Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : AUTOMATIQUE - PRODUCTIQUE

Arrêté ministériel : 25 mai 2016

Présentée par

Pietro GRANDINETTI

Thèse dirigée par **Carlos CANUDAS DE WIT** , Directeur de Recherche
et codirigée par **Federica GARIN**, INRIA

préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Contrôle de vaste réseau de trafic

Control of large scale traffic network

Thèse soutenue publiquement le **11 septembre 2017**,
devant le jury composé de :

Madame Simona SACONE
Professeur associé, University of Genova, Rapporteur
Monsieur Eduardo CAMACHO
Professeur, University of Seville, Rapporteur
Monsieur Giovanni DE NUNZIO
Ingénieur de recherche, IFP Energies nouvelles, Examinateur
Madame Paola GOATIN
Directeur de recherche, INRIA, Examinateur
Monsieur Alamir MAZEN
Directeur de recherche, CNRS, Président
Madame Federica GARIN
Chargé de recherche, INRIA, Examinateur

Abstract

The thesis focuses on traffic lights control in large scale urban networks. It starts off with a study of macroscopic modeling based on the Cell Transmission model. We formulate a signalized version of such a model in order to include traffic lights description into the dynamics. Moreover, we introduce two simplifications of the signalized model towards control design, one that is based on the average theory and considers duty cycles of traffic lights, and a second one that describes traffic lights trajectories with the time instants of the rising and falling edges of a binary signal. We use numerical simulations to validate the models with respect to the signalized Cell Transmission model, and microsimulations (with the software Aimsun), to validate the same model with respect to realistic vehicles behavior.

We propose two control algorithms based on the two models above mentioned. The first one, that uses the average Cell Transmission model, considers traffic lights' duty cycles as controlled variables and it is formulated as an optimization problem of standard traffic measures. We analyze such a problem and we show that it is equivalent to a convex optimization problem, so ensuring its computational efficiency. We analyze its performance with respect to a best-practice control scheme both in MatLab simulations and in Aimsun simulations that emulate a large portion of Grenoble, France. The second proposed approach is an optimization problem in which the decision variables are the activation and deactivation time instants of every traffic light. We employ the Big-M modeling technique to reformulate such a problem as a mixed integer linear program, and we show via numerical simulations that the expressivity of it can lead to improvements of the traffic dynamics, at the price of the computational efficiency of the control scheme.

To pursue the scalability of the proposed control techniques we develop two iterative distributed approaches to the traffic lights control problem. The first, based on the convex optimization above mentioned, uses the dual descent technique and it is provably optimal, that is, it gives the same solution of the centralized optimization. The second, based on the mixed integer problem aforesaid, is a suboptimal algorithm that leads to substantial improvements by means of the computational efficiency with respect to the related centralized problem. We analyze via numerical simulations the

convergence speed of the iterative algorithms, their computational burden and their performance regarding traffic metrics.

The thesis is concluded with a study of the traffic lights control algorithm that is employed in several large intersections in Grenoble. We present the working principle of such an algorithm, detailing technological and methodological differences with our proposed approaches. We create into Aimsun the scenario representing the related part of the city, also reproducing the control algorithm and comparing its performance with the ones given by one of our approaches on the same scenario.

Résumé

La thèse concerne le contrôle de feux tricolores dans de larges réseaux urbains. Le point de départ est l'étude d'un modèle macroscopique se basant sur le Cell Transmission model. Nous avons formulé une version du modèle intégrant les feux tricolores à sa dynamique. De plus, nous avons introduit deux simplifications à ce modèle orientées vers la conception des techniques de contrôle ; la première se base sur la théorie de la moyenne et considère le pourcentage de vert des feux tricolores, la seconde décrit les trajectoires des feux tricolores en utilisant les instants d'activation et de désactivation d'un signal binaire. Nous utilisons des simulations numériques pour valider les modèles en les comparant avec le Cell Transmisson model intégrant les feux tricolores, ainsi que des simulations microscopiques (avec le logiciel Aimsun) afin de valider les mêmes modèles en les comparant cette fois-ci à un comportement réaliste des véhicules.

Nous proposons deux techniques de contrôle à partir des deux modèles mentionnés ci-dessus. Le premier, qui utilise le modèle moyen de transmission de véhicules, considère les pourcentages de vert des feux tricolores comme variables contrôlées, et il est formulé comme un problème d'optimisation des mesures de trafic standards. Nous analysons un tel problème et nous montrons que cela équivaut à un problème d'optimisation convexe, afin d'assurer son efficacité de calcul. Nous analysons sa performance par rapport à un best-practice control à la fois dans des simulations MatLab, et dans des simulations microscopiques, avec un modèle Aimsun qui reproduit une grande partie de Grenoble, en France. La deuxième approche proposée est un problème d'optimisation dans lequel les variables contrôlées sont les instants d'activation et de désactivation de chaque feu tricolore. Nous utilisons la technique de modélisation Big-M dans le but de formuler un tel problème comme un programme linéaire avec variables entières, et nous montrons par des simulations numériques que l'expressivité de cette optimisation conduit à des améliorations de la dynamique du trafic, au prix de l'efficacité de calcul.

Pour poursuivre la scalabilité des techniques de contrôle proposées nous développons deux algorithmes itératifs pour le problème de contrôle des feux de signalisation. Le premier, basé sur l'optimisation convexe mentionnée ci-dessus, utilise la technique dual descent et nous prouvons qu'il est

optimal, i.e., il donne la même solution que l'optimisation centralisée. Le second, basé sur le problème d'optimisation entier susmentionné, est un algorithme sous-optimal qui mène à des améliorations substantielles par rapport au problème centralisé connexe, concernant l'efficacité de calcul. Nous analysons par des simulations numériques la vitesse de convergence des algorithmes itératifs, leur charge de calcul et leurs performances en matière de mesure du trafic.

La thèse est conclue avec une étude de l'algorithme de contrôle des feux de circulation qui est utilisé dans plusieurs grandes intersections dans Grenoble. Nous présentons le principe de fonctionnement d'un tel algorithme, en détaillant les différences technologiques et méthodologiques par rapport aux approches proposées. Nous créons dans Aimsun le scénario représentant la partie intéressée de la ville, en reproduisant également l'algorithme de contrôle et en comparant ses performances avec celles de l'une de nos approches sur le même scénario.

Contents

List of Figures

Chapter 1

Introduction

In this introductive chapter we will begin with a general overview on the topic of urban traffic control. Then, the objectives of the thesis will be specified, along with the scope of the conducted research. We will finally discuss the author's original contributions and outline the organization of this manuscript.

1.1 The traffic problem

At the time of writing Europe is the most urbanized continent in the world: over 80% of its population lives in towns and cities [112]. The run towards the urbanization has as direct consequence development and expansion of cities, which, in return, leads to increased vehicular traffic. The underlying motivation that brings people to congregate in large urban areas also causes often unbearable levels of traffic congestion in urban roads. When the demand for a roadway infrastructure is such that it exceeds the available supply then congestions arise. Also, congestion can be seen as the direct consequence of the obstruction caused by vehicles to each others movements; this is often the case when the whole infrastructure is operating at some condition close to its maximum capacity. Not that geographically different areas experience better situations; for instance in the United States of America traffic congestions from 2012 on are said to cause 5.5 billion hours of traffic delays, resulting in 2.9 billion gallons of wasted fuel, producing 56 billions lbs of carbon dioxide, and to have a total (estimated) cost of 121 billions of dollars [96].

It has certainly been over the past few years that the number of circulating vehicles has increased dramatically and traffic congestion has become a major problem, not only from a safety point of view, but also from an economic and environmental perspective. Congestion involves queuing, lower speeds, and increased travel times, which, other than imposing costs on the economy, also generates multiple impacts on urban regions and their inhabitants [61].

However, along with increased congestions, recent years have also brought some promising improvements, particularly in the areas where the Intelligent Transportation Systems (ITS) have been put into practice. When dealing with urban traffic, these systems bring innovation into the field concerning intelligent cars, smart traffic lights, variable speed limits and specialized sensing technologies.

This explains why the overall motivations to tackle the problem of improving traffic management are quite strong for traffic researchers and engineers. During the past years we have been witnessing different techniques and technologies adopted to deal with the problem.

1.2 History of road traffic control and scope of the thesis

Traffic control is a wide subject that has offered to engineers and practitioners space for research and innovation. The idea itself, to regulate transportation routes, is indeed very old and perhaps it was born together with the concept of transportation. Well documented sources (see, e.g., [43]) testify examples of *manual* traffic flow control (i.e., organized by traffic officers) on the London Bridge back in the year 1722.

The first mechanical mean of vehicular traffic control was a set of gas–lit traffic lights installed on 9th December 1868 outside the Houses of Parliament, London, as the *green plaque* on site cites, see figure 1.1. The purpose was to control traffic in three main avenues, namely Bridge street, Great George street and Parliament street. The idea belongs to John Peake Knight (1828–1886), who is thereafter credited as the inventor of traffic lights. He was a railway engineer and inventor, and indeed he borrowed the idea of traffic lights from the railway systems and proposed it to use in London in 1866, a year in which statistics say 1102 people were killed and 1334 injured on roads, in London. Its operational life was, unfortunately, very short: due to the poor technology available at that time, one of the traffic lights exploded on 2nd January 1869, killing (or injuring, according to other sources) the policeman who was operating it. This first, historical road traffic control system was therefore removed, falling out of favour.

It took the first two decades of the 20th century for traffic lights to establish themselves as the standard de facto. In 1912, a traffic control device was placed at the top of a tower in Paris at the crossing between Rue Montmartre and Grand Boulevard. This tower signal was manned by a police woman and she operated a revolving four-sided metal box on top of a glass showcase where the word *Stop* was painted in red and the word *Go* painted in white [43]. In the same year, on the other side of the Atlantic ocean, precisely in Salt Lake City, Utah, Lester Wire developed what is said

Figure 1.1 – The plaque in memory of J.P. Knight, credited of the invention of traffic lights.

to be the first electric traffic light using red and green lights. From that moment on, traffic lights never abandoned our streets.

Another old control mean for traffic system is speed limit. Like for traffic lights, United Kingdom was the field for innovation: after recognizing the *offense of furious driving*, the first speed limits were created in the Locomotive Acts, a series of Acts of Parliament regulating the use of mechanically propelled vehicles on British public highways. The first limits are certified to be 6 km/h in rural areas and 3 km/h in town.

These two ancient means of control, along with the construction of new infrastructures (roads, roundabouts, etc.) have been the lonely used technologies employed for a long time. However, space for constructions run out till the point that building a new road with the only purpose to ease traffic congestion is considered as a last resource nowadays. Fortunately, as space was running out, technologies and engineering were developing, leading to a huge boost to the earlier mentioned control systems as well as the development of novel and innovative techniques to deal with the traffic problem. We talk about to the so–called Intelligent Transportation Systems (ITS).

While we are used to think about ITS as incredibly complex systems connected via the internet, the first experiment of such dates back to mid–1960 [9], when prototypes of vehicles equipped with Driver Aided Information and Routing System (DAIR) was realized. Such a system could send an emergency message to a service center, including information on road conditions. The system relied on magnets buried at regular intervals along the road (generally between 3 to 5 miles apart) and used binary code to communicate location information. DAIR included a display panel on the car's dashboard that would show warning messages regarding road hazards and had a system that could guide a driver along a pre-determined route. General Motors installed this technology in 1966 in two vehicles and tested it at their testing center in Detroit, Michigan. Ultimately, General Motors could not muster the resources necessary to deploy the system. The two DAIR-equipped cars were never tested outside General Motors' facilities.

A real leap was achieved with the advent of map–matching algorithms,

Figure 1.2 – The installation of a traffic signal in San Diego in 1940. Source [117].

in the 1970s. Networks of roads were modeled in a digital map database, in which a particular route could be programmed mathematically and an onboard computer was used to analyze the current state of the vehicle and match its path to the programmed route. The first developer of such a system was Robert L. French, in 1971, who made it for a newspaper delivery route and included a pre–recorded voice message played at appropriate points during the journey.

The great development in the 1980s, 1990s and early 2000s brought a number of technologies we are now accustomed to. Loop detectors have become the most widely used sensors to measure (or estimate) speed, flow and occupancy. Dynamic message signs (electronic traffic signs that provide information and warnings to the travelers) are used everywhere for a variety of messaging purposes. Ramp management (or *metering*) is the usual way to control the access to highways. Traffic management centers, hubs dedicated to collect and process data about traffic in highway as well as in towns, are built in almost all mid–large cities. The Global Positioning System (GPS), originally designed during the 1960s for military purposes, is an easily accessible resource, constantly used to track vehicles.

The impressively diversified set of technologies available today leaves the question *what's next* open. Autonomous vehicles are certainly not a mirage anymore (see [59], [103] and [100]); interconnecting applications for best path selection are on the market and flagship products of millionaire entreprises ([116, 106]); the need for an accurate mapping of the available routes has been satisfied (e.g., [42], [54] and [80]); car–sharing, intelligent traffic lights and adaptive speed limit are already in place. The real challenge is to choose how to use them.

1.2.1 Scope of the thesis

Road traffic control is commonly divided into two categories, that are highway control and urban control [84]. The two often share some technological point. Even though highways were initially conceived so as to allow unlimited mobility to road users, the increasing traffic demand has led to recurrent congestions also there. Control means that are typically employed are ramp metering, link control and guidance systems. Urban traffic control, instead, deals with the strategies employed to account for congestion within towns.

The large number of control strategies for urban traffic may be partitioned into vehicle control and infrastructure control (notice that such a distinction could be used also for highway control, leading there to different techniques). The term vehicle's control includes a broad set of approaches and applications, such as recommended systems for routing, trajectory selection, platooning, that rely on in–car sensing and processing. Infrastructure control, instead, deals with managing the available roads' infrastructure that is, in a way, imposed to travelers.

Two examples of infrastructure control are adaptive speed limits and smart traffic lights. The idea of adaptive speed limit is to equip roads with dynamic message signs showing the current maximum allowed speed limit on that road. The controller adjusts such limit according to its processing.

This thesis is positioned within the frame of smart traffic lights. With this name we intend the software and hardware system that relies on detectors information from the roads which are given as input to a processing unit, whose purpose is to decide values of green time for each traffic light so as to deal with traffic congestion. Typically, traffic lights control consists of either single intersection (isolated) control, or coordinated control, that can be realized via a centralized or a distributed architecture.

1.3 Objectives and contributions of this work

This thesis has been developed as part of the european project Scalable Proactive Event-Driven Decision-making (SPEEDD, EU-FP7 619435)[1]. The goal of the proactive traffic management use case within this project is to make decisions in order to attenuate congestions. Indeed, eliminating urban congestion is neither an affordable, nor feasible goal. Nevertheless, a lot can be done to reduce its occurrence and its impact on road users. The idea of proactive traffic management is to employ modern software and hardware to mitigate the congestions' effect, particularly when the traditional approach of expanding the existing infrastructure to increase capacity, although helpful, results to be too costly, besides being ecologically

[1] *http://speedd-project.eu*

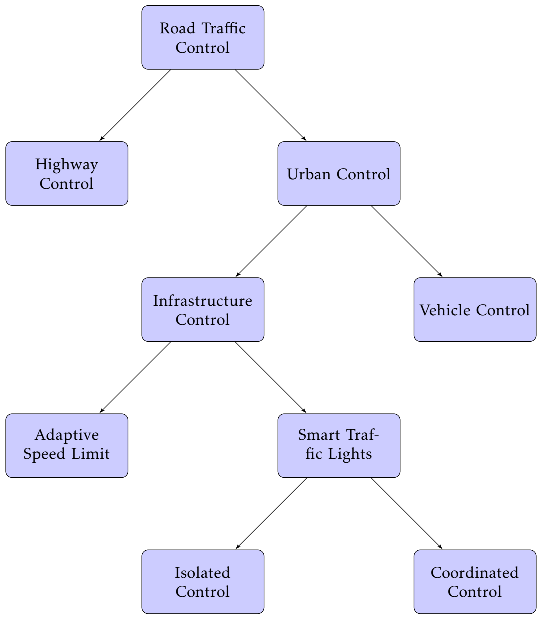


Figure 1.3 – Scope of the thesis.

intrusive and space demanding. In fact, many cities all over the world have already taken actions, or plan to, in order to address these problems and to achieve their objectives, typically about reducing road accidents, prioritizing public transport, reducing emissions, etc.

The primary objective of this thesis is to develop strategies for urban traffic control that are efficient by means of their computational cost and reproducible in the field. The goal of these control strategies is to provide an improvement of the system's behavior with respect to existing strategies, in terms of standard traffic performances (e.g., travelling time, travelled distance, total time spent, density balancing). Such indexes are standard measures of infrastructures effectiveness, and they will be discussed more in detail in Chapter 4.

As for the selected actuators, we focus on the control of traffic signals split (i.e., the green time allocated to every traffic lights). All of the proposed control strategies use macroscopic model–based prediction of the network state evolution (see figure 1.4).

The computational efficiency of the controllers will be enhanced not only by developing optimization–based strategies that can be efficiently computed, but also by proposing a distributed architecture. In constrast to a classical centralized controller, a distributed design allows us to compute the optimal decision by using only local state information and some supplementary information arriving from the neighboring controllers. As a result, the total length of the communication channel will be reduced, the probability of losing information will decrease, and instead of solving a

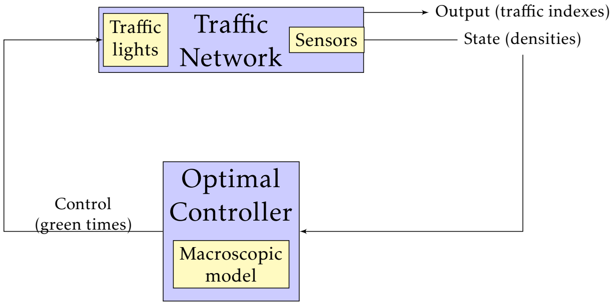


Figure 1.4 – Feedback strategy for traffic networks controlled via traffic lights.

very large optimization problem the task will be split into several, smaller, problems. Overall, this type of design ensures a better scalability of the system.

1.3.1 Contributions

The contributions of this work to the urban traffic control lie in the traffic lights management, regarding both isolated and coordinated control, and it can be summarized as follows.

First, we propose a solution for a global efficient controller of traffic lights green time. We develop a macroscopic representation of traffic dynamics that includes the average value of green time over a fixed assigned cycle time. Then, we illustrate how standard urban traffic metrics can be used as objective functions of an optimization problem that considers short–term predictions of vehicles density based on the aforesaid model; most importantly, we also show that such an optimization problem can be rewritten as an equivalent convex problem, that guarantees its computational efficiency. These results are partially published in [44].

Secondly, we propose a solution for a global controller of traffic lights scheduling. Revisiting the macroscopic modeling, we develop a variant of it that is much more expressive than the average representation before mentioned. By doing so, it is possible to use similar traffic metrics to formulate an optimization problem that uses a longer prediction horizon to foresee value of vehicles density in the network. This second approach gives important benefits regarding the network's performance, but it results very costly. This part of our research has been published in [46].

To account for the scalability of the algorithms we focus on the development of distributed control techniques. We design a distributed algorithm based on the average macroscopic model using techniques of convex and distributed optimization. We prove that this control algorithm is optimal, that is, we show that its solution is equivalent to the solution of the corresponding centralized problem. We continue along the same direction proposing a distributed algorithm for the latter control technique mentioned in the earlier paragraphs. Unlike the previous distributed strategy, this one is suboptimal in terms of cost function value, but leads to great benefits in terms of computational costs. These results are partially presented in [45, 46].

Lastly, we present an in–depth study of an industrial traffic lights controller that is used in many important intersection in Grenoble, France. This city is also considered as study case for the european project. Our contribution consists in a reverse engineering of the industrial controller, its replication in a microscopic simulated environment and in comparing its performance with one of our proposed approach. In particular, we show how the proposed control strategy based on the average macroscopic model can be used to design an isolated control that can be compared with the industrial one (which is an isolated control, too). These results are still unpublished, although an article in this respect is planned.

1.4 Structure of the manuscript

The organization of the rest of this manuscript is as it follows:

- In chapter 2 we present a review of the state of the art of scientific works on the road traffic control. The chapter follows the partition proposed in figure 1.3, focusing large parte of its content on scientific literature about traffic lights control, which is the most relevant for our work;
- Chapter 3 is devoted to presenting the elements of traffic modeling, including the novel models we designed towards control applications. We will start by clarifying what are the several levels of abstraction when modeling traffic flow. We will continue by describing the highest level of abstraction of traffic flow dynamics, based on a mass conservation law. Then, we will present successive approximations of such a law, showing their strenghts and weaknesses, and what they have been designed for. Eventually, we will present a microscopic traffic simulator, discussing in what regards it is useful, and present comparisons between the various models;
- Chapter 4 serves as preliminary for the presentation of the control schemes. We present and discuss the metrics most commonly used as traffic indexes, and also used for evaluation in our work, as well a general overview on the control task that we seek to adress;
- In chapter 5 we present the first of the proposed solutions for the traffic lights control. The main idea is to simplify the optimization task by considering a simplified dynamics that, on one hand, preserves the

main features of the traffic behavior, on the other it includes green times as decision variables, used therefore as control input;

- In chapter 6 we present the second proposed solution for the centralized traffic lights control. This will be an algorithm in charge of deciding the traffic lights schedule by using model prediction that are very accurate but also more expensive, due to the model precision and to the prediction horizon;
- Chapter 7 illustrates our design of distributed strategies. We develop a distributed version of the first proposed solution, that is provably optimal, and a distributed suboptimal algorithm related to the second of the centralized proposed algorithms.
- In chapter 8 we present our work about experiments with the industrial traffic lights plan that is deployed in many large intersections in the city of Grenoble, and compare it with one of the proposed techniques.

Every chapter also contains numerical experiments (via either macroscopic or microscopic simulation, or both), as well as comments and illustrations of the obtained numerical results.

Chapter 2

Review of previous research

In this chapter we give a summary of the previous approaches to the road traffic control problem. The chapter follows the partition proposed in section 1.2.1 (see figure 1.3), focusing large parte of its content on scientific literature about traffic lights control, which is the most relevant for our work.

2.1 Highway ramp metering

The first forms of ramp control were applied in Chicago, Detroit and Los Angeles areas [8]. Since then, a variety of both heuristic and optimal strategies have been developed.

Many ramp metering systems are based on fixed-time strategies. This means that the design is based on the analysis of the historical demand profiles only, and implemented without using the real-time measurements. Moreover, fixed-time strategies are usually activated only in the peak-hours. One of the first proposed approaches, authored by Wattleworth [115], relied on the simple policy of regulating the entering flows such to guarantee that the mainstream flow at each of the sections does not exceed the capacity. This idea was then extended, and some variants of it have been proposed in [113], [97] and [114]. As pointed out in [84], due to the absence of real-time measurements, the fixed-time strategies may overload the mainstream flow resulting in congestion. On the other hand, if the expected (through the historical data) demand is overestimated, then the ramp metering may lead to freeway underutilization.

Nowadays, several existing ramp metering systems use local (isolated) or coordinated regulators fed with real-time measurements, and typically their the goal is to keep the traffic state close to some pre-defined value. Tthe most famous example of local feedback ramp-metering strategy is perhaps ALINEA, introduced by Papageorgiou et. al [82]. ALINEA was shown to be remarkably simple, highly efficient and easy to implement, and due to these facts it became one of the most commonly used ramp metering strategy. The idea is to regulate the local vehicles density at the set point that is usually represented by the critical density. As a result, the local (downstream to the on-ramp) flow is meant to be kept around the capacity. It was shown [81] that ALINEA is not very sensitive to the choice of the regulator parameter.

Some extensions to ALINEA have been proposed. METALINE [82] can be seen as a multivariable extension of ALINEA: the metering rate of each ramp is computed based on the change in the measured occupancy of each freeway segment and on the deviation of the occupancy from the critical occupancy for each segment. Like the ALINEA algorithm, the METALINE algorithm is theoretically robust and easy to implement. The main challenge in METALINE is the proper choice of the control matrices and the target occupancy vector [120].

Proposed by Papamichail et al. [85], HERO is another extension to ALINEA. HERO employs a feedback regulator where the target is the critical occupancy for throughput maximization which is considered more robust than targeting a pre-specified capacity value. As reported by the authors, HERO outperforms uncoordinated local ramp metering and gets close to the efficiency of sophisticated optimal control methods.

Model–based control techniques have received numerous attentions for application in highway networks. For instance in [17] the receding horizon approach is used to control ramp metering and speed limits in freeway traffic with several classes of vehicle. The seeked optimization deals with the minimization of the total time spent by the vehicles in the network and with regularization of the controlled variables, but the general approach results to be quite flexible in terms of the objective cost; for example similar ideas allow in [86] to take into account the emissions of a two–class freeway network, and to design a nonlinear optimal control for emissions' reduction. Variations of this technique have also been explored, for example event–triggered model predictive control for freeway systems has been studied in [34]. In this case, the receding horizon problem is solved via mixed–integer programming and, to reduce the computational load, the optimization is run only when some events are triggered. The combination of predictive control and integer optimization has often been exploited in traffic control, because it fits well the modeling side of traffic flow; for example, in the recent work [35] these techniques have been used to take decisions about the innovative system of reversible lanes. The work uses a discrete model predictive control that minimizes the total time spent of the modeled network within some constraints for the maximum values of the generated bottleneck queues.

Several fuzzy logic controllers have also been tested for the ramp metering applications. Fuzzy logic algorithms aim to convert the experimental knowledge about a traffic system into fuzzy rules. The traffic conditions as occupancy, flow rate, velocity and ramp queue are split into categories (e.g., small, average and large). Then, a set of rules are designed to relate traffic conditions with the metering rates. For the interesting results, the reader is referred to the papers [66], [60].

2.2 Vehicle control

The largest part of the scientific literature about vehicle control offers, from high perspective, two approaches: path optimization, related to the selection of the best path for the vehicles, by means of time spent or distance traveled and generally achieved via vehicles routing, and enviroment optimization, related to reduction of emissions and ECO–friendly vehicles management.

The Vehicle Routing Problem (VRP) historically plays a central role in the fields of physical distribution and logistic, and more recently in the field of dynamical traffic assignment. Approaches to this problem generally start from the Dijkstra's algorithms [30], constantly revisited and extended to increasingly complex scenarios. For instance, a broad class of tractable routing algorithms in stochastic networks has been studied in [94]; solutions to the *on–time arrival* problem via routing have been studied in [14], by exploiting the ever increasing penetration rate of mobile systems and by the application of the Bellman's principle [79]. Theoretical understanding of such a problem and techniques for efficient solutions have been developed in [95].

Important theoretical properties of distributed routing algorithms, such as robustness, responsiveness and resilience, have been extensively analyzed by Como et al. [23, 24] in dynamical networks (i.e., networks whose evolution is described by a system of ordinary differential equations), that generalize the traffic networks case.

The System Optimum Dynamic Traffic Assignment (SO–DTA) problem, that was introduced in [76] and [77], has received significant interest from the transportation research community, see [87] for a review. Differently from its original purpose, SO–DTA is also being increasingly used as a framework to compute optimal control for traffic flow over freeway networks (see, e.g. [78]), when traffic controllers aim to minimize a global cost of the whole network – hence Social Optimality – as opposed to the single-vehicle oriented user equilibrium modeling frameworks. Recent approaches to the problem propose a combination of several control actions (variable speed limit, ramp metering and routing) to design optimal controls for traffic flows over a given time horizon; we refer the reader to [22].

Environment optimization is also named vehicle–ECO management. The main purpose is the adoption of an energy-aware driving style, since the driving behavior can have a big impact on emissions, as demonstrated by several studies [109]. A possible classification of these techniques is [99]:

- *Pre–trip systems.* These are systems integrated within navigation systems. For any given start and destination points, and for defined time windows for departure or arrival, the system calculates optimal start time and route, based on car and driver's characteristics, so as to minimize environmental impact of the journey. Pre-trip advising devices include in-vehicle equipment, the internet, phone services, mobile devices.

- *In–trip systems.* They are part of the broader category of Advanced Driver Assistance Systems (ADAS). As a main interface, the majority of systems use visual displays, while audible alerts and haptic gas pedal are applied in only a few solutions. In-trip advice can be provided either as an information-only message or as an explicit route recommendation. Interestingly, survey results witness that travelers prefer the less intrusive information and to be able to make their own decisions [107].

- *Post–trip systems.* These systems are an attempt to increase the driver's motivation for eco-driving by displaying encouraging results, and generating summaries and statistics that can be compared to other drivers.

Most approaches are based on heuristic rules of thumb or good practices that are associated with an energy-efficient drive. Moreover, only a few are predictive, that is, based on estimates of future external conditions, while the rest is solely based on current driving information. However, several concepts are emerging in Vehicle Eco-Management that attempt at implementing eco-driving in a more rigorous framework. We refer the interested reader to the Ph.D. thesis [27].

2.3 Urban infrastructure control

Traffic congestion is created, to put it simply, every time the demand exceeds the capacity of the road and/or the intersection. This is the fundamental reason why basically all infrastructure control policies aim at reducing the demand/supply ratio.

Some advanced ideas which involve interaction with the drivers have been tested in practice, in form of advice systems provided to the users of the network. The key idea is that most drivers are familiar with the traffic conditions in the network because, for example, they travel it on a daily basis. Therefore, they naturally optimize their individual routes based on their past experience, thus leading to a sort of user-equilibrium condition. However, several events can perturb such an equilibrium, with the drivers not necessarily aware of it. For example, daily varying demands, changing weather conditions, exceptional aggregation events (sport events, fairs, concerts, etc.) and, most importantly, incidents, create conditions very different from the ones the daily experience would suggest, and thus they may lead to an under-utilization of the overall available facilities. In these contexts, a type of recommended system that is gaining popularity is the display on electronic message panels of the expected travel times in particular locations of the network. This kind of information is easy to understand and can help with the choice of the path. An example of such, about 350 variable message signs are installed on the Boulevard Périphérique of Paris and on its on-ramps.

Other than information on the travel time, displays with speed suggestion have been already tested in the 80s (and became more popular with time). Cleary, this kind of method relies on the compliance of the drivers, but in urban scenarios, where following such advices can allow to avoid to stop at traffic lights, it is reasonable to expect a high compliance, which in turn makes us talking of speed limit control.

The physical means most often used for this purpose are signs displaying electronically controlled digits, placed at the side of the road about a hundred meters past the previous traffic light. Such signs suggest a certain speed to the drivers, in order to catch the green light at the forthcoming signalized intersection. Quite simply, the suggested speed value is calculated from the knowledge of how many seconds are left before the traffic light turns green. Under proper working conditions, the complying drivers will catch the green; those who reject the suggestion will most likely fail to catch it, and meet instead a red.

Clearly, the suggested speed is higher the farer from the intersection the panel is, until a speed greater than the legal maximum would be necessary to catch the end of the present green cycle. If that is the case, the driver is advised with the speed that will get her to the first seconds of the next green phase.

It is interesting to compare this strategy with traffic–responsive signal timings control. The latter usually favor vehicles stream with high density, and a weakness of them is traffic dispersion that often results in some stream to be forced to stop. On the other hand, speed advisory signs work well against the dispersion phenomenon, increase traffic flow and hence increase also the capacity of the considered infrastructure. In general, smoother traffic flow, more relaxed and less competitive drivers, increased safety, and reduced energy consumption and pollution are among the benefits of this strategy.

One of the very first test of this strategy took place in Düsseldorf, Germany, more than 30 years ago [11]. Even though no detailed evaluation

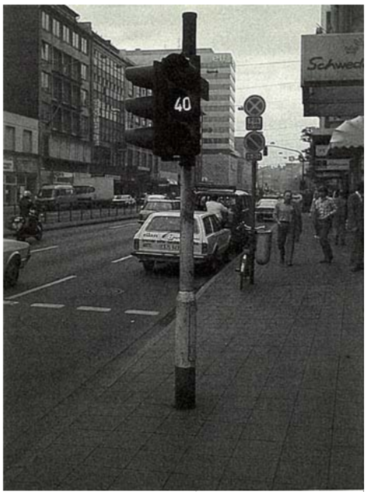


Figure 2.1 – From [11]: «Advisory speed signs have worked well in Dusseldorf for 25 years.»

of its performances was documented, it is reported that, after some time, almost 95% of the drivers were complying to the suggested speed, which in a sense gives an idea of its effectiveness.

It must also be said that the more drivers comply with the advisory system, the better the strategy works, and consequently its implementation appears more meaningful. However, all kind of practical problems can arise: first of all, some people will not be willing to drive slowly when they see an open road ahead; also, when the advisory panel goes temporarily blank (before giving the next suggestion to catch the next green phase), drivers may be tempted to speed up to reach the preceding platoon and try to pass through the signalized intersection at an amber light or, even worse, a red light. Furthermore, a number of drivers can always be expected to be willing to drive slower than the signs indicate. Moreover, busses or pedestrians can easily interfere with the system, especially when changing lane is not easy (e.g., in heavy traffic conditions).

Speed limit control, when adressed from the infrastructure point of view, presents also the interesting side of making the driver feel part of an intelligent, co-operative network. We have presented only short comments about it; recent works have attempted to evaluate the impact of lowering the speed limits in urban areas, and demonstrated benefits in terms of safety and energy consumption [7].

2.3.1 Traffic lights control

Traffic lights are the most forcing type of control, in the sense that the control is imposed over the system with the underlying assumption that the compliance of the users is almost 100% (which is quite true in practice). There exist very comprehensive and exhaustive surveys on road traffic control strategies, such as [84]. In the following, the classification will be made according to the level and the complexity of communication and information that the different strategies rely on. Specifically, we will distinguish among isolated management strategies, which use only local information, and coordinated strategies, which use pieces of information coming from the network as input parameters to the algorithm that has to take decisions about the control measures.

The former of the two aforesaid types of policy is known as *isolated strategy*. These can be either fixed-time or responsive. The fixed-time isolated strategies are known to be applicable to under–saturated traffic conditions, and they are stage–based, when they are in charge of selecting the optimal splits and cycle time so as to minimize the total delay (i.e. the time wasted due to congestion) or maximize the intersection capacity, or phase–based, when they also decide the optimal sequence of phases. This last feature can be quite important for complex intersections, although the controller's task is much more difficult in such cases.

The most famous among the stage-based strategies are perhaps SIGSET and SIGCAP, developed by Allsop in the 1970s' [5, 4]. Both algorithms are fed with predefined stages and are in charge of determining splits and cycle times at one intersection only. Two important features of theirs are the capacity constraints (set in order to avoid queue building) and constraints on maximum cycle and minimum green duration. As for their realization, SIGSET solves a nonlinear programming problem to minimize the total intersection delay for given traffic demands, while SIGCAP solves a linear programming problem and may be used to maximize the intersection capacity.

The stage-base approach is extended by the phase-based approaches [56], in order to consider different staging combinations. These are more complex strategies that are based on integer optimization. However, since the objective is to define a fixed–time combination of stages, the computation is executed offline and therefore the inefficiency is considered a minor drawback. In any case, the decision variables are the optimal staging, splits, and cycle time, and the objective is to minimize total delay or maximize the intersection capacity; the solution is typically searched with branch–and–bound methods.

An important reference about isolated strategies is represented by the study of the over–saturated condition and delay minimization for a single intersection, initially adressed by Gazis in the early 1960s [38]. Prior to his work, it was known that, in presence of saturation arising in both directions, the minimum delay objective is achieved by exhausting both queues at the same time (for fixed-time scheduling). Gazis proposed a minimum–delay algorithm for the off-peak hours, and a maximum–capacity strategy for peak hours. The former should be replaced by the second as soon as congestions appear in the incoming roads. Gazis' work has been recently improved, due to an inexact solution to the minimum delay problem, and therefore a new optimal control for the isolated intersection has been proposed [51, 58].

A different approach to the single intersection managament problem is based on the concept of self–organizing traffic lights [40, 41, 63]. The ideas shared by all methods in this family are the fully decentralized policy and the preference assigned to vehicles that have been waiting longer, or to larger groups. In some scenario these policies proved to be more efficient than simple fixed–time strategies; however, some instability issues may occur in the case of saturation of the traffic conditions, since the traffic lights are not coordinated. So, an extension to the previously cited works have been proposed in [64], that includes a supervisory control level in order to increase the robustness of the strategy for higher traffic flows.

We will now continue the exposition by discussing coordinated strategies. Coordinated signalized intersections control is of central importance when facing the task to schedule traffic lights in a network of urban arterials. For example, under ideal (free flow) conditions, if coordinated properly, traffic lights can give raise to the so–called *green waves*, the phenomenon where catching a green light at one intersection allows to catch it at the downstream junction too. In presence of congestion, that is admittedly the most common case, green waves are not possible anymore and different measures are required.

Historically, great contribution to the subject was given by invention of the first fixed–time methodologies. According to these strategies, intersection management is applied on the network at pre–set time instants, which are computed (or we should rather say, suggested) by empirical observations and historical statistical data about peak and off-peak hours. The most famous instances of fixed–time, coordinated, strategies are MAXBAND and TRANSYT.

MAXBAND [70, 71], designed by the pioneer John Little, is also one of the first works of this kind. It considers a two-way arterial with a given number of intersections, and is based on a concept known as *arterial bandwidth*: the maximum number of vehicles that can catch the green light at every intersection, therefore not stopping, and in given speed range, in the two directions of travel. MAXBAND works with the objective to find values for the offset that increase the arterial bandwidth. In the original formulation some variables represent the arrival times at the intersection and it was therefore necessary to introduce additional decision variables, binary valued. This lead Little to formulate a binary-mixed-integer-linear-programming problem. Ideas to improve the computational burden speeding up the branch–and–bound solution method by using specific properties of the problem are illustrated in [21]. MAXBAND has been applied in several road networks in North America, and a few significant extensions have been proposed, also by Little himself, trying to deal with finer aspects such as time of clearence of existing queues, left–turn movements, etc. (see MULTIBAND, [37, 102]).

TRANSYT is perhaps the most famous among the offline (fixed–time) traffic signals control, and was developed by Robertson et al. [93]. The main ingredients of this control technique are a model that estimates evolution in time of the queues length and an heuristic optimization algorithm, based on hill-climbing. Its objective is to minimize the sum of the average queues length in the chosen area, and the decision variables are splits, offsets, and cycle time. The scheme implemented by TRANSYT is iterative: for given values of the decision variables the dynamic network model calculates the corresponding performance index (e.g., the total number of vehicle stops) and the heuristic hill-climbing searches for variations of the decision variables that cause an update to the model; these two steps are repeated until a (local) minimum is found. The key knowledge for this algorithm to work properly is the data on average flows, maximum flows, cruise times and so on that have to be provided as inputs to the program. The main limitation of this strategy is that these data are collected most of the time physically by workers on the roadside, and such a task is tedious as well as time consuming. Moreover, this information is stored and not systematically updated, hence the "offline" nature of this strategy. Nevertheless, the first field implementations of TRANSYT produced savings of about 16% of the average travel time through the network.

Clearly, the main drawback of fixed-time strategies is that their settings are based on historical rather than real-time data. This is related to the important question about periodicity of the traffic conditions: even though one would expect traffic behavior to exhibit repetitive features (for example along same hours of the day, in different days), very often the small inevitable diversities can lead to substantially different behaviors. Not to mention that they can also change occasionally due to special events. Static strategies result to be a blind approach to the traffic control problem for several reasons, therefore a better solution is definitely provided by the so-called *traffic–responsive* algorithms, which are designed to trigger appropriate measures in response to the actual traffic conditions. However, more effective strategies do not come for free, and they need more advanced technologies: usually, they use sensors to retrieve traffic data (for instance, by using magnetic loops embedded in the asphalt) and central control rooms for the processing of the data. Also, installation and maintenance costs have to be considered. Differently said, dynamic strategies are more reliable and efficient, but also more costly. The most famous among the traffic responsive strategies are SCOOT and SCATS, which we will describe shortly.

SCOOT [55] was developed by the same team that authored TRAN-

SYT, and indeed was thought to be a dynamical version of the latter. It has been very successful: the number of reported deployment is more than 150, among others in cities such as Santiago of Chile, Beijing, Sao Paulo, and several in the United Kingdom. It is based on a dual design: architecturally, SCOOT is runnig on a central machine; functionally, it is decentralized. SCOOT's core is a traffic model that runs in real time and an optimization procedure that is in charge to decide splits, cycle times and offset, as the name itself suggests (Split Cycle Offset Optimization Technique). The real time measurements are retrieved by sensors located well upstream from stop lines at junctions and ideally right after the previous intersection, and consist of the average one-way flow of vehicles past an intersection during each cycle time. Under good working operations, these sensors' position allows to have a very early information about arrivals in the monitored section, and also detect if the queue fills up completely the road. Using this data, the model calculates the length of the queues in the whole controlled area and the optimization works by looking at different directions: the split optimizer computes whether it is beneficial to advance, to delay or to unchange the scheduled splits; the offset optimizer calculates whether the objective (the sum of the average queues in an area, as in TRANSYT) can be reduced around each junction by changing the offset. Splits and offsets are adapted immediately, cycle times of a group of intersections are similarly modified every few minutes. SCOOT makes a large number of decisions, approximately ten thousand per hour on a network counting one hundred intersections. However, this system is not robust in case of over-saturated traffic conditions. If the queue reaches the upstream detector, SCOOT is not able to understand the presence of stationary vehicles, and it may mistake the situation for a lower demand, decrease the duration of green time at the downstream intersection and cause in turn an increase in congestion.

SCATS [74] is another successful industrial controller that has been developed at the same time as SCOOT, although in a diametric opposite place (Australia), and it is indeed a shortname for Sydney Coordinated Adaptive Traffic System. Fruitful applications of SCATS have been in Honk Kong and in several cities in Europe, Australia, and USA. Let us describe the main differences between SCOOT and SCATS, in order to detail SCATS' working principles, by classifying the main features across the two of them. From a model point of view, SCOOT uses a model of the entire network under exams, and it measures flows in real time. SCATS, instead, splits the whole system into several regional subsystems, such that each of them contains several junctions. Moreover, in every subsystem one junction is elected as the critical intersection. From computational point of view, SCOOT update the internal model, that represents the entire network, about once every five seconds. Moreover, such computations are executed on a central machine. On the other hand, SCATS needs to decide about the critical

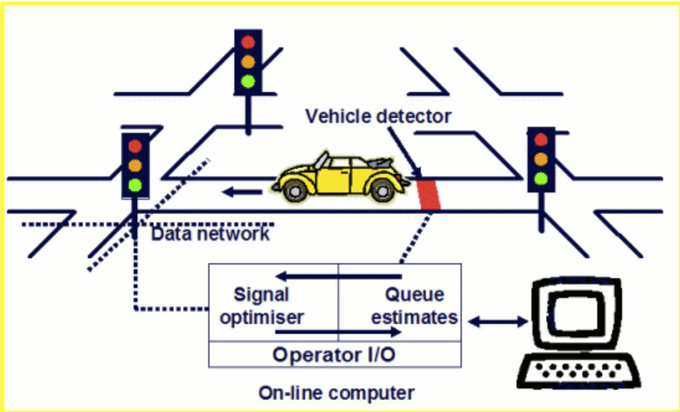


Figure 2.2 – How SCOOT works, from [98]. Robertson, the main author of both TRANSYT and SCOOT, more than 20 years ago said: «I find it difficult to believe that, as we approach the end of this century, traffic engineers and drivers will continue to tolerate signals with green and red times that were decided by the flows and queues that happened to be observed on one day many years earlier, rather than in the last five minutes» [92].

junction for every subsystem, that may change over time; furthermore, the traffic lights plan is optimized for the critical junction, and therefore plans for the other junctions have to be computed separately. Regarding the optimization process, SCOOT computes novel values of cycle time every two to five minutes, of splits every stage and of offsets every cycle; SCATS, instead, computes the cycle times and the splits every cycle, while the offsets are not optimized and simply precomputed. Regarding the measurements needed by the two controllers, SCOOT uses one type only of sensors, placed upstream the intersection to measure flow, occupancy and queues length. SCATS uses rather two types of sensors, one near the stop line at the intersection, to measure flow and occupancy, and one (optional) upstream to measure queues. As for the tuning of the parameters, SCOOT needs some model's parameters such as the travel time between every detector and the stop lines, that, however, are set offline and do not need any update. SCATS' documentation does not report information about the model parameters. An investigation conducted by the Australian research board showed no advantage of SCATS in terms of travel time with respect to TRANSYT. There was however a reduction in the number of stops, improved by about 9% in the central area, and 25% on main arterials. As for SCOOT, results show a total delay reduction of about 12% as compared to TRANSYT, and this should get better with time, as the fixed-time plans become outdated.

Following the wave created by SCOOT and SCATS, which both remain the most used industrial controllers, a number of model-based, more rigorous, traffic responsive algorithms have been developed: OPAC [36], RHODES [101], PRODYN [53], CRONOS [13], UTOPIA [75]. Unlike SCOOT and SCATS, these newer strategies consider the total time spent in the network by the vehicles as the main objective, that has to be minimized. The optimization is carried out using a typical receding horizon control approach, where the rolling horizon adopted in the online optimization is rather long (i.e., about sixty seconds), but the results of the optimization are applied on a much shorter time window (i.e., about four seconds), before the algorithm is run again using new data information. Furthermore, they do not consider explicitly cycle times, offsets and splits, but rather tries to find the traffic lights scheduling, consisting of the switching sequence in the near future. In these control policies the optimization is in charge of solving an online dynamic problem subject to realistic traffic model with a sample time of the order of few seconds and fed with traffic measurements. Discrete variables are employed to describe traffic lights behavior, and several constraints are imposed on the inputs, such as minimum and maximum green signals duration, etc.

Thus, the combination of a traffic model, a receding horizon approach, and complex decision variables (the switching sequence) causes the presence of discrete variables to model the alternating nature of the traffic lights, which requires exponential complexity algorithms for a global optimization. This means that, in practice, a global optimization is not possible and therefore alternative solutions are searched by the control. For instance, OPAC utilizes an online optimizer based on a complete enumeration to generate integer switching sequences, while PRODYN and RHODES use dynamic programming; in both cases, the computational burden is such that an online implementation for more than one intersection is not feasible and therefore these three strategies are decentralized. In the case of PRODYN and RHODES, the optimization task is split into multiple optimization subsystems, whose results are coordinated by a supervisory control layer. In particular, in RHODES, the upper level of the control strategy also performs a prediction of future traffic conditions to allow a proactive behavior of the controllers at the lower level. Clearly, when facing such a difficult task there is plenty of space for original heuristic. CRONOS keeps the optimization at the global level but uses a heuristic method with polynomial complexity which allows for simultaneous management of several junctions, although it results in a local minimum. UTOPIA, instead, is based on the concept of a fully variable cycle on a rolling horizon. In it, the local controller first of all uses a branch-and-bound technique over a horizon of two minutes that searches for the optimization of a local cost function accounting the queue length integrals and exceeding link storage capacity at all incoming roads of the junction. Then, each local controller receives traffic lights timing plans from the neighboring junctions, and information about queues, capacity, saturation flows and a reference nominal cycle for the specific area. Using this information, the local controller is able to assess how its signal timings affect downstream intersections and how neighboring nodes influence its released traffic, so deciding actions. UTOPIA is currently active in Turin, Italy.

It is interesting to summarize the points that differentiate these newer algorithms from their ancestors SCOOT and SCATS. Mainly, the former have been designed in order to overcome the limitations of the latter, often moved by the critisms, towards SCOOT and SCATS, about their lack of a real traffic responsiveness during rapidly changing conditions, and the poor robustness to oversaturated conditions. In the attempt to solve these issues, UTOPIA and the others are based on algorithms relying on computationally heavy optimization and on a dual working principle that prescribes a decision–making module decentralized at intersection level and a centralized coordination level. As a results, none of the aforementioned queue management strategies is truly scalable and applicable to large-scale networks, which perhaps is the reason why SCOOTS and SCATS are still the most used by far. A second important point, that has given further motivation to the scientific reasearch on traffic lights control, is that the industry–oriented scope of these solutions makes it difficult to fully understand their functioning principles, also because of their complex hierarchical and layered architecture. Therefore, we devote the remainder of the section to review traffic lights control strategies that are well documented in the literature.

The Traffic Urban Control (TUC) design [29] was motivated by the limitations of sensors failure and large scale applicability of the industrial solutions previously mentioned. This control solution is based on the concept of feedback control, which should preserve a good implementation simplicity, at the same time not compromising its efficiency [31, 83], and on the store–and–forward modeling approach [38]. From technological point of view, it is based on the real–time measurement of occupancy only, because the algorithms encloses a mathematical model to calculate the average vehicles density from the measured occupancy. Moreover, the measurement is needed only one time per cycle since one of the optimization modules, the split optimization, is the one in charge to decide values for the green splits once per cycle. There are three more modules in TUC, one for the cycle length decision, one for the offset, and one more to take care of public transport priority. In particular the split and the cycle optimization are based on global information and applied network–wide: this should guarantee a truly optimal solution, and a better robustness to sensors failure. TUC found practical implementation and employment in Glasgow and Southampton, UK, and Chania, Greece, often outperforming the resident traffic control strategies (e.g., SCOOT in the UK). The key mathematical ideas have been extended in subsequent works by the same authors [2, 3].

The store and forward modeling has received numerous attentions, but there exist several propositions of different approaches. More in detail, scientific works that deal with traffic lights control via a different modeling approach always look for macroscopic modeling techniques, since the microscopic models are usually too complex and do not fit model–based control applications. It is very often the case of works proposing a traffic model strictly related to a control scheme designed at the same time. For example, a macroscopic model, embedding enough details so as to include more specific features of traffic networks, such as different types of vehicles, pedestrian movements and amber phases, has been proposed in [10] and further extended in [32]; these works also propose control algorithms dedicated to minimize the number of vehicles in the road network by acting on splits of the traffic lights. Other modeling and control solutions for a network–wide control of traffic signals have been proposed in [68, 69]. In these works, the controller makes use of model–based prediction in order to minimize the total time spent by the vehicle in the network.

In order to solve the issues of scalability and real-time applicability of the control measures, only in recent times a few interesting distributed control strategies have been presented. In [18] the traffic network has been linearly modeled and the control objective is to discharge vehicle queues while penalizing deviation from a nominal control plan, by acting on the stage times of the traffic lights. In [105], the Cell Transmission Model has been used to develop a distributed control strategy via the alternating direction method of multipliers.

Traditionally, the problem of traffic lights control (or scheduling) has been investigated by the control system community as well as the computer science one. A recent proposition that tried to merge the two, and inspired further research is the max–pressure algorithm, defined for networks of signalized intersection in [119, 110, 111]. The max–pressure is indeed based on the scheduling algorithms used for packet transmission in wireless networks, and is a fully decentralized algorithm based on local calculations that only require the knowledge of queues length in the adjacent roads, at a given time. Unlike in the model–based approaches, where both the average demands and the split ratios need to be known and where biased model–based predictions lead to biased results of the optimization, the max–pressure requires the knowledge of the turning percentage only. Also, possible extensions to relax this assumption have been studied, e.g., [47]. Moreover, one the main feature of the max–pressure is that it is a stabilizing control which maximizes the throughput; however, it can be successfully computed only if the stabilization of the network is possible. By contrast, model–based optimization techniques do not offer guarantees about the stabilization, but are always capable to give a feasible solution to the problem.

Yet another different perspective has been explored in the so–called perimeter control [39, 1, 50]. Based on experimental results about the existence of a macroscopic fundamental diagram (MFD) for geographical areas, the idea is to partition the city into regions, each corrisponding to a different MFD. Then, the controller operates on the borders between the regions and manipulates the percentages of flows that transfer between the regions such that the number of trips that reach their destinations is maximized.

Chapter 3

Macroscopic representation of urban traffic

Accurate descriptions of traffic flow over transportation networks requires a good understanding of traffic operations. In particular, insights into what causes congestions, and how they propagate through the network, are of central importance. Theory and experiments require enormous efforts to be put into the field, therefore modeling and numerical simulations play a role of dominant importance as well. They represent extremely valuable tools to describe and replicate congestion phenomena, as well as to support and improve the design, test and validation of the traffic management strategies.

The discussion in this chapter will go through the various elements of traffic modeling used in the thesis. We will start by clarifying what are the several levels of abstraction when modeling traffic flow. We will continue by describing the highest level of abstraction of traffic flow dynamics, based on a mass conservation law. Then, we will present successive approximations of such a law, showing their strenghts and weaknesses, and what they have been designed for. Eventually we will present a microscopic traffic simulator, discussing in what regards it is useful, and illustrate comparisons between the various models illustrated.

3.1 Modeling levels

Traffic theory and practice distinguish between three levels of abstraction for traffic dynamics:

Microscopic modeling. It includes car-following models and most cellular automata. All models in this category describe individual vehicles as "particles". Each and every particle is characterized by instantaneous reaction of every driver, e.g., accelerating, braking, lane-changing. In exchange for such a fine degree of description, these models are based on complex equations and they require high computation effort.

Macroscopic modeling. It describes traffic flow analogously to fluids in motion. Indeed, models in this class are sometimes called hydrodynamic models. Macroscopic models are able to describe collective phenomena such as the evolution of congested regions or the propagation velocity of traffic waves. Furthermore, they are useful when specific aspects of the traffic, such as lane change, different classes of vehicles, etc., do not need to be considered, and when computation time is critical.

Mesoscopic modeling. It combines microscopic and macroscopic approaches to a hybrid model, partly using features from both of them. The level of detail for a mesoscopic model is less than a microscopic and greater than a macroscopic model, since it does not distinguish individual vehicles, but specify the behavior of *individuals*. To this end, the transportation elements are divided in small groups, in which elements are considered homogeneous. A typical example is vehicle platoon dynamics and household-level travel behavior.

The work in this thesis deals with microscopic and macroscopic modeling. The precise context will be clarified in a later section, here we limit ourselves to say that macroscopic models will be used for control synthesis, while a microscopic one will be used for validation and simulative tests.

3.2 The Cell Transmission Model

The conventional macroscopic approach to simulate traffic behavior over networks keeps track of the number of vehicles in discrete sections of the network as time passes. The vehicle occupancy in each section is increased by the number of vehicles allowed to enter the section in each time interval, and is decreased by the number of vehicles allowed to leave it. This reminds the modeled behavior of a gas into a finite volume.

The most used instance of a continuous macroscopic traffic model is the Lighthill-Whitham-Richards (LWR) model [67, 90]. Let us define

- x, the spatial coordinate,
- t, the time coordinate,
- ρ, the density of vehicles,
- f, the flow of vehicles.

The LWR is a continuous model of the form:

$$\frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x} f(\rho(x,t)) = 0, \tag{3.1}$$

where density and flow are linearly related, i.e., $f(\rho) = \rho v(\rho)$, with v the average velocity of vehicles.

Under such definitions, the outflow becomes a function of the occupancy of x only, and it is not explicitly dependent on the occupancy downstream. This is unreasonable in several cases and can produce erroneous results, because traffic could be sent forward even when there is no space available downstream. In order to overcome this deficiency, Daganzo introduced the Cell Transmission Model (CTM) [25, 26], which is undoubtedly one of the main and most famous traffic models to date. It is a macroscopic model too, and it is based on a first order Godunov approximation of the LWR [65].

In the CTM, roads are divided into homogeneous sections (or, cells), indexed by natural numbers i starting from 1. Every cell is described by a set of phisical properties:

$\rho_i^{\max}$, the maximum density, often referred as *jam* density,

$f_i^{\max}$, the capacity flow,

v_i, the maximum velocity of vehicles in the cell, said the *free–flow* velocity,

w_i, the speed of the congestion wave's backpropagation.

These quantities are correlated according to the *fundamental diagram*, see figure 3.1 and the explanation therein.

We will now present the model following Daganzo's footsteps. Let us define

$S_i(t)$, the supply of cell i, i.e., the maximum flow that can enter cell i,

$D_i(t)$, the demand of cell i, i.e., the maximum flow that can leave cell i.

For the sake of clarity, we notice that Daganzo's seminal work presents the CTM by using the number of vehicles $n_i(t)$ as main variable. However, the notions of density and number of vehicles are interchangeable, being $\rho_i(t) = n_i(t)/L_i$, where L_i is the length of cell i.

The formal definitions of demand and supply are as it follows: both are affine in the density with a saturation imposed by the capacity flow, i.e.,

$$D_i(t) = \min\left(v_i\rho_i(t), f_i^{\max}\right) \tag{3.2a}$$

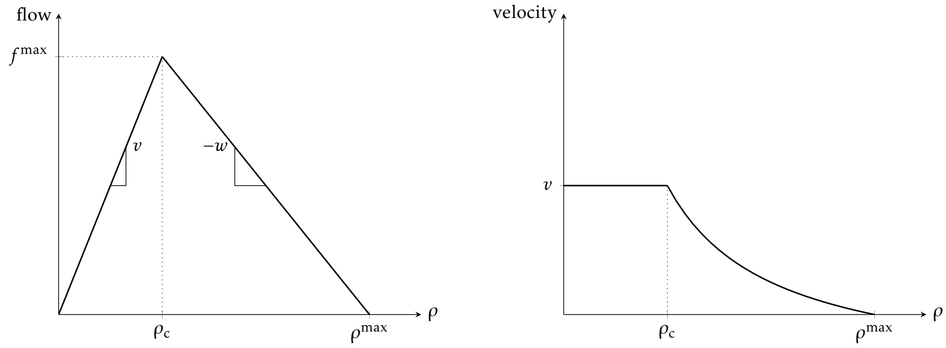


Figure 3.1 – Fundamental diagrams density-flow and density-speed that describe correlation between important quantities in traffic systems. The value of density ρ_c is usually said *critical density*, and it represents the switching point between a free-flow condition and a congested one.

$$S_i(t) = \min\Big(w_i(\rho_i^{\max} - \rho_i(t)), f_i^{\max}\Big) \tag{3.2b}$$

Let us start by considering the case of a single road divided into cells (for example, a highway): in this case, every cell is connected exactly to one cell both upstream and downstream, except for the boundary cells. Then, the flow that can pass from cell $i-1$ to cell i is

$$f_i^{\text{in}}(t) = \min\Big(D_{i-1}(t), S_i(t)\Big).$$

Similarly, the flow passing from cell i to cell $i+1$ is computed as

$$f_i^{\text{out}}(t) = \min\Big(D_i(t), S_{i+1}(t)\Big),$$

and, of course, in this simple case without junctions, it holds $f_{i-1}^{\text{out}}(t) = f_i^{\text{in}}(t)$. Under the assumption of fundamental triangular diagram, it is also possible to express the flow inside a cell as

$$f_i(t) = \min\Big(v_i\rho_i(t), w_i(\rho_i^{\max} - \rho_i(t))\Big). \tag{3.3}$$

As such, the derivative of the density of vehicles in a cell evolves according to the following rule:

$$\dot\rho_i(t) = \frac{1}{L_i}\Big(f_i^{\text{in}}(t) - f_i^{\text{out}}(t)\Big). \tag{3.4}$$

Equation (3.4) creates a continuous-time model that is mathematically bounded, thanks to the demand–supply paradigm. This means that spillbacks are formally not possibles; in practice, queues of vehicles cannot grow without limits. From a different point of view, (3.4) gives a model that is simpler to simulate and analyze than the original LWR model. To do so, the continuous equation is discretized with the Euler method. This completes the discretization of the original variables x and t both in space (via the cells definition) and time, and it gives

$$\rho_i(t+T_s) = \rho_i(t) + \frac{T_s}{L_i}\Big(f_i^{\text{in}}(t) - f_i^{\text{out}}(t)\Big), \tag{3.5}$$

where T_s represents the chosen time step length, and t is now meant to be the discrete time coordinate.

It is worth mentioning that, under the fundamental triangular diagram, the CTM's cells operate by switching among two states: a free-flow state, where vehicles move with speed v and density is lower than the critical density ρ_c, and a congested state, where congestion backpropagates with speed w, and density is greater than ρ_c. When a cell is in the first condition, its inflow will be bounded by its maximum capacity, while its outflow will be bounded by $v_i\rho_i$. This means that the eigenvalue of the discrete–time system (3.5), for the i–th cell, will be given by $1 - \frac{T_s}{L_i}v_i$. Therefore, to ensure the stability of the discrete–time CTM, the sampling time has to chosen such as to respect

$$\frac{T_s}{L_i}v_i \leq 1. \tag{3.6}$$

On the other hand, since $v > w$, the same condition also guarantees that $\frac{T_s}{L_i}w_i \leq 1$, that is the stability condition for the congested cells.

In [25] important properties of the CTM are highlighted, among which:

- The model (3.5) is a discrete approximation of the original model given by Lighthill, Whitham and Richards (3.1);

- The density profile in the CTM moves with the wave speed accordingly to the hydrodynamic theory, i.e., $\rho(x,t) = \rho(x - wt, 0)$;

- The values of density that are computed at every time instant as novel values (i.e., computed at time t in order to be assigned as $\rho(t+T_s)$) do *not* depend on the order in which cells are considered. This feature arises because the model prescribes that the number of vehicles entering a cell is unrelated to the number of vehicles that leave it at the same time instant. Such a property will be extremely important when extending the CTM to complex networks that, as a matter of fact, contain loops.

3.3 The signalized CTM for urban networks

We introduce an extension of the original Cell Transmission model in order to take into account two key features of traffic networks: the splitting of flows and the presence of traffic lights in proximity of junctions.

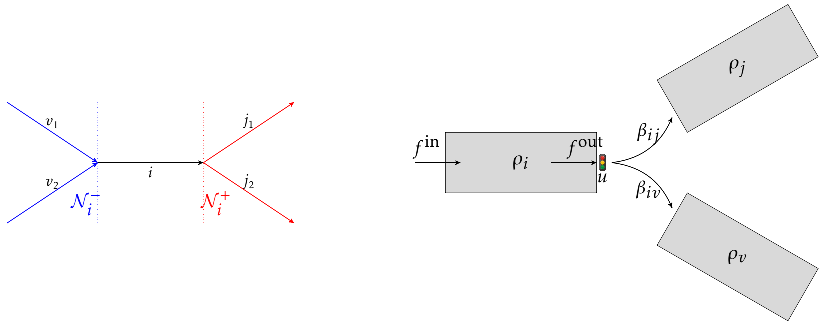


Figure 3.2 – The notation used in the urban traffic model.

An urban network is a collection $\mathcal{R}$ of roads, among which there are entering roads ($\mathcal{R}^{\text{in}}$) and exiting roads ($\mathcal{R}^{\text{out}}$). The locations where two or more roads merge are called *intersections*. Such intersections have no capacity storage and they are regulated by traffic lights, which guarantee the safe merging of flows. Formally, a traffic light is a function

$$u : \mathbb{N}_+ \to \{0,1\},$$

where $\mathbb{N}_+$ is the set of all time instants. Hence, the traffic flow exiting each road i is regulated by the traffic light u_i, which alternates green and red phases within a cycle time of length T.

Whether two roads i and j are connected to the same intersection in such a way that flow can exit i and enter j, we say that i belongs to the set of upstream neighbors of j, a set indicated with $\mathcal{N}_j^-$. Similarly, the set of downstream neighbors of i, $\mathcal{N}_i^+$, is defined as the set of all roads that may directly receive flow exiting i.

In order to describe the splitting of flows at intersections we associate to every pair of roads i,j a value $\beta_{ij} \in [0,1]$ (called *split ratio*, or at times turning proportion), which expresses the percentage of the flow exiting i which turns into j. Then, $\beta_{ij} = 0$ if and only if $j \notin \mathcal{N}_i^+$ and $\beta_{ij} = 1$ if and only if $\mathcal{N}_i^+ = \{j\}$. Clearly, it must be $\sum_{j\in\mathcal{N}_i^+} \beta_{ij} = 1, \forall\, i \in \mathcal{R} \setminus \mathcal{R}^{\text{out}}$; see figure 3.2 for an illustration of these notations.

The simplest CTM form can be extended to account intersections by adopting a first-in first-out rule, see also [26]. In other words, the outflow of a road i will be the maximum flow respecting constraints given by the demand D_i as well as by the supply S_j for all roads $j \in \mathcal{N}_i^+$. Formally, this corresponds to the solution of the following problem,

$$\begin{aligned} f_i^{\text{out}}(t) &= \max \phi \\ \text{subj. to: } \phi &\le D_i(t) \\ \beta_{ij}\phi &\le S_j(t) \,\forall j \in \mathcal{N}_i^+, \end{aligned}$$

that is a simple linear program whose solution is given by (assuming for simplicity all split ratios are stricly greater than zero)

$$f_i^{\text{out}}(t) = \min\left(D_i(t), \left\{\frac{S_j(t)}{\beta_{ij}}\right\}_{j\in\mathcal{N}_i^+}\right). \tag{3.7}$$

Equation (3.7) expresses the flow that can leave a road according to the demand-supply paradigm extended to intersections with a *firtst-in first-out* (FIFO) rule. It is worth noticing that this is not the only choice, and different proposals have been made, e.g., [73]. One of the consequences of choosing the FIFO approach, is that if, at some instant, one of the roads downstream some intersection has supply equal to zero, then the outflows of all roads upstream the same intersection will be zero as well. This means that this modeling choice cannot capture the fact that, if a road is completely full of vehicles, drivers can simply decide to change their path. However, it is known in the related literature that all approaches regarding this choice have some drawback.

The effect of traffic lights in an intersection is simply to allow the maximum flow to pass through or to block it. Therefore, the signal $u_i(t)$ multiplies at each time instant the value given by (3.7). Also, traffic lights located at the same intersection have to respect the constraints about the safe crossing: this means that at every time instant only one of them can have green light. These constraints will be hereafter called *collision avoidance constraints*, and they can be imposed in the following way,

$$\forall\, i \in \mathcal{R}\setminus\mathcal{R}^{\text{in}}, \forall\, t \in \mathbb{N}_+ \quad \sum_{j\in\mathcal{N}_i^-} u_j(t) \le 1. \tag{3.8}$$

Regarding the flow of vehicles entering a road from an intersection upstream, it is implicitely defined by the flows exiting the roads entering the same intersection, i.e.,

$$f_i^{\text{in}}(t) = \sum_{j\in\mathcal{N}_i^-} \beta_{ji} f_j^{\text{out}}(t) u_j(t). \tag{3.9}$$

The definitions given so far are valid for roads that have both the upstream and the downstream intersections within the network. In particular, definition (3.7) is valid for every road $i \in \mathcal{R}\setminus\mathcal{R}^{\text{out}}$, while definition (3.9) is valid for roads in $\mathcal{R}\setminus\mathcal{R}^{\text{in}}$. Let us therefore define the external supplies of the network, S_i^{out}, as the supply that is downstream road $i \in \mathcal{R}^{\text{out}}$, and the external demands of the network D_i^{out}, as the demand of vehicles that wants to enter every road $i \in \mathcal{R}^{\text{in}}$. Hence, for roads in $\mathcal{R}\setminus\mathcal{R}^{\text{out}}$ the outflow will be the minimum between the demand of the road and the external supply; similarly, for roads in $\mathcal{R}\setminus\mathcal{R}^{\text{in}}$, the inflow will be the minimum between the supply of the road and the external demand.

Putting all above developments together, the signalized CTM (S-CTM) assumes the following form,

$$\rho_i(t+T_s) = \rho_i(t) + \frac{T_s}{L_i}\Big(f_i^{\text{in}}(t) - u_i(t) f_i^{\text{out}}(t)\Big) \tag{3.10a}$$

$$f_i^{\text{out}}(t) = \begin{cases} \min\Big(D_i(t), S_i^{\text{out}}(t)\Big),\ i \in \mathcal{R}^{\text{out}} \\ \min\left(D_i(t), \left(\frac{S_j(t)}{\beta_{ij}}\right)_{j\in\mathcal{N}_i^+}\right),\ i \in \mathcal{R}\setminus\mathcal{R}^{\text{out}} \end{cases} \tag{3.10b}$$

$$f_i^{\text{in}}(t) = \begin{cases} \min\Big(D_i^{\text{in}}(t), S_i(t)\Big),\ i \in \mathcal{R}^{\text{in}} \\ \sum_{j\in\mathcal{N}_i^-} \beta_{ji} f_j^{\text{out}}(t) u_j(t),\ i \in \mathcal{R}\setminus\mathcal{R}^{\text{in}} \end{cases} \tag{3.10c}$$

$$D_i(t) = \min\Big(v_i\rho_i(t), f_i^{\max}\Big) \tag{3.10d}$$

$$S_i(t) = \min\Big(f_i^{\max}, w_i(\rho_i^{\max} - \rho_i(t))\Big). \tag{3.10e}$$

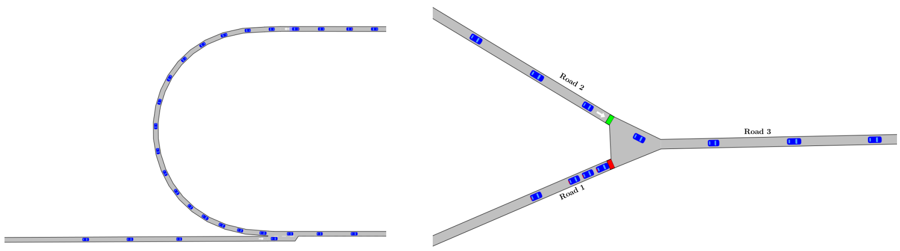
3.4 Calibration and validation via microscopic simulator

In this section we introduce some of the tests we performed with the software that has been used through the whole thesis for numerical experiments: Aimsun [108]. Aimsun is an integrated transport modelling software, developed and marketed by the Transport Simulation Systems (TSS). It is used by government agencies, municipalities, universities and consultants worldwide for traffic engineering, traffic simulation, transportation planning and emergency evacuation studies. TSS has shipped over 3500 licenses in 70 countries.

We have made extensive use of Aimsun both for modeling and for control. In this section we show some illustrative example regarding the calibration of the signalized CTM and its numerical validation against the microscopic model implemented into Aimsun, which is considered as ground truth.

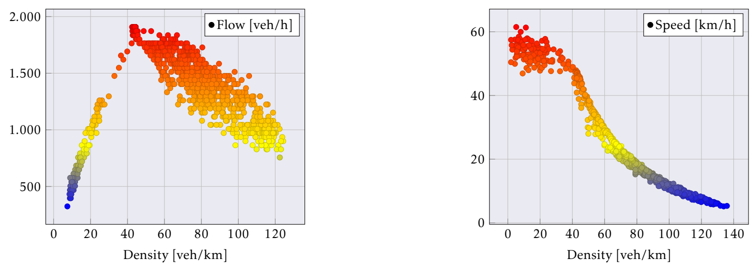
In figure 3.3a it is shown the Aimsun model we created to calibrate the signalized CTM. The roads receives variable external demands in such a way that free-flow and congested status alternate across time, and enough data for fitting the fundamental diagram is generated. The results of this calibration are shown in figure 3.4. In figure 3.4a the triangular shape of the fundamental diagram can clearly be distinguished. The same holds for the profile in figure 3.4b, to be compared with the theoretical plot in figure 3.1.

In figure 3.3b we depict the Aimsun model we created to compare the microscopic model against the macroscopic S-CTM (3.10), in the case of an intersection where merging flows are regulated by traffic lights. The S-CTM has been calibrated with the results of the previously mentioned test,



(a) (b)

Figure 3.3 – Aimsun models used for calibration and validation of the signalized CTM.



(a) (b)

Figure 3.4 – Results of the calibration of the signalized CTM.

and all numerical values are reported in table 3.1. Figure 3.5 reports the results about this additional test. Given the formal discrepancy between a microscopic model and the S-CTM, we believe that the resulting approximation is very good.

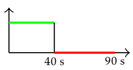
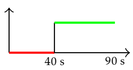
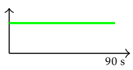
3.5 Model simplifications towards control synthesis

The S-CTM includes the traffic lights dynamics as binary signals that can switch from red to green and viceversa at every time instant. It has been remarked in several works, e.g. [84], that this approach leads to extremely complex algorithms when tackling the problem of synthesizing traffic signals schedule. This is why we extended our research looking for model simplifications that can ease the control problem while providing reasonable approximation of the signalized CTM.

We dealt with this from two diametrically opposed points of view; we analyze:

- An approach that simplifies the S-CTM the least, that is, a light approximation that can nevertheless bring benefits for the optimization;

Table 3.1 – Numerical set-up of the validation test for the S-CTM.

Parameter	Road 1	Road2	Road3
L [m]	250	250	250
D^{in} [veh/h]	600	800	-
S^{out} [veh/h]	-	-	$f_3^{\max}$
v [km/h]	50	50	50
w [km/h]	20	20	20
$f^{\max}$ [veh/h]	2000	2000	2000
u			

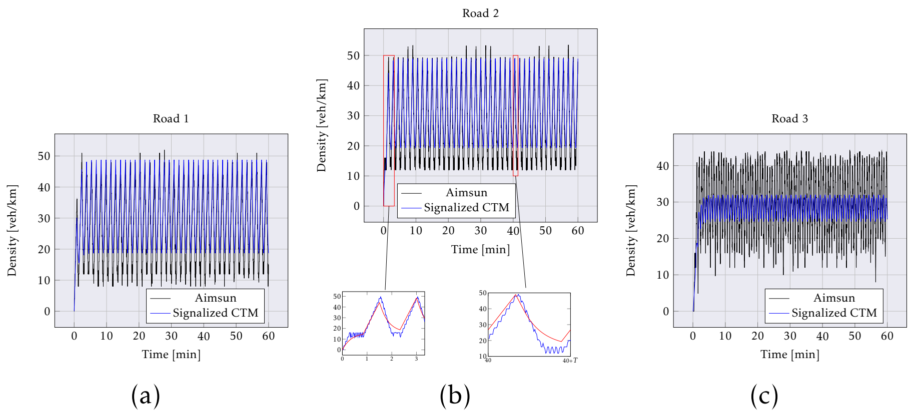


Figure 3.5 – Results of the validation of the signalized CTM against the Aimsun microscopic model.

- A radical approach that heavily simplifies the S-CTM, but still captures the main characteristic of the traffic flow dynamics and adds important properties that ease the control task.

3.5.1 A two degrees of freedom model

The first of our proposed model simplifications is based on the natural assumption, which indeed holds true in many practical cases, that a traffic light can have at most one rising edge (a switch from red to green), within the assigned cycle length T. Under this assumption is then natural to define two variables $\sigma^{(1)}, \sigma^{(2)}$, that are the two time instants capable of giving full description of the traffic light's trajectory during the cycle, as it follows,

$$u_i(t) = \begin{cases} 1 & \text{if } nT + \sigma_i^{(1)} \leq t \leq nT + \sigma_i^{(2)} \\ 0 & \text{otherwise,} \end{cases} \tag{3.11}$$

where $n = \lfloor t/T \rfloor$, see also figure 3.6. Hence, the two degrees of freedom signalized CTM (2DoF-CTM) is formally equivalent to (3.10) plus the constraints given by (3.11) for every traffic light.

Some comments are worth noticing about this model:

- If the assumption, mentioned earlier, about the maximum number of rising edges of the traffic lights holds true, then the 2DoF-CTM is equivalent to the S-CTM, in the sense that the former can describe all trajectories described by the latter;

- The 2DoF-CTM, even though is based on an approximation, is still capable of capturing all main features of the traffic lights, most notably the length of the green phase and the *offset*, that is the temporal shift of activation of the green light, with respect to the beginning of the cycle;

- The simplification introduced by the 2DoF-CTM reduces the model complexity by adding a structure to the signals' shape representing the traffic lights. We will exploit further this structure in chapter 6.

3.5.2 The average cell transmission model

We propose in this section a model simplification approach based on a change of variable. The main idea is inherited from the average theory [62] and it consists in defining a new dynamical system, which we call average CTM (Avg-CTM), whose mathematical description emulates the S-CTM regarding the demand-supply and replaces the binary traffic lights

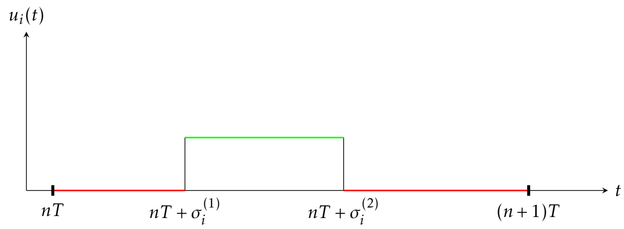


Figure 3.6 – The two degrees of freedom dynamics for traffic lights.

with the following signal,

$$\bar{u}_i(t) = \frac{1}{T/T_s} \sum_{k=1}^{T/T_s} u_i(t + kT_s). \tag{3.12}$$

The signal (3.12) is commonly said to be the *duty cycle* of the original signal $u(t)$, and for traffic lights it corresponds to the percentage of green time within a cycle. While u was a binary valued signal, $\bar{u}$ is a signal that takes a constant value, in the interval $[0,1]$, through an entire cycle. The collision avoidance constraints (3.8) become for the duty cycles:

$$\forall\, i \in \mathcal{R} \setminus \mathcal{R}^{\text{in}}, \forall\, t \in \mathbb{N}_+ \quad \sum_{j \in \mathcal{N}_i^-} \bar{u}_j(t) \leq 1. \tag{3.13}$$

The state variables of the Avg-CTM, which we will call average densities and will indicate with $\bar{\rho}$, evolve according to the dynamics described by the following set of equations, for every $i \in \mathcal{R}$,

$$\bar{\rho}_i(t+T_s) = \bar{\rho}_i(t) + \frac{T_s}{L_i}\left(f_i^{\text{in}}(t) - \bar{u}_i(t) f_i^{\text{out}}(t)\right) \tag{3.14a}$$

$$f_i^{\text{out}}(t) = \begin{cases} \min\left(D_i(t), S_i^{\text{out}}(t)\right), \; i \in \mathcal{R}^{\text{out}} \\ \min\left(D_i(t), \left(\frac{S_j(t)}{\beta_{ij}}\right)_{j \in \mathcal{N}_i^+}\right), \; i \in \mathcal{R} \setminus \mathcal{R}^{\text{out}} \end{cases} \tag{3.14b}$$

$$f_i^{\text{in}}(t) = \begin{cases} \min\left(D_i^{\text{in}}(t), S_i(t)\right), \; i \in \mathcal{R}^{\text{in}} \\ \sum_{j \in \mathcal{N}_i^-} \beta_{ji} f_j^{\text{out}}(t) \bar{u}_j(t), \; i \in \mathcal{R} \setminus \mathcal{R}^{\text{in}} \end{cases} \tag{3.14c}$$

$$D_i(t) = \min\left(v_i \bar{\rho}_i(t), f_i^{\max}\right) \tag{3.14d}$$

$$S_i(t) = \min\left(f_i^{\max}, w_i(\rho_i^{\max} - \bar{\rho}_i(t))\right). \tag{3.14e}$$

with $\bar{u}_i$ subject to (3.13) for every $i \in \mathcal{R}$ and to

$$\forall\, i \in \mathcal{R}, \forall\, t \in \mathbb{N}_+ \quad 0 \leq \bar{u}_i(t) \leq 1. \tag{3.15}$$

The proposed Avg-CTM, with respect to other models based on similar ideas, e.g., the store and forward [2], is more faithful to the original CTM, in the sense that it relies on the concept of demand and supply, and it is able to reproduce congestions spillbacks. In different words, the average densities are strongly bounded by the demand and the supply, and therefore stability property are implicit in the model definition (as they are with the 2DoF-CTM, too). With hindsight, this feature will be important when we will present optimal control strategy that are based on optimization and model–based predictions, rather than on feedback stabilization. The idea of the average description has also been used in [28], although not for traffic light's duty cycle control (like in our work).

3.5.3 Validation of the Avg-CTM against the S-CTM

The approach we propose with the definition of average CTM is to define a new system that, formally, is not linked to the S-CTM in any way. However, it turns out that the average densities are good approximations of the signalized trajectories when the two systems start from the same initial condition (and are subject to the same external conditions). We will show in this section the relevant results about numerical validation. The theoretical problem behind this empiric fact has received fewer attention to date (an exception is [52]) and is worth of future investigation.

Using the MatLab environment [57], we simulate a network with 40 roads connected by standard 4–ways intersections, which ideally represents a Manhattan-like grid. For ease of visualization we assume that all roads have the same parameters: $f_i^{\max} = f^{\max} = 2000$ veh/h, $\rho_i^{\max} = \rho^{\max} = 200$ veh/km, $v_i = v = 50$ km/h, $w_i = w = 12.5$km/h, $L_i = L = 0.5$ km, for every road i. Split ratios are chosen as follows. If a road i has two downstream neighbors, then the neighbor along the same (horizontal or vertical) orientation as i is assigned a split ratio of 0.6 plus a small random number, extracted uniformly in $[-0.05, 0.05]$; the split ratio for the second downstream road is then assigned so that the two sum to one. If there is only one downstream neighbor, then it gets split ratio 1 The network is simulated in two different cases, a first one where it evolves according to the S-CTM, and a second one where it does according to the Avg-CTM.

We are interested in:

- Evaluating how well the average system reproduces the distinction between free flow roads (i.e., with density lower than the critical density) and congested roads that is given by the signalized system;
- How close is the average system to the S-CTM, by means of comparison between the state's trajectories.

Regarding the second point, we compare the time trajectories of the two systems, and also of the formal integral average of the signalized CTM

Table 3.2 – Validation with different cycle lengths.

Error [veh/km]		$T = 45$ s	$T = 60$ s	$T = 90$ s	$T = 120$ s
(3.14) vs (3.16)	Mean	1.9	2.03	1.69	1.8
	Worst	12	12	10	11
(3.14) vs (3.10)	Mean	2.5	2.7	3	4.7
	Worst	13	15	14	22

trajectory, defined as [62]:

$$\hat{\rho}(t) = \frac{1}{T}\int_t^{t+T} \rho(\tau)\,d\tau. \tag{3.16}$$

We remark that, due to the nonlinearity in the expression of $\rho(t)$ in (3.10), the value of $\bar{\rho}(t)$ is not exactly the same as $\hat{\rho}(t)$, but it is a good approximation of the latter.

We run simulations where each traffic light (at each intersection) is a periodic given signal and the split ratios cause some asymmetry in the traffic evolution. Outside the network, demands (supplies) are generated for all the entering (exiting) roads. Representative examples of our results are reported in figure 3.7. Notice that:

- The averaged system succeeds in discriminating the status (free or congested) of a high percentage of roads. The mean error is around 10%, as it results from figure 3.7b;

- The precision of the averaged model results to be fully satisfying in approximating the density over all roads, as shown in figure 3.7c, where the only significant difference in densities evolution is that oscillations are switched off, as expected.

Results about quantitative comparisons are given in table 3.2. We replicate the simulation varying the length of the traffic lights' cycle and we measure the mean (over time and over all roads) and the worst error made by the average CTM with respect to signals (3.10) and (3.16). It is worth noticing that in all cases the mean error is very low and we believe that the worst cases are also fully acceptable.

In figure 3.8 we show one selected example, among all roads in the simulated network, of the earlier mentioned signals' time trajectories, visually explaining why the average CTM is considered to be a reasonable approximation of (3.10) and (3.16).

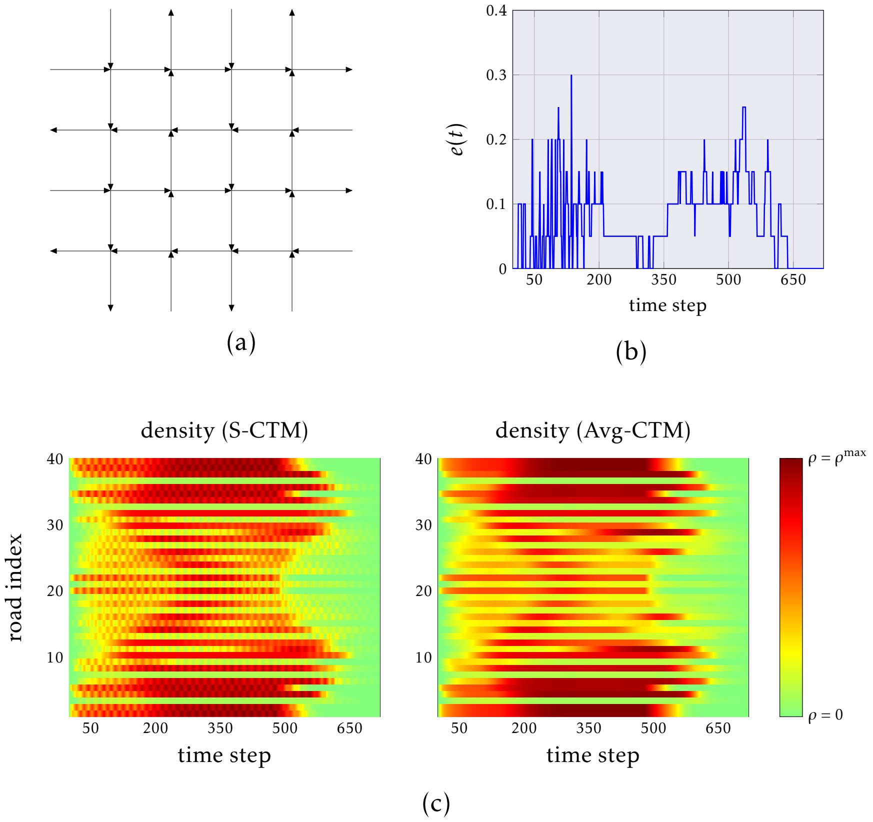


Figure 3.7 – Results of numerical validation regarding the accuracy of the Avg-CTM. In Figure 3.7a the idealized network of a Manhattan-like grid. We compare the evolution of the Avg-CTM and the S-CTM subject to the same external demands and supplies. Every road can have a status either free or congested: in Figure 3.7b is depicted the percentage of status that are not correct (in the sense that in the Avg-CTM they differ from the S-CTM), as $\frac{\#\text{wrong status}}{\#\text{roads}}$. In Figure 3.7c is depicted the time evolution of densities for all roads in the network; the figure shows that the average densities are very close to the signalized densities.

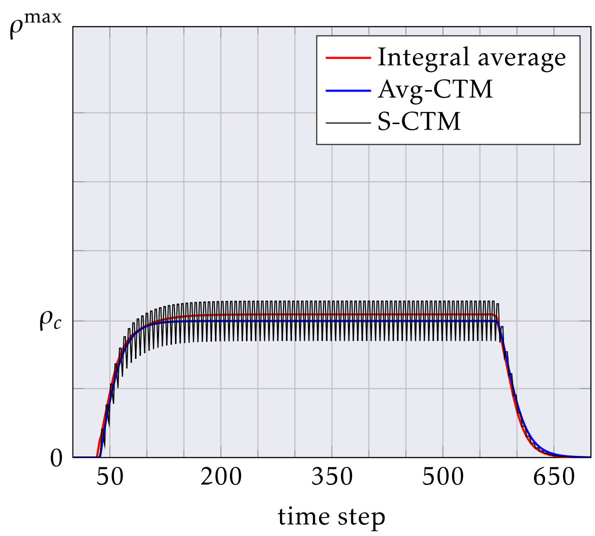


Figure 3.8 – Example of time evolution of different functions. For one selected road of the network, the figure shows the trajectories of the three signals used for comparison.

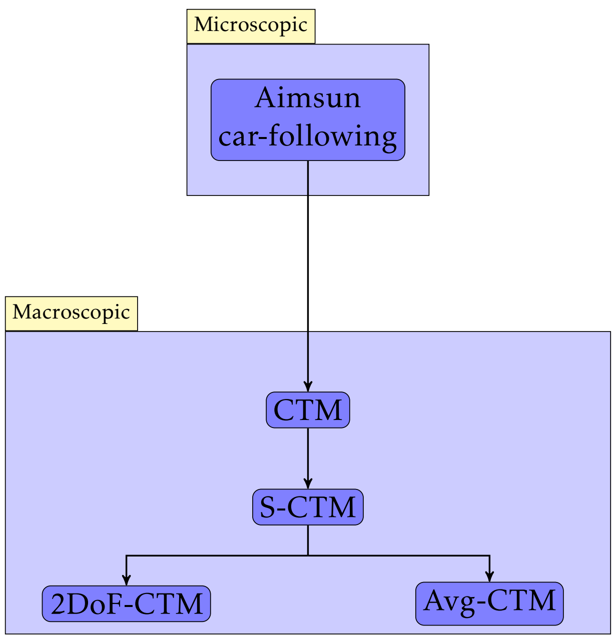


Figure 3.9 – The chain of model approximation presented in this chapter.

3.6 Final comments on the chapter

We have presented in this chapter the models that have been used throughout the thesis. The microscopic model implemented into Aimsun has been used for reproducing scenarios as close as possible to the reality, in order to test the control strategies that we will present later. The inherent complexity of such a model is well known and not practical for algorithms synthesis. Therefore, for the design of the control policies, we rely on macroscopic model. Starting from the classical CTM, we have extended it in order to include pieces of information that are crucial to networks, in particular the traffic lights. These give a macroscopic model with bynary variables, which we desire to simplify further to pursue computational efficiency. We propose two simplifications of the signalized CTM, one quite powerful, that describes all relevant traffic lights features, and another, which seeks to capture only the main characteristic. We will show in the following chapters how these two final models can be used to design optimization–based control schemes.

Chapter 4

Optimal control in urban signalized traffic

In this chapter we introduce the framework that characterizes the control strategies we will present in the later chapters. We will discuss the traffic indexes that are used as objectives to be improved and others that are used as evaluation metrics for the analysis. We will continue by giving a general formulation of the control problem. Such a formulation will then be detailed in chapter 5 and chapter 6 from two different points of view.

We recall some of the notation given in the previous chapters, in order to ease the reading:

ρ_i, ρ, ρ', denote the density of the i–th section, the vector of all roads density, and its transpose, respectively,

$\bar{\rho}$, indicates the density in the Avg-CTM system,

f_i, f_i^{in}, denote the flow inside and entering the i–th section, respectively,

T_s is the sampling time used for the system,

$\mathcal{N}_i^+$, $\mathcal{N}_i^-$, indicate the set of roads connected downstream and upstream, respectively, to road i,

$\mathcal{N}_i$, is defined as $\mathcal{N}_i^+ \cup \mathcal{N}_i^-$,

u_i is the traffic lights connected to road i, and $\bar{u}_i$ its duty cycle over the traffic lights' cycle T.

4.1 Traffic performance metrics

Traffic behavior has to be evaluated and rated with respect to traffic performance metrics properly defined. We give in this section a list of well-

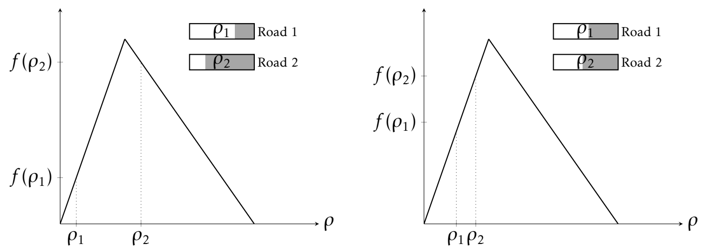


Figure 4.1 – Illustration of the density balancing idea. Considering the figure on the left, a value of TTD is obtained for $J = f(\rho_1) + f(\rho_2)$ which corresponds to an uneven distribution of vehicles in the two roads. Balancing the densities improves this by keeping a similar value of J but better distributing the total charge.

known traffic indexes that have been used in this work.

Total travel distance (TTD)

It is a measure of how efficiently the infrastructure is used in terms of occupancy and traveling velocity. It is desired to maximize this metric in order to have as many vehicles as possible traveling at the maximum allowed velocity. In other words, the purpose is to have a high utilization of the infrastructure at the same time avoiding congestions.

The TTD is defined and adapted to our framework as

$$\mathrm{TTD}(t) = \sum_{k=0}^{\lfloor t/T_s \rfloor} \sum_{i \in \mathcal{R}} f_i(kT_s). \tag{4.1}$$

where f_i is defined according to (3.3). From this definition, it is clear that increasing the TTD means to increase the flow traveling the section. From a different perspective, we already discussed that the flow is maximized when the system reaches the critical density ρ_c, therefore maximizing TTD can be seen as an attempt to bring the system towards such a value.

Density balancing (Bal)

The idea of balancing was introduced in traffic scenarios in [89]. A better balanced density avoids to have some overly congested roads and underutilized others. It can be regarded as an equal interdistance between vehicles, which makes travels smoother and safer and reduce emissions [88]. As objective function the balancing works nicely in relation to the TTD and the triangular fundamental diagram: in fact, the sum of flows can be maximized even if the vehicles diffusion is not well equilibrate. Balancing density helps in this respect (see figure 4.1 for an illustration).

A measure of densities balancing is given by

$$\mathrm{Bal}(t) = \sum_{k=0}^{\lfloor t/T_s \rfloor} \sum_{i \in \mathcal{R}} \sum_{j \in \mathcal{N}_i^+} (\rho_i(kT_s) - \rho_j(kT_s))^2. \tag{4.2}$$

Equation (4.2) can be expressed in a compact form as

$$\mathrm{Bal}(t) = \sum_{k=0}^{\lfloor t/T_s \rfloor} \rho'(kT_s)\mathcal{L}\rho(kT_s), \tag{4.3}$$

where $\mathcal{L}$ is a laplacian matrix of the network, defined as

$$\begin{aligned} \mathcal{L}_{ii} &= |\mathcal{N}_i|, \\ \mathcal{L}_{ij} &= \begin{cases} -1 & \text{if } j \in \mathcal{N}_i \\ 0 & \text{elsewhere.} \end{cases} \end{aligned} \tag{4.4}$$

Traffic lights regularization

The structure of the CTM with FIFO rule is such that numerical optimization can lead to abrupt changes in the control dynamics even for a short-term disturb on the system's input. Consider, for instance, a scenario where two roads entering the same intersection share the cycle length equally (e.g., half of the cycle is assigned to each of them). In such a case, if the flows entering the roads drop significantly (for example, to zero) even for a short amount of time (for example, one cycle), then the controller will try to reduce both duty cycle at the same time, in order to keep the TTD and the density balancing close to the optimal values. This will leave both traffic lights red for an unjustified amount of time.

This type of behaviors is commonly addressed in optimization by including a regularization term into the cost function. Regularizing means penalizing abrupt variations of the decision variables in order to obtain smoother results across time. Regarding the traffic lights' duty cycles this translates into adding the following term to the cost function,

$$\|\bar{u}(t) - \bar{u}(t-T)\|_2^2, \tag{4.5}$$

where $\bar{u}(t)$ is the duty cycle for the upcoming cycle (which can be a decision variable) and $\bar{u}(t-T)$ is the duty cycle applied in the last cycle.

Total travel time (TTT)

The total travel time (also the *total time spent* by the vehicles in the network) is one of the most used and informative traffic metrics to assess nature of traffic and vehicles behavior. The TTT is a global information and is only influenced by the number of vehicles and the density evolution inside the section across time. It is defined as

$$\mathrm{TTT}(t) = \sum_{k=0}^{\lfloor t/T_s \rfloor} \sum_{i \in \mathcal{R}} \rho_i(kT_s) \tag{4.6}$$

The minimization of total travel time in a traffic network is equivalent to the maximization of throughput. In other words, the earlier the vehicles are able to exit the network (by appropriate use of the available control measures) the smaller TTT will be [2].

Service of demand (SoD)

Urban traffic networks continuously receive demand from outside. This demand cannot be ignored just to favour the inner quality of the system, because the external request will end up growing with several undesired effects, due to the bigger and bigger queues arising outside.

For this reason the number of vehicles (users) served can be used as quality of the service [44]:

$$\mathrm{SoD}(t) = \sum_{k=0}^{t} \sum_{i \in \mathcal{R}^{\mathrm{in}}} f_i^{\mathrm{in}}(kT_s), \tag{4.7}$$

where f_r^{in} is the boundary flow as defined in (3.10c). In general, one would like to maximize the service of demand, since this will have the effect of decreasing the number of vehicles waiting to enter the network.

Queues length

The first instance of measure of urban traffic operation is the number of vehicles waiting for a red traffic light to become green. Such a measure is available if a microscopic model is used, as all vehicles can be tracked by these complex models.

Stop time

It is another microscopic traffic measure, that gives the amount of time every vehicle has been stopped (due to traffic red light, or due to congestions) in a network. Like for the queues length, it is possible to measure the stop time if a microscopic model is available.

A priori, optimal urban policies should bring benefits to all the mentioned metrics. In our set–up, we have chosen some among them to be part of the optimization procedure, in the sense that they are explicitly part of the numerical algorithms, and others to be post–simulation metrics. More precisely:

- The total travel distance and the density balancing will be considered the main objectives. In particular, choosing TTD at the place of the TTT suits well urban large-scale scenarios, where equal distribution of the vehicles is considered more important than the throughput; for this reason, TTD and Bal mix well together;

- Traffic lights regularization will be used to smooth out the trajectories imposed by the controller, if the controlled variables are the duty cycles;

- The other mentioned metrics will be used for a posteriori evaluation; some will be computed by the microscopic model implemented inside Aimsun.

4.2 Problem formulation

In this section we will state a general problem formulation, that includes both particular cases studied much more in detail in the next chapters.

Strategies candidate to address the traffic problem in urban scenarios need to be adaptive, i.e., capable to react smartly to changes in the local demand; they need to act in order to optimize traffic behavior, e.g., the fluidity of traffic within cities; they have to be numerically tractable, in the sense that a trade-off between the price of optimality and the real time feasibility is needed; they must be scalable, i.e., a distributed implementation of them has to be achievable, in order to avoid single point of failure and, as already stressed, to avoid that the computational load increases with the network's size; also, they have to be evaluated in scenarios close to reality.

Let J be a chosen traffic metric, or a combination of some among them, and let us make the following assumptions:

1. The state of the system is fully available, i.e., vehicles densities can be measured in every road.

2. Values of external demands D^{in} and supplies S^{out} are known, possibly as average values over short temporal horizon in the upcoming future.

We believe these assumptions are not severe limitations of our approach. Vehicles density can be measured by several technological means, for example, cameras installed at intersections, loop detectors installed at the beginning and at the end of the roads of interest, floating car data taken from various sources (e.g., [106, 54]). On the other hand, reasonable estimations of external demands and supplies are easily available, and in most cases already known to the field experts for a given scenario. This is due to the empirical fact that traffic behavior exhibits recurring patterns over time (e.g., similar patterns for similar days, a different one for holidays, etc., [19]).

A control policy intended to improve urban traffic behavior may be verbalized as it follows: *find values for traffic lights to be applied in the signalized intersections, such that they optimize the selected metric J over a chosen time horizon K in the future, given the current state of the network at time t.*

A first formalization of this approach will be

$$\begin{aligned} &\underset{u(kT_s)}{\text{optimize}} \sum_{k=1}^{K} J(t+kT_s) \\ &\text{subj. to signalized traffic dynamics over } K. \end{aligned} \tag{4.8}$$

Several comments about problem (4.8) are worth considering. It immediately catches the eye that the problem is one that requires great computational power: first of all, we have not given yet specific information about the cost function, that in general do not come with nice geometrical properties which could help the optimization task. Therefore, not all desired traffic metrics can be taken into account in J at the same time, if the efficiency of the solution method is a requirement. Secondly, the objective of predicting traffic behavior using the signalized dynamics brings further complexity into the problem: the decision variable become a sequence of K integer (binary) values, and the correlation between predicted densities is strongly nonlinear, due to (3.10b)–(3.10e). This is the reason why we introduced the 2DoF-CTM and the Avg-CTM in the previous chapter.

We will continue the exposition by studying the problem from two different perspectives:

- We will use the Avg-CTM in order to simplify drastically the prediction task. In particular, by setting $K = 1$ it is possible to design an optimization problem that takes care of TTD, Bal and traffic lights regularization within the cost function J and still it provably is a convex problem. This, in return, gives us the possibility to design a distributed version of the same control algorithm that is also optimal, i.e., it provably converges to the same solution of the initial problem.

- We will use the 2DoF-CTM in order to design a control scheme that makes predictions over multiple steps. In particular, by choosing $K = T/T_s$ it is possible to predict densities' values over an entire traffic light cycle using an optimization technique known as Big-M. This does not come for free, and the resulting problem will have some integer variables, therefore only the TTD will be used as cost function in this case, in order to synthesize a problem with linear cost, for which at least good numerical algorithms are available. On the other hand, for non-convex problem such as the mixed-integer problems (MIP) ones, no optimal distributed scheme is known to date, therefore we will design a custom distributed approach that is sub-optimal, in order to ease the computation burden and for which we are interested in studying the performances with respect to the optimal MIP.

4.3 Final comments on the chapter

In this brief chapter we outlined the general set-up for urban traffic optimization that we seek. We introduced several important traffic performance metrics, that we will use either as part of the optimization process or as evaluation indexes. We also gave a first formalization of the control problem we intend to solve, which is a very demanding one. As it is often the case, the problem one would like to solve proves to be prohibitive in practice. Therefore, we sketched out the reasoning we will follow while searching for feasible solutions to it, and that will be analyzed more in details in the chapters to follow.

Chapter 5

One step ahead traffic control via convex optimization

In this chapter we will present the first of our approaches to the traffic lights control problem. The main idea is to simplify the prediction task by using the Avg-CTM which, as shown in chapter 3, can give a good approximation of the signalized dynamics.

To start the presentation we recall the first formalization of the problem we are interested in solving, that was given in (4.8):

$$\begin{aligned} &\underset{u(kT_s)}{\text{optimize}} \sum_{k=1}^{K} J(t+kT_s) \\ &\text{subj. to signalized traffic dynamics over } K. \end{aligned} \tag{4.8 revisited}$$

5.1 Choice of cost function, constraints and decision variables

In the following section we will present a control algorithm that is based on optimization of the total travel distance (TTD) (4.1) and of density balancing (Bal) (4.2), and that uses as decision variables the duty cycles of the traffic lights. The optimization acts searching for duty cycles' values that optimize the traffic behavior in the near future, by means of the selected traffic metrics. Therefore, our scheme will resemble a model–predictive control, tailored for the traffic scenario. More in particular, in order to avoid abrupt changes of the duty cycles from one cycle to the next one, we will also use the traffic lights regularization term (4.5) in the cost function. We use positive constant values k_{bal} and k_{ttd} to weight TTD and Bal in such a multiobjective optimization.

The model-based predictions will be computed by using the Avg-CTM (3.14), and we generalize the problem by considering lower bounds l_i that are assigned for every traffic light's cycle according to roads and intersections features (size, number of lanes, capacity, etc.). If $l_i = 0$, then this will be equivalent to constraints (3.15). The decided duty cycles have also to respect the collision avoidance constraints (3.13). Hence, we formulate the problem as it follows,

$$\begin{aligned}\min_{\bar{u}} \quad & \sum_{k=1}^{K}\left(k_{\text{bal}}\bar{\rho}'(t+kT_s)\mathcal{L}\bar{\rho}(t+kT_s) - k_{\text{ttd}}\sum_{i\in\mathcal{R}} f_i(t+kT_s)\right) + \|\bar{u}-\bar{u}(t-T)\|_2^2 \\ & \text{subj. to: } (3.13), (3.15),\ l_i \leq \bar{u}_i \leq 1\ \forall i.\end{aligned} \tag{5.1}$$

The principle of functioning of our approach is the following: just before a traffic light's cycle begins novel values for the duty cycles are computed by solving (5.1). Indeed, the decision variable in (5.1) is a vector gathering all duty cycles. Such values are then applied into the network for the entire upcoming cycle; then, this process is repeated at the beginning of every cycle. This approach leads to other comments worth mentioning:

- The chosen cost function uses model–based predictions for the TTD and Bal, while the decision variables are regularized only with respect to the value they had in the last cycle; this is due to the "cycle-to-cycle" functioning scheme above mentioned;

- The portion of the objective cost related to TTD is multiplied by -1, since TTD is a metric that has to be maximized;

- The goal of optimizing a cost function given by the model–based predictions over some time horizon and the entire network is, however, still computationally heavy, due to the nonlinearities in the Avg-CTM given by the demand and the supply (3.14b)-(3.14e). Therefore, in order to achieve an opimization problem that is computationally tractable a trade–off is required. We will hence limit the predictions computed at time t to one step ahead only, choosing $K = 1$, i.e., to values depending on $\bar{\rho}(t+T_s)$.

Clearly, restricting the predictions to such a short horizon is a limitation of this approach. However, the intent of our work consists in developing a general framework that integrates concepts from modeling and optimization theory in order to synthesize an optimal urban traffic lights control that satisfies requirements about efficiency and scalability, so that it can have an impact in real scenarios. The limitation we introduce with the one step ahead control pursues these objectives: we will in fact show that a convex formulation can be obtained for this problem; this also gives the possibility to design a distributed version of the same algorithm that is provably equivalent, and that will be subject of a later chapter. An alternative approach for multi step ahead traffic control is instead discussed in the next chapter.

5.2 A convex formulation for the control problem

We will now illustrate the path to achieve a convex formulation for the one step ahead optimal traffic control. The main obstacle to overcome is given by the nonlinearities in the objective function.

The first point worth noticing is that, when using the measured values $\bar{\rho}(t)$, every term $f^{\text{out}}(t)$, at every instant t, is a real number that can be explicitly computed by using (3.14b).

In second place, and before proceeding to show the convex formulation of the problem, it is useful to explicit the correlation between the one step ahead density and the duty cycles: expanding equations from (3.14b) to (3.14e) into (3.14a) it yelds,

$$\bar{\rho}_i(t+T_s) = \bar{\rho}_i(t) + \frac{T_s}{L_i}\left(\sum_{j\in\mathcal{N}_i^-} \beta_{ji} f_j^{\text{out}}(t)\bar{u}_j - \bar{u}_i f_i^{\text{out}}(t)\right), \tag{5.2}$$

where every term $f_i^{\text{out}}(t)$ is indeed a real number given by (3.14b). Thus, it can be seen from (5.2) that the correlation between the one step ahead densities $\bar{\rho}(t+T_s)$ and the duty cycles $\bar{u}$ is affine. Let us therefore define the following notation: let

$$\bar{u}_{[i,v\in\mathcal{N}_i^-]}$$

be a vector containing duty cycles of road i and roads $v \in \mathcal{N}_i^-$ and, similarly, let $\bar{\rho}_{[i,j\in\mathcal{N}_i]}$ be a vector containing average densities of road i and roads $j \in \mathcal{N}_i = \mathcal{N}_i^+ \cup \mathcal{N}_i^-$, as defined earlier. Using such notations, (5.2) can be rewritten as

$$\bar{\rho}_i(t+T_s) = h_i\Big(\bar{\rho}_{[i,j\in\mathcal{N}_i]}(t)\Big)\bar{u}_{[i,v\in\mathcal{N}_i^-]} + \bar{\rho}_i(t) \tag{5.3}$$

where h_i is a row-vector of real numbers computed using the measured values of $\bar{\rho}_{[i,j\in\mathcal{N}_i]}(t)$, and consistent dimensions. Then, the link between the whole vector of one step ahead densities and the whole vector of duty cycles can be written as an affine dependency, i.e.,

$$\bar{\rho}(t+T_s) = H(t)\bar{u} + c(t),$$

where the matrix H gathers the vectors h_i that are built as indicated in (5.3).

At this point it should also be clear why only the one step ahead prediction has nice geometrical properties. If one wants to express $\bar{\rho}_i(t+2T_s)$ then, according to (5.3), this is given by

$$\bar{\rho}_i(t+2T_s) = h_i\Big(\bar{\rho}_{[i,j\in\mathcal{N}_i]}(t+T_s)\Big)\bar{u}_{[i,v\in\mathcal{N}_i^-]} + \bar{\rho}_i(t+T_s)$$

But the values of $\bar{\rho}_{[i,j\in\mathcal{N}_i]}(t+T_s)$ are affine functions of $\bar{u}$ (that, it is worth remembering, are the decision variables), and are given as argument to the function h_i, which, due to (3.14), is based on the nonlinear operator min. Therefore, not only $\bar{\rho}(t+2T_s)$ would not be affine with respect to $\bar{u}$, but it would not have any other regular geometrical property (such as convexity), and the results we are going to show would not hold either.

Let us resume the analysis of the objective function in (5.1). Its portion related to the density balancing and to variables regularization can be rewritten as

$$\begin{aligned} &k_{\text{bal}}\bar{\rho}'(t+T_s)\mathcal{L}\bar{\rho}(t+T_s)+\|\bar{u}-\bar{u}(t-T)\|_2^2 = \\ &\quad k_{\text{bal}}\Big(H\bar{u}+c\Big)'\mathcal{L}\Big(H\bar{u}+c\Big)+\bar{u}'\bar{u}-2\bar{u}'(t-T)\bar{u}+\bar{u}'(t-T)\bar{u}(t-T) = \\ &\qquad \bar{u}'Q\bar{u}+p'\bar{u}+c'\mathcal{L}c+\bar{u}'(t-T)\bar{u}(t-T). \end{aligned} \tag{5.4}$$

The last two terms in (5.4) do not depend on $\bar{u}$ and therefore do not count in the optimization process. Hence, the sum of density balancing and traffic lights smoothing can be written as the following quadratic form:

$$\bar{u}'Q\bar{u}+p'\bar{u},$$

where

$$Q = k_{\text{bal}}H'(t)\mathcal{L}H(t)+I, \tag{5.5}$$

$$p = \Big(2k_{\text{bal}}c'(t)\mathcal{L}H(t)-\bar{u}'(t-T)\Big)'. \tag{5.6}$$

Let us now looking at the TTD. By substituting (3.14b) in (3.14c) and then into (3.14a), every element of the cost function becomes a nonlinear function of the duty cycles, expressed as

$$\begin{aligned} f_i(\bar{u}) = \min\Bigg(& v_i\Bigg(\bar{\rho}_i(t)+\frac{T_s}{L_i}\Big(\sum_{j\in\mathcal{N}_i^-}\bar{u}_j\beta_{ji}f_j^{\text{out}}(t)-\bar{u}_if_i^{\text{out}}(t)\Big)\Bigg), \\ & w_i\Bigg(\rho_i^{\max}-\bar{\rho}_i(t)-\frac{T_s}{L_i}\Big(\sum_{j\in\mathcal{N}_i^-}\bar{u}_j\beta_{ji}f_j^{\text{out}}(t)+\bar{u}_if_i^{\text{out}}(t)\Big)\Bigg)\Bigg). \end{aligned} \tag{5.7}$$

Rewriting TTD $= \mathbb{1}'f(\bar{u})$, where f is the vector containing all f_i and $\mathbb{1}$ is the vector containing all ones, we arrive to the following optimization problem:

$$\begin{aligned} \min_{\bar{u}} \quad & \bar{u}'Q\bar{u}+p'\bar{u}-k_{\text{ttd}}\mathbb{1}'f(\bar{u}) \\ \text{subj. to, } \forall i: \; & l_i \le \bar{u}_i \le 1 \\ & \sum_{j\in\mathcal{N}_i^-}\bar{u}_j \le 1. \end{aligned} \tag{5.8}$$

The nonlinearity in the objective function of (5.8) can be moved onto the constraints by defining a set of auxiliary variables y, along with the constraints $y_i = f_i(\bar{u}), \forall\, i$, thus giving the following,

$$\begin{aligned} \min_{\bar{u}} \quad & \bar{u}'Q\bar{u} + p'\bar{u} - k_{\text{ttd}}\mathbb{1}'y \\ \text{subj. to, } \forall i: \; & l_i \leq \bar{u}_i \leq 1 \\ & \sum_{j \in \mathcal{N}_i^-} \bar{u}_j \leq 1 \\ & y_i = f_i(\bar{u}). \end{aligned} \tag{5.9}$$

Then, the new constraints, given their special structure due to the piecewise linearity of every f_i, can be relaxed, hence giving the following convex program.

$$\begin{aligned} \min_{\bar{u}, y \geq 0} \quad & \bar{u}'Q\bar{u} + p'\bar{u} - k_{\text{ttd}}\mathbb{1}'y \\ \text{subj. to, } \forall i: \; & l_i \leq \bar{u}_i \leq 1 \\ & \sum_{j \in \mathcal{N}_i^-} \bar{u}_j \leq 1 \\ & y_i \leq v_i\left(\bar{\rho}_i(t) + \frac{T_s}{L_i}\left(\sum_{j \in \mathcal{N}_i^-} \bar{u}_j f_j^{\text{out}}(t) - \bar{u}_i f_i^{\text{out}}(t)\right)\right) \\ & y_i \leq w_i\left(\rho_i^{\max} - \bar{\rho}_i(t) - \frac{T_s}{L_i}\left(\sum_{j \in \mathcal{N}_i^-} \bar{u}_j f_j^{\text{out}}(t) + \bar{u}_i f_i^{\text{out}}(t)\right)\right). \end{aligned} \tag{5.10}$$

It is worth noticing that (5.10) is an instance of quadratic program with linear constraints, which belongs to the class of tractable problems and can be efficiently solved with Newton–like (interior point) methods, see [15]. Moreover, and most importantly, (5.10) can be used to find an optimal solution of problem (5.8), because the two turn out to be equivalent, as the following proposition shows (see also figure 5.1 for a graphical illustration).

Proposition 1. *Let Q and $D \in \mathbb{R}^{n\times n}$, and $p, b, g \in \mathbb{R}^n$, with $b \geq 0$. Let $f : \mathbb{R}^n \to \mathbb{R}^n$ be a concave positive function defined as $f(u) = \min\left(H_1 u + c_1, H_2 u + c_2\right)$, for given $H_1, H_2 \in \mathbb{R}^{n\times n}$ and $c_1, c_2 \in \mathbb{R}^n$.*

Let (5.11) be the following optimization problem

$$\min_{u \geq 0} \quad u'Qu + p'u - b'f(u) \tag{5.11a}$$

$$Du \leq g, \tag{5.11b}$$

and (5.12) be the following one

$$\min_{u, y \geq 0} \quad u'Qu + p'u - b'y \tag{5.12a}$$

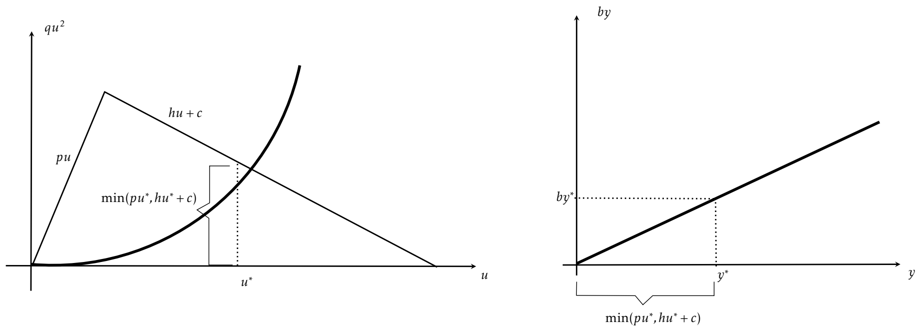


Figure 5.1 – Graphical illustration of Proposition 1, in the case of two scalar variables u and y for the minimization of $qu^2 - by$. In the figure on the left, the thick black line represents the function qu^2. Whatever is the optimal value for the first variable, say u^*, it will define a constraint for the second variable in the form of $y \leq \min(pu^*, hu^* + c)$. In the y–space (figure on the right) the purpose is to achieve large values for by (since the minimization is about $-by$). Therefore, y will always takes the value at the limit of such a constraint, because this gives the largest admissible value for y. Thus, the constraint will be satisfied by equality, that is the condition required to prove the statement.

$$Du \leq g \tag{5.12b}$$

$$y \leq H_1 u + c_1 \tag{5.12c}$$

$$y \leq H_1 u + c_2. \tag{5.12d}$$

Then (5.11) *is equivalent to* (5.12).

Proof. The statement holds true if one among (5.12c) and (5.12d) is satisfied by equality for every optimum of (5.12). By contradiction, suppose (u^*, y^*) is an optimal solution for (5.12) and there exist $\varepsilon > 0$ and a natural $i \leq n$ such that

$$y_i + \varepsilon = \min(h_{1i} u + c_{1i}, h_{2i} u + c_{2i}),$$

where $h_i i$ (h_{2i}) is the i–th row of H_1 ($H2$). Then, increasing the i–th component of y^* by ε one obtains a new admissible solution that provides a better value of the objective function than (u^*, y^*), which, therefore, could not be optimal. □

Proposition 1 is a well-known techinique to deal with some non convex constraint. In the context of traffic networks it has been employed for variable speed limit control and vehicle (split ratios) control, see [78, 22]. Here we have used it in relation with the triangular shape of the fundamental diagram for the traffic lights' duty cycle synthesis.

5.2.1 Remark on the cost function

In the previous section we have shown a multiobjective optimization mainly directed to two well known traffic measures, the total travelled distance and the density balancing, together with a regularization term. It is worth mentioning that if one decides to include in the objective the TTD only, then it is possible to use the same reasoning in order to prove that the optimization problem is a linear program (LP), which in average is computationally even easier to solve than a QP. This concept was exploited by us already in [44].

However, the several benefits of the selected cost function have already been stressed. TTD and density balancing suit well to urban scenarios, with the purpose to achieve equilibrated flows. Moreover, we have shown that the resulting problem is equivalent to a convex optimization; besides the efficiency, this property allows the design of iterative distributed schemes that are provably optimal, that is, the iterations converge to the same variables' value of (5.10). One of the approach we will present in chapter 7 is indeed suitable for the multiobjective optimization (5.10). Nevertheless, it is possible to obtain a distributed optimal algorithm for the LP that considers TTD only, by using different algorithms such as those proposed in [91, 16].

5.3 Computational details

In this section we intend to clarify the process of variables computation in the problem formulation, by exploiting the affine dependency between the one step ahead density and the duty cycles (as the latter are optmization variables). According to (5.2) $\bar{\rho}_i(t+T_s)$ depends linearly only on decision variables $\bar{u}_i$ and $\bar{u}_v$, for $v \in \mathcal{N}_i^-$. The coefficients of this linear combination, gathered in h_i, according again to (5.2) are computed with the knowledge of $\bar{\rho}_i(t), \bar{\rho}_v(t)$ and $\bar{\rho}_q(t)$, $q \in \mathcal{N}_v^+$.

Thus, it would look like road i needs pieces of information also from roads $q \in \mathcal{N}_v^+$. However, the computation of the terms f^{out} allows to avoid this further communication exchange. Indeed, the f^{out} are the only constant involved in the computation of the one step ahead prediction. More precisely, the prediction $\bar{\rho}_i(t+T_s)$ can be carried out locally from road i with the following steps (see also figure 5.2):

1. Compute f_i^{out}, by knowing $\rho_i(t)$ and $\rho_j(t)$ for $j \in \mathcal{N}_i^+$ only, from (3.14b);
2. Get f_v^{out} from roads $v \in \mathcal{N}_i^-$;
3. Compute the prediction by using (5.2).

This scheme shows why every road i need communications only in $\mathcal{N}_i^- \cup \mathcal{N}_i^+$ to compute $\bar{\rho}_i(t+T_s)$.

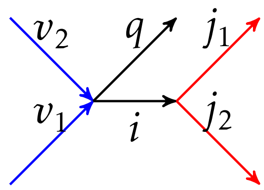


Figure 5.2 – Illustration of roads involved in the same intersection and how they are labelled. Here $\mathcal{N}_i^- = \{v_1, v_2\}$ and $\mathcal{N}_i^+ = \{j_1, j_2\}$. According to (5.2), the one step ahead density of road i is an affine function of duty cycles u_i and $u_{[v \in \mathcal{N}_i^-]}$. The real numbers that are the coefficients of such a relation, according to the FIFO-CTM, depend also on densities measured in road q (and in all roads in $\mathcal{N}_{v_1}^+$), but, as explained in section 5.3, they can be computed by road i with bits of information from roads in $\mathcal{N}_i^- \cup \mathcal{N}_i^+$ only.

5.4 Simulations and results

We implemented and tested under various scenarios the one step ahead traffic control (OSA-TC). In this section we will show our results obtained from numerical simulation of the S-CTM (simulated in MatLab) and of the microscopic model (simulated in Aimsun). We compare the performances of the control strategy against a best–practice policy for traffic lights, that is defined later.

For the implemented set–up, since the cost function of the one step ahead control includes terms with different magnitudes (i.e., flows in the TTD and densities in the balancing measure), we set the constant values k_{bal} and k_{ttd} such that both terms are normalized. In different words, every term contributing to the TTD, that is y_i, is normalized by the constant $f_i^{\max}$, and every term $(\bar{\rho}_i - \bar{\rho}_j)^2$ is normalized by $\rho_i^{\max}$.

Two points are worth recalling: first, in both cases the control signal is computed by using the Avg-CTM and then it is applied into a different system, the S-CTM or the Aimsun model. In other words, the Avg-CTM is used as a tool to ease the computational burden. At the beginning of every traffic light cycle, once the results of the optimization is available, it is applied into the signalized model (or the Aimsun model) which then evolves until the beginning of the next cycle, when new measurements are taken from it. Therefore, the performance of the algorithm is evaluated in the signalized system and in the Aimsun scenario. Secondly, the goal of the OSA-TC is to decide about the duty cycle (i.e., the green split) of every traffic lights in every intersection. The controller does not decide on the sequence of execution of such duty cycles. In other words, the order of the traffic lights within every cycle is given and the objective of the optimal control is to assign the length of every traffic light's green time.

5.4.1 Macroscopic simulations

Most cities apply a simple choice for traffic lights' green time, i.e., selecting fixed duration of them based on the known turning proportions and the statistical knowledge of external demands, as it is implemented in the earlier traffic lights control policies, e.g., [92]. We compare our feedback strategy with a fixed best pratice policy based on the same concepts: we collect data from the simulated network and we use it to compute the mean density experienced within every road during the simulation. Given this information, fixed duty cycles are assigned proportionally to the mean densities.

We used for these tests a network with same shape of the one used for the model validation (see figure 3.7). For the sake of results visualization we limited the network size to 40 roads. The simulation is run for 700 time steps, and till the 550-th step, an external demand requires to enter the network. By doing so, we test the algorithm both in the case where the network is filling up and where it is emptying. We used the numerical values already used for the numerical experiments described in section 3.5.3.

Representative results are reported in figure 5.3. We show, for every road and for every time step, the distance between the road density and ρ_c, which would be the best point in the state space. Also, we show graphical visualization of the two optimized indexes (TTD and density balancing), from which the improvement can be seen. Moreover, we plot the time evolution of another well known macroscopic traffic index, the service of demand (4.7). We remark that, even though this index was not included in the optimization process, an improvement is also achieved in its respect.

5.4.2 Microscopic experiments

We also tested the OSA-TC algorithm in the commercial traffic microsimulator Aimsun. In this way, the scenario the control is tested on is much more realistic than the S-CTM we implemented in the MatLab environment (see previous section). On the other hand, it worth recalling that the control strategy is based on the Avg-CTM, which is only an approximation of the complex model implemented in the Aimsun software (see the discussion about model validation in chapter 3). Hence, the idea behind our microscopic experiments is the following: we run simulations in which, while the software is acting, the one-step ahead optimal control is computed at the beginning of every traffic light's cycle and is then injected into the software itself, that will continue the simulation with the new values of green times. Other simulations are run, where traffic lights are instead controlled by the best practice earlier explained. Also, to create a scenario as realistic as possible, cycle lengths depend on the size of the intersection, and we have four different values of those (45, 60, 90, and 120 seconds).

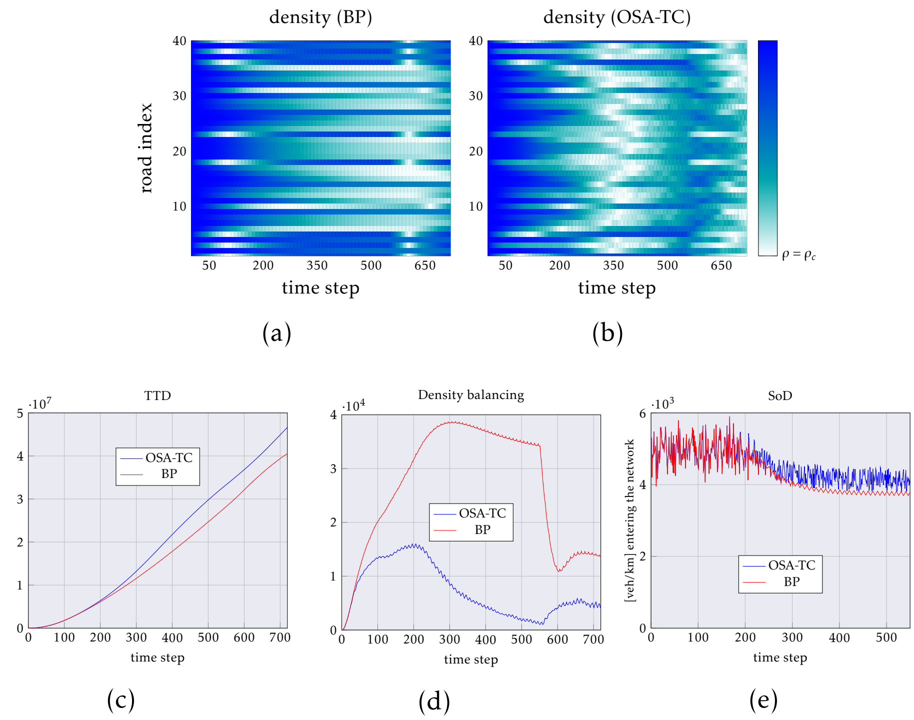


Figure 5.3 – Indexes comparison between the best practice control (BP) and the feedback strategy proposed in this paper (OSA-TC). In figures 5.3a and 5.3b we compare the system with best–practice policy and the system where the adaptive traffic control is our algorithm, by means of distance from the best working point, that is the critical density ρ_c. The improvement achieved in terms of TTD is shown in figure 5.3c, and the improvement in terms of balancing in figure 5.3d. In figure 5.3e we show the improvement over another traffic index, the service of demand, that the algorithm obtains even though such an index is not explicitly part of the cost function of the optimization problem.

The optimization problem is then solved for the entire network every time any cycle starts, but the resulting variables are applied only in the intersections where the cycle is actually starting.

To generate the scenarios, we used an Aimsun model that reflects the downtown portion of Grenoble, France, in every major aspect (see figure 5.4). The computation of the optimal control was done in the Python language, as Aimsun supports Python scripting through its API. The solver [6] has been used again, like in the MatLab simulations.

The calibration of the flows entering the Aimsun model is done using real flows measurement obtained from the loop detectors installed in the city. Some of these loop detectors are placed near the boundary of the town, and from them we measured the external input values, that are also injected into the Aimsun model (in terms of vehicles per hour). We have selected a typical day and set up the simulation as to represent the stream of vehicles corresponding to the morning peak-hours, from 7a.m. to 10a.m. Each simulation is then run for three (simulated) hours. In figure 5.5 we report some example of these data profiles.

The other detectors, placed more towards the center of the city, have been used to reproduce realistic values of split ratios into Aimsun: by measuring the flow in a road from a detector, and the input flows in the downstream neighbors of the same road, we could compute the split ratios for almost all pairs of roads. In the few cases where this was not possible (due to the absence of a detector) we assigned the splits proportionally to the size of the downstream roads (i.e., to the number of lanes): for example, in an intersection with three exiting roads with 1,2 and 3 lanes respectively, they would be assigned split ratios equal to 0.16, 0.34 and 0.5, respectively, from each of the roads entering the same intersection. Such a rule is meant to reproduce the fact that larger streets are more oftenly used by vehicles, which we know to be true thanks to our on-the-field experience, in Grenoble. Finally, we extracted the CTM parameters from the Aimsun UI, after we imported the (open street) map of the model.

We report about experiments conducted in two different scenarios. In both, the entering flows are the same, but in the second scenario we force the software to create incidents between vehicles, which in return gives increasing congestions in some areas. At the end of every simulation, Aimsun computes microscopic traffic measures that we use to evaluate the performance of the proposed strategy. Such results are shown in Table 5.1. We remark that every microscopic index is improved by using the presented control algorithm: travelled distance is increased, in average, of around 13%; travel time is reduced by more than 17% in the first scenario, and almost 20% in the second; the mean queue length (that the software computes as sum all over the network) is reduced by 10%; last, the stop time (computed as the time the vehicles were stopped, per kilometer) is reduced by 21% in the first scenario, and by 19% in the second.

Table 5.1 – Evaluation of the one-step ahead traffic control (OSA-TC) against the best practice policy (BP), via microscopic experiments and resulting traffic indexes computed by the microscopic simulator Aimsun.

Index	Scenario 1		Scenario 2	
	BP	OSA-TC	BP	OSA-TC
Travelled distance [km]	23396	26471	19772	17003
Travel time [h]	1775	1462	1955	1583
Mean queue [veh]	496	441	627	564
Stop time [sec/km]	123	97	172	139

A demo of these experiments is available online at the url [48].

5.5 Final comments on the chapter

In this chapter we presented the design of an efficient algorithm for traffic lights control. We used the Avg-CTM in order to compute short–term predictions of the state of the system (i.e., vehicles density) and select duty cycles values of the traffic lights that optimize a chosen metric.

The proposed approach has strengths and limitations: the algorithm is extremely efficient, since it can be solved as a convex program; also, it is scalable, in the sense that a distributed iterative version of the same algorithm can be synthesized, one that is provably convergent to the global optimal value. This will be the subject of a later chapter. Moreover, our tests in a scenario very close to reality, carried out in a microsimulator calibrated for the city of Grenoble, indicate that the algorithm can give benefits and have impact in the field: standard traffic indexes, both macroscopic and microscopic, have large improvement with respect to a common practitioners policy for traffic lights scheduling.

As for its limitations, the algorithm needs measurements of currents densities value as input to the optimization process. However, we believe that these can be easily obtained with the already existing technology. More important, we believe, is the shortness of the horizon prediction. It has been chosen in order to give the highest numerical efficiency to the optimization task. We have explored a different approach to this problem, that is the subject of the next chapter.

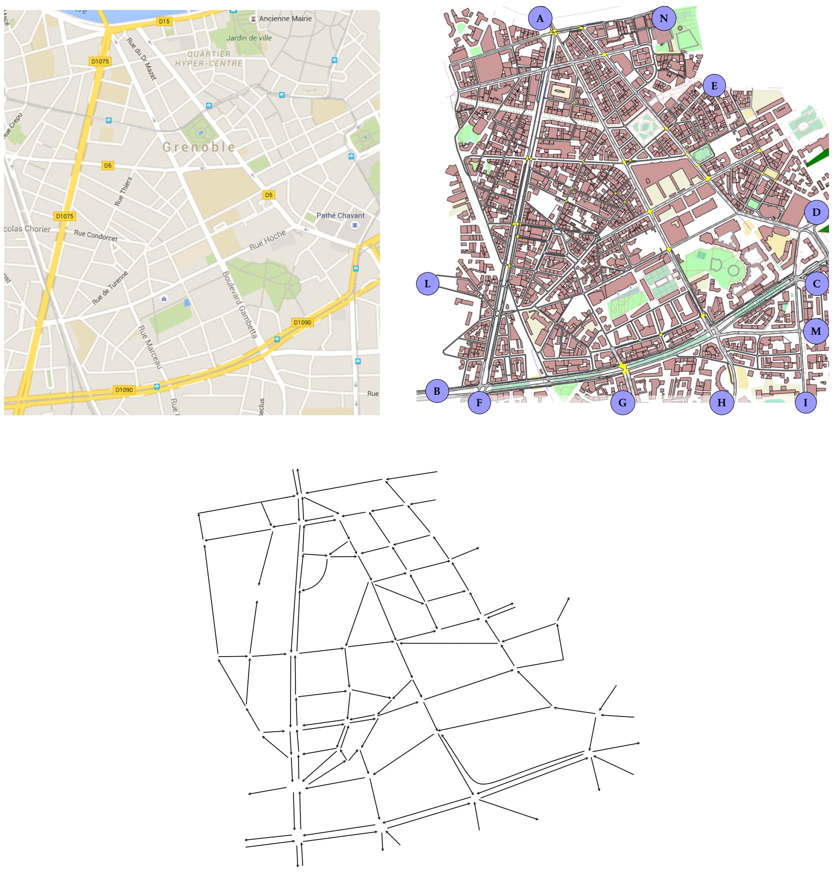


Figure 5.4 – From top left, clockwise: an image of Grenoble, from Google Maps; the Aimsun model used in the microscopic experiments, with markers for the detector positions, that correspond to the inflows of the model (see also Figure 5.5); the graph we superimposed to the network, in order to represent connectivity between roads. A demo of our experiments is available online at the url [48].

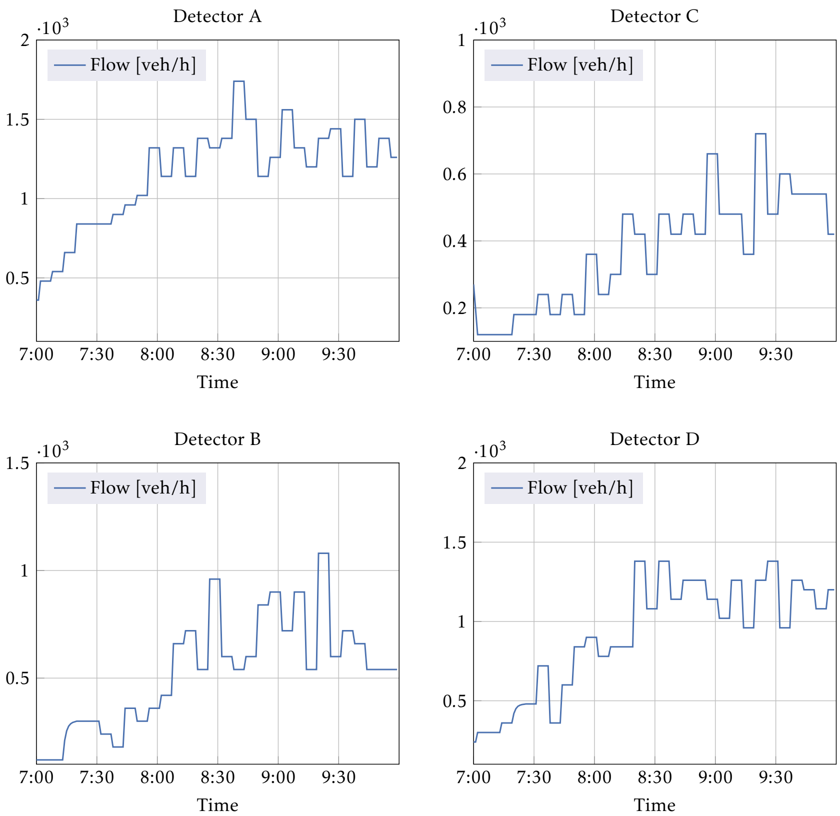


Figure 5.5 – Some of the real data profiles that have been measured by detectors installed in the city and that we used as input for the Aimsun model. The considered loop detectors A,B,C,D, from which the data shown is taken, are located as in figure 5.4.

Chapter 6

Multi step ahead traffic control via mixed integer programming

In this chapter we study the traffic lights control problem from a different perspective than chapter 5. There, the focus was on a model simplification that allows the design of a control policy low demanding. Here, we discuss the results of our research related to the design of a control algorithm that uses a more complex model, and therefore is computationally heavier.

To start the presentation we recall the first formalization of the problem we are interested in solving, that was given in (4.8):

$$\begin{aligned}&\underset{u(kT_s)}{\text{optimize}} \sum_{k=1}^{K} J(t+kT_s)\\&\text{subj. to signalized traffic dynamics over } K.\end{aligned} \tag{4.8 revisited}$$

6.1 Choice of cost function, constraints and decision variables

In the following section we will present a control algorithm that is based on the 2DoF-CTM described in detail in chapter 3. With respect to the algorithm presented in the previous chapter, that was based on a drastic simplification of the model (i.e., the Avg-CTM) leading to highly efficient numerical operations, here we construct an optimal control that still uses model–predictive ideas, but one that makes predictions based on richer dynamics, given by the 2DoF-CTM. It is worth recalling that this model introduces only a slight simplification into the S-CTM, by using the assumption that every traffic light has only one switch from red to green

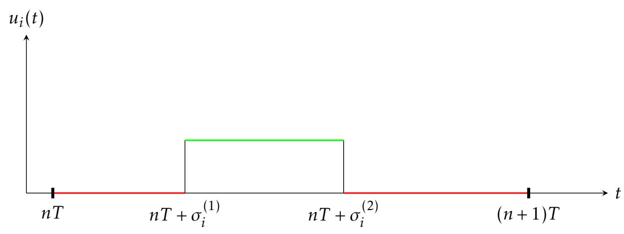


Figure 3.6 – The two degrees of freedom dynamics for traffic lights (repeated from page 38).

during the cycle T. By doing so, traffic lights' dynamics within the cycle can be fully described by two time instants, as we showed in figure 3.6, reported in this chapter in page 68 to ease the reading.

The time instants $\sigma^{(1)}$ and $\sigma^{(2)}$ decide the rising and falling edges of the traffic signals, and they are therefore appointed as decision variables. Phisical constraints on the minimum green time that a traffic light needs to have can be imposed by the inequality $\sigma_i^{(2)} - \sigma_i^{(1)} \geq \sigma_i^{\min}$, where $\sigma_i^{\min}$ is the minimum duration of green light for traffic light i. The optimization will take care of the model–based predictions that are, in this case, made by the 2DoF-CTM: thus, the main constraints in the optimization problem are given by the two degrees of freedom dynamics described in section 3.5.1 and entirely given as,

$$\rho_i(t+T_s) = \rho_i(t) + \frac{T_s}{L_i}\left(f_i^{\text{in}}(t) - u_i(t) f_i^{\text{out}}(t)\right) \tag{6.1a}$$

$$f_i^{\text{out}}(t) = \begin{cases} \min\left(D_i(t), S_i^{\text{out}}(t)\right), \ i \in \mathcal{R}^{\text{out}} \\ \min\left(D_i(t), \left(\frac{S_j(t)}{\beta_{ij}}\right)_{j\in\mathcal{N}_i^+}\right), \ i \in \mathcal{R}\setminus\mathcal{R}^{\text{out}} \end{cases} \tag{6.1b}$$

$$f_i^{\text{in}}(t) = \begin{cases} \min\left(D_i^{\text{in}}(t), S_i(t)\right), \ i \in \mathcal{R}^{\text{in}} \\ \sum_{j\in\mathcal{N}_i^-} \beta_{ji} f_j^{\text{out}}(t) u_j(t), \ i \in \mathcal{R}\setminus\mathcal{R}^{\text{in}} \end{cases} \tag{6.1c}$$

$$D_i(t) = \min\left(v_i \rho_i(t), f_i^{\max}\right) \tag{6.1d}$$

$$S_i(t) = \min\left(f_i^{\max}, w_i(\rho_i^{\max} - \rho_i(t))\right) \tag{6.1e}$$

$$u_i(t) = \begin{cases} 1 & \text{if } nT + \sigma_i^{(1)} \leq t \leq nT + \sigma_i^{(2)} \\ 0 & \text{otherwise,} \end{cases} \tag{6.1f}$$

where in (6.1f) $n = \lfloor t/T \rfloor$.

It is immediately evident that the dynamics expressed by (6.1) has a higher complexity than the Avg-CTM used for the synthesis of the one step ahead traffic control. However, in the remainder of this chapter, we will show that the nonlinear constraints given by (6.1) can be linearized at the price of adding some integer variable into the problem. Even though, from theoretical point of view, this does not reduce the complexity, mixed integer linear problems (MILP) are a deeply studied subject and therefore there exist good numerical solvers in the market. Such an increase in complexity is a first price to pay in order to extend the prediction horizon from one to multi steps ahead. In particular, our optimization will consider T/T_s steps of predictions, hence forecasting the traffic evolution over an entire traffic lights cycle.

The chosen approach, to bring the problem into the MILP framework, also introduces limitations in terms of the cost function that can be accepted by the optimizer: we will show that TTD, still the main metric in urban scenarios, can be reformulated in order to suit the MILP framework; we will also show that the SoD can be used, if desired; these two metrics will be weighted by positive constants k_{ttd} and k_{sod}. However, such a reformulation cannot be carried out for the density balancing index, since it intrinsically is a quadratic form. Using Bal would mean to accept another, further, increase in the computational load, as the problem would become a quadratic integer program.

According to the discussion above, the multi steps ahead traffic control is expressed as,

$$\begin{aligned} \max_{\sigma} \quad & \sum_{k=1}^{T/T_s} \left(k_{\text{ttd}} \sum_{i \in \mathcal{R} \setminus \mathcal{R}^{\text{in}}} f_i(t + kT_s) + k_{\text{sod}} \sum_{i \in \mathcal{R}^{\text{in}}} f_i^{\text{in}}(t + kT_s) \right) \\ \text{subj. to,} \; \forall\, i: \; & \text{dynamics (6.1)}, \\ & \text{collision avoidance (3.8)}, \\ & \sigma_i^{(1)} \geq T_s, \sigma_i^{(2)} \leq T/T_s \\ & \sigma_i^{(2)} - \sigma_i^{(1)} \geq \sigma_i^{\min} \end{aligned} \tag{6.2}$$

The mathematical tool we use to transform (6.2) into a MILP is known in literature as Big–M technique. We will therefore start by briefly explaining this approach.

6.2 The Big–M technique

Big–M is a modeling tool used in optimization theory. In the context of automatic control it has been formalized in [12] by proposing a complete framework for modeling and controlling systems described by interdependent physical laws, logic rules and operating constraints, that are known as mixed logical dynamical (MLD) systems. MLD systems include linear hybrid systems, finite state machines, constrained linear systems and nonlinear systems that can be approximated by piecewise linear functions.

The Big–M approach is formally defined using notation from the logical calculus [118], i.e., with predicates holding either a *true* value or a *false* one, and *connectives* such as

$\wedge$, the logical conjunction of predicates (*and*),

$\vee$, the logical disjunction of predicates (*or*),

$\sim$, the logical negation (*not*),

$\rightarrow$, the logical implication (*if... then...*),

$\leftrightarrow$. the logical *if and only if*.

It is well known that all connectives can be defined in terms of a minimal subset of them (for example, $(\vee, \sim)$ is a minimal subset of such); however, we will use a non-minimal set of those, for the sake of clarity.

The foundation of the Big–M is the following: let δ_1, δ_2 be two binary variables. Then, the following properties are true [118, p. 176]:

$$X_1 \vee X_2 \text{ is equivalent to } \delta_1 + \delta_2 \leq 1, \tag{6.3a}$$
$$X_1 \wedge X_2 \text{ is equivalent to } \delta_1 = 1, \delta_2 = 1, \tag{6.3b}$$
$$\sim X_1 \text{ is equivalent to } \delta_1 = 0, \tag{6.3c}$$
$$X_1 \rightarrow X_2 \text{ is equivalent to } \delta_1 - \delta_2 \leq 0, \tag{6.3d}$$
$$X_2 \leftrightarrow X_2 \text{ is equivalent to } \delta_1 - \delta_2 = 0. \tag{6.3e}$$

Properties (6.3) allow to establish a link between dynamical systems and logical constraints. Suppose, for instance, that the predicate X is defined as the property $[f(x) \leq 0]$, where $f : \mathcal{X} \rightarrow \mathbb{R}, \mathcal{X} \subset \mathbb{R}^n$, and define $\delta \in \{0, 1\}$ and

$$M = \max_{x \in \mathcal{X}} f(x), \quad m = \min_{x \in \mathcal{X}} f(x).$$

It is then possible to prove that the following statements are true,

$$[f(x) \leq 0] \wedge [\delta = 1] \text{ holds true iff } f(x) - \delta \leq -1 + m(1 - \delta), \tag{6.4a}$$
$$[f(x) \leq 0] \vee [\delta = 1] \text{ holds true iff } f(x) \leq M\delta, \tag{6.4b}$$
$$[f(x) \leq 0] \rightarrow [\delta = 1] \text{ holds true iff } f(x) \geq \varepsilon + (m - \varepsilon)\delta, \tag{6.4c}$$
$$[f(x) \leq 0] \leftrightarrow [\delta = 1] \text{ holds true iff } \begin{cases} f(x) \leq M(1 - \delta) \\ f(x) \geq \varepsilon + (m - \varepsilon)\delta, \end{cases} \tag{6.4d}$$

where ε is a small tolerance (typically the machine precision) beyond which the constraint is considered violated (see [12]).

Big–M builds a bridge between the worlds of logic and dynamics, and hence it is useful to model control problems with logic state/input constraints. It has already received attention in scientific works dealing with traffic systems, e.g., [33, 46, 121]. Many other, more complex, logic constraints can be expressed with this techinique, as it is shown in [12]. We will continue the exposition by tailoring this approach for the multi steps ahead traffic control problem (6.2); we will analyze every single term of the optimization and show how they can be transformed into mixed-integer linear constraints.

6.3 Traffic lights trajectory

We want to express the 2DoF trajectory via linear constraints. Let us introduce the binary variables $\delta_i^{(1)}(t)$, $\delta_i^{(2)}(t)$, then the constraint expressed by (6.1f), for a given time instant t, is equivalent to the following:

$$\left[\delta_i^{(1)}(t) = 1\right] \leftrightarrow \left[\sigma_i^{(1)} \leq t\right] \tag{6.5a}$$

$$\left[\delta_i^{(2)}(t) = 1\right] \leftrightarrow \left[t \leq \sigma_i^{(2)}\right] \tag{6.5b}$$

$$\left[u_i(t) = 1\right] \leftrightarrow \left[\delta_i^{(1)}(t) = 1 \wedge \delta_i^{(2)}(t) = 1\right]. \tag{6.5c}$$

Let $M^{(1)}$ and $m^{(1)}$ be upper and lower bounds such that $m^{(1)} < \sigma_i^{(1)} - t < M^{(1)}$, for every $\sigma_i^{(1)}$; then (6.5a) is equivalent to the following constraints:

$$\sigma_i^{(1)} - t \leq M^{(1)}(1 - \delta_i^{(1)}(t)) \tag{6.6a}$$

$$\sigma_i^{(1)} - t > m^{(1)}\delta_i^{(1)}(t). \tag{6.6b}$$

Similarly, let $M^{(2)}$ and $m^{(2)}$ be such that $m^{(2)} < \sigma_i^{(2)} - t < M^{(2)}$, for every $\sigma_i^{(2)}$; then (6.5b) is equivalent to the following constraints:

$$\sigma_i^{(2)} - t \leq M^{(2)}(1 - \delta_i^{(2)}(t)) \tag{6.7a}$$

$$\sigma_i^{(2)} - t > m^{(2)}\delta_i^{(2)}(t). \tag{6.7b}$$

Finally, the constraint (6.5c) is equivalent to the following:

$$u_i(t) - \delta_i^{(1)}(t) \leq 0 \tag{6.8a}$$

$$u_i(t) - \delta_i^{(2)}(t) \leq 0 \tag{6.8b}$$

$$\delta_i^{(1)}(t) + \delta_i^{(2)}(t) - u_i(t) \leq 1. \tag{6.8c}$$

6.4 State constraints

The dynamics defined in (6.1) is non linear due to the min operator and to the product between flows and traffic lights' values. Our aim is to show how such a dynamics may be reformulated with mixed integer linear constraints.

Let σ be the vector containing the variables σ_i for every road i, and let $\bar{f}_i^{\text{in}}$ and $\bar{f}_i^{\text{out}}$ be the following modified flows,

$$\bar{f}_i^{\text{in}}(t) = \begin{cases} \sum_{j \in \mathcal{N}_i^-} \beta_{ji} \bar{f}_j^{\text{out}}(t), & ß \in \mathcal{R} \setminus \mathcal{R}^{\text{in}} \\ f_i^{\text{in}}(t), & i \in \mathcal{R}^{\text{in}} \end{cases} \tag{6.9a}$$

$$\bar{f}_i^{\text{out}}(t) = \begin{cases} u_i(t) f_i^{\text{out}}(t), & i \in \mathcal{R}^{\text{in}} \setminus \mathcal{R}^{\text{out}} \\ f_i^{\text{out}}(t), & i \in \mathcal{R}^{\text{out}}. \end{cases} \tag{6.9b}$$

We now show a scheme to rewrite (6.9b), i.e., min of several functions multiplied by a binary variable. For every road $i \in \mathcal{R} \setminus \mathcal{R}^{\text{out}}$ let $d^{(\text{i})}(t) \in \{0,1\}^h$ be a vector of binary variables, where $h = 1 + |\mathcal{N}_i^+|$. Then the definition of outflow (6.1b) is equivalent to the following constraints:

$$\left[d_0^{(\text{i})}(t) = 1\right] \rightarrow \left[f_i^{\text{out}}(t) = D_i(t)\right] \tag{6.10a}$$

$$\left[d_q^{(\text{i})}(t) = 1\right] \rightarrow \left[f_i^{\text{out}}(t) = \frac{S_j(t)}{\beta_{ij}} \, \forall j \in \mathcal{N}_i^+\right] \tag{6.10b}$$

$$\left[d_0^{(\text{i})}(t) = 1\right] \rightarrow \left[D_i(t) \le \frac{S_j(t)}{\beta_{ij}} \, \forall j \in \mathcal{N}_i^+\right] \tag{6.10c}$$

$$\left[d_j^{(\text{i})}(t) = 1\right] \rightarrow \left[\frac{S_j(t)}{\beta_{ij}} \le \frac{S_v(t)}{\beta_{iv}} \, \forall j \ne v\right] \tag{6.10d}$$

$$\sum_{l=1}^{h} d_l^{(\text{i})}(t) = 1. \tag{6.10e}$$

Given the additional upper and lower bounds, $l_0 < f_i^{\text{out}}(t) - D_i(t) < L_0$, $l_j < f_i^{\text{out}}(t) - S_j(t)/\beta_{ij} < L_j$, $\psi_0 < D_i(t) < \Psi_0$, $\psi_j < S_j(t)/\beta_{ij} < \Psi_j$, logical constraints (6.10a)–(6.10d) are equivalent to the following linear ones:

$$f_i^{\text{out}}(t) - D_i(t) \ge l_0(1 - d_0^{(\text{i})}(t)) \tag{6.11a}$$

$$f_i^{\text{out}}(t) - \frac{S_j(t)}{\beta_{ij}} \ge l_j(1 - d_j^{(\text{i})}(t)) \tag{6.11b}$$

$$f_i^{\text{out}}(t) - D_i(t) \le L_0(1 - d_0^{(\text{i})}(t)) \tag{6.11c}$$

$$f_i^{\text{out}}(t) - \frac{S_j(t)}{\beta_{ij}} \le L_j(1 - d_j^{(\text{i})}(t)) \tag{6.11d}$$

$$D_i(t) \le \frac{S_j(t)}{\beta_{ij}} + (\Psi_0 - \psi_j)(1 - d_j^{(\mathrm{i})}(t)) \tag{6.11e}$$

$$\frac{S_j(t)}{\beta_{ij}} \le \frac{S_v(t)}{\beta_{iv}} + (\Psi_j - \psi_v)(1 - d_j^{(\mathrm{i})}(t)) \tag{6.11f}$$

$$\forall\, j \in \mathcal{N}_i^+, \forall\, v \neq j$$

$$d^{(\mathrm{i})}(t) \in \{0,1\}^h,$$

for every road i and $j \neq i$.

Finally, given $u_i(t) \in \{0,1\}$, setting

$$\bar{f}_i^{\mathrm{out}}(t) = u_i(t) f_i^{\mathrm{out}}(t) = \begin{cases} f_i^{\mathrm{out}}(t) & \text{if } u_i(t) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{6.12}$$

is equivalent to

$$g(1 - u_i(t)) + \bar{f}_i^{\mathrm{out}}(t) \le f_i^{\mathrm{out}}(t) \tag{6.13a}$$

$$-G(1 - u_i(t)) - \bar{f}_i^{\mathrm{out}}(t) \le -f_i^{\mathrm{out}}(t) \tag{6.13b}$$

$$-G u_i(t) + \bar{f}_i^{\mathrm{out}}(t) \le 0 \tag{6.13c}$$

$$g u_i(t) - \bar{f}_i^{\mathrm{out}}(t) \le 0, \tag{6.13d}$$

where $g < f_r^{\mathrm{out}}(t) < G$.

6.5 Objective functions

Both terms in the objective function of (6.2) are made of nonlinear functions based on the min operator. Thus, we use the same scheme presented in the previous section to move such nonlinearities into linear constraints, with addition of binary variables.

SoD

Using the scheme illustrated by (6.10)–(6.13), the expression (4.7) is equivalent to:

$$\pi_1(1 - b_1(t)) \le \mathrm{SoD}_i(t) - D_i^{\mathrm{in}}(t) \le \Pi_1(1 - b_1(t)) \tag{6.14a}$$

$$\pi_2(1 - b_2(t)) \le \mathrm{SoD}_i(t) - f_i^{\max} \le \Pi_2(1 - b_2(t)) \tag{6.14b}$$

$$\pi_3(1 - b_3(t)) \le \mathrm{SoD}_i(t) - w_i(\rho_i^{\max} - \rho_i(t)) \tag{6.14c}$$

$$\mathrm{SoD}_i(t) - w_i(\rho_i^{\max} - \rho_i(t)) \le \Pi_3(1 - b_3(t)) \tag{6.14d}$$

$$D_i^{\mathrm{in}}(t) \le f_i^{\max} + (P_1 - p_2)(1 - b_1(t)) \tag{6.14e}$$

$$D_i^{\mathrm{in}}(t) \le w_i(\rho_i^{\max} - \rho_i(t)) + (P_1 - p_3)(1 - b_1(t)) \tag{6.14f}$$

$$f_r^{\max} \le D_i^{\mathrm{in}}(t) + (P_2 - p_1)(1 - b_2(t)) \tag{6.14g}$$

$$f_i^{\max} \le w_i(\rho_i^{\max} - \rho_i(t)) + (P_2 - p_3)(1 - b_2(t)) \tag{6.14h}$$
$$w_i(\rho_i^{\max} - \rho_i(t)) \le D_i^{\text{in}}(t) + (P_3 - p_1)(1 - b_3(t)) \tag{6.14i}$$
$$w_i(\rho_i^{\max} - \rho_i(t)) \le f_i^{\max} + (P_3 - p_2)(1 - b_3(t)) \tag{6.14j}$$
$$b(t) \in \{0,1\}^3,$$

where Π, P (π, p) are upper (lower) bounds consistently chosen.

TTD

Similarly, expression (4.1) is equivalent to the following:

$$\gamma_1(1 - c_1(t)) \le \mathrm{TTD}_i(t) - v_i\rho_r(t) \tag{6.15a}$$
$$\mathrm{TTD}_i(t) - v_i\rho_r(t) \le \Gamma_1(1 - c_1(t)) \tag{6.15b}$$
$$\gamma_2(1 - c_2(t)) \le \mathrm{TTD}_i(t) - w_i(\rho_r^{\max} - \rho_i(t)) \tag{6.15c}$$
$$\mathrm{TTD}_i(t) - w_i(\rho_i^{\max} - \rho_i(t)) \le \Gamma_2(1 - c_2(t)) \tag{6.15d}$$
$$v_i\rho_i(t) \le w_i(\rho_i^{\max} - \rho_i(t)) + (\Theta_1 - \theta_2)(1 - c_1(t)) \tag{6.15e}$$
$$w_i(\rho_i^{\max} - \rho_i(t)) \le v_i\rho_i(t) + (\Theta_2 - \theta_1)(1 - c_2(t)) \tag{6.15f}$$
$$c(t) \in \{0,1\}^2,$$

where Γ, Θ (γ, θ) are upper (lower) bounds consistently chosen.

6.6 Comments on the final problem formulation

In the previous sections we have shown the path to reformulate problem (6.2) into the MILP framework, by using the Big–M technique. The original problem includes constraints about the 2DoF-CTM dynamics, that are nonlinear. However, the piecewise linearity of those allows us to transform the same constraints into linear ones, by adding some integer variables to the problem. Thus, the mixed–integer linear formulation is given as,

$$\begin{aligned}
\max_{u,\delta,d,b,c} \quad & \sum_{k=1}^{T/T_s}\left(k_{\text{ttd}} \sum_{i \in \mathcal{R}\setminus\mathcal{R}^{\text{in}}} f_i(t + kT_s) + k_{\text{sod}} \sum_{i \in \mathcal{R}^{\text{in}}} f_i^{\text{in}}(t + kT_s)\right) \\
\text{subj. to } \forall\, i: \quad & \text{Big–M constraints } (6.6), (6.7), (6.8), \\
& \qquad (6.11), (6.13), (6.14), (6.15) \\
& \text{collision avoidance } (3.8), \\
& \sigma_i^{(1)} \ge T_s, \sigma_i^{(2)} \le T/T_s \\
& \sigma_i^{(2)} - \sigma_i^{(1)} \ge \sigma_i^{\min}
\end{aligned} \tag{6.16}$$

Some comments are worth remarking about this formulation:

- Problem (6.16) is *equivalent* to problem (6.2): the proposed approach does not introduce any relaxation into the formulation, therefore the result of (6.16) is expected to give better performances than the OSA-TC presented in the previous chapter. This is due to two main factors: first, the model–based predictions in (6.16) are computed using the 2DoF-CTM, which is in general more accurate than the Avg-CTM used in the one step ahead control. In second place, (6.16) carries out these predictions across multiple steps ahead in time, precisely over an entire traffic light cycle made of T/T_s steps;

- On the other hand, (6.16) is expected to be largely less efficient than the OSA-TC, because of the complexity introduced by the integer variables;

- For the proposed multi step ahead traffic control (MSA-TC), the finer the sampling time T_s, the finer decisions can be made about the optimization variables. However, as we already discussed in chapter 3, the modeling constraints inherited from the CTM do not give too much freedom in the choice of the sampling time. For instance, in the set–up we tested the OSA-TC (see previous chapter) the sampling time was 15 seconds, which gives 7 choices available for the σs over a traffic lights cycle of 90 seconds (0,15,30,. . . ,90). This reduces the improvement that the MSA-TC can bring with respect to the OSA-TC, and makes an argument in favor of the latter;

- The working principle of the MSA-TC is unchanged with respect to the OSA-TC: at the beginning of every traffic light cycle, the optimization problem (6.16) is solved and the activation instants are obtained as result. They are applied into the system for the upcoming cycle, and the procedure is repeated every new cycle.

6.7 Simulation and comparisons

The proposed approach has been tested via numerical simulations in MatLab environment, using software [49, 72] to solve the mixed integer linear programs. The simulated scenario is the following: all roads in the network have same physical properties (v,w,$f^{\max}$,$\rho^{\max}$) and each simulation is run for a virtual time of 45 minutes, when traffic lights' cycle is 90 seconds and sampling time is 15 seconds; as explained in the previous sections, the optimization problem is solved at the beginning of each cycle. The constant $\sigma^{\min}$ is set equal to 1 sample steps, so the minimum green slot is 15 seconds. Outside the network, time-varying demands and supplies are randomly uniformly generated in the interval $[0.5, 1]f^{\max}$ for the entire simulated time; by doing so, the network state changes during the simulation (from overall free to overall congested, and viceversa), and the controller is thus tested in different circumstances.

The same scenario has been run a second time, with the OSA-TC as controller, and we used the results from the two different simulations in order to compare various macroscopic indexes as well as the computation time of the algorithms. We would like to stress once more that the MSA-TC we propose is able to solve a very general problem, despite the computational inefficiency. The proposed traffic lights representation, along with the numerical optimization, guarantees optimal behavior by means of the chosen objective index, as it may cleverly choose phase shifts between lights (usually called *offset*) as well as green time for each of them. Therefore, we believe the results obtained applying this techinique should be considered as benchmark (upper bounds) for evaluation, from traffic and computational performance point of view. We generate the afore mentioned scenario for two sample networks, shown in figure 6.2.

Representative results of the simulations are reported in table 6.1, that lead to further comments:

- The MSA-TC obtains better performances by means of all considered macroscopic indexes, as expected. We compare the TTD, which is part of the optimization in both cases, as well as Bal and SoD, that are cost functions only in the OSA-TC and only in the MSA-TC, respectively. The improvement is up to 15% for the TTD and 8-10% for the other two metrics;

- While the OSA-TC is extremely efficient from computational point of view, the MSA-TC appears to be heavy even for the two small scenarios proposed. As earlier mentioned, this is simply due to the fact that the MSA-TC tries to solve a very complex problem, without passing through any model simplification. As such, we believe its usability in real scenarios is very limited. We explored, in this regards, the possibility to design a heuristic distributed approach for the MSA-TC, that reduces consistently the numerical burden, but it is suboptimal. It will be the subject of chapter 7.

6.8 Final comments on the chapter

In this chapter we presented a control algorithm (MSA-TC) for traffic lights scheduling that is based on the 2DoF-CTM and uses a modeling technique called Big-M in order to transform the nonlinearities of the model as integer linear constraints.

We analyzed the proposed approach, also with respect to the control strategy proposed in the previous chapter (OSA-TC), from two different

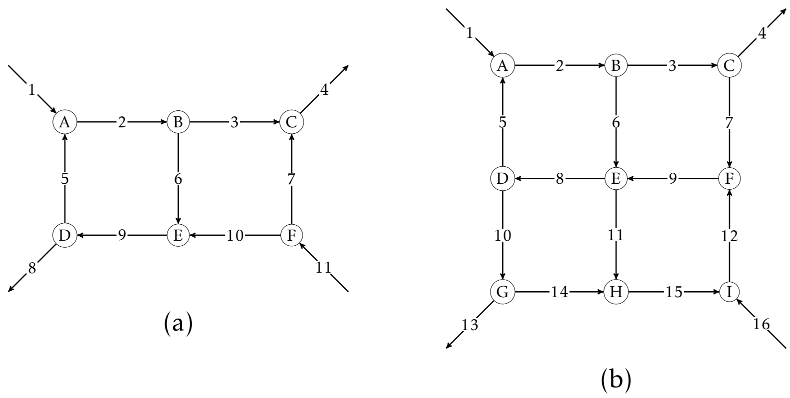


Figure 6.2 – Networks used for simulations and comparison between between the one step ahead traffic control and the multi step ahead traffic control.

Table 6.1 – Comparison between the one step ahead traffic control (OSA-TC) and the multi step ahead traffic control (MSA-TC), in the scenarios proposed in Figure 6.2. The table shows normalized values of macroscopic traffic indexes (total travel distance, density balancing and service of demand), as well as measured computation time, averaged over all run optimizations, for the two control algorithms.

	Network 6.2a		Network 6.2b	
	OSA-TC	MSA-TC	OSA-TC	MSA-TC
TTD	88%	100%	85%	100%
Bal	90%	100%	90%	100%
SoD	93%	100%	92%	100%
cpu time (avg)	0.2 sec.	3 min	0.25 sec	6.4 min

points of view: as for traffic performances, the MSA-TC gives good improvements over the OSA-TC; as for the computational efficiency, MSA-TC results very heavy in constrast to the highly efficient OSA-TC, questioning the practical usability of the former.

Chapter 7

Distributed approaches

In the previous chapters we have stressed in multiple occasions what an important feature scalability is. It is generally regarded as the capability of a system, network, or process to handle a growing amount of work, or its potential to be enlarged in order to accommodate that growth. Thus, in the context of urban traffic networks, which can have very large dimensions, and even more for traffic lights control algorithms, which need to be reactive and efficient enough to be deployed into the field and have an impact, scalability becomes crucial.

The way we addressed the scalability problem is to search for distributed algorithms. These are algorithms in which the computation task is indeed distributed among several entities in order to obtain different perks: unlike a centralized computation, where a failure in the single processing unit means a failure of the entire system, a distributed one is more robust because even if one of the involved entities fails the task can be recovered; moreover, distributed computing naturally tends towards parallel deployment, typically executed concurrently with separate parts of the algorithm being run simultaneously on independent processors, and having limited information about what the other parts of the algorithm are doing. On one hand, this approach clearly improves the computational performance of large scale systems; on the other, one of the major challenges in developing and implementing distributed algorithms is to successfully coordinate the behavior of the independent parts in order to obtain a correct solution to the problem.

In this chapter we will present our results about two distributed implementations:

- A distributed optimal algorithm based on the OSA-TC, that was discussed in chapter 5;
- A distributed suboptimal algorithm based on the MSA-TC, that was illustrated in chapter 6.

Regarding the first point, we will design an algorithm that is the equivalent distributed version of the convex program (5.10). Its convexity property allows us to use results from distributed optimization theory to construct an iterative distributed procedure that is provably optimal, i.e., its solution is the same of the centralized algorithm (5.10). Proven the optimality of such a scheme, we will analyze numerically its convergence speed in terms of number of iterations.

As for the second item, we are interested in developing a distributed version of the control problem (6.16). In this case, such a problem does not have geometrical properties that allow to use known results from optimization theory (as we discussed in the previous chapter, it is a MILP); therefore we designed an *ad hoc* distributed algorithm that, however, is suboptimal. Thus, we will analyze numerically how far is the distributed solution from the global optimum given by (6.16).

7.1 Distributed optimal one step ahead traffic control

The algorithm we designed is based on decomposition of the original centralized optimization in subproblems of smaller dimension. In order to reconstruct the global optimal solution, the subproblems follow an iterative convergent scheme which includes some exchange of information. These exchanges take place according to a scheme, established a-priori, that is called communication graph. The purpose in doing so is that subproblems will have a fixed size and therefore scalability can be obtained if they can be solved in parallel.

We will first introduce the communication graph. Then, we will show how the centralized problem can be reformulated to allow a distributed algorithm to exist and work properly. Finally, we will illustrate such an algorithm and comments about its features such as initialization and scalability.

7.1.1 Communication graph

Our intent is to construct one subproblem for every road; in this way, each road is in charge to select a value for its own traffic light. To do so, however, it has to consider values of some other optimization variable by evaluating pieces of information that come through the communication graph; in other words, the communication scheme states, for every road, from which other roads it can obtain information about the optimization variables.

We define the set $\mathcal{S}_i$ of roads that are allowed to share information with road i as

$$\mathcal{S}_i = \mathcal{N}_i^- \cup \mathcal{N}_i^+ \cup \mathcal{I}_i,$$

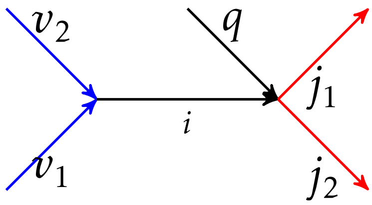


Figure 7.1 – Illustration of the required sharing of information for the variables $\bar{u}$. The subproblem created for road i is defined in a small neighborhood that involves upstream roads in $\mathcal{N}_i^- = \{v_1, v_2\}$, downstream roads in $\mathcal{N}_i^+ = \{j_1, j_2\}$, and roads in $\mathcal{I}_i = \{i, q\}$. In order to solve the problem distributedly, variables need to be shared within this neighborhood.

where

$$\mathcal{I}_i = \{q : \mathcal{N}_q^+ \equiv \mathcal{N}_i^+\}.$$

It is worth noticing that the set $\mathcal{S}_i$ speficies connections between roads that are geographically close to each other, and therefore has some advantage also in view of a practical implementation, see figure 7.1 for a graphical illustration.

We also remark that the choice of $\mathcal{S}_i$ is naturally given by the shape of the cost function we have chosen: to express the one step ahead density of road i we need the variable $\bar{u}_i$ and variables $\bar{u}_v$ for every road v upstream i. Then, to express the density balancing with downstream roads j, we need variables $\bar{u}_j$, and also variables $\bar{u}_q$, for every q upstream j (the latter are needed to express the one step ahead density of roads j).

Finally, notice that $i \in \mathcal{S}_i$ and if $p \in \mathcal{S}_i$ then $i \in \mathcal{S}_p$. Hence we can define the communication graph as the undirected graph $\mathcal{G}$ whose vertices are roads and in which an edge $\{i, j\}$ exists whenever $j \in \mathcal{S}_i$. The natural assumption that the directed road network is strongly connected implies that $\mathcal{G}$ is connected.

7.1.2 Problem formulation in the distributed set–up

A common set–up in distributed optimization ([20]) deals with problems with a separable objective function and local constraints. Regarding the optimization presented in chapter 5, we will show that we can rewrite the (centralized) problem (5.10) as

$$\begin{aligned} \min_{\bar{u}} \quad & \sum_{i \in \mathcal{R}} g_i\left(\bar{u}_{[p \in \mathcal{S}_i]}\right) \\ \text{s.t.} \quad & \bar{u}_{[p \in \mathcal{S}_i]} \in \mathcal{X}_i, \forall\, i \in \mathcal{R}, \end{aligned} \tag{7.1}$$

where each set $\mathcal{X}_i$ is compact and convex, and each function g_i is strictly convex. We used again the notation $\bar{u}_{[p \in \mathcal{S}_i]}$ to indicate a vector containing duty cycles of roads $p \in \mathcal{S}_i$.

The reason why this is convenient lies in the existence of efficient algorithms to solve (7.1) in a distributed manner. More specifically, the separation into subproblems is naturally suggested by the separability property of cost function and constraints. However, one more step is needed to formulate subproblems that do not have any variable in common. To accomplish this requirement, local copies of the decision variables are defined in every subproblem: therefore, we define the variable $\bar{u}_p^{(i)}$ as the copy of the global variable $\bar{u}_p$ kept in memory locally by subproblem i. The fully decentralized optimization is then based on new constraints that guarantee the consistency between all local variables, as it follows:

$$\min_{\bar{u}} \quad \sum_{i\in\mathcal{R}} g_i\Big(\bar{u}^{(i)}_{[p\in\mathcal{S}_i]}\Big) \tag{7.2a}$$

$$\text{s.t. } \bar{u}^{(i)}_{[p\in\mathcal{S}_i]} \in \mathcal{X}_i, \forall\, i \in \mathcal{R}, \tag{7.2b}$$

$$\bar{u}_i^{(i)} = \bar{u}_i^{(p)}, \quad \forall\, i \in \mathcal{R}, \forall\, p \in \mathcal{S}_i \setminus i \tag{7.2c}$$

$$\bar{u}_p^{(i)} = \bar{u}_p^{(p)}, \quad \forall\, i \in \mathcal{R}, \forall\, p \in \mathcal{S}_i \setminus i. \tag{7.2d}$$

The equivalence between (7.1) and (7.2) is guaranteed by the connectedness of the communication graph $\mathcal{G}$.

We will now tailor the specific form of g_i and $\mathcal{X}_i$ that make the one step ahead traffic optimization separable. To ease the notation we omit the time dependency from $\bar{\rho}(t)$, $f^{\text{out}}(t)$ and $h(t)$, which are measured (the first one) or simply computed (the last two, by using (3.14b) and (5.3)). The term $\bar{u}(t-T)$ is also known and is the vector containing the duty cycles applied during the last cycle. Let g_i be defined as

$$\begin{aligned} &g_i\Big(\bar{u}_{[p\in\mathcal{S}_i]}\Big) = \\ &-k_{\text{ttd}} \min\Bigg\{ v_i\Bigg(\bar{\rho}_i + \frac{T_s}{L_i}\Bigg(\sum_{j\in\mathcal{N}_i^-} \bar{u}_j \beta_{ji} f_j^{\text{out}} - \bar{u}_i f_i^{\text{out}}(t)\Bigg)\Bigg), \\ &\qquad w_i\Bigg(\rho_i^{\max} - \bar{\rho}_i - \frac{T_s}{L_i}\Bigg(\sum_{j\in\mathcal{N}_i^-} \bar{u}_j f_j^{\text{out}} + \bar{u}_i f_i^{\text{out}}\Bigg)\Bigg)\Bigg\} + \\ &\qquad k_{\text{bal}} \sum_{j\in\mathcal{N}_i^+} \Big(h_i \bar{u}_{[i,v\in\mathcal{N}_i^-]} + \bar{\rho}_i - \\ &\qquad\qquad h_j \bar{u}_{[j,v\in\mathcal{N}_j^-]} - \bar{\rho}_j\Big)^2 + \\ &\qquad \sum_{p\in\mathcal{S}_i} \frac{1}{|\mathcal{S}_p|}(\bar{u}_p - \bar{u}_p(t-T))^2, \end{aligned} \tag{7.3}$$

and $\mathcal{X}_i$ be defined as the set of all $\bar{u}_{[p\in\mathcal{S}_i]}$ satisfying

$$\begin{aligned} \bar{u}_p &\geq l_p \quad \forall\, p \in \mathcal{S}_i \\ \bar{u}_p &\leq 1 \quad \forall\, p \in \mathcal{S}_i \\ \sum_{q\in\mathcal{I}_i} \bar{u}_q &\leq 1. \end{aligned} \tag{7.4}$$

Based on the previous discussion, two remarks matter regarding definitions (7.3) and (7.4):

- Every set $\mathcal{X}_i$ is compact and convex, since it is given by linear inequalities;
- Every function g_i is stricly convex in every variable u_p, $p \in \mathcal{S}_i$. To see this, notice that each g_i is given by the sum of three terms: the first is convex because composition of convex functions (notice that the min operator is multiplied by a negative constant); the second is convex because it is a quadratic form; the third is stricly convex because it is a quadratic form with a positive definite matrix (in particular, it is the identity matrix). Hence their sum is a stricly convex function.

Thanks to such definition of g_i and $\mathcal{X}_i$ and to their properties we have remarked, our problem is in the set–up specified by [20] whose steps we will now adapt to the traffic optimization problem.

7.1.3 A distributed algorithm for the one step ahead traffic optimization

We will now present a distributed algorithm to solve problem (7.2) that uses the well known techinque of dual decomposition. In (7.2), the objective function and the constraints (7.2b) are already separable. Therefore, the dual decomposition will consider the coupling constraints (7.2c) and (7.2d) only. To this aim, we introduce the lagrangian multipliers, denoted $\lambda_i^{(i,p)}$ and $\lambda_p^{(i,p)}$ and associated with constraints (7.2c) and (7.2d), respectively. The partial lagrangian, via simple manipulation, can be written as it follows:

$$\begin{aligned} L = \sum_{i\in\mathcal{R}} L_i = \sum_{i\in\mathcal{R}} \Bigg[g_i\Big(\bar{u}_{[p\in\mathcal{S}_i]}\Big) + \bar{u}_i^{(i)} \sum_{p\in\mathcal{S}_i\setminus i} \Big(\lambda_i^{(i,p)} - \lambda_i^{(p,i)}\Big) + \\ \sum_{p\in\mathcal{S}_i\setminus i} \bar{u}_p^{(i)} \Big(\lambda_p^{(i,p)} - \lambda_p^{(p,i)}\Big)\Bigg]. \end{aligned} \tag{7.5}$$

The dual decomposition is based on the iteration of two steps: the first is the minimization of the partial lagrangian, and the second is the update of the lagrangian multipliers λ via a gradient descent update. We now present the iterative algorithm, by illustrating the scheme that has to be executed by every subproblem i:

Initialization. Set $k = 0$. Create local variables $\bar{u}_p^{(i)}$ for every $p \in \mathcal{S}_i$, and $\lambda_i^{(i,p)}$, $\lambda_p^{(i,p)}$ for every $p \in \mathcal{S}_i \setminus i$, and initialize them such that for every i and for every $p \in \mathcal{S}_i \setminus i$

$$\lambda_p^{(i,p)}(0) = -\lambda_p^{(p,i)}(0). \tag{7.6}$$

Primal update. Update primal variables $\bar{u}_p^{(i)}(k+1)$ by minimizing the i-th term of the partial lagrangian:

$$\begin{aligned} \min_{\bar{u}_{[p\in\mathcal{S}_i]}} \quad & g_i\Big(\bar{u}_{[p\in\mathcal{S}_i]}\Big) + \\ & \bar{u}_i^{(i)} \sum_{p\in\mathcal{S}_i\setminus i} \Big(\lambda_i^{(i,p)} - \lambda_i^{(p,i)}\Big) + \\ & \sum_{p\in\mathcal{S}_i\setminus i} \bar{u}_p^{(i)}\Big(\lambda_p^{(i,p)} - \lambda_p^{(p,i)}\Big) \\ \text{s.t. } \quad & \bar{u}_{[p\in\mathcal{S}_i]} \in \mathcal{X}_i. \end{aligned} \tag{7.7}$$

Transmission. Send primal local variables to requiring neighbors $p \in \mathcal{S}_i$, and collect the most recent values of $\bar{u}_i^{(p)}$, $\bar{u}_p^{(p)}$ from them.

Dual update. Update the lagrangian multipliers, i.e., for every $p \in \mathcal{S}_i \setminus i$ set

$$\lambda_p^{(i,p)}(k+1) = \lambda_p^{(i,p)}(k) + \alpha\Big(\bar{u}_i^{(i)}(k+1) - \bar{u}_i^{(p)}(k+1)\Big). \tag{7.8a}$$

$$\lambda_i^{(i,p)}(k+1) = \lambda_i^{(i,p)}(k) + \alpha\Big(\bar{u}_p^{(i)}(k+1) - \bar{u}_p^{(p)}(k+1)\Big). \tag{7.8b}$$

Stop condition. If the solution has numerically stabilized stop, otherwise increment k and go to 1.

The convergence property of this algorithm is stated in the next proposition.

Proposition 2. *Let $\bar{u}^*$ be the unique optimal solution of the centralized optimization problem (5.10). There exists $\alpha^* > 0$ such that, if $0 < \alpha < \alpha^*$, then*

$$\lim_{k\to\infty} \bar{u}_i^{(i)} = \bar{u}_i^*,$$

for all $i \in \mathcal{R}$.

Proof. We have shown that $\mathcal{X}_i$ are compact and convex sets, and that functions g_i are stricly convex, thus satisfying the assumptions of [20, Theorem 3.3]. □

7.1.4 Comments

It is worth clarifying three points regarding the introduced algorithm. First, the initialization of local variables is subject to condition (7.6). There are three main possibilities that satisfy this requirement:

- A simple initialization to zero values, every time the optimization starts;
- A preliminary round of agreement between neighbors, that decide on the values to be assigned to the dual variables. Notice that in this case the neighborhood given by the set $\mathcal{S}_i$ is sufficient to reach this goal.
- A zero initialization for the first optimization and a *warm start* for the following ones. Since in our set-up the optimimization needs to be solved at the beginning of every traffic light's cycle, is it possible to use the optimal values computed for one cycle as initial condition for the new problem. Such a warm start choice is guaranteed to respect (7.6), because the update rule (7.8) preserves the same property. This approach can give practical benefits in terms of convergence speed.

In second place, it is important to notice that to construct problems (7.7) one only needs information of some of the neighbors in $\mathcal{S}_i$, as it can be seen from (7.3). This implies that only very few exchanges of message are needed, with some neighbor (those are actually a strict subset of $\mathcal{S}_i$) and about their current density.

Last, we would like to stress that the local optimization program stated in (7.7) can be rewritten as a convex (quadratic) program, using the same strategy outlined in section 5.2, with the introduction of auxiliary variables y_i (see proposition 1). Therefore every local primal update is a convex program with a much smaller size than the centralized algorithm. The size of such local problems will not increase with the dimension of the network, only the number of subproblems will. If an appropriate hardware infrastructure is available, i.e., every local problem can be solved in a different processing unit, and those units can communicate as prescribed by the communication graph $\mathcal{G}$, then the algorithm is scalable to large networks' size.

7.1.5 Numerical experiments

Regarding the developed distributed OSA-TC (Dist-OSA-TC), proposition 2 answers the most natural question: Dist-OSA-TC finds the same optimal value as OSA-TC, and therefore their performance in terms of traffic behavior are the same. We have presented our results in this regards in section 5.4.1, about macroscopic simulations, and in section 5.4.2 about microscopic experiments with the simulator Aimsun.

In this section we want to show results about tests we have done in order to analyze the convergence speed of the iterative distributed procedure.

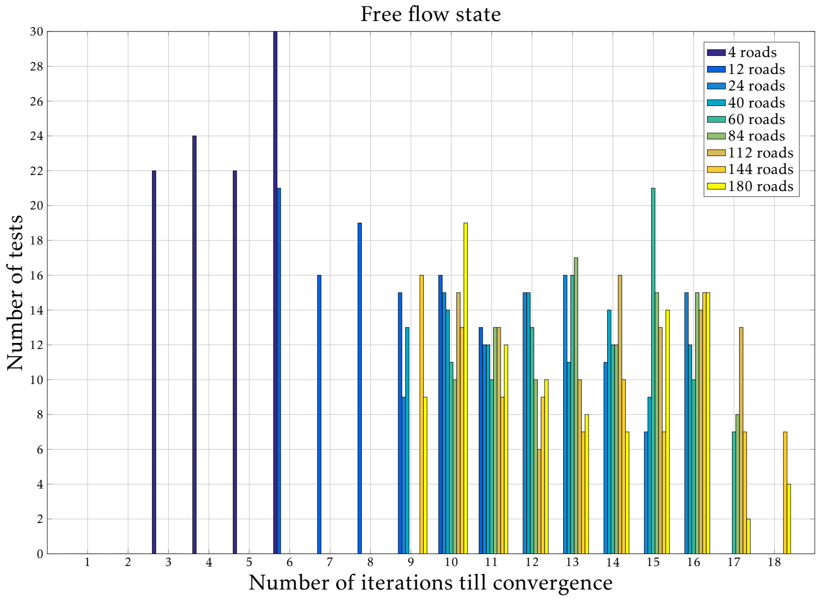
We have run a test bench of two thousand seven hundred simulations, with different network sizes and density (initial) conditions. The tests are organized as follows: the network's size (whose shape is again Manhattan grid–like) varies among nine values, starting from a total of 4 roads up to a total size of 180 roads; for each of these dimensions, we run 100 simulations with initial state of every road pseudo–randomly generated every time. Such a test is repeated three times: first, the initial roads state is free flow for all roads ($\rho(0) < \rho_c$); second, it is congested ($\rho(0) > \rho_c$); last, every road is initialized with a value that can be either free or congested ($0 < \rho(0) < \rho^{\max}$). The rationale is to test cases where roads have similar states (either free or congested) and where they have not (intuition would say that in the latter case a consensus requires more iterations).

Concerning roads' physical features, we defined homogeneous roads all over the network, with free flow speed of 50 km/h, maximum density of 200 vehicles/km and critical density of 40 vehicles/km. Thus, the maximum flow is 2000 veh/hour and the speed of backpropagation of the congestion is 12.5 km/h.

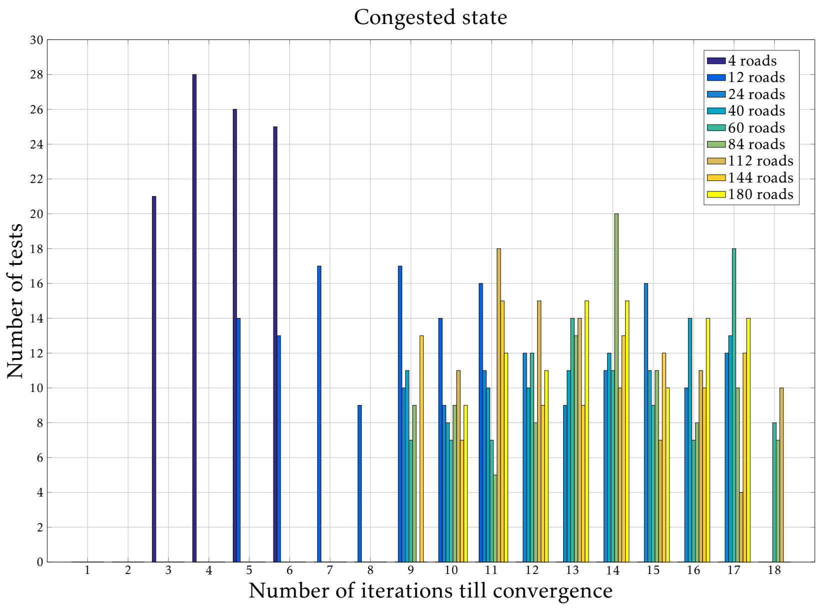
The test bench was run in the MatLab environment, using software [72, 6] to solve the optimization problems.

The results are reported in the histograms in Figure 7.2. We declared that the distributed algorithm has converged when the change in its solution from one iteration to the next one is less (in absolute value) than 1e-3. Firstly, and most importantly, we notice that the maximum number of iteration needed to the algorithm to converge, overall all run simulations, is less than 30, that we consider very efficient since every iteration takes around 0.05 seconds to run. This time would also be approximately the overall time needed for an iteration over the entire set of subproblems, assuming that each of them has a separate hardware available (e.g., each traffic light is integrated with a microprocessing unit). Time needed for communication should be taken into account too; however, in our algorithm communication exists only between roads physically close, hence the time needed for all iterations results to be comparable to the time needed by the centralized algorithm, in which the solution, once it has been computed, needs to be sent from the single computing machine to roads that are also very far.

We also notice that, in case of someway balanced network (both free and congested) this number is actually even lower, with a maximum number of iterations equal to 18. When densities all over the network have more spread values, then the number of iteration grows a bit, as we were expecting, but still stays in a fully acceptable range.

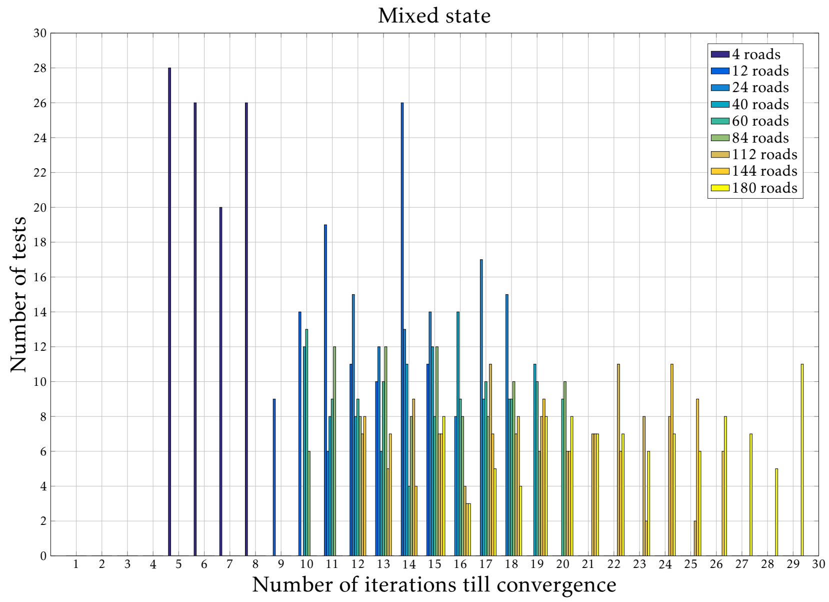


(a)



(b)

Figure 7.2 – Results of the test bench on convergece speed of the distributed algorithm.



(c)

Figure 7.2 – Results of the test bench on convergece speed of the distributed algorithm (cont.).

7.2 Distributed multi step ahead traffic control

In chapter 6 we discussed the design of a control technique based on prediction and optimization that uses the 2DoF-CTM. Such an algorithm was formulated as a mixed integer linear program thanks to the Big-M tool, that allows to reformulate the nonlinearities of the model as linear constraints with integer variables. The numerical tests we presented in section 6.7 show that the control scheme is powerful in terms of traffic performance, but it is not as for its computational efficiency. This is yet another reason why we searched for a distributed version of it.

However, linear integer optimization does not have geometrical properties (such as, e.g., convexity) that concede the use of well established techniques for distributed optimization (for example, in the design of the Dist-OSA-TC we used the dual descent). Therefore we designed a custom solution for this particular problem, being aware that the obtained solution will be suboptimal.

Before presenting the algorithm, it is worth recalling that in the MSA-TC the decision variables are the time instants $\sigma_i^{(1)}, \sigma_i^{(2)}$, corresponding to the switching instants from red to green and viceversa, for traffic light i. In order to use a common reasoning in distributed procedures, the algorithm we propose is based on two main ingredients: suboptimal solutions for local problems and agreement policy between local solutions.

7.2.1 Local problems

The optimization procedure is decentralized among intersections, each of which solves a local MILP. For every local problem, say over intersection A, a receding horizon approach is used and the predicted densities belong to the set

$$\mathcal{R}_{\mathrm{A}} = \Big\{ i \in \mathcal{R} : i \text{ entering or exiting A} \Big\}, \tag{7.9}$$

while the optimization variables are all traffic lights in the following set:

$$\Omega_{\mathrm{A}} = \Big\{ \sigma_i : i \in \mathcal{R}_{\mathrm{A}} \Big\} \cup \Big\{ \sigma_q : q \in \mathcal{N}_i^-, i \text{ entering A} \Big\}. \tag{7.10}$$

Furthermore, densities for roads in $\mathcal{R} \backslash \mathcal{R}_{\mathrm{A}}$ are considered constant (equal to the last measured values). This is the main reason of suboptimality, since the objective function maximized in every subproblem is computed over the set $\mathcal{R}_{\mathrm{A}}$, i.e.,

$$J_{\mathrm{A}} = \sum_{k=1}^{T/T_s} \Bigg(k_{\mathrm{ttd}} \sum_{i \in \mathcal{R}_{\mathrm{A}} \backslash \mathcal{R}^{\mathrm{in}}} \mathrm{TTD}_i(t + kT_s) + k_{\mathrm{sod}} \sum_{i \in \mathcal{R}_{\mathrm{A}} \cap \mathcal{R}^{\mathrm{in}}} \mathrm{SoD}_i(t + kT_s) \Bigg). \tag{7.11}$$

As illustrative example look at figure 7.3, where we consider the optimization solved by intersection A. In such a scenario density predictions are carried out only for roads 1–4, and the problem is solved with respect to the variables $\sigma_1, \dots, \sigma_8$ (notice that $\sigma_5, \dots, \sigma_8$ are needed to compute inflows for roads 1 and 2).

It is worth noticing that, among all σ's considered by an intersection, only a subset of them fulfills hard constraints for collision avoidance within the local optimization: referring again to figure 7.3, intersection A guarantees such fulfillment only for σ_1 and σ_2. For the other variables considered by A (marked in red in the figure) there might be some hard constraint which is not included within the local problem (in figure 7.3 this happens for σ_3 and σ_4); Therefore, the values computed by A for these variables can be interpreted only as suggestions that A would like to advice its neighbor intersections, to optimize its own objective. How such suggestions are considered is explained in the next section.

7.2.2 Agreement policy

To let local problems taking care of advices provided by neighbors the algorithm prescribes to save the result of every local optimization. For instance, when A solves its subproblem, we save the variables $\sigma_{i,\mathrm{A}}$, for every i involved in such problem. Once these informations are available we can identify, for every road i, the set $\mathcal{O}_i$ of all intersections whose local problem involves σ_i. Then the average suggestion given by intersections about σ_i is

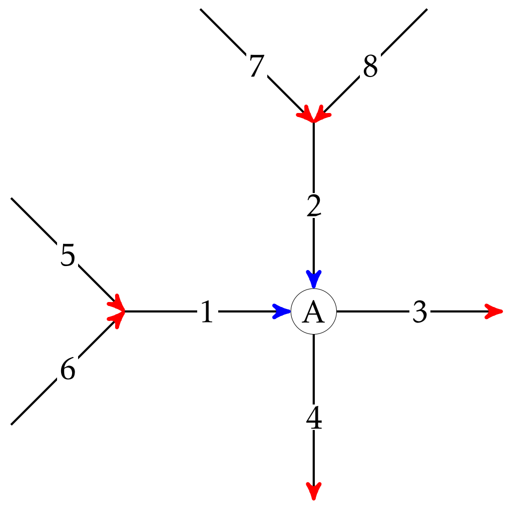


Figure 7.3 – Illustrative example of local subproblems for the Dist-MSA-TC.

computed as:

$$\hat{\sigma}_i = \frac{1}{|\mathcal{O}_i|} \sum_{\mathrm{I} \in \mathcal{O}_i} \sigma_{i,\mathrm{I}}. \tag{7.12}$$

We now use the values $\hat{\sigma}_r$ to modify the cost function of every local problem, as the following:

$$\hat{J}_{\mathrm{A}} = J_{\mathrm{A}} - a_3 \sum_{\sigma_i \in \Omega_{\mathrm{A}}} \|\sigma_i - \hat{\sigma}_i\|_1, \tag{7.13}$$

where a_3 is a real number used to weight neighbors suggestions with respect to A's own objective. Notice also that the 1–norm can be turned in a mixed integer linear formulation adding two binary variables for every $\sigma_i \in \Omega_{\mathrm{A}}$. This is the computational prize to pay in order to take into account suggestions by neighbors, rather then ignoring them; the problem, however, is still significantly numerically more efficient than its centralized version. This idea can be iterated N_{it} times, in order to let suggestions spread among intersections. The scheme we implemented is the following:

0. Let N_{it} be assigned;
1. For every intersection A solve the local MILP with cost function J_{A} and save the resulting $\sigma_{i,\mathrm{A}}$;
2. If $N_{\mathrm{it}} = 0$ then stop;
3. For every signalized road i compute the average suggestions $\hat{\sigma}_i$;
4. For every intersection A solve the local MILP with cost function $\hat{J}_{\mathrm{A}}$, and use the result to update $\sigma_{i,\mathrm{A}}$;

5. If for every i and for every A there was no update in $\sigma_{i,\mathrm{A}}$ then stop;

6. Decrement N_{it} by 1, goto 2.

Whenever the algorithm stops, the value assigned to every traffic light is

$$\sigma_i^* = \sigma_{i,\mathrm{A}}, \text{ where } i \text{ enters A}, \tag{7.14}$$

since every intersection guarantees the fulfillment of collision avoidance (hard) constraints only over roads entering it.

We underline that if $N_{\mathrm{it}} = 0$ is given, this means the suggestions from neighbors are ignored. An important feature of our scheme is that the subproblems' complexity does *not* depend on the network size; therefore the algorithm complexity is linear with respect to the number of intersections and to the chosen number of iteration, which is a tunable parameter. In our numerical tests, presented in the next section, the algorithm stops already after five iterations as no changes appear in the solution.

Another benefit of this decentralized strategy is that step 1 (the same applies to step 4), requires to solve a set of optimization problems that are completely indipendent of each other. Therefore the procedure can be implemented even more efficiently in a parallel architecture, where there is a controller at every intersection that exchanges σ's values with the others.

7.2.3 Performance analysis

The distributed multi step ahead traffic control (Dist-MSA-TC), unlike the Dist-OSA-TC, is not convergent to the global optimum of the corresponding centralized problem (6.16). Therefore, it is interesting to analyze how far from the global optimum the solution provided by Dist-MSA-TC is.

For this, we used again the scenarios in figure 6.2, whose centralized solution had already been computed and analyzed in section 6.7, and we tested on them the Dist-MSA-TC. The results of these tests are reported in table 7.1.

- First and foremost, we remark that the computational burden of the MSA-TC is extremely reduced in the Dist-MSA-TC, from an average time of 3 minutes, in the smallest scenario, and 6.5 minutes, in the largest, to 4 seconds in both (such a value is computed as the average execution time among subproblems times N_{it};

- Regarding traffic performances, the solution is suboptimal, as expected, and the loss is between 8% and 13%. Interestingly, this make the Dist-MSA-TC closer to the OSA-TC (whose results are also reported in table 7.1, to ease the reading), although the former is still

Table 7.1 – Comparison between the multi step ahead traffic control (MSA-TC) and its distributed suboptimal version (Dist-MSA-TC), in the scenarios proposed in figure 6.2. The table shows normalized values of macroscopic traffic indexes (total travel distance, density balancing and service of demand), as well as measured computation time, averaged over all run optimizations, for the two control algorithms. Also, the same indexes regarding the OSA-TC are reported from table 6.1.

	Network 6.2a			Network 6.2b		
	MSA-TC	Dist-MSA-TC	OSA-TC	MSA-TC	Dist-MSA-TC	OSA-TC
TTD	100%	92%	88%	100%	87%	85%
Bal	100%	91%	91%	100%	89%	90%
SoD	100%	89%	89%	100%	90%	92%
cpu time (avg)	3min.	4sec.	0.2sec.	6.5min.	4sec.	0.26sec.

slightly better by means of the macroscopic indexes, and the latter is in terms of traffic performances.

7.3 Final comments on the chapter

In this chapter we have shown the results of our research about distributed algorithms for traffic lights control. Our motivation lies in the need for scalabilty of such algorithms, both for security and for performance reasons: large scale urban networks need scalable traffic lights control algorithms in order to be deployed in the real world and have a positive impact on it.

Thus, we searched for distributed implementations of the algorithms presented in earlier chapters, the OSA-TC and the MSA-TC. Regarding the fist one, its geometrical properties, in particular about convexity of cost function and constraints, allow us to use results from distributed optimization in order to develop a distributed algorithm that is provably equivalent to it, i.e., the solutions given from the two are the same. Hence, traffic performances are ensured to be the same, and our numerical tests show that the distributed one, that is based on an iterative scheme, converges to such a solution in a small amount of iterations, that guarantees also its computational efficiency.

As for the MSA-TC, we developed a custom distributed algorithm based on local subproblems and agreement policy among them, that however is suboptimal, the Dist-MSA-TC. The MSA-TC was designed as a difficult problem taking care of a complex model, and it is not surprising that a suboptimal solution has to be accepted to improve the efficiency. This results in a loss between 8% and 13% with respect to the centralized solution.

A further, interesting, comparison arises from these tests. While the MSA-TC was significantly better than the OSA-TC (due to the model complexity of the first one), the Dist-MSA-TC goes much closer to the OSA-TC, with some small difference in favor of the former, regarding traffic performances, and in favor of the latter, about computational efficiency.

Chapter 8

Realistic implementation and comparison with the Grenoble *Plan de Feux*

In this chapter we will present our work about experiments with the industrial traffic lights plan (*plan de feux*, in French) that is deployed in many large intersections in the city of Grenoble. We obtained the detailed specification thanks to an agreement with the public office in charge, the Grenoble-Alpes Métropole[1], to whom we are grateful for the several conversations and explanations on the subject, too.

The study of the plan de feux reveals several elements that we believe are of utmost importance for the analysis of other approaches and their usability. Notably, as we will explain in detail in the chapter, the plan de feux shows the existance of several hard constraints that have to be respected by any other controller and that leave only a small window for an optimization approach. These include, for example, the minimum and maximum green time that every traffic light needs to have, as well as the high priority assigned to public transport vehicles.

We will start by describing the scenario in which the analyzed plan de feux is used: it consists of eight large intersections along one of the main arterials in the core of the city, and includes one tram line running over the same arterial, as well as two other tram lines intersecting it, and several bus stops. The aforesaid intersections are signalized to guarantee the safe crossing of vehicles, trams and pedestrians, and the traffic lights are regulated by the plan de feux, of which we did a reverse engineering work. We will hence detail its working principle and the technological and methodological differences with respect to the optimization–based approaches that have been presented in the earlier chapters. Then, we will show how one among such approaches can be adapted in order to respect the imposed

[1] *http://www.lametro.fr*

constraints. The chapter is concluded with the presentation of the simulative tests, once more in Aimsun, that we executed in order to compare the performance of the plan de feux against the one step ahead approach adapted to this scenario.

8.1 Scenario description

The studied plan de feux is used in Grenoble for the intersections along one of the main arterials of the city, called *Grand Boulevard*. It is a long and wide road of approximately 4 km that crosses the city from east to west, with two or three lanes for each of the two directions. Its main features are (see also figure 8.1):

- A public tram line follows the arterial from east to west and viceversa, with one reserved lane per direction, and two other tram lines cross the arterial in different points;
- Several busses run along the Grand Boulevard and in proximity of it, and numerous bus stops are located nearby the intersections;
- Eight large signalized intersections are currently controlled by the plan de feux, in order to guarantee the safe crossing of cars, trams, busses and pedestrians.

This scenario has been reconstructed in Aimsun in every minute detail: the dimensions of lanes, roads and intersections have been measured, as well as the distance between traffic lights and intersections. The speed limits have been imposed as they are in the city in every road.

The reserved lanes for trams and busses are also included into the Aimsun model, and we generated the timeline for public transports copying it from the city's timeline from 7a.m. till 10a.m. (which is available online). Thanks to the collaboration with the public office in charge of studying the mobility in Grenoble[2] we also had access to data collected by loop detectors installed in the cities. These are around 230 loop detectors, located as in figure 8.2, that give information about flow of vehicles; we used such pieces of information to calibrate the demand into the Aimsun model and to calibrate also every junction with reasonable values of split ratios.

Thus, the simulated scenario emulates the real traffic behavior in Grenoble from 7a.m. to 10a.m. very closely. Furthermore, we designed a piece of software that reproduces the real behavior of the plan de feux, and integrated it into the Aimsun simulation. In this way, not only external conditions (demand, split ratios, public transport timelines) but also the reactive behavior of traffic lights corresponds to the real one. We detail this subject in the next section.

[2] *http://www.metromobilite.fr*

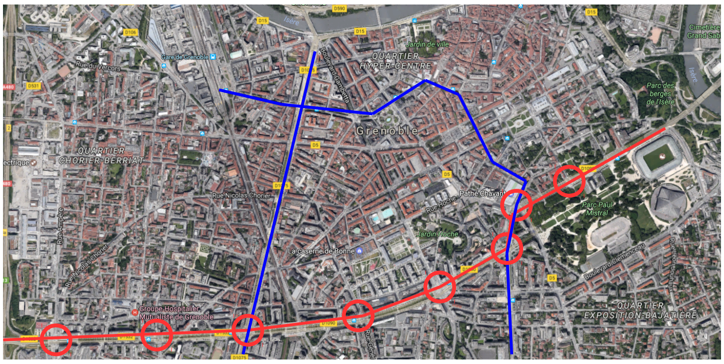


Figure 8.1 – Illustration of the scenario in which the plan de feux is deployed and compared with the proposed approach. In red, the *Grand Boulevard*, made of two or three lanes in both directions. Along the same red line, a public tram is also running, with an average frequency of 1 tram every 5 minutes from 6a.m. to 7p.m. In blue, the two other lines of public tram that cross their path with the analyzed scenario. All over the scenario, several busses serve the citizens with bus stops in proximity of every intersection. The red circles represent the intersections where the plan de feux controls the traffic lights, installed for the safe crossing of cars, busses, trams and pedestrians. The view of the city is taken from Google Earth.

Figure 8.2 – Positions of the loop detectors installed in Grenoble. We used the measured data about flow of vehicles in order to set up the demand into the Aimsun model and to calibrate every junction with reasonable values of split ratios. The view of the city is taken from Google Earth.

8.2 The Grenoble plan de feux

The Grenoble plan de feux is a logical controller, whose working principle recalls the one of a finite state machine. The main technological elements are the following:

- Every intersection is equipped with several detectors. These are on–off detectors located in proximity of the intersection, that communicate to the processing unit the presence (or the absence) of vehicles only;

- Every intersection in the Grand Boulevard is controlled by a *single–intersection controller*, that is, the decision about traffic lights in an intersection is taken without considering traffic lights in the others. This due to the current absence, to date, of a technology that would allow communication between intersections along the Grand Boulevard;

- Different type of sensors are used to know the position of trams and busses and adjust the plan de feux accordingly. More precisely, a fusion of data from busses' GPS and magnetic detectors for tram is used in order to say to the controller *how many seconds is the vehicle (tram or bus) away from the junction*.

The above points lead to some comments worth underlining about the technological differences between this industrial controller and our proposed approaches. First of all, the algorithms we developed are based on density measurements rather than on–off detections. However, as we discussed also in the earlier chapters, we believe this is not a technological limitation: for instance, the same loop detectors could be used for density measurements if they are installed at the beginning and at the end of a road, via a simple numerical integration scheme (based on the difference between measured outflow and measured inflow); also, a camera installed in the intersection could be used, along with image processing algorithms, to count the number of vehicles with high accuracy, and hence the density would be easily computed. Moreover, we assume that the same alternative technology is also used to measure the flow that entered a section during the last 60 seconds.

In second place, the single-intersection controller is a simplified instance of the algorithms illustrated in the previous chapters, which were either centralized optimization (hence collecting information globally) or distributed with exchange of messages between neighbors. On one hand, the single-intersection approach is expected to have worse performance than a global controller; on the other hand, the optimization task is much easier (simply, the problem to be solved has smaller size). However, we respected this technological constraint and adapted the one-step ahead traffic control to fulfill it.

Third, information about bus and tram sensors needs to be considered. As we will explain, this is a crucial point for the plan de feux, since it assignes the highest priority to the public transport; therefore, we assumed to have the same technology at our disposal, and complied the need for high priority to trams and busses.

Regarding the methodological features of the plan the feux, the following are the most relevant:

- In each intersection, the set of all traffic lights is divided into subsets, called *phases*. Each phase corresponds to a group of traffic lights that can have green at the same time. The division into phases is decided by traffic engineering experts, and it is based on the topology of the intersection;
- To every phase a minimum and a maximum green time are assigned, as well as the amber time; these parameters depend on the speed limit, the number and size of lanes, the size of the intersections, etc.

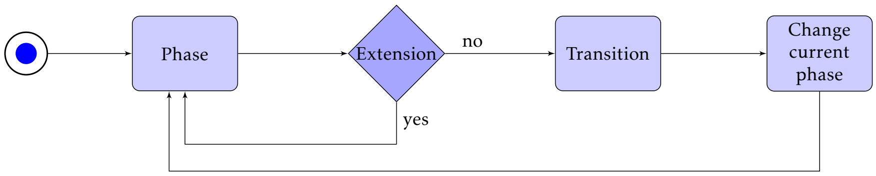
It is worth mentioning that the second of the above points is a type of constraint already considered in our previous formulations, see e.g., (5.10). As for the first one, we comply with the choice of grouped signal (phases) made by the traffic engineers for this specific scenario. Thus, the adapted one step ahead control will decide about duty cycles for phases, rather than for single traffic light.

The plan de feux is based on a recursive scheme phase–transition–phase, explained in depth in the next section. Roughly speaking, at every time instant a phase is active and the controller takes a decision about extending this phase or switching to a transition. A transition is a fixed scheme of green lights chosen in order to prepare the transit to another phase (see figure 8.3 and the explanation therein). The decision is made every second.

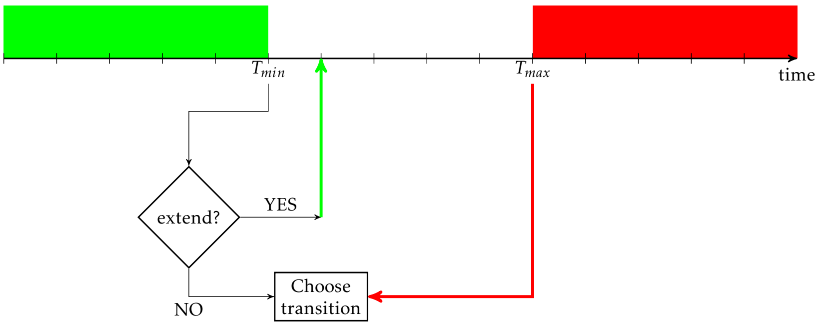
8.2.1 Reverse engineering of the plan de feux

The controller that decides the plan de feux actions is a logic program based on a dialect of the T language (see, e.g., [104]). For the Grand Boulevard scenario, every intersection controller comes with its formal description in a technical manual; the set of manuals for the eight considered intersections was kindly conceded by the Grenoble-Alpes Métropole. These manuals contain information about

- The topology of the intersection, and the choice of phases;
- Constant parameters, such as the amber time;



(a)



(b)

Figure 8.3 – Top level view of the plan de feux's working principle. As shown in figure 8.3a, the plan de feux is a recursive scheme that at every time instant decides whether to extend the phase currently active, or to switch to a transition. A transition is a fixed combination of traffic lights (a few seconds long) that prepares the activation of the next phase. In figure 8.3b more details are given about this scheme: mainly, the active phase cannot be interrupted before the expiration of its minimum green time, and it cannot be extended if it reaches its maximum green time. Moreover, the controller takes the decision at every sampling time, that is 1 second.

- Constrains about minimum and maximum green time for each phase;
- Formal description of the logic rules for switching, in T language.

After studying the manuals, common elements between the intersections appear:

- The intersections' topology are quite similar; each of them has at least three phases, and no more than six;
- Standard minimum green time for phases is between 10 and 20 seconds; instead, maximum green time is usually between 40 and 55 seconds;
- The trams have a very strong priority: some phases for cars are interrupted even 10 seconds in advance if a tram is about to arrive. This is a directive of the city hall.

We will now explain in depth the plan de feux algorithm by examining one representative intersection among the ones of the Grand Boulevard. This is unformally called *Foch Ferrié* and it approximately corresponds to the coordinates 45°10′51.3″ N, 5°43′23.4″ E.

This intersection counts 12 traffic lights, among which two are for the reserved tram lanes and ten for the other vehicles (busses included). The dimensions and the shape of the intersection and related roads influence the choice of the phases, that are four: roughly speaking, one phase is for trams, one for the horizontal crossing, one for the vertical, and one more for the left–turns. See figure 8.4 for an illustration.

As earlier mentioned, the main elements of the recursive scheme, deduced from the manual, are the extension rule and the switching choice.

Phases extension logic

The rules adopted in the plan de feux for a phase extension are in general of low complexity. For instance, phase A in the Foch Ferrié intersection is extended when the following statement (reported fully from the related intersection manual) holds true,

$$\text{(DM4-2 OR DM8-2) AND dA>100,} \tag{8.1}$$

where DM4-2 and DM8-2 are shortnames for "detected measure", plus the identifier of the detectors, and dA stands for estimated time left before any tram arrives at the intersection. Thus, (8.1) states that the phase has to be extended if any vehicle (notice the *or* condition) is detected from the related sensors and if the tram is still far away. Most of the phases have a similar rule for the extension. The most interesting exception is represented by the phases related to the trams (for example, phase T in Foch Ferrié), whose extension is equivalent to the statement *the tram already completed the crossing and no other tram is close enough*, where the "close enough" is intersection–dependent.

Choice of the transition

If the extension is not available for a phase then the controller chooses a transition. Transitions are fixed slots of time during which the traffic lights are assigned according to a pre–defined choice. For example, figure 8.5 shows the transition Tr3 in Foch Ferrié. This transition realizes the bridge from phase A to phase T (see table 8.1) and, according to the intersection's manual, is activated if

$$dA \leq 11,$$

which explains why the transition is 12 seconds long and during the last seconds green lights are assigned to T1 and T2. If the condition does not hold true then the controller will choose Tr1 from A to B (we are still assuming that A cannot be extended). However, the plan de feux has a strong tendency to favor the phases related to the trams: indeed, in this example, if the controller passes to phase B, then from this phase it can again switch to phase T via transition Tr13, which is activated if

$$dA \leq 9.$$

For the sake of clarity we would like to underline that the rule for selecting the transition can be very complex at time, in particular in the intersections where different tram lines merge. We have presented the simplest only, in order to illustrate the type of logic the plan de feux uses.

Replication of the plan de feux

A major goal of this study is to be able to compare the plan de feux and the optimization–based control in the same scenario. Therefore, we replicated the design of the plan de feux in an object–oriented architecture that can be run within Aimsun, which allows online programming in the Python language. Thus, the replicated plan de feux is a separate piece of software that is called by the Aimsun environment at every second of simulation (i.e., every sampling time of the real plan de feux). When it is activated, the replicated plan de feux uses the Aimsun API in order to simulate the detectors that are used by the original plan de feux, as well as the trams distance from the intersections. Using these bits of information it makes decision based on the same logic rules about extension and switching, that we reproduced fully from the plan de feux manuals into the software.

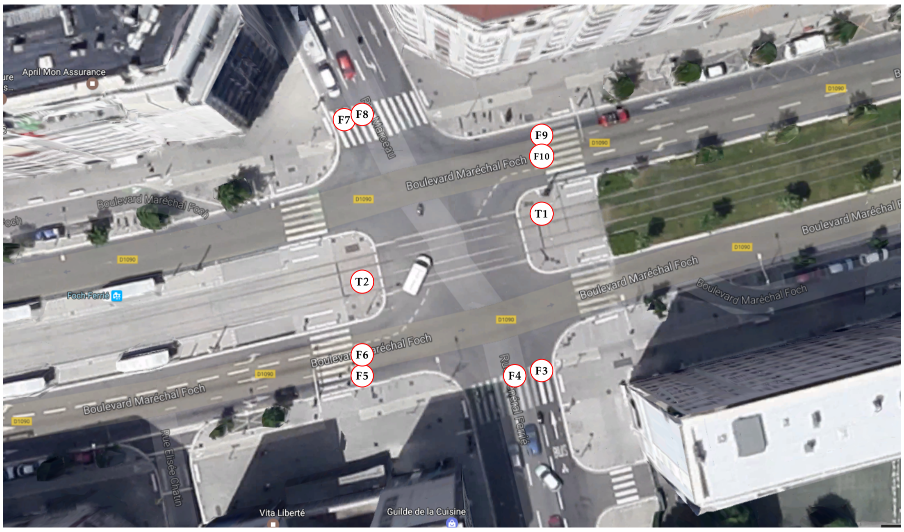


Figure 8.4 – Topology and traffic lights for the illustrative intersection. The background image is taken from Google Earth.

Table 8.1 – Choice of the phases (on the left) and phase-transition-phase scheme (on the right) for the illustrative intersection (in figure 8.4).

Phase	Traffic lights
A	F3,F4,F7,F8
B	F5,F9
C	F6,F10
T	T1,T2

From\to	A	B	C	T
A	×	Tr1	×	Tr3
B	Tr8	×	Tr11	Tr13
C	Tr22	×	×	Tr24
T	Tr25	×	Tr27	×

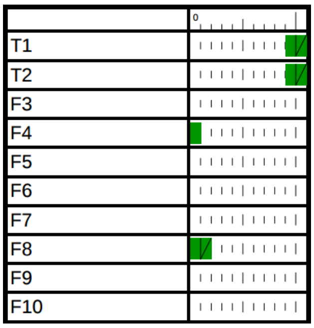


Figure 8.5 – Example of transition. The figure shows the transition Tr3 from phase A to phase T in the intersection Foch Ferrié (see table 8.1), and it is taken from the manual of the plan de feux.

In the same objected–oriented architecture we also implemented the optimization–based control (explained in detail in the next secion, and based on chapter 5). In this way, the two different controllers can be readily tested and compared in the same scenario.

Finally, we added to the architecture a module that is in charge to retrieve data of interest for the a–posteriori analysis. For example, this module tracks every vehicle in order to measure its waiting time at every intersection.

8.3 Adaptation of the one step ahead traffic control

We will now present how the OSA-TC is adjusted to fit the scenario and the imposed constraint, in order to compare its performance with the plan de feux. As mentioned in the previous section, the optimization is a single–intersection controller. Let us define $\mathcal{P}$ as the set of phases in a given intersection, and for every phase $p \in \mathcal{P}$ and every road i connected to the same intersection let $p(i)$ be the phase that includes the traffic light u_i. Also, for every phase p let $\bar{\eta}_p$ be the duty cycle of it. Let us also partition the set of all roads connected to the intersection into the set $\mathcal{I}$ of roads entering it, and the set $\mathcal{J}$ including the remaining (hence exiting) ones.

Density predictions are therefore computed as follows,

$$\begin{aligned}
\forall\, i \in \mathcal{I}: \quad & \bar{\rho}_i(t+T_s) = \bar{\rho}_i(t) + \frac{T_s}{L_i}\Big(f_i^{\text{in}}(t) - f_i^{\text{out}}(t)\bar{\eta}_{p(i)}\Big) \\
\forall\, j \in \mathcal{J}: \quad & \bar{\rho}_j(t+T_s) = \bar{\rho}_j(t) + \frac{T_s}{L_j}\Big(\sum_{i\in\mathcal{I}} \beta_{ij} f_i^{\text{out}}(t)\bar{\eta}_{p(i)} - f_j^{\text{out}}(t)\bar{u}_j\Big),
\end{aligned}$$

where

- The exiting flows f^{out} are computed according to the Avg-CTM (3.14),
- The entering flows f^{in} are an average over T_s of the measured flows (see section 8.2),
- The terms $\bar{\eta}$,$\bar{u}$ are the variables to be decided.

Thus, the optimization problem, based on the OSA-TC (5.10) and adapted for this scenario, is,

$$\begin{aligned}
\min_{\bar{u},\bar{\eta}} \quad & k_{\text{bal}} \sum_{i\in\mathcal{I}} \sum_{j\in\mathcal{J}} \Big(\bar{\rho}_i(t+T_s) - \bar{\rho}_j(t+T_s)\Big)^2 - k_{\text{ttd}} \sum_{v\in\mathcal{I}\cup\mathcal{J}} f(\bar{\rho}_v(t+T_s)) \\
\text{subj. to,}\ & \forall\, j \in \mathcal{J},\ 0 \le \bar{u}_j \le 1 \\
& \forall\, p \in \mathcal{P},\ 0 \le \bar{\eta}_p \le 1 \\
& \forall\, (p_1,p_2) \in \mathcal{P}\times\mathcal{P},\ \bar{\eta}_{p_1} T_{\max,p_2} \ge \bar{\eta}_{p_2} T_{\min,p_1} \\
& \qquad \bar{\eta}_{p_2} T_{\max,p_1} \ge \bar{\eta}_{p_1} T_{\min,p_2}
\end{aligned} \tag{8.2}$$

Some comments are worth discussing about such a formulation:

- The decision variables of (8.2) are the duty cycle for roads entering and exiting the intersection. However, collision avoidance constraints cannot be guaranteed for roads exiting the intersection, because this will require to consider also duty cycles of other roads. In section 7.2 we discussed a distributed scheme that adress this problem and that requires some exchange of messages between intersections. Given the technological limitation of the plan de feux, we avoid sharing of information between intersections, in order to fairly compare the two approaches, and the variables $\bar{u}$ are simply not used. Referring to the discussion in section 7.2, this would correspond to set $N_{\text{it}} = 0$.

- The resulting $\bar{\eta}$ are the duty cycles for each of the phases. However, the plan de feux does not specify a fixed cycle length, therefore after every optimization is completed we also select a value for the cycle length as it is explained below.

The only constraint that can be deduced from the plan de feux regarding the cycle length T_{cycle} is about its bounds,

$$\sum_{p\in\mathcal{P}} T_{\min,p} \le T_{\text{cycle}} \le \sum_{p\in\mathcal{P}} T_{\max,p}. \tag{8.3}$$

However, once the $\bar{\eta}$s have been computed, they give other constraints regarding the cycle length: indeed, to guarantee minimum and maximum green time for every phase it must be

$$\forall\, p \in \mathcal{P}, \quad \frac{T_{\min,p}}{\bar{\eta}_p} \le T_{\text{cycle}} \le \frac{T_{\max,p}}{\bar{\eta}_p}. \tag{8.4}$$

Combining (8.3) with (8.4) a narrow window of possible values is left, and we execute a linear search within it. Let us illustrate the general idea with a simple example. Suppose in an intersection there are two phases, $\mathcal{P} = \{\text{A},\text{B}\}$, for which green times T_{A} and T_{B} must be decided such that $15 \le T_{\text{A}} \le 45$ and $20 \le T_{\text{B}} \le 40$. Then, from (8.3) one can only infer $35 \le T_{\text{cycle}} \le 85$. Let us suppose the results of the optimization (8.2) are $\bar{\eta}_{\text{A}} = 0.3$ and $\bar{\eta}_{\text{B}} = 0.7$. So, in order to ensure $T_{\text{A}} \ge 15$ it must be $T_{\text{cycle}}\bar{\eta}_{\text{A}} \ge 15$, hence $T_{\text{cycle}} \ge 50$; to ensure $T_{\text{A}} \le 45$ it must be $T_{\text{cycle}}\bar{\eta}_{\text{A}} \le 45$, hence $T_{\text{cycle}} \le 150$. Using the same reasoning about T_{B} it yelds $T_{\text{cycle}} \ge 28.5$ and $T_{\text{cycle}} \le 57$. Thus, combining all constraints, it must be

$$50 \le T_{\text{cycle}} \le 57,$$

and a linear search (with 1 second step, that is the sampling time of the plan de feux) is applied within this window, looking for the value of T_{cycle}

that results in the lowest value of the cost function in (8.2). Notice that this last operation is very fast, since it consists in computing a few terms only (as the problem's size is small for a single intersection) and compare their numerical values.

One final comment about this approach regards its feasibilty. First of all, the existence of a solution for problem (8.2) is always ensured by the values of T_{max} and T_{min} assigned a priori to each phase. In second place, it must be avoided that the procedure contain conflicting constraints such that they generate an empty feasible set of values for the conclusive linear search. This is guaranteed by the last couple of constraints in (8.2), which ensure the constraints generated by the minimum green times do not conflict with the ones generated by the maximum green times, i.e, for every pair of phases (p_1, p_2) it must be

$$\frac{T_{\text{min},p_1}}{\bar{\eta}_{p_1}} \leq \frac{T_{\text{max},p_2}}{\bar{\eta}_{p_2}}.$$

Overall, this is a simple adaptation of the OSA-TC that fulfills the imposed technological constraints.

8.3.1 Implementation details

We conclude this section with two comments on the practical implementation of the optimization–based control. In first place, amber times for the intersection are not explicitly part of the optimization procedure. This is due to the fact that their values are stricly imposed by the topology of each intersection, and they are stated in the plan de feux's manual. Therefore, we used the same values, that are applied after the green time for a phase ends, and before to start a new phase.

Second, it is important to respect the desired priority for the public transport (in the Grand Boulevard scenario, mainly the tram should be addressed, since it has reserved lanes). We therefore implemented a reactive behavior to the trams arrival: we suppose to have the same technology used by the plan de feux to measure the tram distance, and when a tram is approaching this is seen as a disturbance of the scheduled phase; so, the other phases are stopped to let the tram cross. When the tram has completed the crossing the optimization problem is solved again, resulting in a new scheduling that is deployed. A different approach such as to include explicit constraints about the trams arrival in the optimization would make the modeling task much more difficult and would hardly be accurate, considering the random variability in such arrivals.

8.4 Simulation and comparisons

In this section we will present the results of our simulative tests meant to compare the performance of the optimization–based control with respect to the plan de feux, in the Grand Boulevard scenario.

As earlier said, we used real data measured from loop detectors to perform a first set of simulations. Using such data, the microsimulator is calibrated as to generate input demand for three hours of simulated time that corresponds to the measured flows at the borders of the scenario. The measured internal flows have been used to set, in every intersection, reasonable values of split ratios.

First of all, we compare the numerical values that the microsimulator computes regarding various traffic indexes. Such results are reported in table 8.2; we notice that all relevant indexes are improved when the intersections are controlled with the optimization–based approach, rather than with the plan de feux. Macroscopic measurements such as TTD, TTT and input flow are improved, and even more so the microscopic indexes such as the mean queue (from 152 to 113 vehicles, that is 25% better) and the stop time (from 110 to 83 seconds/km, that is 24% better).

In order to make a finer analysis, we measured the average waiting time (or the lost time) at every intersection. Given the specific topology of the scenario, we separate the measurements of vehicles that are traveling from west to east and viceversa. A graphical illustration of these results is given in figure 8.6, and the numerical values are reported in table 8.3. We remark that when the optimal control is used, it results in a save of time in all intersections, from a minimum of five seconds (Hotel de ville, west to east) to a maximum of 25 seconds (Ampere Vallier, west to east, and Hotel de ville, east to west).

For this scenario, calibrated with real data, we report also the measurements of the queue length cumulated over the entire network. As it can be seen from figure 8.7, if the proposed optimization–based control is in charge of regulating the traffic lights then the queue length is constantly improved across the three hours of simulation.

For a broader analysis, we performed a different set of eight simulations, each with a different value of input demand, from a minimum of 3 thousands vehicles in one hour, to a maximum of 10 thousands. Each of these simulations is run for 60 minutes. In figure 8.8 we report the measured values of travel time for vehicles that travel the entire length of the Grand Boulevards, in any of the two directions. Comparing the two different controllers, we notice that the optimal control leads to lower travel times, with a gain of around 50 seconds in the least congested case, up to a gain of around 7 minutes in the most congested ones.

Table 8.2 – Comparison of traffic indexes in the Grand Boulevard scenario. The reported values result from a simulation of three hours where the external demands correspond to the data measured by the loop detectors installed in the city from 7a.m. to 10a.m.

Index	Plan de feux	Optimal control
Input Flow [veh/h]	6097	6151
Mean queue [veh]	152	113
Stop time [sec/km]	110	83
TTD [km]	16307	16411
TTT [h]	794	680
Veh. waiting [veh]	333	240

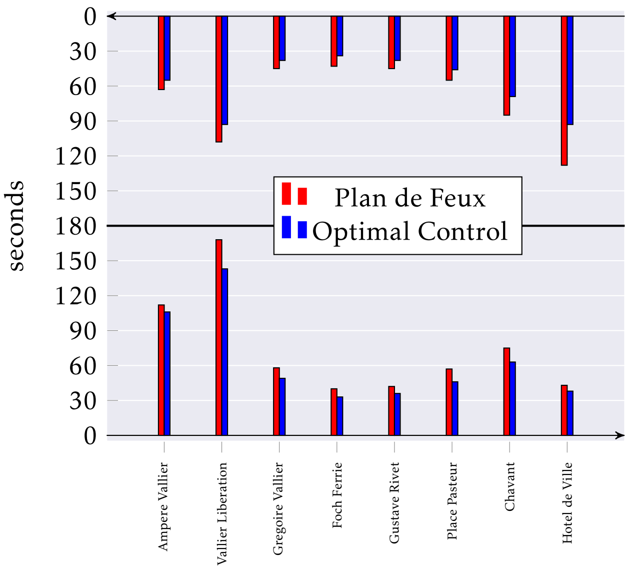


Figure 8.6 – Waiting time per intersection in the Grand Boulevard scenario, in the two main directions. The reported value is the average waiting time, for each intersection, of vehicles traveling east-west (top of the figure) and west-east (bottom of the figure). The numerical values are reported in table 8.3.

Table 8.3 – Waiting time table for figure 8.6.

Intersection	Waiting time [sec]			
	west-east		east-west	
	PdF	Opt.	PdF.	Opt.
Ampere Vallier	112	106	63	55
Vallier Liberation	168	143	108	93
Gregoire Vallier	58	49	45	38
Foch Ferrie	40	33	43	34
Gustave Rivet	42	36	45	38
Place Pasteur	57	46	55	46
Chavant	75	63	85	69
Hotel de Ville	43	38	128	93

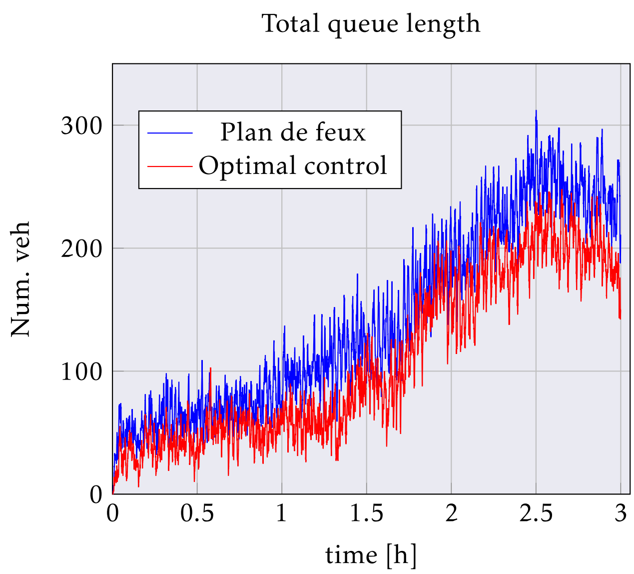


Figure 8.7 – Queue length in the Grand Boulevard scenario. The figure shows the cumulated queue length (across all intersections) during the simulated rush hours, from 7a.m. to 10a.m., and compares the plan de feux performance with the optimization–based control.

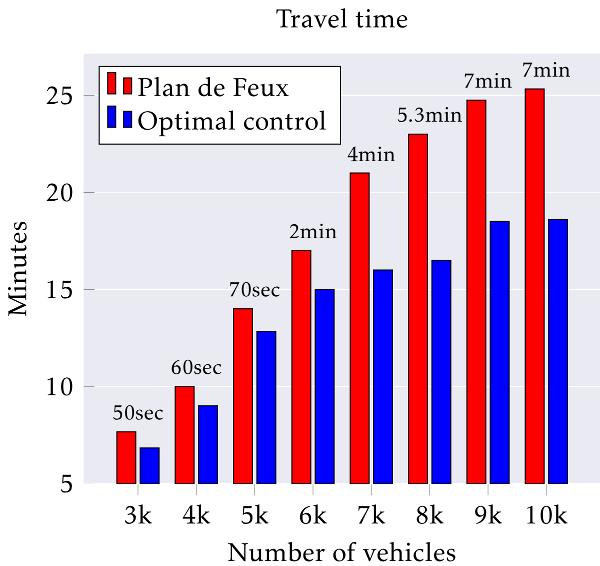


Figure 8.8 – Travel time with different input demand in the Grand Boulevard scenario. The scenario is simulated with eight different values of external demand, from 3k veh/h to 10k veh/h. For each of these values, the average travel time from the left border to right border (and viceversa) is measured in two different simulation, one where the intersections are controlled with the plan de feux, and a second one where the proposed control is acting. The improvement in terms of saved time is marked on the top of the bars, for every simulation.

8.5 Final comments on the chapter

In this chapter we have presented the study of an industrial traffic lights controller and compared against it the optimization–based approached that we developed, in a realistic scenario.

We were motivated by the need for any theoretical study of traffic lights control, like the ones proposed in the previous chapters, to be evaluated in a specific scenario and compared against another controller that has been designed for it too. The studied plan de feux fits perfectly this need, as it has been designed specifically for the Grand Boulevard scenario in Grenoble.

We first presented in detail the working principle of the plan de feux, via a reverse engineering approach, and discussed its main technological and methodological features. We analyzed the differences, in both regards, with the optimization approaches we proposed, and illustrated how the one step ahead traffic control can be adapted, in order to fulfill the scenario constraints and to be fairly compared with the industrial controller.

We did numerous tests in a microsimulator calibrated as to reproduce very closely the analyzed scenario, and that reproduces the plan de feux in all its features. We used the same microsimulated scenario to deploy the optimal control and to compare the two approaches. Our results show that the optimal control gives benefits by means of all traffic indexes. The realism of the scenario, combined with the fully reproduced plan de feux, allows us to claim that our approach could have a real positive impact if deployed into the town.

Conclusions and future perspectives

In this thesis we addressed the problem of traffic lights control in large scale urban networks. We proposed solutions that differ each other from methodological, computational and operational point of view. The proposed control approaches belong to the set of infrastructure controls and therefore their deployment in the field has to be done at the same level: the infrastructure must be equipped with appropriate hardware and software.

From hardware point of view, all proposed strategies rely on measurements of vehicles density; therefore such data has to be given as input to the algorithm. This definetely is not a limitation in terms of technology: for example, loop detectors installed at the beginning and at the end of a road are sufficient to measure the number of vehicles inside the same road, and hence the density; a camera and an appropriate image-processing system could easily serve at the same scope; floating car data from various sources (e.g., GPS) may be used too. However, some additional costs have to be considered, for example for the maintenance of the chosen detectors.

Among our propositions there are centralized, distributed and isolated algorithms. Depending on the chosen solution, the software infrastructure will have to be determined. In case of a distributed algorithms, traffic lights need to be equipped with a small processing unit, enabled to communicate within the neighbors distance and to receive density data from the neighborhood. In case of a centralized solution, a single, much more powerful, computer should be available and enabled to receive density data from all over the network as well as to send the decided schedule to traffic lights. Finally, if an isolated controller is selected, every traffic lights needs to have a small processing unit but there is no need to communicate with the others, and only local density values are necessary.

Review of the contributions

The thesis started by presenting a detailed study of traffic modeling techniques. In particular, starting from the well known Cell Transmission model, we designed two variations of the same model that include traffic lights. Both models are tailored towards control applications. We presented an extensive numerical analysis that includes validation of the models with respect to a classical binary representation of the traffic lights, and also with respect to the results of a microsimulated traffic system (via the commercial software Aimsun).

The first of the proposed control techniques is a centralized algorithm that decides percentage of green times, over a fixed cycle length, for all traffic lights in the network. The decision is made in order to optimize a cost function designed as a linear combination of standard traffic indexes and based on a short-term prediction of the system state. Such predictions are computed using the average model that describes a smoother traffic dynamics; thus, the short-term prediction offers a reasonable approximation of the future system evolution. We prove that this optimization problem is equivalent to a convex optimization, and it can therefore be efficiently solved with known gradient–based techniques. We tested this controller in a microsimulated environment that replicates a large portion of Grenoble, France, and compare its performance against a best practice approach. Several tests have shown that traffic indexes such as travelled distance, travel time, queues length and stop time are improved of about 13%, 15%, 10% and 15%, respectively.

The second of the designed controllers is a centralized algorithm that decides traffic lights schedule by means of activation and deactivation time instants of the green phases. In this case too, the schedule is decided in order to optimize a cost function based on densities prediction. However, this set–up allows a better expressivity in predicting traffic dynamics and therefore results in a better performance from traffic point of view, on one side, and in a much higher cost from computational point of view, on the other. In particular, simulative tests show that macroscopic traffic indexes are improved by more than 10% with respect to our first approach, but the optimization task is very heavy and the average computation time of a single run is about 3 minutes, even for a small network.

A further contribution of this work has been the development of distributed algorithms, based on the two centralized schemes aforesaid. We were motivated by the several benefits that a distributed control can lead to, mainly in terms of scalability of the approach to large scale traffic networks. Regarding the first of the proposed centralized controllers, we used a distributed optimization technique known as dual descent in order to design an iterative distributed algorithm that is optimal: in fact, we proved that the solution of the iterative scheme converges to the solution of the centralized optimization. The key points of this distributed control are its numerical efficiency and the lesser number of communications. As for the second of the proposed centralized controllers, we designed a distributed variant that improves greatly the computational efficiency, although it is suboptimal in terms of traffic performances.

Our last contribution lies in an in–depth study of the industrial traffic lights controller that is used in several important junctions in Grenoble. We have detailed the working principle of such a regulator, that can serve as an important benchmark for evaluation of the other strategies. Indeed, we also showed as the first of our proposed approach can be modified to become a single intersection controller, hence fulfilling the technological constraints imposed by the scenario, and compared against the industrial solution. Our tests, once more in a microsimulated environment, show that the scenario benefits better from the optimization–based approach rather than the from the industrial controller.

Future perspectives

The first part of our contributions dealt with centralized control of traffic lights. We first explored the possibility of a lightweight optimization, whose efficiency is mainly given by the self–imposed limitation in the prediction horizon. Then, we studied how to extend this approach for a longer horizon, but this implied to complicate the traffic model and lead to much more costly operations. An interesting direction for future works is about searching for model reductions in the spirit of the average CTM that nevertheless allow a multi step ahead predictions without compromising the efficiency. However, we believe that some model assumption has to be made in this respect. For instance, the controller can be designed in order to keep the system in free flow and, retroactively assuming that this always holds true, predictions can be carried out only for the free flow state space (which is described by a linear dynamics, according to the CTM). Of course this assumption does not come without costs: the most realistic scenario where it could actually be fulfilled is a network where traffic lights are installed also at the boundary, and the external demand is completely stopped if some internal road is about to become congested; this might cause long queues just outside the network.

In a second part of the thesis we proposed distributed techniques that are based on the centralized controls earlier introduced. We designed an optimal algorithm based on the average CTM and a suboptimal one based on the 2DoF-CTM. Interestingly enough, the suboptimality of the latter is someway mitigated by the better accuracy of the used model. As a results, the two distributed schemes have similar performances from traffic point of view. This means that if a better heuristic can be found for the Dist-MSA-TC then also better traffic performances can be obtained. We believe this would be an interesting direction for future works.

The last part of our research was focused on the study of an industrial traffic lights controller and on comparing its performance with an adapted version of the first among our propositions. Our simulative tests show that the latter brings concrete benefits to the scenario. Therefore, we believe that an additional work of engineering in order to put in production the algorithm could have a real impact and bring benefits to the city of Grenoble.

Bibliography

[1] K. Aboudolas and N. Geroliminis. “Perimeter and boundary flow control in multi-reservoir heterogeneous networks”. *Transportation Research Part B: Methodological* (2013).

[2] K. Aboudolas, M. Papageorgiou, and E. Kosmatopoulos. “Store-and-forward based methods for the signal control problem in large-scale congested urban road networks”. *Transportation Research Part C: Emerging Technologies* (2009).

[3] K. Aboudolas et al. “A rolling-horizon quadratic-programming approach to the signal control problem in large-scale congested urban road networks”. *Transportation Research Part C: Emerging Technologies* (2010).

[4] R.E Allsop. “SIGCAP: A computer program for assessing the traffic capacity of signal-controlled road junctions”. *Traffic Engineering & Control* (1976).

[5] R.E. Allsop. “SIGSET: a computer program for calculating traffic signal settings”. *Traffic Engineering & Control* (1971).

[6] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1.* 2015. URL: *http://docs.mosek.com/7.1/toolbox/index.html*.

[7] J. Archer et al. *The impact of lowered speed limits in urban/metropolitan areas*. 2008.

[8] E.D. Arnold. “Ramp metering: A review of the literature”. *Virginia Transportation Reasearch Council* (1998).

[9] A. Auer, S. Feede, and S. Lockwood. *History of Intelligent Transportation Systems*. 2016.

[10] A. Barisone et al. “A macroscopic traffic model for real-time optimization of signalized urban areas”. *41st IEEE Conference on Decision and Control* (2002).

[11] A. Bell. “If you always caught the green light”. *ECOS Magazine* (1984).

[12] A. Bemporad and M. Morari. “Control of systems integrating logic, dynamics, and constraints”. *Automatica* (1999).

[13] F. Boillot et al. "Optimal Signal Control of Urban Traffic Networks". *Road Traffic Monitoring* (1992).

[14] P. Borokhov et al. "An adaptive routing system for location-aware mobile devices on the road network". *IEEE Conference on Intelligent Transportation Systems (ITSC)* (2011).

[15] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge Press University, 2004.

[16] M. Burger et al. "A distributed simplex algorithm for degenerate linear programs and multi-agent assignments". *Automatica* (2012).

[17] C. Caligaris, S. Sacone, and S. Siri. "Model predictive control for multiclass freeway traffic". *European Control Conference* (2009).

[18] E. Camponogara, H.F. Scherer, and L.V. Moura. "Distributed optimization for predictive control with input and state constraints: Preliminary theory and application to urban traffic control". *IEEE International Conference on Systems, Man and Cybernetics* (2009).

[19] C. Canudas De Wit et al. "Grenoble traffic lab: An experimental platform for advanced traffic monitoring and forecasting". *IEEE Control Systems* (2015).

[20] R. Carli and G. Notarstefano. "Distributed partition-based optimization via dual decomposition". *52nd IEEE Conference on Decision and Control* (2013).

[21] E. Chang et al. "MAXBAND-86: Program for optimizing left-turn phase sequence in multiarterial closed networks". *Transportation Research Record* (1988).

[22] G. Como, E. Lovisari, and K. Savla. "Convexity and Robustness of Dynamic Network Traffic Assignment for Control of Freeway Networks". *Transportation Research Part B: Methodological* (2016).

[23] G. Como et al. "Robust distributed routing in dynamical networks – Part I: Locally responsive policies and weak resilience". *IEEE Transactions on Automatic Control* (2013).

[24] G. Como et al. "Robust distributed routing in dynamical networks – Part II: Strong resilience, equilibrium selection and cascaded failures". *IEEE Transactions on Automatic Control* (2013).

[25] C. Daganzo. "The Cell Transmission Model: A dynamic representation of highway traffi consistent with the hydrodynamic theory". *Transportation Research Part B* (1994).

[26] C. Daganzo. "The Cell Transmission Model, Part II: Network traffic". *Transportation Research Part B* (1995).

[27] G. De Nunzio. "Traffic eco–management in urban traffic networks". PhD thesis. University of Grenoble Alpes, 2015.

[28] G. De Nunzio, C. Canudas de Wit, and P. Moulin. "Urban Traffic Eco-driving: a macroscopic steady-state analysis". *13th IEEE European Control Conference* (2014).

[29] C. Diakaki. "Integrated control of traffic flow in corridor networks". PhD thesis. Technical University of Crete, 1999.

[30] E. Dijkstra. "A note on two problems on connection with graphs". *Numerishe Mathematik* (1959).

[31] V. Dinopoulou, C. Diakaki, and M. Papageorgiou. "Applications of the urban traffic control strategy TUC". *European Journal of Operational Research* (2006).

[32] M. Dotoli, M. Fanti, and C. Meloni. "A signal timing plan formulation for urban traffic control". *Control engineering practice* (2006).

[33] A. Ferrara, S. Sacone, and S. Siri. "Distributed Model Predictive Control for MLD systems: application to freeway ramp metering". *IEEE American Control Conference* (2014).

[34] A. Ferrara, S. Sacone, and S. Siri. "Event-triggered model predictive schemes for freeway traffic control". *Transportation Research Part C: Emerging Technologies* (2015).

[35] J.R.D. Frejo et al. "Macroscopic Modeling and Control of Reversible Lanes on Freeways". *IEEE Transactions on Intelligent Transportation Systems* (2016).

[36] N. Gartner. "OPAC: A demand-responsive strategy for traffic signal control". *Transportation Research Record* (1983).

[37] N. Gartner et al. "A multi-band approach to arterial traffic signal optimization". *Transportation Research Part B: Methodological* (1991).

[38] D.C. Gazis and R.B. Potts. "The oversaturated intersection". *2nd International symposium on the theory of traffic flow* (1963).

[39] N. Geroliminis, J. Haddad, and M. Ramezani. "Optimal perimeter control for two urban regions with macroscopic fundamental diagrams: A model predictive approach". *IEEE Transactions on Intelligent Transportation Systems* (2013).

[40] C. Gershenson. "Self-organizing traffic lights" (2004).

[41] C. Gershenson and D.A. Rosenblueth. "Self-organizing traffic lights at multiple-street intersections". *Complexity* (2012).

[42] *Google Maps*. URL: *www.google.com/maps/*.

[43] M. Gordon. *Traffic devices: historical aspects thereof*. Institute of Traffic Engineers, 1971.

[44] P. Grandinetti, C. Canudas de Wit, and F. Garin. “An efficient one-step-ahead optimal control for urban signalized traffic networks based on an averaged Cell-Transmission model”. *14th European Control Conference* (2015).

[45] P. Grandinetti, C. Canudas de Wit, and F. Garin. “Distributed optimal traffic lights design for large-scale urban networks” (2017). Submitted to the IEEE Transaction on Control Systems Technology.

[46] P. Grandinetti, F. Garin, and C. Canudas de Wit. “Towards scalable optimal traffic control”. *54th IEEE Conference on Decision and Control* (2015).

[47] J. Gregoire et al. “Back-pressure traffic signal control with unknown routing rates”. *19th IFAC World Congress* (2014).

[48] *Grenoble Traffic Lab – Aimsun Demo*. *http://gtl.inrialpes.fr/aimsun-control-demo-accident/index.html*.

[49] Gurobi Optimization, Inc. *Gurobi Optimizer Reference Manual*. 2015. URL: *http://www.gurobi.com*.

[50] J. Haddad and N. Geroliminis. “On the stability of traffic perimeter control in two-region urban cities”. *Transportation Research Part B: Methodological* (2012).

[51] J. Haddad et al. “Optimal steady-state control for isolated traffic intersections”. *IEEE Transactions on automatic control* (2010).

[52] K. Han et al. “On the continuum approximation of the on-and-off signal control on dynamic traffic networks”. *Transportation Research Part B: Methodological* (2014).

[53] J. Henry, J. Farges, and J. Tuffal. “The PRODYN real time traffic algorithm”. *4th IFAC-IFIC-IFORS Conference on Control in Transportation Systems* (1984).

[54] *HERE. Maps for Life*. URL: *www.m.here.com*.

[55] P.B. Hunt et al. *SCOOT - A traffic responsive method of coordinating signals*. Tech. rep. 1981.

[56] G. Improta and G.E. Cantarella. “Control system design for an individual signalized junction”. *Transportation Research Part B: Methodological* (1984).

[57] The Mathworks Inc. “MATLAB Version: 8.5.0.197613 (R2015a)” (2015).

[58] I. Ioslovich et al. “Design of optimal traffic flow control at intersection with regard for queue length constraints”. *Automation and Remote Control* (2011).

[59] M. Isaac. “Uber Expands Self-Driving Car Service to San Francisco”. *The New York Times* (2016-12-14).

[60] T. Jiang and X. Liang. "Fuzzy self-adaptive PID controller for freeway ramp metering". *International Conference on Measuring Technology and Mechatronics Automation* (2009).

[61] I. Kaparias et al. "State-of-the-art of urban traffic management policies and technologies" (2010).

[62] H.K. Khalil. *Noninear Systems*. Prentice-Hall, 1996.

[63] S. Lammer and D. Helbing. "Self-control of traffic lights and vehicle flows in urban road networks". *Journal of Statistical Mechanics: Theory and Experiment* (2008).

[64] S. Lammer and D. Helbing. "Self-stabilizing decentralized signal control of realistic, saturated network traffic". *Saturated Network Traffic* (2010).

[65] J.P. Lebacque. "The Godunov scheme and what it means for first order traffic flow models". *Internaional symposium on transportation and traffic theory* (1996).

[66] X. Liang and Z. Li. "Freeway ramp control based on genetic PI and fuzzy logic". *IEEE Conference on Computational Intelligence and Industrial Application* (2008).

[67] M.J. Lighthill and G.B. Whitham. "On Kinematic waves. II. A theory of traffic flow on long crowded roads". *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* (1955).

[68] S. Lin et al. "Efficient network-wide model-based predictive control for urban traffic networks". *Transportation Research Part C: Emerging Technologies* (2012).

[69] S. Lin et al. "Integrated urban traffic control for the reduction of travel delays and emissions". *IEEE Transactions on Intelligent Transportation Systems* (2013).

[70] J. Little. "The synchronization of traffic signals by mixed-integer linear programming". *Operations Research* (1966).

[71] J. Little, M. Kelson, and N. Gartner. "MAXBAND. A versatile program for setting signals on arteries and triangular networks". *Transportation Research Record: Journal of the Transportation Research Board* (1981).

[72] J. Löfberg. "YALMIP : A Toolbox for Modeling and Optimization in MATLAB". *IEEE Computer Aided Control System Design* (2004).

[73] E. Lovisari, G. Como, and K. Savla. "Stability of monotone dynamical flow networks". *53rd IEEE Conference on Decision and Control* (2014).

[74] P.R. Lowrie. “The Sydney coordinated adaptive traffic system: Principles, methodology, algorithms”. *International Conference on Road Traffic Signalling* (1982).

[75] V. Mauro and C. Di Taranto. “UTOPIA”. *IFAC Control, Computers, Communications in Transportations* (1989).

[76] D. Merchant and G. Nemhauser. “A model and an algorithm for the dynamic traffic assignment problems”. *Transportation science* (1978).

[77] D. Merchant and G. Nemhauser. “Optimality conditions for a dynamic traffic assignment model”. *Transportation Science* (1978).

[78] A. Muralidharan and R. Horowitz. “Optimal control of freeway networks based on the link node cell transmission model”. *IEEE American Control Conference* (2012).

[79] Y. Nie and X. Wu. “Shortest path problem considering on-time arrival probability”. *Transportation Research Part B: Methodological* (2009).

[80] *Open Street Maps*. URL: *www.openstreetmap.org*.

[81] M. Papageorgiou and F. Hadj-Salem H.and Middelham. “ALINEA local ramp metering: Summary of field results”. *Transportation Research Record* (1997).

[82] M. Papageorgiou, H. Hadj-Salem, and J. Blosseville. “ALINEA: A local feedback control law for on-ramp metering”. *Transportation Research Record* (1991).

[83] M. Papageorgiou et al. “Application of the signal control strategy TUC in three traffic networks: Comparative evaluation results”. *IEEE International Conference on Information and Communication Technologies* (2006).

[84] M. Papageorgiou et al. “Review of road traffic control strategies”. *Proceedings of the IEEE* (2003).

[85] I. Papamichail et al. “Heuristic ramp-metering coordination strategy implemented at monash freeway, australia”. *Transportation Research Record* (2010).

[86] C. Pasquale et al. “Two-class freeway traffic regulation to reduce congestion and emissions via nonlinear optimal control”. *Transportation Research Part C: Emerging Technologies* (2015).

[87] S. Peeta and A. Ziliaskopoulos. “Foundations of dynamic traffic assignment: The past, the present and the future”. *Networks and Spatial Economics* (2001).

[88] D. Pisarski. “Collaborative ramp metering control: application to Grenoble south ring”. PhD thesis. University of Grenoble Alpes, 2015.

[89] D. Pisarski and C. Canudas de Wit. “Optimal balancing of road traffic density distributions for the cell transmission model”. *IEEE 51st Conference on Decision and Control* (2012).

[90] P.I. Richards. “Shock waves on the highway”. *Operation research* (1956).

[91] D. Richert and J. Cortes. “Robust distributed linear programming” (2014). Available at *http://arxiv.org/pdf/1409.7140v1.pdf*.

[92] D.I. Robertson. “Research on the TRANSYT and SCOOT Methods of Signal Coordination”. *ITE Journal* (1986).

[93] D.I. Robertson. “TRANSYT: a traffic network study tool”. *Road Research Laboratory Report* (1969).

[94] S. Samaranayake, S. Blandin, and A. Bayen. “A tractable class of algorithms for reliable routing in stochastic networks”. *Transportation Research (Part C)* (2012).

[95] S. Samaranayake, S. Blandin, and A. Bayen. “Speedup Techniques for the Stochastic on-time Arrival Problem”. *12th Workshop on Algorithmic Approaches for Transportation Modelling, Optimization, and Systems* (2012).

[96] D. Schrank, B. Eisele, and T. Lomax. “TTI’s 2012 urban mobility report”. *Texas A&M Transportation Institute. The Texas A&M University System* (2012).

[97] S.C. Schwartz and H.H. Tan. *Integrated control of freeway entrance ramps by threshold regulation*. Tech. rep. 1977.

[98] *SCOOT’s website*. URL: *http://www.scoot-utc.com/*.

[99] P. Seewald et al. “User acceptance evaluation approach of energy efficient driver assistance systems”. *9th ITS European Congress* (2013).

[100] *Self driving cars - Google*. URL: *https://waymo.com/*.

[101] S. Sen and K.L. Head. “Controlled optimization of phases at an intersection”. *Transportation science* (1997).

[102] C. Stamatiadis and N. Gartner. “MULTIBAND-96: A program for variable-bandwidth progression optimization of multiarterial traffic networks”. *Transportation Research Record: Journal of the Transportation Research Board* (1996).

[103] TESLA. “All Tesla Cars Being Produced Now Have Full Self-Driving Hardware” (2016-10-14). URL: *www.tesla.com/blog/all-tesla-cars-being-produced-now-have-full-self-driving-hardware*.

[104] *The T language*. *https://en.wikipedia.org/wiki/T_(programming_language)*.

[105] S. Timotheou, C. Panayiotou, and M. Polycarpou. “Distributed Traffic Signal Control Using the Cell Transmission Model via the Alternating Direction Method of Multipliers”. *IEEE Trans. Intelligent Transportation Systems* (2015).

[106] *TOMTOM*. URL: www.tomtom.com.

[107] S. Trommer and A. Holtl. “Perceived usefulness of ECO–driving assistance systems in Europe”. *IET Intelligent Transport Systems* (2012).

[108] TSS - Transportation Simulation Systems. *Aimsun 8 Dynamic Simulators Users' Manual*. 2015.

[109] J. Van Mierlo, E. Van de Burgwal, and R. Gense. “Driving style and traffic measures influence on vehicle emissions and fuel consumption”. *Journal of Automobile Engineering, Part D* (2004).

[110] P. Varaiya. “Max pressure control of a network of signalized intersections”. *Transportation Research Part C: Emerging Technologies* (2013).

[111] P. Varaiya. “The max-pressure controller for arbitrary networks of signalized intersections”. *Advances in Dynamic Network Modeling in Complex Transportation Systems*. Springer, 2013.

[112] M. Wallström. *Reclaiming city streets for people: chaos or quality of life*. Ed. by European Commission Directorate-General for the Environment. 2007.

[113] C.F. Wang. “On a ramp-flow assignment problem”. *Transportation Science* (1972).

[114] J.J. Wang and A. May. “Computer model for optimal freeway on-ramp control”. *Highway Research Record* (1973).

[115] J. A. Wattleworth. “Peak period analysis and control of a freeway system”. *Highway Research Record* (1967).

[116] *Waze*. URL: www.waze.com.

[117] Wikipedia. *Traffic light history*. URL: wikipedia.org/wiki/Traffic_light.

[118] H.P. Williams. *Model building in mathematical programming*. John Wiley & Sons, 2013.

[119] T. Wongpiromsarn et al. “Distributed traffic signal control for maximum network throughput”. *15th International IEEE Conference on Intelligent Transportation Systems* (2012).

[120] M. Zhang et al. “Evaluation of on-ramp control algorithms”. *Institute of Transportation Studies, California PATH report* (2001).

[121] Y. Zhang, R. Su, and K. Gao. “Urban road traffic light real-time scheduling”. *54th IEEE Conference on Decision and Control* (2015).