[81] M. Papageorgiou. Some remarks on macroscopic traffic flow modeling. *Transportation Research Part A*, 32(5):323–329, 1998.

[82] M. Papageorgiou, J. Blosseville, and H. Haj-Salem. Modelling and real-time control of traffic flow on the southern part of boulevard peripherique in paris: Part II: Coordinated on-ramp metering. *Transportation Research Part A: General*, 24(5):361 – 370, 1990.

[83] M. Papageorgiou, J.M. Blosseville, and H. Haj-Salemn. Modelling and real-time control of traffic flow on the southern part of Boulevard Peripherique in Paris: Part I: Modelling. *Transportation Research Part A: General*, 24:345–359, 1990.

[84] M. Papageorgiou and M. Cremer. Parameter identification for a traffic flow model. *Automatica*, (17):837843, 1981.

[85] M. Papageorgiou, C. Diakaki, V. Dinopoulou, A. Kotsialos, and Yibing Wang. Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12):2043–2067, December 2003.

[86] M. Papageorgiou, E. Kosmatopoulos, and I. Papamichail. Effects of variable speed limits on motorway traffic flow. *Transportation Research Record*, 2047(-1):37–48, 2008.

[87] M. Papageorgiou, I. Papamichail, A. Messmer, and Y. Wang. *Traffic Simulation with METANET*, volume 145 of *International Series in Operations Research and Management Science*. Springer, 2010.

[88] Markos Papageorgiou, Habib Hadj-Salem, and Jean-Marc Blosseville. ALINEA: A local feedback control law for on-ramp metering. *Transportation Research Record*, (1320), 1991.

[89] Europa Press. El V Centenario soporta el doble de tráfico que cuando se abrió. 18th February 2015.

[90] M. Boero L. Breheret C. Di Taranto M. Dougherty K. Fox S. Algers, E. Bernauer and J. F. Gabard. Smartest - final report for publication. *Tech. rep., ITS, University of Leeds*, 2000.

[91] Y. Xi S. Lin, B. De Schutter and H. Hellendoorn. Fast model predictive control for urban road networks via milp. *Transactions on Intelligent Transportation Systems*, 1-11, March 2011.

[92] H. Sarimveis and G. Bafas. Fuzzy model predictive control of non-linear processes using genetic algorithms. *Fuzzy Sets and Systems*, 139(1):59 – 80, 2003.

[93] O. Slupphaug, J. Vada, and B. Foss. MPC in systems with continuous and discrete control inputs. American Control Conference, June 1997.

[94] A. Spiliopoulou, M. Kontorinaki, M. Papageorgiou, and P. Kopelias. Macroscopic traffic flow model validation at congested freeway off-ramp areas. *Transportation Research part C*, 41:18–29, 2014.

[95] X. Sun and R. Horowitz. A set of new traffic-responsive ramp-metering algorithms and microscopic simulation results. In *85th Annual Meeting of Transportation Research Board*, November 2005.

[96] H. Taale and F. Middelham. Ten years of ramp-metering in the netherlands. In *Road Transport Information and Control, 2000. Tenth International Conference on (Conf. Publ. No. 472)*, pages 106–110, 2000.

[97] C. Tampere, R. Corthout, D. Cattrysse, and L. H. Immers. A generic class of first order node models for dynamic macroscopic simulation of traffic flows. *Transportation Research Part B: Methodological*, 45(1):289–309, January 2011.

[98] C. Tampere and L.H. Immers. Traffic state estimation and prediction using the cell transmission model with implicit mode switching and dynamic parameters. In *Proceedings of the 86th Annual Meeting of the Transportation Research Board*, Washinton DC, USA, Jan 2007.

[99] E. van den Hoogen and S. Smulders. Control by variable speed signs: results of the dutch experiment. In *Road Traffic Monitoring and Control, 1994., Seventh International Conference on*, pages 145 –149, apr 1994.

[100] J.H. van der Lee, W.Y. Svrcek, and B.R. Young. A tuning algorithm for model predictive controllers based on genetic algorithms and fuzzy decision making. *ISA Transactions*, 47(1):53–59, 2008.

[101] A. Venkat, J. Rawlings, and S. Wright. Distributed model predictive control of large-scale systems. In Rolf Findeisen, Frank Allgower, and Lorenz Biegler, editors, *Assessment and Future Directions of Nonlinear Model Predictive Control*, volume 358 of *Lecture Notes in Control and Information Sciences*, pages 591–605. Springer Berlin / Heidelberg, 2007.

[102] E.I. Vlahogianni, M.G. Karlaftis, and J.C. Golias. Statistical methods for detecting nonlinearity and non-stationarity in univariate short-term time-series of traffic volume. *Transportation Research Part C: Emerging Technologies*, 14(5):351 – 367, 2006.

[103] Y. Wang, P. Coppola, A. Tzimitsi, A. Messmer, M. Papageorgiou, and A. Nuzzolo. Real-time freeway network traffic surveillance: Large-scale field-testing results in southern italy. *IEEE Transactions on Intelligent Transportation Systems*, (2):548–562.

[104] P. B. Wolshon and L. Lambert. *Convertible roadways and lanes: a synthesis of highway practice*. Transportation Research Board. National Research Council, 2004.

[105] L. Xiao-Yun, T.Z. Qiu, P. Varaiya, R. Horowitz, and S.E. Shladover. Combining variable speed limits with ramp metering for freeway traffic control. In *American Control Conference (ACC), 2010*, pages 2266 –2271, 30 2010-july 2 2010.

[106] L. Xiao-Yun, P. Varaiya, and R. Horowitz. An equivalent second order model with application to traffic control in transportation systems. In *12th IFAC Symposium on CTS*, 2009.

[107] Y. Yin. Robust optimal traffic signal timing. *Transportation Research Part B: Methodological*, 19(4):708–719, 2010.

[108] H. Zackor. *Speed Limitation on Freeways: Traffic-Responsive Strategies*. Pergamon Press, Oxford, United Kingdom, 1991.

[109] S.K. Zegeye, B. De Schutter, J. Hellendoorn, E.A. Breunesse, and A. Hegyi. A predictive traffic controller for sustainable mobility using parameterized control policies. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1420–1429, 2012.

[110] J. Zhihai, X. Zhang, and H. Yao. Analysis of variable speed limits on freeway calibration of a traffic simulation model by using factorial experiments and response surfaces. In *2004 IEEE Intelligent Transportation Systems Conference Washington, D.C.*, 2004.

Glossary and Acronyms

Acronyms

CT	Computation Time.
CTM	Cell Transmission Model.
FC-MPC	Feasible Cooperation based MPC.
FD	Fundamental Diagram of traffic flow.
GA	Genetic Algorithms.
LN-CTM	Link-Node Cell Transmission Model.
MILP	Mixed Integer Linear Programming.
MINLP	Mixed Integer NonLinear Programming.
MPC	Model Predictive Control.
QP	Quadratic Programming.
SQP	Sequential Quadratic Programming.
ST-MPC	MPC with Spatially and Temporally constrained VSL.
T-MPC	MPC with Temporally constrained VSL.
TR	Percent TTS Reduction.
TTS	Total Time Spent.
VMS	Variable Message Signs.
VSL	Variable Speed Limits.

Macroscopic Traffic Modeling

m	Link index.
i	Segment index.
N_m	Number of segments in freeway link m..

L_m	Length of the segments in link m (km).
λ_m	Number of lanes in freeway link m.
k_m	Discrete time index for the process model.
T_m	Simulation time step size.
$\rho_i(k_m)$	Density of segment i at time step k_m (veh/km/lane).
$v_i(k_m)$	Speed of segment i at time step k_m (veh/km/lane).
$q_i(k_m)$	Flow leaving segment i at time step k_m (veh/km/lane).
$v_{free,i}$	Free flow speed on segment i.
$q_{ramp,i}(k)$	Traffic flow entering the freeway from an on-ramp on segment i.
$\beta_i(k)$	Split ratio of an off-ramp on segment i.
$D_i(k_m)$	Traffic demand at origin o or at ramp on segment i at time step k_m (veh/h).
$w_i(k_m)$	Queue at origin o or at ramp on segment i at time step k_m (veh/h).
K_i	METANET speed anticipation term parameter (veh/km/lane).
τ_i	Time constant of the METANET speed relaxation term (h).
μ_i	METANET speed anticipation term parameter (km*km/h).
$\mu_{i,h}$	Anticipation constant for a downstream density that is higher that the density of the segment.
$\mu_{i,l}$	Anticipation constant for a downstream density that is lower that the density of the segment.
$V(\rho_i(k_m))$	Speed of segment i on a homogeneous freeway as a function of the density $\rho_i(k_m)$ (km*km/h).
δ_i	METANET parameter for the speed drop term caused by merging at an on-ramp on segment i.
ϕ_i	METANET parameter for the speed drop term caused by weaving at a lane drop on segment i.
$V_{c,i}$	Speed limit applied in segment i.
$V_{wVSL}(\rho_i(k_m))$	Desired speed without considering VSL (km).
α_i	Parameter expressing the non-compliance of drivers with the displayed VSL.
a_i	Parameter of the fundamental diagram.
b_i	Model parameter equivalent to a_i but only defined for the congested part..
$\rho_{crit,i}$	Critical density (the density corresponding to the maximum flow in the Fundamental Diagram).
$Q(\rho_i(k_m))$	Flow leaving segment i on a homogeneous freeway as a function of the density$\rho_i(k_m)$ (km*km/h).

$\bigtriangledown_{\mathrm{r}} v_i(k_m)$	Negative term added to speed equation where there is an on-ramp.
$\bigtriangledown_{\mathrm{d}} v_i(k_m)$	Negative term added to speed equation where there is a lane-drop.
$C_{\mathrm{ramp},i}$	Capacity of an on-ramp on segment i.
$\rho_{\mathrm{m},i}$	Maximum density for segment i (veh/km/lane).
$r_i(k_m)$	Ramp metering rate of on-ramp on segment i at time step k_m.
$v_{\mathrm{lim}}(k_m)$	Speed that limits the flow of the main-stream origin (km/h).
$q_{\mathrm{o},i}(k_m)$	Flow entering from the main-stream origin at time step k_m (veh/h).
$v_o(k_m)$	Virtual speed of a main-stream origin at time step k_m (km/h).
$\rho_{N+1}(k_m)$	Virtual density upstream the last segment N (veh/km/lane).
$q_{\mathrm{max},o}(k_m)$	Maximal inflow of a main-stream origin at time step k_m (veh/h).
$\bar{\lambda}$	Number of lanes with the reversible lanes open.
Δ_λ	Number of reversible lanes.
k_C	Time step when the reversible lanes are closed.
$D_c(k)$	Length of the revesible lane that is car-free.
$v_j(k_m)$	Speed of the segment which has a reversible lane partially closed at time step k_m.
$\widehat{\lambda}_i(k_m)$	Equivalent number of lanes during the period of time that there are still cars leaving the closed lanes..
k_O	Time step when the reversible lanes are open.
$D_o(k)$	Length of the revesible lane that is occupied by cars.
$\hat{x}(k_m)$	Measured state variables.
N	Total number of measures.
$R_i(k_m)$	CTM Demand Function.
$S_i(k_m)$	CTM Supply Function.
W_i	Congestion wave speed.
$\rho_{J,i}$	Jam density.

Freeway Traffic Control

N_{p}	Prediction horizon lenght.
k	Discrete time index for the controller.
N_{u}	Control horizon lenght.
T	Controller time step size.

$x(k_{\mathrm{m}})$	State variables.
$u(k_{\mathrm{m}})$	Controllable Input Variables.
$d(k_{\mathrm{m}})$	Non-controllable input vector.
$J(x_{\mathrm{t}}(k), u_{\mathrm{t}}(k), d_{\mathrm{t}}(k))$	Cost function.
$x_{\mathrm{t}}(k)$	State predictions along the prediction horizon N_p.
$d_{\mathrm{t}}(k)$	Non-controllable input predictions along the prediction horizon N_p.
$u_{\mathrm{t}}(k)$	Control sequence along the control horizon N_u.
$u^{\mathrm{c}}(k)$	Discrete input vector.
$u^{\mathrm{d}}(k)$	Continuous input vector.
$\mathbb{X}$	Physical or operational constraints for the state components.
$\mathbb{U}$	Physical or operational constraints for the continuous input components.
$\mathbb{S}$	Set of possible control value for the discrete input components.
$h(u_t(k), x_t(k), d_t(k)) \in \mathbb{D}$	Other linear and non-linear constraints.
$\Delta_{vel}, \Delta_{den}, \Delta_q$	Constraints tunning parameters.
x_k	Last measured state at time step k..
γ_i	Maximum change of a VSL during a controller sample time.
ζ_i	Maximum difference between the VSL values of two adjacent segments.
$V_{\mathrm{cST},i}(k)$	VSL of segment i computed by ST-MPC controller.
$r_{\mathrm{cST}}(k)$	ST-MPC solution for the ramp metering rates.
$N_l(N_{\mathrm{u}})$	Number of leafs.
θ	Maximum difference between the VSL discretization and the ST-MPC solution.
$R(k)$	Variable that indicates when the reversible lanes are open or closed.
N_{R}	Number of reversible lanes.
$L_c(k)$	Congestion Length.
V_{cg}	Speed threshold below which the system can be considered congested.
χ	Logic controller for reversible lanes parameter.
$q_{b,I}(k), q_{b,D}(k)$	Flows through the bottleneck in the increasing and decreasing directions.