PhD-FSTM-2022-017

The Faculty of Science, Technology and Medicine

DISSERTATION

Defence held on 25/02/2022 in Esch-sur-Alzette

to obtain the degree of

DOCTEUR DE L'UNIVERSITÉ DU LUXEMBOURG

EN SCIENCES DE L'INGÉNIEUR

by

Xavier MAZUR

Born on 03 March 1994 in Thionville (France)

CONTROLLABILITY OF COMPLEX FLOW NETWORKS

Dissertation defence committee

Prof. Dr Francesco Viti, Dissertation supervisor
Associate Professor, University of Luxembourg

Prof. Dr Marco Rinaldi
Assistant professor, Delft University of Technology

Prof. Dr Francesco Corman
Professor, ETH Zürich

Dr Richard D. Connors, Vice-chairman
Research scientist, University of Luxembourg, University of Leeds

Prof. Dr Thomas Engel, Chairman
Professor, University of Luxembourg

Acknowledgments

It is still hard to believe that I arrived at this moment, writing the last words of my PhD thesis and how fast it arrived. These four years were full of new exciting experiences and passed faster than expected. Even if this PhD was full of ups and downs, it was an exciting and interesting experience nonetheless, in which I learned a lot about research, transportation, and also myself. The last thing I would like to do is to thank all the wonderful peoples I met during this journey.

First and foremost, I want to express my gratitude to my supervisors, Francesco Viti and Marco Rinaldi. Thank you for the guidance you both provided during these years, for helping me all along, even though it was not always an easy task, all of this was made possible thanks to you. Next, I would like to thank Francesco Corman and Richard Connors, who always provided me with plentiful of valuable advice and feedback. I am particularly grateful to all members of the Mobilab group that quickly became more than colleagues to me. I would like to thank Georgios, who welcomed me warmly in the group and was always there to help me out, and that started the long list of multilingual armchairs on my whiteboard. Then, I want to thank Giulio, who was always helpful and shared a lot of music bands. I would like to give special thanks to Ariane, that started her PhD almost at the same time as me, and with who I am more than grateful I shared an office for all this time, and with who I love to chat, maybe too much sometimes. Next, I want to thank Claudia for sharing so many good laughs and Piergiorgio, who is always so cheerful. Thanks also to Carolina for bringing music to the office.

I would also like to express my gratitude to the more recent members who integrated me into their group and with whom I shared a lot of fun moments. Thank Federico for putting your life at risk with all these smoking breaks that helped me think and all these things you organized (special thanks for Florence). Thanks to Nicola and Yumeng for sharing all these pleasant moments and laughs.

I also want to thank my friends from Metz, Sarah, Etienne, and Pierre, who, even if we saw each other less than before, were always supportive and with who I shared happy moments.

Again, thank you everyone, for sharing this important chapter of my life. I am grateful I met all of you.

Merci beaucoup à vous tous !

Xavier Mazur

Index

List of Figures

List of Tables

Summary

When left uncontrolled, complex flow networks are susceptible to negative externalities and tend not to be used to their full potential. This work focuses more specifically on the specific instance of transportation networks that are subject to constantly increasing demand. Control strategies, based on increasingly promising technological advancements, have been developed with the aim to exploit the full potential of the existing transportation infrastructure. To improve the current state of transportation networks, control strategies rely on control technologies to impact road users on networks and redirect them, such as to improve the situation by avoiding delays, for example. However, the problem of identifying the required controller numbers, types, and locations has received little attention in the current literature. Existing research works proposed approaches to the problem but often either do not provide complete control over the considered network or lack scalability, thus are not applicable on any type and size of networks.

In this dissertation, we aim at filling this gap by providing a general methodology and proposing various approaches to this problem. The first part of this work focuses specifically on studying the problem of fully controlling a transportation network and provides various approaches. Their capacity to actually impact and control transportation networks is assessed empirically, showing that the proposed approaches can fully control small networks.

The second part studies the problem of scalability and provides a new method that is proved to be able to provide an efficient set of pricing controllers while being scalable. This approach is later improved by integrating flow information and demonstrated to be more reliable in the specific case where the demand is irregularly distributed over the considered network, which is a common setting in real transportation networks. Additionally, the proposed methods are applied and tested over the network of Luxembourg, demonstrating the scalability of the approaches and their capacity to improve the current state of large realistic networks subject to heavy congestion.

Chapter 1

Introduction

Complex flow networks are structures used for transporting flows of some elements, such as water, electricity, or vehicles. Such networks are present all around us and are employed to provide numerous commodities in our daily lives. They represent an essential part of our technical infrastructures, such as the electrical power grid and transportation networks. To perform at their full potential, such networks often rely on control to redirect flows, such as to achieve the desired objective. More specifically, this dissertation focuses on transportation networks that constantly witness a rise in demand due to a continually growing urbanization trend and a concurrent increase of the urban sprawl phenomenon, which contributes to transforming our cities into large and complex systems. This transformation challenges the current transport infrastructure capacities, resulting in lost times, reduced service reliability, and stress for road users. However, before resorting to capacity expansions, which may be beneficial in the short term but bring in the long term many negative impacts (more car trips due to the induced demand phenomenon and to mode shifts from more sustainable modes of transport, increased pollution, more car accidents, increased car ownership rates,...) the existing infrastructure is not always used to its full potential, which produces additional negative impacts on road users, such as delays due to, for instance, inefficient handling of the conflicting flows at intersections or congestion at bottlenecks.

Transportation systems are composed of various modes of transportation; however, in this dissertation, we focus solely on road networks used by vehicles (cars, trucks,...) and the corresponding traffic flows. Table 1.1 shows the extent of the heavy level of congestion in Germany, for example. It displays the average amount of hours lost in congestion per road user over a year and the corresponding cost for this lost time. We can observe that these numbers are very high, indicating that a significant amount of time and money is unnecessarily lost in traffic. Congestion will provoke slower traffic and can produce a stop-and-go effect (Yeo and Skabardonis [50]) that will result in additional negative externalities, such as increased pollutant emissions or a loss in productivity that could be reduced with better network management.

In order to mitigate these negative effects and efficiently use the existing network infrastructure, various control strategies have been developed in the literature with diverse approaches to control transportation networks. In this dissertation, we focus on control strategies that aim at controlling the flows of road users as opposed to strategies that aim at controlling the demand. For instance, some strategies focus on local control policies, such as managing traffic lights over one single intersection to reduce the congestion in the considered area. In contrast, other research works attempt to coordinate controllers over multiple locations to achieve a common goal, such as generating green wave effects (Hunt et al. [18]; Lowrie [24]; de Oliveira and Camponogara [29]; Hoogendoorn et al. [16]). Additionally, some control strategies are developed to control a particular portion of a transportation network that possesses some specific characteristics that require adapted control actions, such as for a section of a highway (Papageorgiou et al. [32]) or a traffic bottleneck (Gonzales and Christofa [12]). However, in this thesis, we are mainly interested in control strategies that consider the whole network at once instead of a portion. Such network-wide control strategies attempt to achieve a global objective, such as minimizing the total travel time of road users through the combined actions of every individual controller on a network. Previous research works explored the possibility of employing controllers to form one or multiple pricing cordons to divide the considered network into multiple smaller areas, easier to control (Zhang and Yang [52]), thus focusing more on a local strategy. Other control strategies attempted to coordinate toll gantries and pricing levels over a network to fully control the entire network (Verhoef [42]).

Various types of controller technologies can be employed, resulting in distinct ways to impact network traffic, aiming at modifying drivers' route choices, for example. As such, traffic lights are a commonly employed type of controller, often used to improve safety for road users on intersections by adding constraints to the natural behavior of the users. However, they can also be employed to reduce the congestion in the considered area by managing conflicting flows through appropriately redistributing the total available capacity over time for the considered intersection. Pricing controllers are another commonly employed type of controller; their functioning differs from traffic lights as they impose a monetary cost to the location they are placed on, thus making any path using this location less attractive. As demonstrated in previous work (Rinaldi [34]), the capability of control strategies to actually impact a network is strongly dependent on the chosen set of controllers. More precisely, the number, type, and location of controllers employed on the considered network affect the maximal reachable performance for control strategies.

Therefore, several research works aim to determine critical locations for controllers such that the resulting controller set can efficiently control the traffic across the entire network. Some of these approaches rely on flow information to identify such essential locations (Verhoef [42]). In contrast, others are solely based on topological information; that is, they use only information that is independent of the traffic flow, such as the shape of the network (Rinaldi et al. [37]; Mazur et al. [25]). However, existing approaches tend to possess heavy computational complexity. Therefore, these approaches are not easily scalable and applicable to large-scale networks commonly representing real-world transportation networks.

This dissertation aims to provide a methodological basis and multiple approaches capable of producing the needed set of controllers to efficiently control any size of transportation network while minimizing the installation cost. Before formalizing this aim into multiple research

City	Average hours lost in congestion	Cost of congestion per driver	Cost of congestion per city
Berlin	154	€ 1.340	€ 1.7B
München	140	€ 1.218	€ 618.5M
Hamburg	139	€ 1.212	€ 758.2M
Leipzig	108	€ 941	€ 184.6M
Stuttgart	108	€ 938	€ 204.8M
Nuremberg	107	€ 937	€ 167.2M
Frankfurt	107	€ 935	€ 239.7M
Dusseldorf	100	€ 874	€ 187.3M
Cologne	99	€ 867	€ 322.0M
Bremen	96	€ 839	€ 163.7M

Table 1.1: Hours spent in road congestion annually by the average driver in Germany, in 2018. The cost of congestion is calculated based on the average hourly wage per capita. (Source: INRIX Global Traffic Scorecard, February 2019)

objectives, we first introduce the background of this problem. For this purpose, section 1.1 presents the various characteristics of this problem. Section 1.2 discusses challenges as well as opportunities associated with this problem. Section 1.3 provides a description of the research scope and objectives of this dissertation. Section 1.4 details the contributions of this thesis.

1.1 Problem characteristics

As described previously, transportation networks require control to be used efficiently. However, network-wide traffic control strategies rely on controllers to actually impact the flow distribution to reach various objectives. This section first describes the problem of controlling transportation networks, followed by a description of the characteristics of the various types of controllers considered in this dissertation.

1.1.1 Control of transportation networks

To mitigate negative effects, such as delays, transportation networks rely on control systems. For instance, intersections with high demand are often regulated by traffic lights to constrain the natural behavior of road users, with the primary aim to improve safety in the considered intersection. However, by appropriately distributing the available capacity where needed, traffic lights can also reduce the emergence and propagation of congestion in the considered area. Similarly, controllers can be exploited to trigger a specific desired behavior, such as better routing choices, to reach some chosen objective. In this work, we are mainly interested in using controllers to reduce congestion; therefore, it is important to consider that congestion is often the result of the demand exceeding the existing supply in a certain portion of a transportation network. The excessive demand in a certain portion of transportation networks results from user behavior, such as departure time, route choice, or even mode choice, which controllers' actions can influence. However, control strategies that do not consider the whole network and focus only on reducing congestion locally can produce unexpected and undesired effects, which are usually neglected when setting the signals. For instance, using ramp metering to reduce the inflow of a highway may produce a spillback effect causing congestion on the underlying urban network (Rinaldi et al. [35]). To address this issue, we focus on control strategies that attempt to coordinate multiple individual controllers with the aim of reducing congestion over the entire network.

While numerous control strategies can be found in the literature, including a few that are conceived for area-wide or network-wide traffic management, an important design choice has received little attention over the past years: determining the amount, type, and location of controllers employed by a control strategy under consideration. This design choice is often overlooked and simply determined based on local needs instead of considering controller locations over the entire network. This can be detrimental to control strategies' performance as the extent to which they can steer the current flow distribution toward the desired one is constrained by the existing set of controllers used on the network. Therefore, to guarantee that maximal performances are reachable, a controller set should be determined such that the quantity, type, and location of controllers properly chosen to be capable of steering the flow distribution toward a desired state. Additionally, the economic aspect of choosing a controller set should also be considered to minimize maintenance and installation costs.

1.1.2 Types of controllers

Various types of controller technologies can be considered in a network-wide traffic control system, and each has its characteristics and differences that need to be considered. This work will focus mainly on two types of control tools, pricing controllers and traffic lights. Their influence on transportation networks and flow distribution will be discussed in this chapter.

Pricing controllers have been commonly implemented in real-world networks via toll gantries which are regularly located on highways. Recently, new forms of control pricing have been developed, such as using automatic number-plate recognition cameras for automatic payment, making pricing control over transportation networks more and more technologically feasible. This type of controller will directly influence the cost of passing through the location they are placed on by imposing a monetary cost or a subsidy to the road users passing through the considered location. Thus, by producing an increase or reduction in price to a location, this controller allows the control strategy to adjust the attractiveness of certain routes and, by extension, to redirect flows toward other routes. In the situation where a pricing controller is located on each and every link of a considered transportation network, we can apply a so-called first-best pricing scheme, which has been proven to be capable of reaching optimal performance on any transportation network, as demonstrated in previous research works (Verhoef et al. [43]).

Traffic lights are one of the most commonly used types of traffic control technologies. Usually, traffic lights are employed to manage the conflicting flows at an intersection by appropriately distributing the available green time. They are primarily used to enhance the safety of the users at an intersection by adding constraints to their natural behavior and reducing the potentially dangerous turning movements. They can also be employed to decrease congestion in the considered area by distributing green times considering the actual demand distributed over the intersection, and aiming to limit the queuing effects. Traffic lights can also be used outside of intersections to add a delay to the cost of passing through a certain location, and by doing so, reducing the attractiveness of this location for road users. Signals near pedestrian crossings, or soft speed control enforcement systems used in different European countries (the Netherlands, Sweden, Portugal) are examples of traffic lights placed within road links.

Other types of controllers exist, like variable speed limits or route advisory and guidance systems. However, we choose to focus only on pricing controllers and traffic lights for this work because they are more commonly found in real networks and for their simple functioning, which will provide a simple way to represent their impact over a network for experiments. Additionally, the impact of pricing controllers can be reasonably translated into other, softer approaches, such as variable speed limits. Nevertheless, the developed approaches can possibly be extended to consider any controller type.

1.2 Challenges and opportunities

Providing a set of controllers capable of fully controlling a transportation network while minimizing the number of controllers employed is not a straightforward task, specifically since the task's difficulty will increase with the size of the considered network. This section will describe the difficulties encountered while working on this problem and the existing opportunities to improve the approaches reviewed from the current literature.

Locating controllers on transportation networks is a complex problem. The first difficulty comes from the ability to assess if a proposed set of controllers is actually capable of fully controlling the underlying network. A transportation network is considered fully controllable when any feasible flow distribution is reachable through a set of actions from a set of controllers. However, the task of verifying if every feasible flow distribution is reachable based on a proposed set of controllers will result in a combinatorial explosion with the network size increase, therefore evaluating the capability of a controller set to actually fully control any given network is not a straightforward task. However, recent research works provided opportunities to face this difficulty. The most relevant innovation for this dissertation is the adaptation of control theory principles to the instance of transportation networks initiated by the recent work of Rinaldi [34]. The author adapted the principle of the controllability gramian matrix, introduced in the work of Kalman et al. [20], to the instance of transportation networks, providing a method to analytically and systematically compute the level of controllability of a network with a given set of controllers. However, the process required to compute the level of controllability exhibits a severe space complexity, bounded by $O(N^4)$; this implies that the amount of data that needs to be stored will increase rapidly with the size of the network. Thus, this process is hardly scalable and therefore cannot be applied on large networks.

Similarly, as for computing the level of controllability, the main difficulty in searching for controller locations lies in the network size. As such, searching for the number, locations, and types of controllers needed to efficiently control the underlying network in real-world urban networks represents an enormous number of possible combinations. Therefore, it is impossible to simply enumerate all combinations to identify which one performs better. In addition to the combinatorial complexity of the problem, demand is often not uniformly and regularly distributed over networks; travel behavior cannot be predicted with certainty and it varies both in time and space. Finally, demand and travel behaviour respond to control and to changes in traffic states. These are supplementary difficulties that make the problem of locating an efficient set of controllers a complex problem. Furthermore, once a suitable set of controllers is determined, coordinating such a high number of variables is also challenging, given the large number of correlated parameters to be optimised.

Some approaches have been developed in the past years to locate an efficient set of controllers on transportation networks (Verhoef [42], Rinaldi et al. [37]). However, existing methods tend to exhibit an important computational complexity, often exponential, which implies that the computation time required to compute a solution will increase exponentially with the considered network's size. Therefore, existing approaches are hardly scalable and cannot be easily applied on large networks, representing a gap in the literature we aim to fulfill with this work.

1.3 Research objective and scope

The main research objective of this dissertation is to develop a general methodology and several approaches to identify controller locations so that we can manage road traffic and reduce congestion in transportation networks. To work towards this objective, we aim at improving existing approaches used to control transportation networks. More specifically, we focus on providing a method capable of identifying the needed number, types, and locations of controllers to control the considered network, such that, through the action of the proposed controllers, we are capable of efficiently redirecting flows aiming at reducing congestion. Additionally, we also consider the scalability of the proposed approaches, such that we are capable of identifying the minimal set of controllers needed to fully control networks of any size. Therefore, in this dissertation, our goal is to develop a method capable of identifying controller locations that fulfill as much as possible the following desirable properties such as to provide the most suitable set of controllers possible:

A1 : The produced controller set should be capable of fully controlling the entire network; that is, through opportune actions of this controller set, all feasible flow solutions are attainable.

A2 : The controller set should contain the minimal number of controllers needed so that the corresponding installation cost is minimal.

A3 : The developed method should be scalable to be applicable to any type and size of networks.

Identifying a minimal controller set brings an additional desirable property, which however we do not fully demonstrate in this dissertation, i.e. by limiting the number of parameters to optimise, solution algorithms may more likely identify efficient solutions in less computation times, hence being more applicable in real-time control systems.

The problem of identifying efficient controller locations on transportation networks possesses many challenges that need to be addressed. To efficiently design such an approach, we decided to limit the scope of our research to the following elements. Firstly, we chose not to consider local control strategies, as they might improve only locally the situation based on myopic decisions that could provoke negative effects in areas of the network that are not considered. Consequently, we focus on control strategies that consider the entire network at once; thus, while searching for ideal control locations, we consider potential locations over the entire network. Additionally, as discussed previously, while various types of controllers exist, we chose to limit our scope to pricing controllers and traffic lights for this dissertation. As we aim to provide a scalable approach applicable to large networks, we propose to include in our scope the study of large-scale traffic networks, either corresponding to real city-sized networks or not.

1.4 Contributions

This dissertation proposes multiple contributions to the scientific literature. In this section, we will briefly describe each contribution presented in the following chapters.

An analysis of the impact of pricing controllers and traffic lights on transportation networks and their potential substitutability is provided in chapter 3. Special attention is paid to studying possible ways to represent the actual impact of these types of controllers. Experiments are carried out to assess the possibility of providing a solution based solely on traffic lights capable of reaching a similar control capability to a pricing-based solution.

To identify pricing controller locations, we propose to study characteristics and rules that favor efficient locations for pricing controllers in chapter 4. Based on this study, we developed a set of heuristics and compared their performances to propose a simple method to locate pricing controllers, particularly aiming to reach full controllability, thus focusing on the first aim previously described (A1).

Based on previous research works made for the similar problem of identifying sensor locations on transportation networks, we propose to adopt the well-known concept in operations research of minimum spanning trees to develop a novel approach to the problem of controller locations in chapter 5. Simulations based on static assignment show that the method is capable of identifying an efficient set of pricing controllers while being easily scalable, thus fulfilling the third aim (A3).

To further extend the research work of the previous chapter, chapter 6 will detail the introduction and exploitation of flow information into the spanning tree approach. The improved method, jointly leveraging topological and flow information, is shown to provide an even more efficient set of pricing controllers especially in the specific but very realistic case where the demand is irregularly distributed over the considered network.

1.5 Thesis outline

Figure 1.1 presents the outline of this thesis. This dissertation is divided into two parts. The first one contains approaches developed with the aim of reaching full controllability. As such, chapter 3 studies the substitutability of controllers while keeping full controllability and chapter 4 provides various heuristics that guarantee full controllability. The second part includes methods developed with the aim of providing a scalable approach. For this purpose, chapter 5 introduces the spanning tree approach, and chapter 6 extends this approach by adding flow information to the process. Finally, chapter 7 concludes this dissertation and discusses possible future research directions.

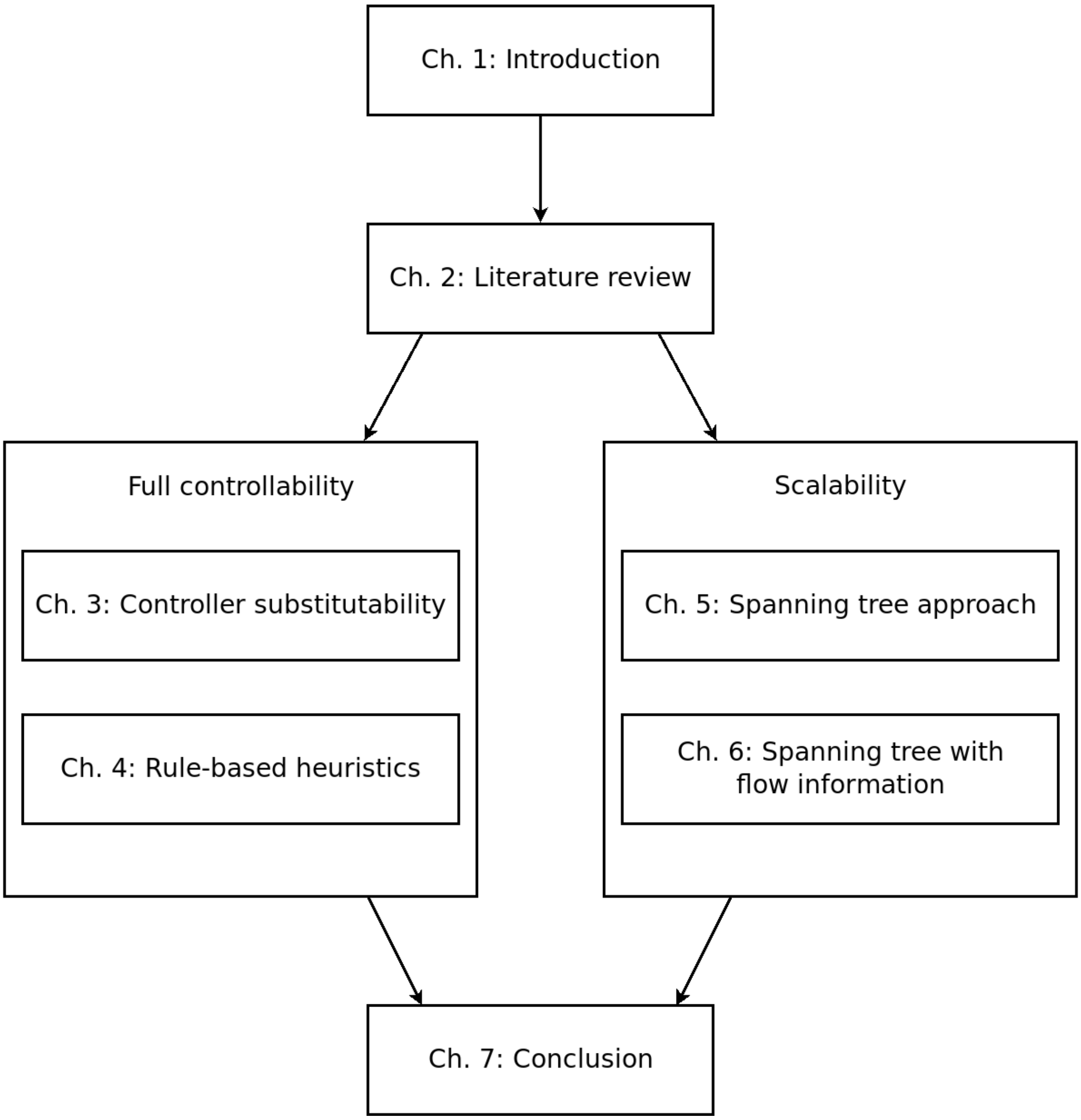


Figure 1.1: Dissertation outline.

Chapter 2

Literature review

As transportation networks rely on control strategies to minimize the severity of traffic externalities, previous research works aimed at developing approaches capable of locating a controller set to ensure that the underlying network could be fully controlled to efficiently reduce delays for road users anywhere they might arise. This section describes existing approaches used to locate an efficient set of controllers, identifying fundamental gaps in the current literature that might hinder application to real-life problems. However, the controller location problem has received comparatively little attention in the literature. Hence, we include approaches developed for sufficiently similar location problems on networks in the review process. We mainly focus on sensor location literature, as this problem can be seen as dual to that of controller location from a theoretical perspective. We aim to identify some approaches developed for the sensor location problem that could be applied or adapted to the controller location problem. This section provides a literature review of methodological approaches employed to locate controllers on transportation networks and methods used in similar location problems.

2.1 Controllability

2.1.1 Cordon pricing approaches

Existing research works consider the problem of controlling a transportation network from multiple perspectives. A common approach to locate controllers, more specifically pricing controllers, is to place them such that they form a pricing cordon that separates the network in multiple areas, thus controlling the entry of road users in each area. In the past years, diverse methods have been developed to solve this specific problem of designing an effective road pricing cordon. In their work, Mun et al. [27] proposed to focus on a specific instance of the problem by considering only monocentric cities. In their approach, the optimal cordon is determined such that the combination of the cordon location and toll values would maximize the resulting social surplus. The authors rely on numerous assumptions to formulate this social maximization objective. The most restrictive is that the considered network must be monocentric, with all trips destined to the central district. This assumption limits the approach's applicability to a precise network shape. Other research works aimed at designing pricing cordons for general networks, as in the work of Zhang and Yang [52], where the authors developed models and algorithms for the cordon-based second-best pricing scheme. They considered the determination of toll levels and locations simultaneously for various forms of cordons such as single-layered cordon, multi-layered cordon, or multi-centered cordon. Once the type of cordon is determined based on the characteristics of the studied city, the toll locations are determined using a genetic algorithm that will naturally select efficient toll locations for a cordon. Through numerical experiments, the authors showed that cordon-based pricing schemes produced by their approach are capable of achieving a significant increase in welfare gain. As many types of approaches exist to generate pricing cordons, Shepherd et al. [41] proposed to identify the most appropriate approach by comparing three methods, a judgmental approach, an optimization approach based on a genetic algorithm, and a short-cut approach which lies in between the two. Their research led to a genetic algorithm that can identify the theoretically best performing cordon for a specifically chosen objective for a given network. However, this approach suffers from scalability concerns due to its computation complexity. The same authors proposed an alternative approach that is a short-cut method that they showed capable of identifying a pricing cordon that can achieve a large portion of the benefits of an optimal cordon.

As travel demand tends to be irregularly distributed over time and space, traditional pricing cordons might not reach their full potential. To address this difficulty, Li et al. [22] presented a methodology that explicitly considers the propagation of traffic congestion over time, thereby adapting to changes in the network status. They showed that the proposed flexibly-located cordon could considerably reduce the amount of congestion over transportation networks while considering the congestion propagation over time and space. While approaches based on cordon pricing are very well suited for specific problems such as subdividing a network into multiple areas, or for some particular network shapes, like monocentric cities, most of the existing approaches aim at second-best solutions. In contrast, we aim to investigate and develop methods capable of fully controlling any given road network.

2.1.2 General pricing approaches

Aiming at a general approach, Verhoef [42] proposed to investigate the potential welfare gain from implementing a second-best pricing scheme over a network. To this purpose, the author developed an indicator to predict the welfare gain from implementing a second-best toll on a specific link of the considered network. Based on this indicator, one is capable of identifying the (single) toll location that has the highest impact over the whole network. In the case where we are considering the selection of multiple toll points, the author proposes three possible strategies. The most straightforward strategy to avoid a full combinatorial exploration is to select the n links that bear the highest predicted scores for implementing a single toll-point. However, this strategy does not consider interactions between multiple toll points; therefore, the selected controller set might contain redundant pricing controllers. A second proposed strategy involves a step-by-step approach, in which the optimal next toll-point is selected given the selection of previous toll-points and the corresponding second-best optimal toll levels previously computed. A drawback of this approach is that toll points that appear efficient at the beginning of the process may become less attractive when the total number of tolls increases. To fully account for the interaction between tolls, the last strategy consists of computing every possible combination of toll points in a network and selecting the combination that bears the highest predicated efficiency gain. However, this approach requires a large number of calculations that increase exponentially with the size of the considered network; thus, this approach is challenging to apply on large networks and is hardly scalable.

In order to reduce the computational complexity, some research works aimed at using solely topological information instead of relying on flow-based information. Topology-based approaches tend to require fewer prior inputs compared to previous methods as they only use the topology of the network, which generally reduces their computational complexity. As such, in his recent work, Rinaldi [34] proposed a general approach based on linear algebra to locate pricing controllers on any type of network. He proposed to adapt the work of Yuan et al. [51] to the instance of transportation networks resulting in an approach capable of determining a minimal set of pricing controller locations while ensuring the full controllability of the underlying transportation network. However, it was later demonstrated in Rinaldi et al. [37] that the proposed approach cannot guarantee that the controller set obtained through this method is actually capable of fully controlling the network under the presence of bi-directional links, which is a typical setting in realistic transportation networks.

2.2 Observability

As the problem of identifying controller locations has received little attention in the literature, we also review methodologies stemming from domains sharing similar characteristics, mainly from the field of observability, as it is a similar location problem over transportation networks that have been extensively studied. In this section, we focus primarily on the problem of sensor location for full traffic flow observability, which is the most similar to our problem of controller location. It consists in identifying important counting sensor locations over a transportation network, such as all the flows over the network are directly or indirectly observable. This problem is often seen as the dual of identifying controller locations. However, sensors are used to simply observe the network without impacting its current state, whereas controllers are employed to modify and steer the existing network's state toward the desired one. Therefore, the set of locations found for the observability problem is not guaranteed to be efficient for the problem of controllability. In what follows, we divide the literature concerning the domain of observability into three broad categories: the first focuses on approaches based on algebraic properties, the second includes methods that rely on topological information, such as network structure and connections, and the last consists of approaches based on optimization algorithms to resolve the location problem.

2.2.1 Linear algebra based approaches

Approaches based on linear algebra mostly rely on appropriate algebraic transformations to extract the minimum subset of links to be observed such that all the unmonitored flows can be estimated. Based on a given network structure, represented by a link-path incidence matrix, and considering static path flows, Hu et al. [17] used the principle of basis vector to define the set of basis links of a network. By definition, basis links are linearly independent links, meaning that all link flows can be expressed as a linear combination of the basis link flows. Therefore, if flows on basis links are observed using sensors, then all unmonitored link flows can be inferred based on the basis link flows. By computing the reduced row echelon form of the link-path incidence matrix, the authors are capable of identifying the basis links. Additionally, the set of basis links obtained is not unique; thus, the set of sensor locations is not unique either. However, such an approach requires a complete path enumeration to obtain the link-path matrix, a process that bears an exponential computation complexity and is therefore only applicable to small networks.

In order to address this issue, Ng [28] proposed to employ the node-link incidence matrix instead of relying on link-path information, thus avoiding the complete enumeration of paths. The node-link incidence matrix only represents which nodes and links are connected to one another. They demonstrated that the reduced row echelon form of the node-link incidence matrix could be subdivided into two sub-matrices, with one being invertible and another one not necessarily invertible. Based on these two sub-matrices, they showed that if links corresponding to the non-invertible matrix are observed, then the flows on links corresponding to the invertible matrix can be inferred. Therefore, the authors can identify a set of links to observe, such that all the link flows over the network can be observed directly or indirectly. However, Castillo et al. [6] demonstrated that node-based methods, such as the one of Ng [28], as opposed to approaches based on path information, only provide an upper bound to the minimum number of link sensors required for full link flow observability.

Following the same idea of providing a scalable method by addressing the complexity of approaches, Castillo et al. [7] proposed to focus on reducing the time and space complexity of the standard algebraic techniques. They proposed a new formulation of the problem by formulating link, OD, and scanned link flows in terms of route flows rather than OD flows. This allows to carry out the computation effort using matrices containing only zeroes, ones, and minus ones in most of the necessary iterations. Thus, allowing the use of ternary arithmetic, which reproduces the algebraic results exactly, without incurring numerical precision problems, leading to faster computation and lower memory requirements. However, the process is not always applicable at every iteration, in which case the slower pure algebraic method is applied. Additionally, the proposed method is not only valid for full observability but also for partial observability problems. However, the size of the incidence matrix increases quickly with the size of the considered network; therefore, the proposed method has a limit in its applicability.

With the same aim to avoid a complete path enumeration, Castillo et al. [6] proposed a new concept for the sensor location problem. They proved that the minimum number of links required to observe all link flows could not be obtained without path information; however, to obtain the minimum number of links to be observed, not all paths are required, a subset of linearly independent paths is sufficient. They provided an algorithm to identify a subset of linearly independent paths to facilitate the process. Based on which they developed a second algorithm capable of identifying the minimum subset of links to be observed for complete link flow observability over the considered network. Even if this approach requires less computation time compared to previous methods, the computational time needed to identify the set of linearly independent links will still grow exponentially with the network size.

2.2.2 Topology based approaches

In order to provide an approach that is actually scalable and applicable on large networks, some research works developed topology-based methods for the counting sensor location problem; the main advantage of these approaches is that they require less prior information compared to previous methods as they only use the topology of the considered transportation network. In their work, Morrison et al. [26] studied the problem of sensor location as described in Bianco et al. [3]. They added a stronger necessary condition by providing a counterexample to the problem, which stipulates that the set of unmonitored links should form a tree to validate this new constraint.

Later, a complete topological method was provided by He [15]. Based on a graph transformation allowing to form a spanning tree on the considered network, they demonstrated that links corresponding to the chosen spanning tree represent the set of links that should be left unmonitored. Therefore, all links that don't belong to the spanning tree need to be equipped with a counting sensor and, based on spanning tree properties and on the law of flow conservation, the link flows of unobserved links that form a spanning tree can be inferred based on the set of links equipped with sensors. The main advantage of this approach is that it only requires knowing the topology of the network, in the form of a graph, to be able to apply it, and as the process of finding a spanning tree has a low computation complexity bound by $O(NlogL)$, thus this approach can be considered easily scalable. Additionally, this approach can easily be adapted for the controller location problem under the assumption that links forming a spanning tree can be indirectly controlled by the remaining set of links equipped with controllers.

2.2.3 Optimization based approaches

Other research works focused on developing approaches based on optimization techniques, such as in Yang et al. [46]. In their work, the authors aimed to observe link flows such as separating as many origin-destination pairs as possible. An origin-destination pair is said separated if the current traffic-counting stations entirely intercept trips between this origin-destination pair. As a first step, the authors proposed a new integer linear programming formulation for the problem; however, due to the problem being NP-hard, they applied a relaxation to it. Then, based on this new formulation, the authors developed a shortest path-based column generation algorithm and a branch-and-bound approach for solving the traffic-counting location problem.

Liu et al. [23] considered a different version of the problem by including spatiotemporal correlation, providing the ability to infer flows considering time. The authors first reformulated the sensor location problem by incorporating spatiotemporal correlation. Based on this new formulation, they developed an ant colony optimization algorithm with a local search procedure specifically designed for this problem. However, for this dissertation, we are going to focus on approaches that only consider fixed time.

A recent research paper (Rodriguez-Vega et al. [38]) proposed to consider the identification of locations for different types of sensors simultaneously. In their work, they considered two types of sensors, one which allows measurement of turning ratios at an intersection and the other that directly measures the vehicle flow on a given road. The authors consider intersections as locations for the turning ratio sensors, under the assumption that the overall number of required sensors is known a priori. To identify suitable locations, they use a greedy algorithm that prioritizes intersections bearing the highest number of connections. After locating turning ratio sensing infrastructure, the authors place the remaining road flow sensors through an algorithm similar to the one introduced by He [15]. The resulting time complexity of the method is $O(n_N log(n_N) + n_\epsilon)$, where n_N is the number of network intersections, and n_ϵ is the number of road links. This approach could be adapted to the problem of controller location by considering a combination of pricing controllers to control roads and traffic lights to manage intersections.

Based purely on optimization techniques, the authors of [13] proposed two new approaches in their work, one based on a branch-and-cut algorithm and a second based on a clustering search heuristic. They showed that their methods could provide optimal solutions on most of the tested networks and that in cases where an optimal solution could not be reached, high-quality solutions were still found. To validate the performance of their approaches, they proposed a comparison with a classic genetic algorithm from Chen et al. [8]. The proposed clustering search heuristic was shown to be able to outperform the state-of-the-art genetic algorithm. Additionally, they applied their approaches to real-life transportation networks in Brazil in order to help decision-makers.

In a recent work (Rubin et al. [40]), the authors proposed an exact algorithm to solve the counting sensor location problem. However, to differ from classical approaches that aim to solve a specific version of the problem, they present an exact algorithm capable of optimally solving the problem in its general form. Specifically, their method allows using any type of information, flow, and sensor placement, such as nodes, links, or a combination, overcoming multiple limitations of other existing methods. For this purpose, they proposed a new formulation of the problem, based on which they developed an implicit hitting set-based algorithm. They validated the performance of their approach by solving the sensor location problem over four existing transportation networks from previous works and showed that their approach is capable of solving the problem on these networks.

In the work of Ortigosa et al. [30], the authors studied the number and location of measurement points needed for implementing an efficient macroscopic fundamental diagram-based perimeter control scheme. Their work revealed that a minimum of 25 % of network coverage is required to ensure the efficiency of the perimeter control scheme. In order to validate these results, the authors tested and validated their approach using a micro-simulation model of the inner city of Zurich.

2.2.4 Sensor location approaches for origin destination trip estimation

Aside from flow observability, sensors can also be used for various other measurement objectives; as such, Jabari et al. [19] developed a method to determine sensor locations which guarantee that the time needed to detect an incident in congested conditions is kept as short as possible.

Other research works concentrated on the relationship between sensor locations and the problem of traffic demand estimation, specifically in terms of the reliability of the estimated origin-destination matrix, which is shown to be heavily dependent on the sensor types and locations employed. Yang et al. [45] proposed a theoretical investigation into the reliability of an estimated OD trip matrix introducing the concept of Maximum Possible Relative Error (MPRE). The proposed MPRE metric represents the maximum possible relative deviation of the estimated origin-destination matrix from the true one. They showed that the chosen number and locations of sensors have an impact on the accuracy of the predicted matrix and, more specifically, that the number of observed independent links should be maximized. Based on the concept of MPRE, Yang and Zhou [47] proposed a set of rules to identify locations for sensors to produce an estimated origin-destination trip matrix as realistic as possible. This set of rules stipulates that sensor locations should be chosen such that all origin-destination pairs are covered. Each sensor should intercept as many flows as possible, and that traffic counting points should be located on the network so that the resulting traffic counts on all chosen links are not linearly dependent. The set of rules they proposed points to critical properties when considering the coverage of a transportation network with counting sensors, rules that can also be relevant for locating controllers on transportation networks.

In this section, we reviewed numerous approaches that have been developed for observability; however, they tend to lack scalability due to heavy computational complexity and/or rely on demanding traffic flow data inputs that are difficult to obtain. Some approaches can potentially be adapted to the controller location problem; however, a substantial difference exists between sensors and controllers. As illustrated in Figure 2.1, sensors are used to simply observe the network and do not exert any impact on the flow distribution. Controllers, as depicted in Figure 2.2, once positioned on a network, are used for steering the flow distribution toward another state, thus changing the current state of the network. Therefore, while we can use the set of locations provided by approaches employed for observability as locations for controllers, there is no guarantee that the resulting controller set will fully control the underlying network. Additionally, controlled flows and controller locations are on a fixed point; thus, both need to be determined simultaneously. Therefore, applying observability methods to controllability is not a straightforward task.

2.3 Framework

In order to develop methods capable of locating an efficient set of controllers, we first need to define a framework that will provide the basis for the development of such approaches. This section will describe a few previous seminal works used as a basis for this dissertation. Primarily, they provide us with a representation of the impact of various types of controllers on a transportation network and a way to evaluate the actual capability of a controller set to fully control the underlying network. Additionally, we also detail an approach developed for controller locations that will be used as a comparison for methods developed in this dissertation.

Throughout this dissertation, a given transportation network is represented by a directed graph $G(N, L)$ comprising of a set N of nodes and a set $L : l \in L = (i, j),\ i, j \in N$ of di-

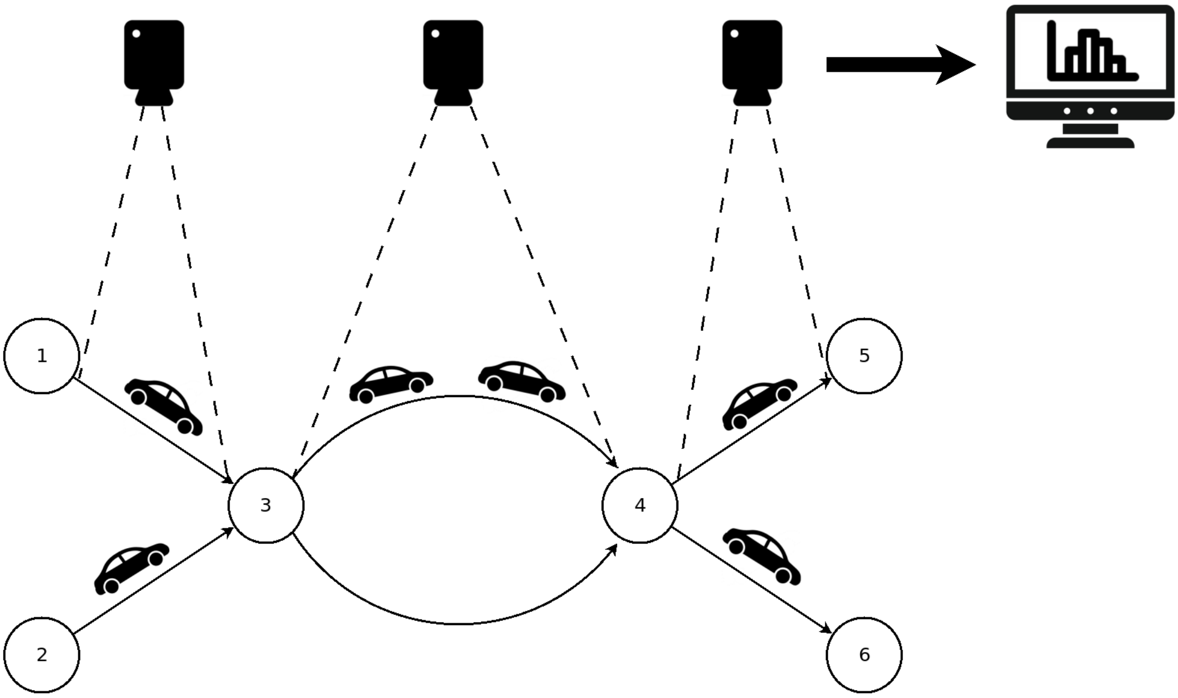


Figure 2.1: Sensor location problem. Observing doesn't change the nature of the observed object; thus, road users' behavior is not impacted.

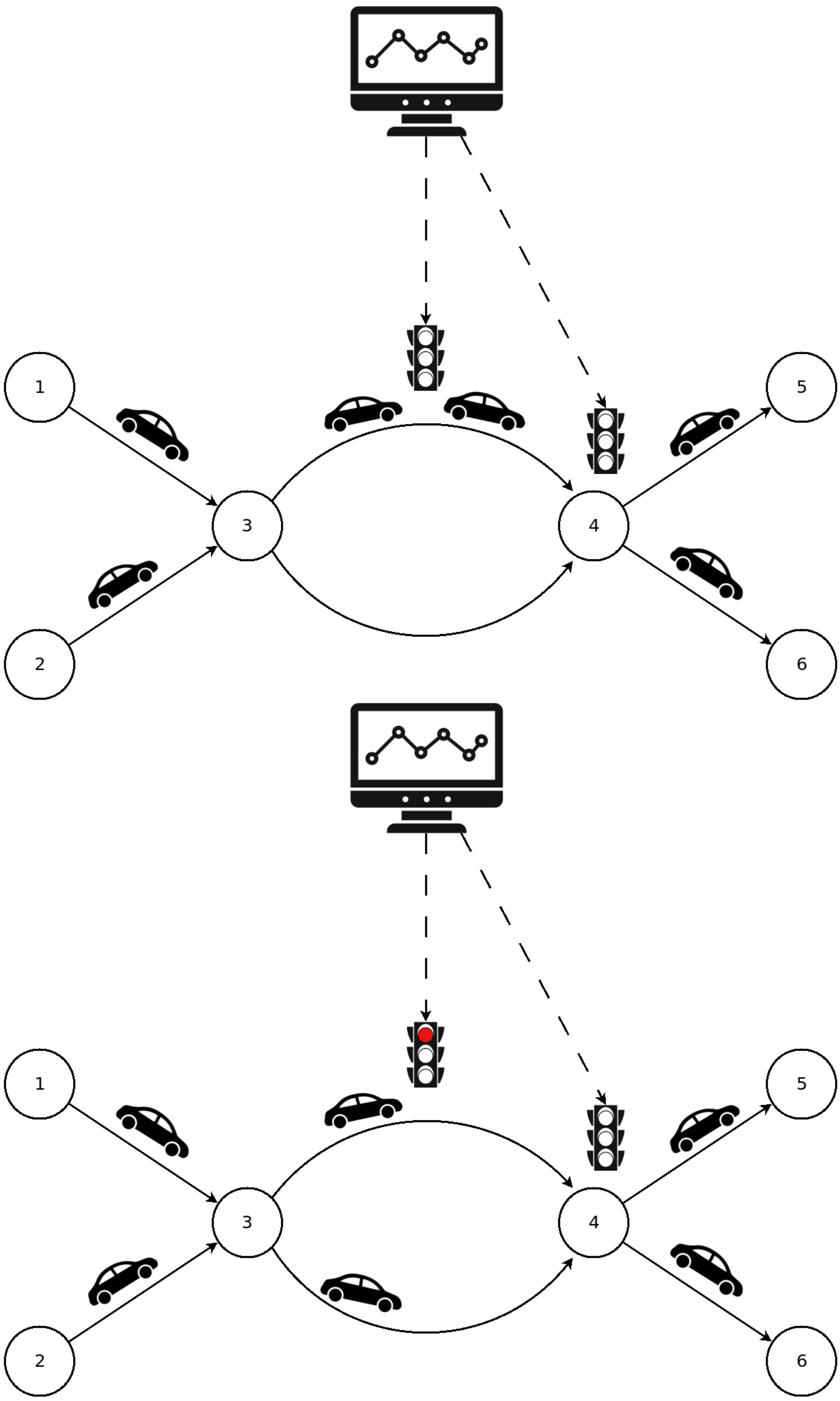


Figure 2.2: Controller location problem. Controller actions have an impact on road users' behavior.

rected links connecting said nodes. To represent user behavior, specific nodes are included to define origin centroids, where traffic flows are produced, and destination centroids, where flows are attracted.

2.3.1 Controller representation

In order to be able to identify efficient controller locations for controlling a transportation network, we first need to understand and be able to represent the action that each type of controller would have on the network. For this purpose, we use the work of Rinaldi [34] as a basis. In this work, the author first proposes to study the dynamics of pricing controllers: these controllers apply a direct influence on the cost of traversing a considered link in the network by adding a monetary cost for road users passing through this link. This added cost has an indirect impact on the number of users passing through this link at a given time since by considering the added cost of the pricing controller, users might decide to use an alternative route that is now comparatively less costly. For the second type of controller, the author details the functioning of traffic lights; a critical characteristic of the author's representation of traffic lights is that, contrary to pricing controllers, traffic lights are located on nodes of the graph, corresponding to intersections in the underlying transportation network. A traffic light equipped on a node will manage the incoming flows to redistribute the available node capacity by routinely assigning green time to the possible flow directions. By doing so, the controller induces indirect costs in the form of delays for road users passing through the considered intersection, making some routes more or less attractive compared to others. Additionally, the indirect cost of added delays can be seen as less constraining from the road user perspective, making traffic lights a more realistic type of controller to employ on real transportation networks. As described, a critical difference between these two types of controllers under this representation is that traffic lights are located on nodes, representing intersections of the network, whereas pricing controllers are located on links of the network. Therefore, these two types of controllers have a different impact on the considered network under this representation.

2.3.2 Level of controllability

To provide an efficient controller set, we need to evaluate the actual capability of a considered collection of controllers to control the underlying network. For this purpose, we studied the work of Rinaldi [34], which provided a framework based on which we can assess if a considered set of controllers is actually capable of fully controlling the underlying network. This work postulates that locations and number of installed controllers should be chosen such that the controller set can achieve full controllability of the network. That is, being capable of steering the system toward any target flow distribution, ensuring the reachability of globally optimal performances for control policies. For this purpose, the author introduced a framework adapting control theory principles to the instance of transportation networks, in which they provided an adaptation of the controllability gramian matrix, presented by Kalman et al. [20]. It allows computing the level of controllability yielded by a set of controllers placed on a given network, thus providing a method to assess the actual capability of a controller set to fully control the underlying transportation network.

To compute this level of controllability for a considered controller set, we first need to define two matrices. The state matrix $A \in \mathbb{R}^{n \times n}$, with n being the number of nodes in the network, represents the influence that a considered node has on adjacent nodes. This matrix is directly based on the network node adjacency matrix, representing which nodes are connected, by a link, to which other nodes. Additional information related to routes can be added in the matrix A as described in the work of Rinaldi [34]. The input matrix $B \in \mathbb{R}^{n \times m}$, with m being the number of controllers on the network, expresses which nodes are affected by the control action of which controllers. For example, if a pricing controller is placed on a node n, as it will only directly affect the node n itself, $B(n, m) = 1$. Whereas if a traffic light is located on a node n then all the predecessor nodes $(p_1, p_2, ...)$ of n will have an equal percentage of 1 in the matrix B such that $B(p_1, m) + B(p_2, m) + ... + B(p_k, m) = 1$. Based on control theoretical principles, the network is considered fully controllable if one of its characteristic descriptive matrices, known as controllability gramian, has an algebraic rank equal to the system's total number of state variables. The controllability gramian can be derived from the two previously described matrices (A, B) as follows:

$$W_c = [B\ AB\ A^2B\ ...\ A^{n-1}B]$$

As detailed in the work of Kalman, a sufficient condition to guarantee full controllability of the system is that the rank of the gramian is equal to the size of the state matrix, thus equal to the number of nodes in the network, such that $rk(W_c) = n$. Therefore we can define the level of controllability as $LevelOfControllability = \frac{rk(W_c)}{n}$, such that $LevelOfControllability = 1$ if full controllability is reached. In this dissertation, the level of controllability is always computed for n as the number of nodes in the entire network unless specified otherwise. Based on this framework, we can assess if a controller set is capable of fully controlling a considered network. This concept will be used throughout this dissertation to evaluate the efficiency of developed methods.

However, as described earlier, to use this concept, controllers need to be considered on nodes; thus, this process is suitable when considering traffic light locations but not straightforward to apply when considering pricing controllers for links. To resolve this incompatibility, we propose to employ the principle of the dual graph transformation introduced by Añez et al. [1]. This graph transformation is obtained by following three main rules:

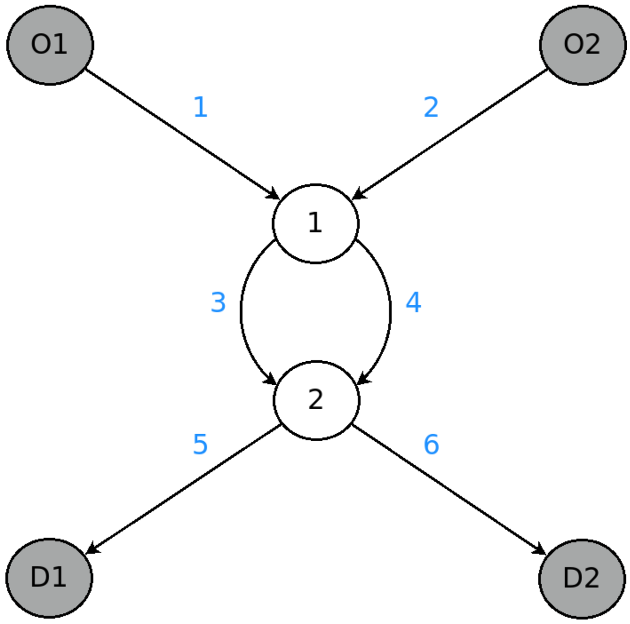
1. Nodes of the dual graph represent links of the primal graph, and they retain all characteristics of the original links.
2. Links of the dual graph represent turning movements.
3. Centroids such as origin and destination nodes are represented by nodes in both primal and dual networks.

To illustrate the dual graph transformation, we consider a simple network (Fig. 2.3a), and we apply this process to obtain its dual representation (Fig. 2.3b). Under this representation, links of the original graph are now represented by nodes; thus, once locations for pricing controllers are identified on the primal graph, the locations can be transferred on dual graph nodes, and the level of controllability held by the set of pricing controllers can be calculated. This is possible as the dual form of a network still represents the same network but with a richer representation, specifically regarding the representation of the turning movements.

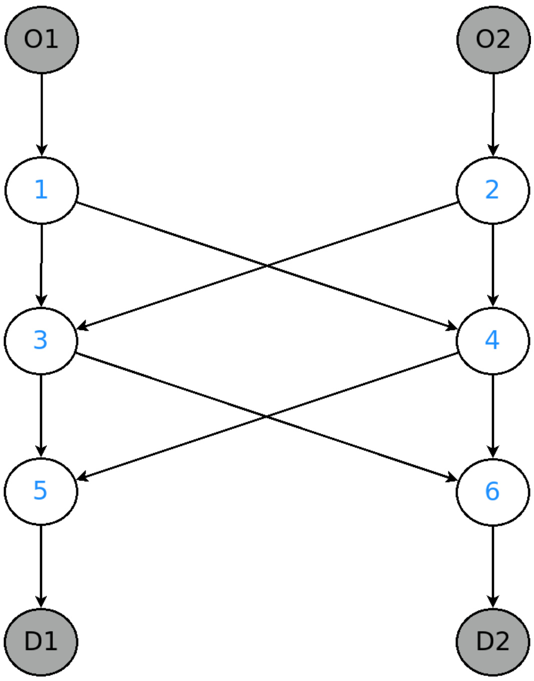
2.3.3 Exact approach

To evaluate if methods developed in this dissertation are efficient, we propose comparing them to an existing approach capable of locating controllers on a transportation network. For this purpose, we chose as a reference the exact method introduced in the work of Rinaldi [34] which is an adaptation of the approach developed by Yuan et al. [51]. This approach is based on the state matrix A; the first step consists in computing its corresponding eigenvalues. Based on the found eigenvalues, the next step consists of identifying each eigenvalue's geometric multiplicity. Once the eigenvalue bearing maximum multiplicity is identified, the matrix can be reordered to isolate the linearly dependent and independent components, then the minimal set of pricing controllers can be determined based on the set of linearly independent links identified.

However, a recent work (Rinaldi and Viti [37]) demonstrated that the exactness of the method can not be guaranteed under the presence of bi-directed links in the network, which is a common setting in transportation networks. Bi-directed links in the network will cause a violation of the algebraic assumptions behind the method due to the self-dependencies introduced by bi-directed links during the computation of the algorithm. In most networks, these dependencies lead to a collapse of the eigenvalue/eigenvector information content, thereby misleading the chosen approach towards solutions with no practical significance. Figure 2.4 demonstrates this effect on a simple grid-like network of 20 nodes containing only one origin-destination pair connected by three routes. We consider that only links belonging to at least a route are necessary to control; thus, in this example, we only consider the sub-network comprising the route set. The first instance of this network only contains mono-directional links (Fig. 2.4a); more specifically, all links in the network are directed from left to right or top to bottom. The exact method can identify a set of pricing controllers that guarantees full



(a) Primal network with links numbered in blue.

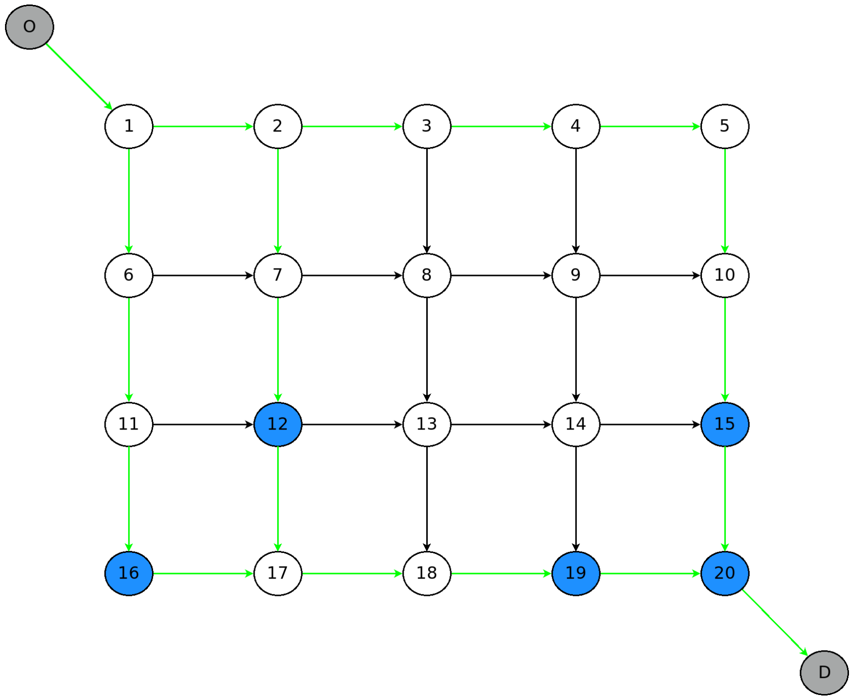


(b) Dual representation of the network.

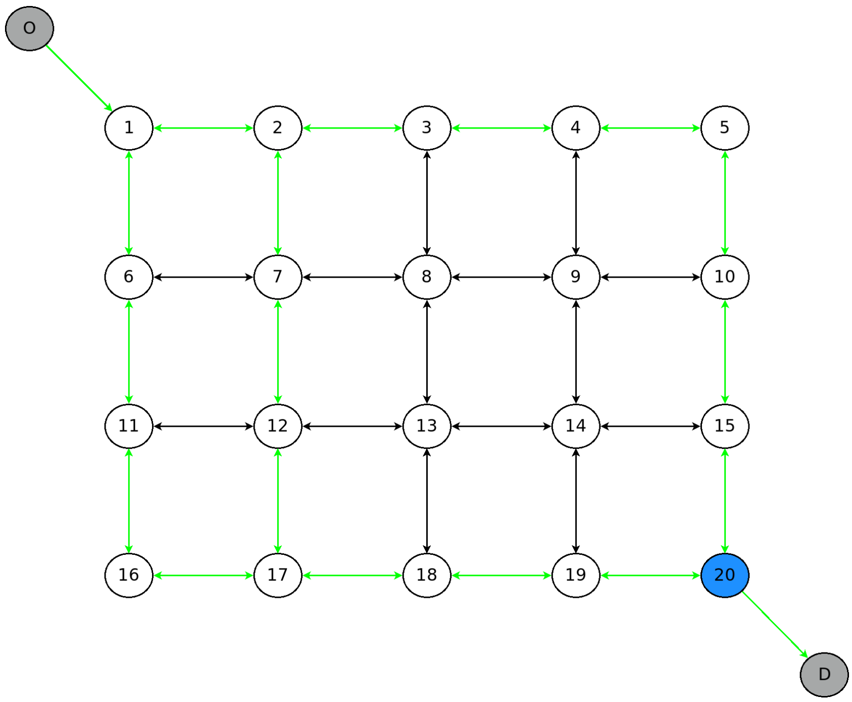
Figure 2.3: A simple network with two origin and two destination nodes and its dual representation.

controllability on this network. However, on the second instance of the network where all links are bi-directional, the method exhibits difficulties in producing a sensible set of pricing controllers as exemplified in Figure 2.4b where the obtained solution contains only one controller. Therefore, we can observe that the introduction of bi-directionality significantly alters the capability of the exact method to produce an efficient set of pricing controllers as the selected set is clearly incapable of effectively influencing the whole network (the measure of the level of controllability yielded by the produced set is indeed not fully controllable).

However, the authors proposed a variation of the method that does not guarantee full controllability but can guarantee that a high level of controllability is reached. For this purpose, they remove from the state matrix conflicting information, thus allowing the computation of a candidate controller set. Unfortunately, by removing information, the method might produce controller sets of lower quality, and the found solution is not guaranteed to be optimal.



(a) All links are mono-directional (from NW to SE). The network is fully controlled.



(b) All links are bi-directional. Only 1 out of 20 nodes is controlled.

Figure 2.4: Yuan's method applied on a 20 nodes graph. Pricing controller locations are marked in blue. Route set is represented in green

Chapter 3

Controller substitutability

In this dissertation, we focus on two types of controllers, traffic lights and pricing controllers, each possessing their characteristics and impacting networks differently. As demonstrated in Verhoef et al. [43], pricing controllers are capable of reaching optimal performance on any transportation network while using a first-best pricing scheme. However, traffic lights represent a more widespread technology and are more realistic to be used in real networks. Therefore we want to study the possible substitutability of pricing controllers by traffic lights on transportation networks to obtain a set of traffic lights as efficient as a set of pricing controllers. Our main objective in this section is to determine whether pricing controllers can be substituted for traffic lights while maintaining complete controllability over the considered network. For this purpose, based on simple atomic grid-like networks, we propose to empirically try to infer through combinatorial search if a topological substitution rule exists.

This section is based on the work done for the following paper: X. Mazur, M. Rinaldi, and F. Viti, On the substitutability of traffic light and pricing controllers in transportation networks. That was published in the proceeding of the conference: 2019 6th International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS).

3.1 Methodology

This subsection discusses the methodology employed to determine whether and how a set of traffic lights can replace a given set of pricing controllers without loss of controllability over the considered network. As described previously, every given transportation network is represented by a directed graph $G(N, L)$. On these graphs, any node $i \in N$ is a potential location for a controller, whether we consider a traffic light or a pricing controller. In this section, we consider that pricing controllers can be located on nodes such that they impact the cost of all outgoing links from the considered node. The differences in functioning between these two types of controllers imply that the substitution of a kind of controller by the other one is not a straightforward task. For example, a traffic light might reproduce the impact of multiple equivalent pricing controllers since it can influence turning flows rather than route flows.

To assess if a considered set of pricing controllers or traffic lights is capable of controlling the underlying network, we used the controllability framework presented in the previous chapter. Based on this framework, we can compute the associated degree of controllability of a set of controllers, and thus we can compare the impact of different controller sets. To provide a complete study, we based our exploratory approach on variously-sized networks. Thus, we developed a simple method to generate different sized square grid networks to have simple networks on which we can study possible substitution rules. Every generated network follows the following set of topological criteria:

1. Every network possesses two origin-destination pairs comprising of two origin nodes on each of the left corners and two destination nodes on the right corners. The first origin-destination pair connects the top-left origin node with the bottom-right destination node. Similarly, the second pair connects the bottom-left origin node to the top-right destination node.

2. Network flows are directed from the left, where origin nodes are placed, to the right, where the destination nodes are placed. All horizontal links are monodirectional, from

left to right.

3. All vertical links are instead bi-directional, as presented in Fig. 3.1.

This graph generator will be used to produce a set of gradually larger networks. More specifically, each generated network will be composed of n squares. These square units are added one by one following an outward spiral pattern, as displayed in Fig. 3.2. Additionally, we consider user behavior on each network in the form of route choice. Each origin-destination pair is connected by a set of routes, which is determined using the K-shortest path heuristic (Yen [49]). Following research works on road user perceptions in terms of route choices (Fiorenzo et al. [9], Bovy et al. [4]), we chose a value of $K = 3$ routes for each origin-destination pair, such as to capture route choice behavior appropriately. In this section, when computing the level of controllability reached by a set of controllers, we only consider nodes belonging to the sub-network resulting from the generated set of routes.

To obtain a candidate set of pricing controllers that is capable of fully controlling a considered network, we decided to employ the exact approach introduced by Rinaldi [34], that was detailed in chapter 2. To investigate if the exact solutions obtained with this approach can be transformed into equivalent traffic light-based solutions without losing the full controllability of the network, we propose to carry a combinatorial exploration. Specifically, we developed an algorithm that explores every possible combinatorial substitution (Algorithm 3.1).

Algorithm 3.1 Given a network description matrix A

compute full controllability with pricing solution S_p
for $k = |m| - 1$ **to** $k = |m|$ **do**
  compute every k-combination K of traffic-light
  for each k-combination K **do**
    if Level of Controllability (K) = Level of Controllability (S_p) **then**
      add K to S_{tl}
    end if
  end for
end for

Based on a full controllability set of pricing controllers S_p, composed of $|m|$ controllers, this

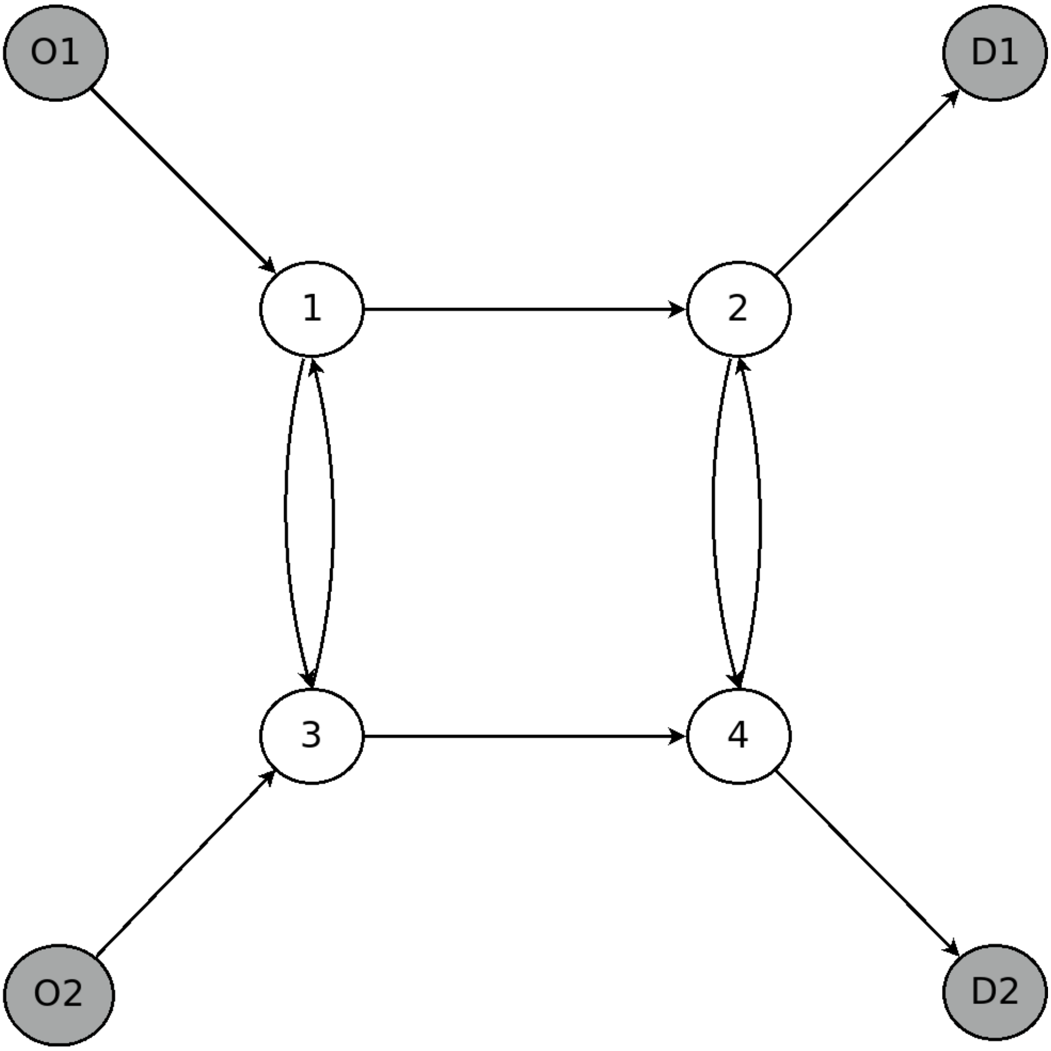


Figure 3.1: Square grid graph example.

algorithm will compute a set of traffic light based solutions by exploring every combination of $|m-1|$ and $|m|$ traffic lights, a set that will contains a total amount of $\binom{|N/\{o \cup d\}|}{|m-1|} + \binom{|N/\{o \cup d\}|}{|m|}$ solutions. Then, only traffic light solutions that reach the same level of controllability as the one achieved by the considered set of pricing controllers S_p are kept. Thus, the final set Stl contains only traffic light-based solutions equivalent to the considered pricing-based solution. By comparing the equivalent traffic light-based solutions to the pricing solution, we aim to infer whether a topological relationship exists due to similarities between the original pricing-based solution and one or more traffic light-based alternatives to identify possible substitution rules.

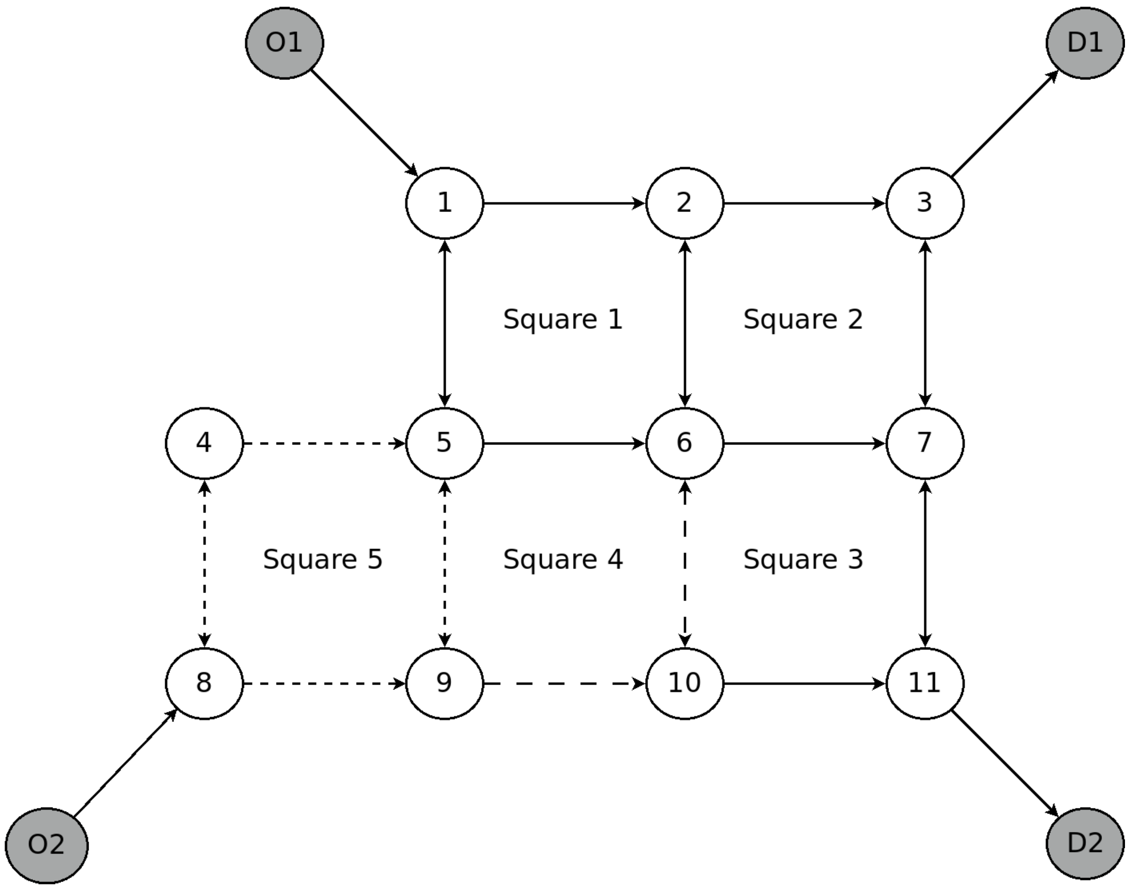
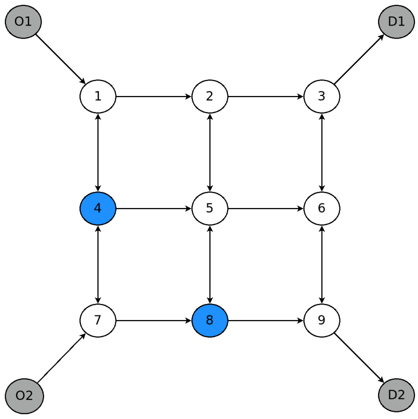


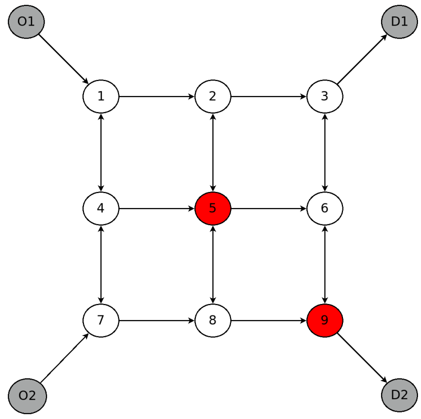
Figure 3.2: Increasing size of square grid graph generator.

3.2 Experimental results

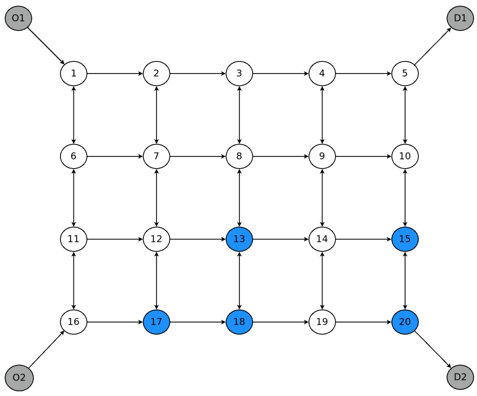
Based on the developed square gird network generator, we propose to focus on a set of three differently sized networks for this experimental setup. The smallest generated graph has a size of three by three nodes and is composed of a four squares grid (Fig. 3.3a). The second considered graph has a size of four by five nodes, resulting in a total of twelve squares (Fig. 3.3c). As for the third network, it is configured as a five by five nodes network, producing a sixteen-squares network (Fig. 3.3e). Based on this set of graphs, we apply the algorithm 3.1 to obtain a set of traffic light-based solutions that are equivalent to the considered pricing solution. By examining the obtained traffic light solution, we searched to highlight similarities between solutions using different controllers. It is interesting to observe that, on the set of considered networks, no traffic light-based solution manages to reach the same level of controllability as the pricing solution while using one less controller (Table 3.1). To infer some simple topological rules, all traffic lights solutions were compared to the pricing solution, seeking to identify topological regularities, such as relationships between predecessor - successor and node distances between the pricing positions and the traffic light positions. Such a topological rule was detected on all three considered network instances: a pricing controller located on a given node i can be replaced by a traffic light placed on a node $j : \exists l, (i, j) \in L$, if node j is a direct successor of node i through a directed link l (Fig. 3.3).



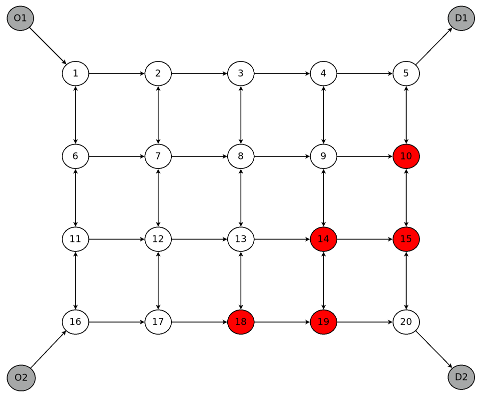
(a) Three by three network with pricing controllers.



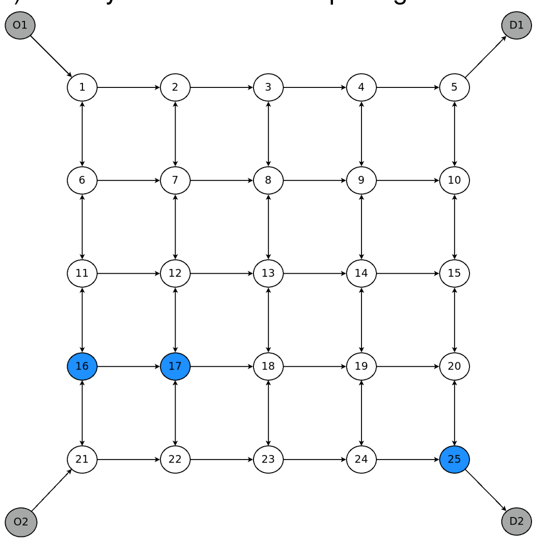
(b) Three by three network with traffic lights.



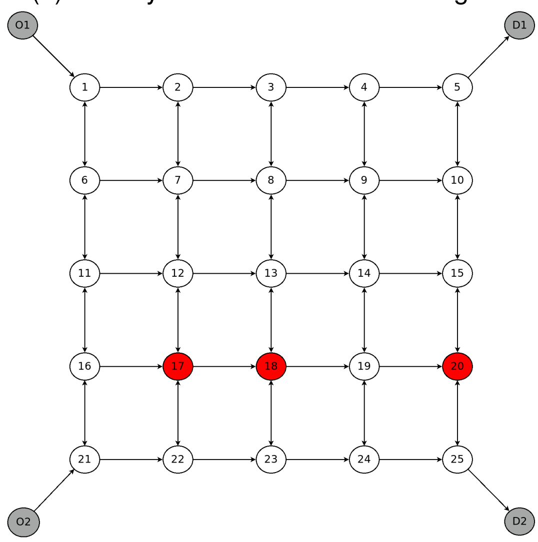
(c) Four by five network with pricing controllers.



(d) Four by five network with traffic lights.



(e) Five by five network with pricing controllers.



(f) Five by five network with traffic lights.

Figure 3.3: Set of three different-sized graphs equipped with pricing controllers and equivalent traffic lights.

Graph size	3 by 3	4 by 5	5 by 5
Amount of pricing controllers $\|m\|$ required to reach full controllability	2	5	3
Total enumerated combinations of $\|m\|-1$ traffic lights	7	3060	253
Number of $\|m\|-1$ traffic light combinations that reach full controllability	0	0	0
Total enumerated combinations of $\|m\|$ traffic lights	21	8568	1771
Number of $\|m\|$ combinations that reach full controllability	2 (∼10%)	2017 (∼24%)	175 (∼10%)

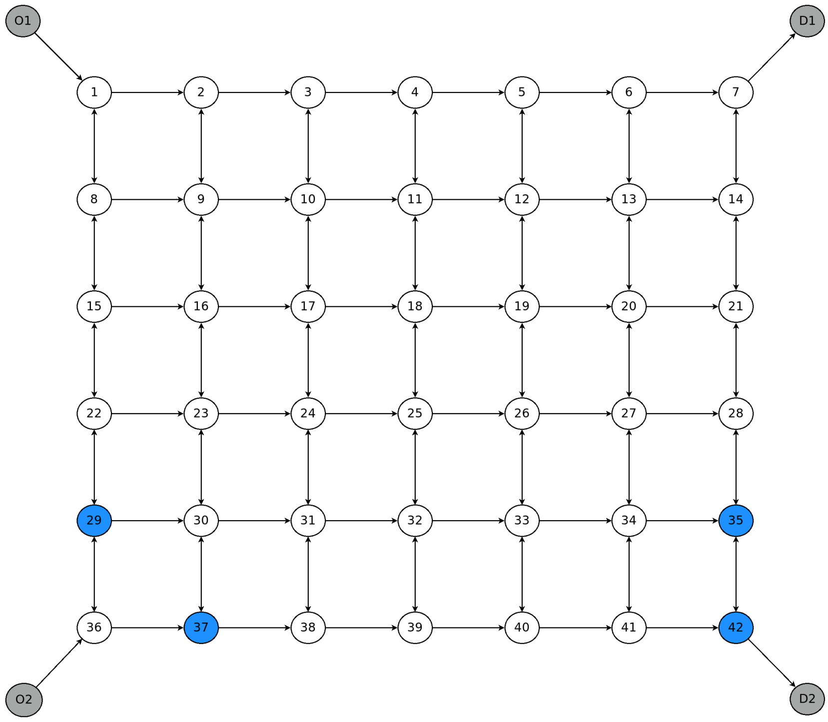
Table 3.1: Controllers needed to reach full controllability.

Additionally, we propose a second experimental setup to validate the topological substitution rule previously identified. We decided to employ a larger network for this second experiment, so we generated a six-by-seven-nodes graph composed of 30 squares. We first applied the exact method to obtain a set of pricing controllers; the found collection is composed of four controllers (Fig. 3.4a). Then we applied the previously described substitution rule to obtain an equivalent set of traffic lights (Fig. 3.4). After computing the level of controllability reached by the produced set of traffic lights, we observed that the full controllability was indeed maintained over the network. This experiment provides a validation that the inferred substitution rule can produce a set of traffic lights that can fully control the underlying network. It also shows that this approach is applicable on large transportation networks where computing every possible combination of traffic lights would have resulted in considerable computation costs.

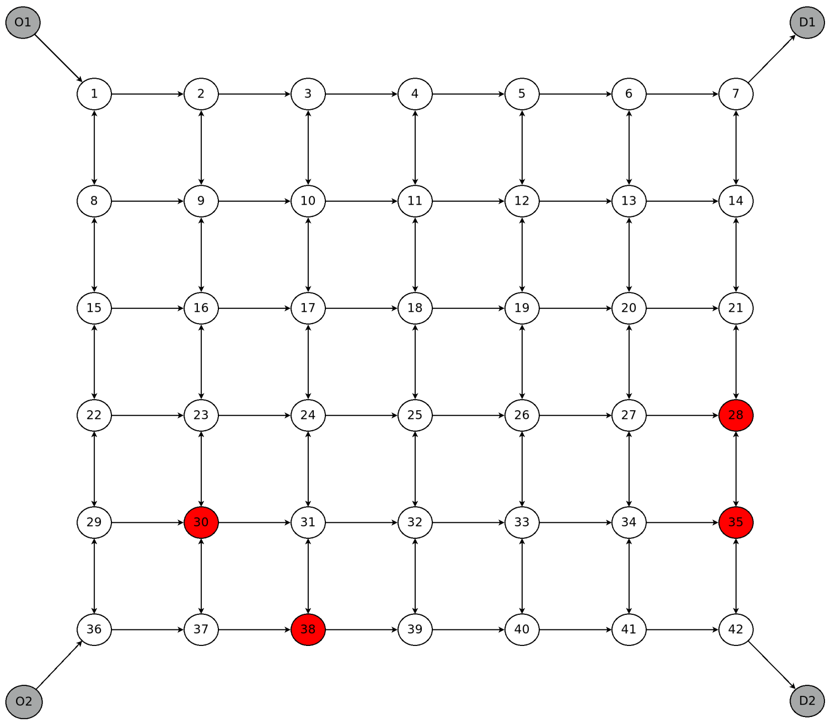
To confirm this result, we propose to consider a set of larger networks. For this purpose, we successively produced networks of increasing sizes through our square grid graph generator, starting from a twenty-five-square-sized network up to a one hundred squares network, for a total of 75 different instances. Out of this set of networks, four were excluded due to unadapted shape preventing the computation of a feasible set of pricing controllers. On 94.3% of the remaining networks, the level of controllability yielded by the substituted set of traffic lights was found equal to the one produced by the pricing approach. The substitution rule could not find feasible traffic lights set on only 4 out of 71 scenarios. These results indicate that the proposed substitution rule can locate an efficient set of traffic lights on square grid networks while avoiding a complex combinatorial exploration. Additionally, we observed over every tested instance that the number of controller nodes required to control the considered network fully is not correlated with the network size in terms of nodes number. Instead, the required number of controlled nodes exhibits a tight distribution of 7% of the network nodes, as displayed in Figure 3.5. Future research will be required to evaluate whether this property depends on the network characteristics, such as the distribution of origin-destination pairs or the general regularity of the tested networks.

3.3 Conclusion

With this study, we explored the possibility of substituting existing pricing controllers with traffic lights on the specific instance of grid-like networks. We identified a simple substitution rule by observing topological similarities between pricing and equivalent traffic light-based solutions. We demonstrated that by following this simple rule, we could identify a set of traffic lights with no loss of controllability compared to a pricing solution. Based on a set of 75 networks, we validated this result, showing that the rule provided valid and lossless solutions in more than 90% of instances.



(a) Pricing locations.



(b) Traffic light locations.

Figure 3.4: Six by seven nodes network.

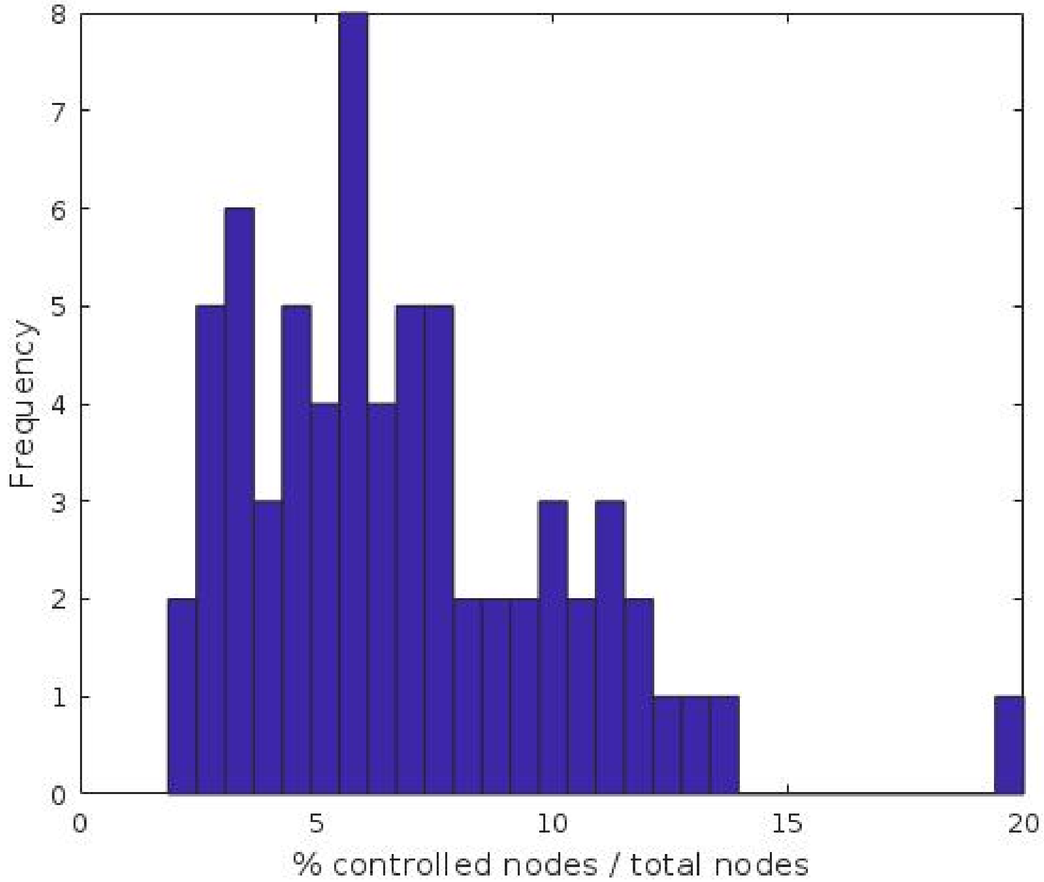


Figure 3.5: Distribution of the percentage of controllers needed.

Chapter 4

Rule-based heuristics

Based on the tools provided in the previous chapters, we propose to study topological properties of transportation networks that would contribute to guide the selection of an efficient set of controllers, such as guaranteeing that the highest level of performance is reachable on transportation networks. For this purpose, we developed several methods based on these properties that can identify essential locations while minimizing the number of controllers employed. Based on the previously presented framework, we provide an experimental setup composed of numerous networks bearing various sizes and characteristics to analyze the performance of the developed heuristics.

This chapter is based on the work done for the following paper: X. Mazur, M. Rinaldi, and F. Viti, Heuristic methods for minimal controller location set problem in transportation networks. This work was presented at the 23rd EURO Working Group on Transportation Meeting, EWGT 2020, 16-18 September 2020, and was published in Transportation Research Procedia 52, January 2021.

4.1 Introduction

As demonstrated in the previous chapter, pricing controllers are capable of fully controlling transportation networks and can be substituted by traffic light controllers under certain conditions. Therefore, we chose to focus solely on the usage of pricing controllers for this and the following chapters. However, their capacity to actually control a network depends on the number and locations of installed controllers. Thus, we aim to identify locations for pricing controllers on transportation networks such that the underlying network is fully controllable while minimizing the number of employed controllers. In general, the problem of controller location lacks a method capable of identifying a set of controllers such that the considered network is fully controllable. To address this issue, we aim to develop heuristic approaches that would identify a satisfying solution while also considering the number of pricing controllers employed to minimize the number of controllers used. Additionally, we propose using topological information mainly to avoid computational complexity faced by algebra-based approaches, such as to provide a scalable approach.

4.2 Methodology

As in previous chapters, a given transportation network is represented by a directed graph $G(N, L)$ comprising of a set N of nodes and a set $L : l \in L = (i, j), i, j \in N$ of directed links connecting them. Origin and destination centroids are introduced in the network to represent route choice behavior, and each resulting origin-destination couple is connected through a given route set, generated with a K-shortest path algorithm (Yen [49]). For this chapter, when computing the level of controllability reached by a set of controllers, we only consider nodes belonging to the sub-network resulting from the generated set of routes. Every node $i \in N$, except origin-destination centroids, is considered a potential location for a pricing controller, as in chapter 3. To solve the minimal controller location set problem, we developed four simple heuristics based on the topological properties of networks. The proposed heuristics are a class of greedy algorithms; therefore, to find a suitable set of controllers, they locate pricing controllers one after the other, trying to identify the most suitable location for the next controller at each step. At each step, the heuristics will compute the level of controllability reached by the current set of controllers to assess if full controllability is achieved, in which case the process will stop. The proposed heuristics are detailed in the following:

- Algorithm 4.1 - Best controller per route: This approach considers one route r from a previously determined route set at each step. A pricing controller is located on the node of r that brings the highest increase in the current level of controllability. A different route r is considered at each step until full controllability is reached. This approach is purely based on myopic decisions. (Alg.4.1)
- Algorithm 4.2 - Node degree weighted: This approach first assigns weights on each node to reflect the network topology characteristics. Each node is weighted by its degree (total amount of inbound and outbound connections) so that nodes densely connected are considered more important. Pricing controllers are then located successively on nodes bearing the highest weight until full controllability is reached. The intuition behind this approach is that nodes with a higher degree of connectivity are likely to represent key pivotal points in the network. (Alg.4.2)
- Algorithm 4.3 - Origin distance weighted: Similarly to the precedent heuristic, this approach first weights each node of the network with their respective topological distance (number of nodes traversed) to the closest origin, following existing routes. Then, pricing controllers are placed one at each step on the node bearing the smallest weight over the network. This approach is inspired by screen line methods following the idea that placing controllers in the vicinity of origin centroids will allow capturing flows directly at their origins, thus yielding an important impact on flows. (Alg.4.3)
- Algorithm 4.4 - Splitting or merging routes: This last approach follows this intuition that positions where routes from the same origin-destination pair are splitting or merging represent critical positions to control the movement of road users. Therefore, at each step, this approach will place a pricing controller on the node where the highest amount of routes are either merging or splitting until full controllability is reached. Suppose if the number of nodes where routes are merging or splitting is not sufficient to fully control the entire network, for example, due to a specific network's shape. In that case, the set of controllers is completed following the first approach until the desired condition is achieved. (Alg.4.4)

Algorithm 4.1 Best controller per route

for each route in the route set **do**
  get node i with max (level of controllability)
  place controller on node i
  if current level of controllability $=$ full controllability **then**
    stop
  end if
end for

Algorithm 4.2 Node degree weighted

for each node **do**
  node's weight = node degree
end for
while current level of controllability $<$ full controllability **do**
  place one controller on the node with max (weight)
end while

Algorithm 4.3 Origin distance weighted

for each node **do**
  node's weight = topological distance to closest origin
end for
while current level of controllability $<$ full controllability **do**
  place one controller on the node with min (weight)
end while

Algorithm 4.4 Splitting or merging routes

for each origin-destination pair **do**
  for each i node where routes are merging or splitting **do**
    if current level of controllability $<$ full controllability **then**
      place controller on node i
    end if
  end for
end for

4.3 Experimental setup

To provide a complete experimental setup, we propose to employ the graph generator introduced in Mireles de Villafranca et al. [44] to create arbitrarily randomized networks. We consider randomized networks to be especially suitable for this experimental setup as they provide a wide range of different networks that bear characteristics close to real urban transportation networks, thus, showcasing that the proposed approaches are applicable to any type and shape of potentially existing networks. First, this algorithm generates a square grid network of the desired size; then, node locations are randomly perturbed while generating bi-directional links between node couples that are sufficiently close, thus reshaping the original network. (Fig. 4.1) The graph is divided into concentric zones (as exemplified by the red circles in Fig.4.1), each representing a different kind of transportation network areas such as the city center, suburban areas, or outside areas with associated characteristics. For example, speed limits are adapted to the considered type of area. A given amount of origin-destination centroids are introduced; in this work, we chose to generate two origin and destination nodes per area. After that, each origin node is connected to all destination nodes belonging to a different zone by a given collection of routes. For this first experimental setup, we introduced a number of $k = 3$ routes per origin-destination couple.

To have a relevant comparison of the efficiency of the proposed methods, based on the described network generator, we produced a set of variably sized networks, ranging from sixteen up to one hundred nodes. For each network size, one hundred different instances are generated based on a different seed, thus producing graphs with similar characteristics but various shapes (as showed in Fig.4.2). Each heuristic approach is applied on every instance to obtain the corresponding candidate set of pricing controllers. To provide a baseline comparison, we also apply the previously presented exact approach adapted from the work of Yuan et al. [51] by Rinaldi [34] on each generated network.

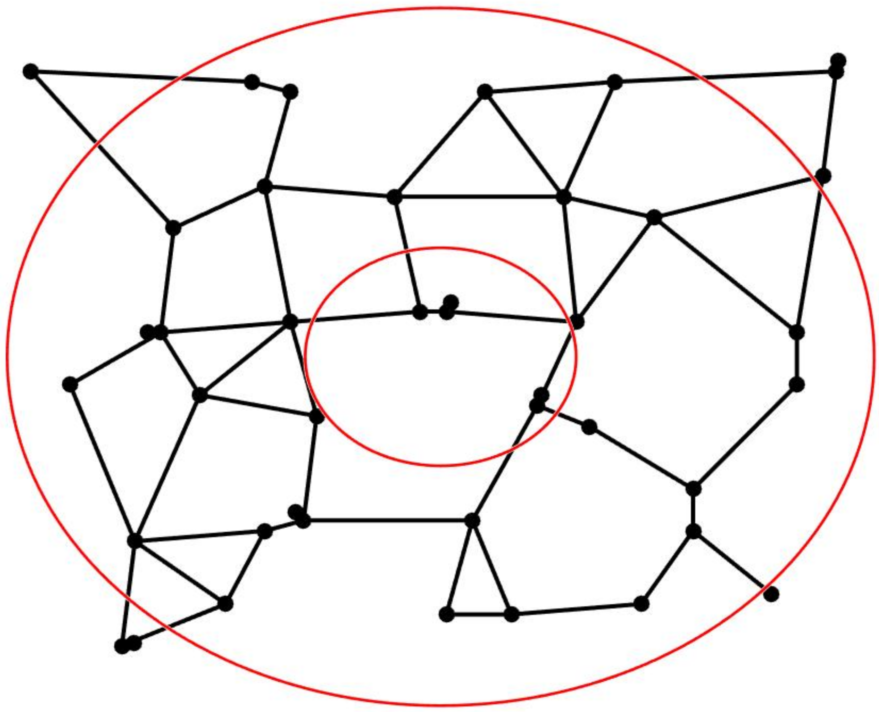


Figure 4.1: Example of a generated network graph.

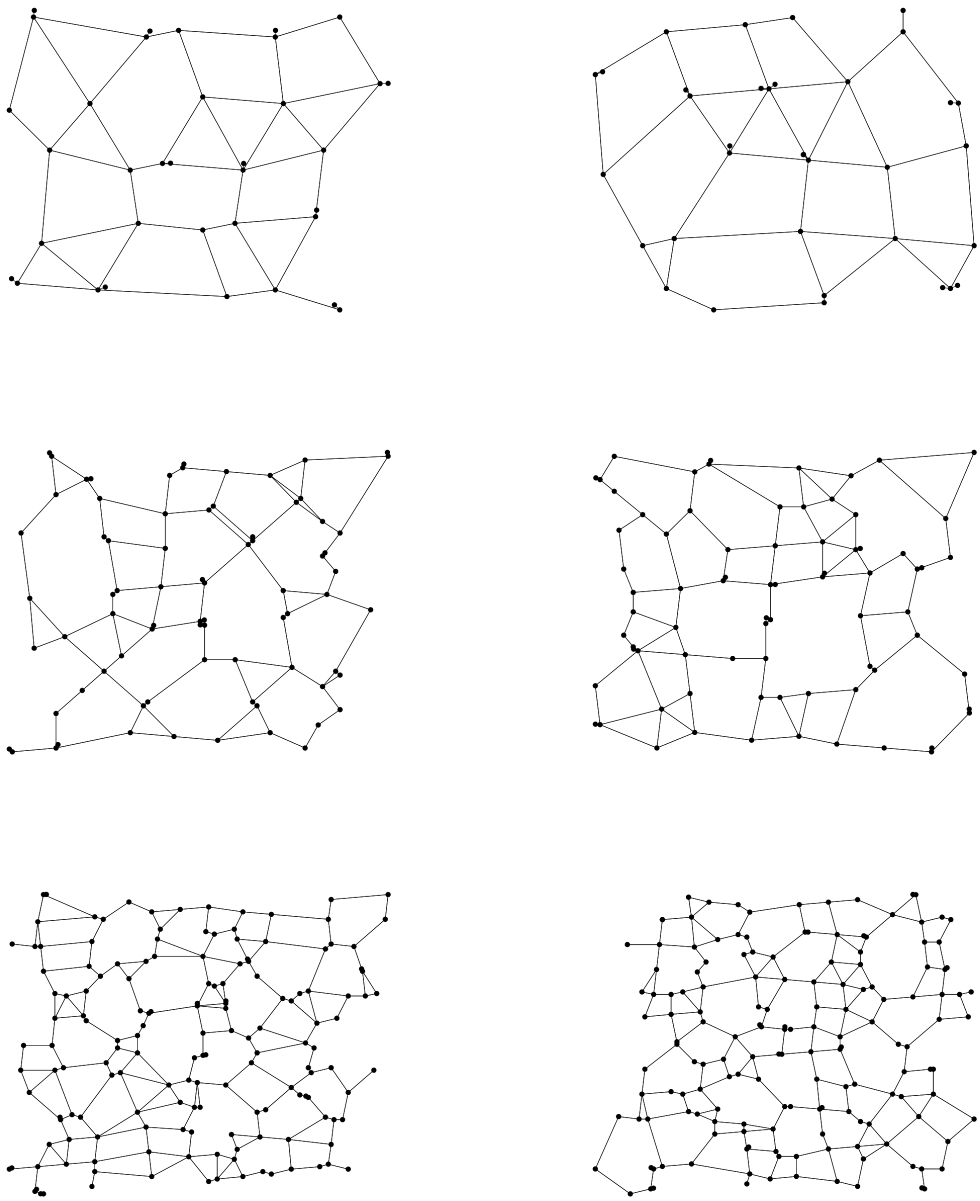
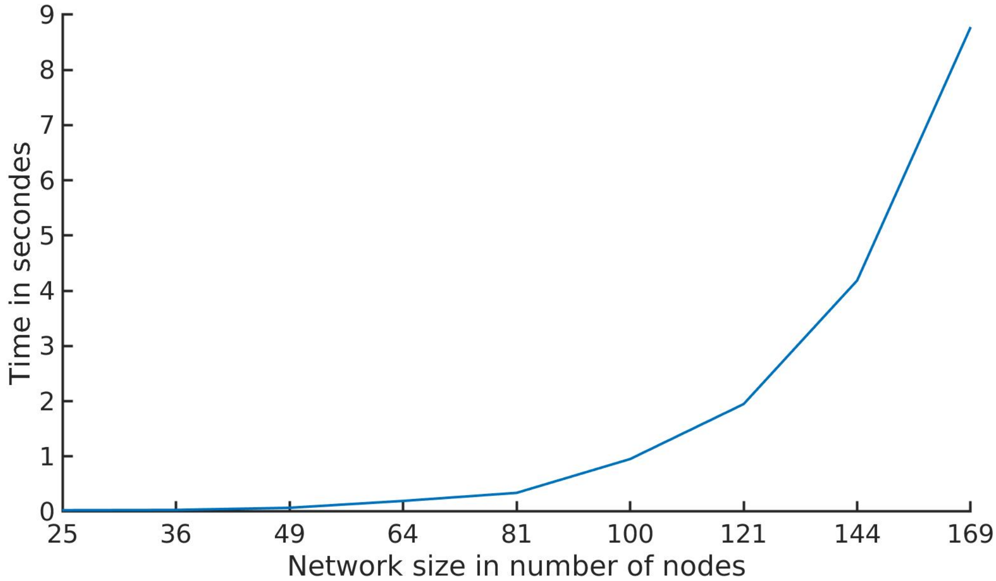


Figure 4.2: Examples of different networks generated.

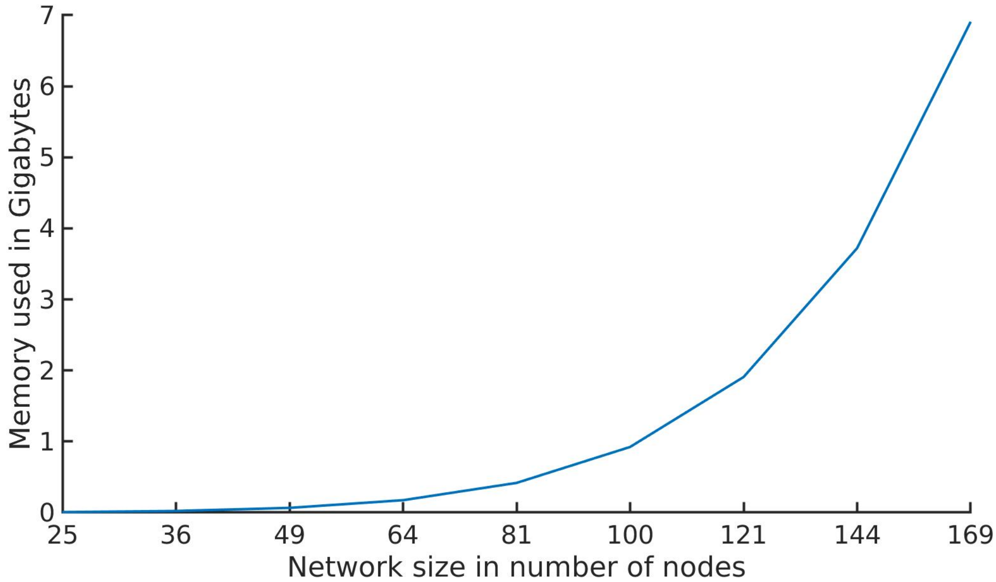
4.4 Results

To evaluate whether the proposed heuristic approaches can reach stratifying results on bi-directional networks, we first observe the percentage of instances where the various methods managed to achieve full controllability, thus satisfying a fundamental constraint of our problem. Table 4.1 displays said comparison over one hundred different randomized networks of 25, 49, and 100 nodes. We can observe that all the proposed heuristics consistently fulfilled the full controllability constraint over every considered instance and network size. As expected, the exact method only managed to provide full controllability over a few instances, thereby confirming the utility of the proposed heuristics to provide a set of controllers capable of fully controlling the underlying network. Additionally, we present the computation time required to produce these results (coded in MATLAB™ on a Dell Latitude-5480, Intel® Core™ i5-7300U CPU). While some approaches appear relatively faster than others, the overall execution times remain satisfactorily low for all methods (Table 4.1). It is essential to consider that execution times obtained with the exact approach are due to erroneous results as this approach will easily find corner solutions and should not be regarded as relevant. Moreover, as displayed in Figure 4.3a the computation time required to compute the level of controllability yielded by a considered set of controllers, which is a process used at each iteration by every of the proposed heuristics, will increase extremely rapidly with the growth of the network size. Similarly, we can observe in Figure 4.3b that the required space, in terms of memory, to compute the level of controllability follows a similar trend and also increase extremely rapidly to reach high values that make this process unusable on large networks. This demonstrates the necessity to further improve the proposed approaches in terms of scalability.

To further assess the heuristic's respective efficiency, we propose a study of the number of controllers employed by each approach to reach full controllability. Figure 4.4 displays the number of controllers used by each approach over one hundred distinct but equally-sized



(a) Evolution of the required computation time.



(b) Evolution of the required memory.

Figure 4.3: Evolution of the computation time and space required to compute the level of controllability over the considered network size.

Network size (in nodes)	Method	Average full controllability percentage	Average computation time	Computation time standard deviation	Average number of controllers
25	Yuan	19 %	0.1065 s	0.0622 s	1.43
	Alg. 4.1 BCR	100 %	0.0436 s	0.0307 s	10.38
	Alg. 4.2 NDW	100 %	0.0153 s	0.0077 s	11.51
	Alg. 4.3 ODW	100 %	0.0182 s	0.0080 s	12.55
	Alg. 4.4 SMR	100 %	0.0758 s	0.0444 s	10.51
49	Yuan	9 %	0.2698 s	0.1117 s	1.33
	Alg. 4.1 BCR	100 %	0.3005 s	0.2107 s	16.24
	Alg. 4.2 NDW	100 %	0.0817 s	0.0454 s	19.51
	Alg. 4.3 ODW	100 %	0.1075 s	0.0510 s	21.48
	Alg. 4.4 SMR	100 %	0.4991 s	0.3478 s	17.12
100	Yuan	17 %	1.3649 s	0.6218 s	2.54
	Alg. 4.1 BCR	100 %	4.2679 s	3.9761 s	26.86
	Alg. 4.2 NDW	100 %	0.8683 s	0.6186 s	34.44
	Alg. 4.3 ODW	100 %	1.2922 s	0.7984 s	35.95
	Alg. 4.4 SMR	100 %	6.9741 s	5.6306 s	28.82

Table 4.1: Approaches performances comparison

networks. The obtained results suggest that some heuristics do perform statistically better than the others for this experimental setup, mainly methods that focus on route information, such as the Algorithm 4.1 ”Best controller per route” and the Algorithm 4.4 ”Splitting or merging routes”. This confirms that additional information brought by route information is significantly relevant for identifying suitable controller locations. As expected, the exact method failed to produce a suitable solution, thus producing a very small controller set that is incapable of fully controlling the underlying network. A similar trend also appears on larger networks, independently of scale, as presented by the results obtained on 49 nodes networks and 100 nodes networks (Fig. 4.5 and Fig. 4.6).

Additionally, we propose to study the evolution of the needed number of controllers by each method with the increase of the number of routes k generated per origin-destination pairs. For this experiment, we generated one hundred networks of twenty-five nodes, Figure 4.7 displays the evolution of the mean number of controllers employed by each approach with the increase of the number of routes. We can observe that the needed number of controllers for every method initially increases with the number of routes up to a certain threshold where a plateau effect appears. In general, the performances of the observed methods are in line with the previous findings. This set of experiments demonstrates that the developed heuristics can produce controller sets that provide full controllability over various shapes and sizes of transportation networks. Additionally, we highlighted the importance of providing appropriate information, such as route information.

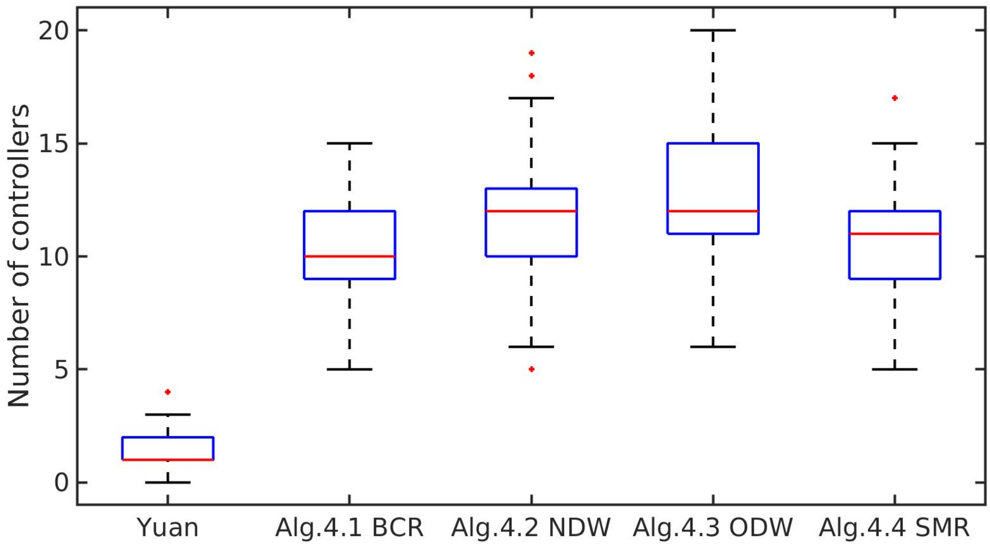


Figure 4.4: Number of controllers used to reach full controllability over 100 different networks of 25 nodes.

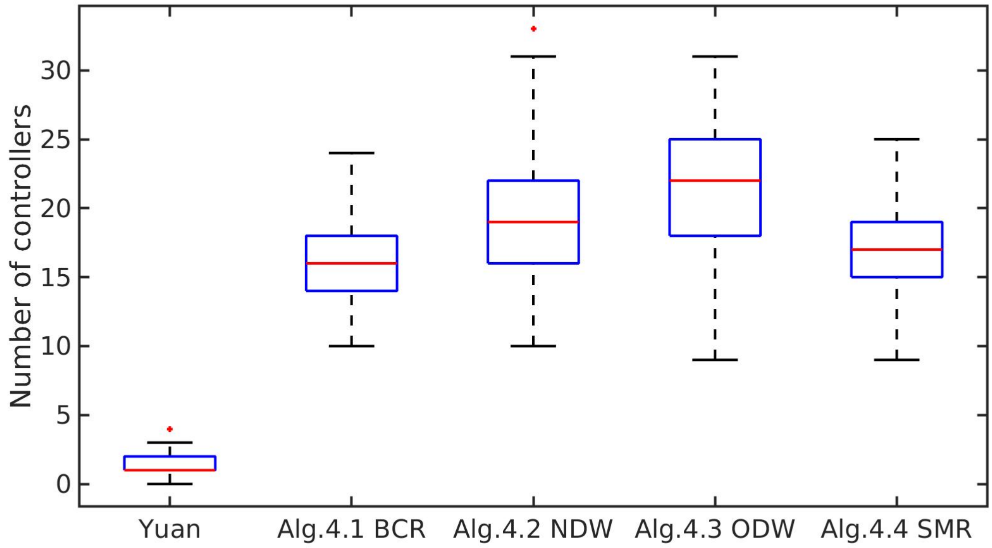


Figure 4.5: Number of controllers used to reach full controllability over 100 different networks of 49 nodes.

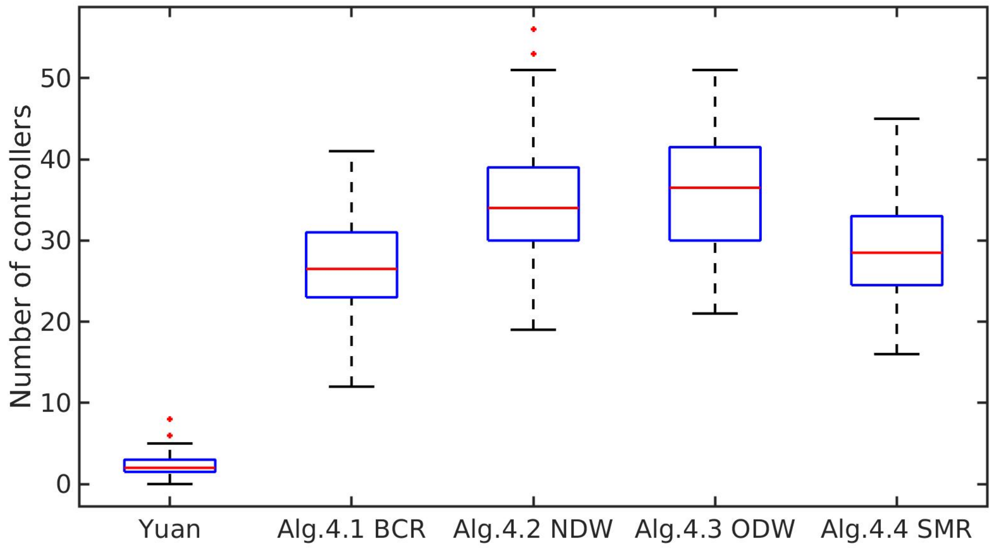


Figure 4.6: Number of controllers used to reach full controllability over 100 different networks of 100 nodes.

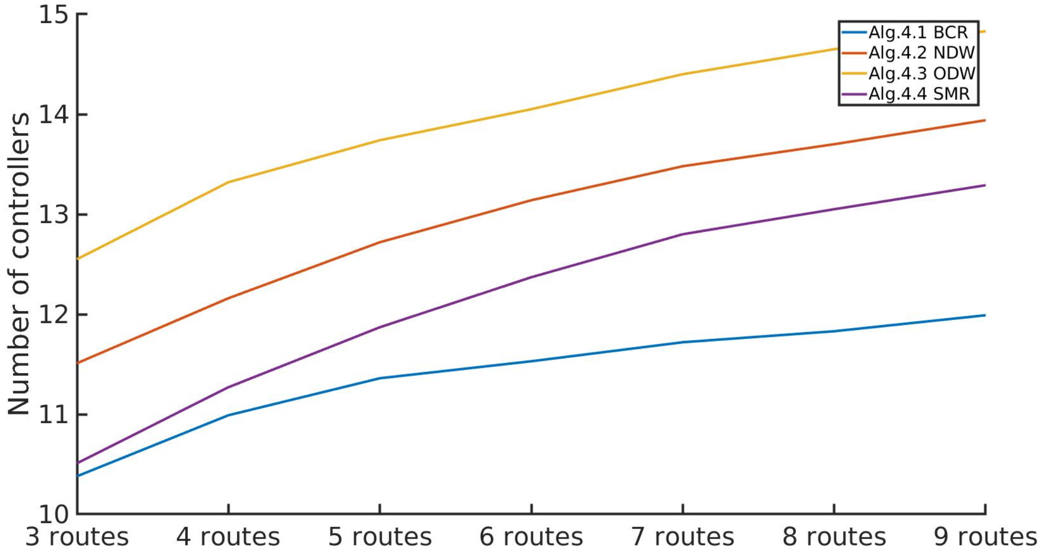


Figure 4.7: Evolution of the average number of controllers employed to reach full controllability for an increasing amount of routes k over 100 networks of 25 nodes.

4.5 Conclusion

In this chapter, we developed several greedy heuristics based on topological rules to solve the problem of identifying the minimal set of pricing controllers needed to control a transportation network fully. To validate the efficiency of the proposed approaches, we compared the respective methods over various network sizes and configurations. They produced adapted controller sets consistently satisfying the full controllability constraint while employing a number of controllers satisfactorily low. However, further work is required to refine the proposed heuristics, most notably to improve their scalability. As for now, it is bound by the necessity to compute the level of controllability at each iteration, which was shown to exhibit severe computational time and space complexity that does not allow the application of the proposed approaches on large networks commonly found in real transportation networks.

Chapter 5

Spanning tree approach

As demonstrated in the previous chapters, the main difficulty of locating an efficient set of controllers is to provide a scalable method. Therefore, this chapter proposes a study of transportation networks' topological properties and details a scalable method, as well as several variations, to locate an efficient set of pricing controllers.

This chapter is based on the following paper that is currently under review for the EURO Journal on Transportation and Logistics: X. Mazur, M. Rinaldi, R. Connors, and F. Viti, A topological approach for identifying pricing controller locations to ensure controllability of transportation networks.

5.1 Introduction

As presented previously, the current literature in controllability lacks a general and scalable method that can locate a set of efficient controllers on any kind and size of a transportation network. We demonstrated in the previous chapter that, based on the concept of level of controllability, it is possible to identify a set of pricing controllers that can fully control a transportation network. However, we also showed that the high computation complexity of this process makes the proposed methods not applicable on large networks. Therefore, in this chapter, we propose to study the topological properties of transportation networks and develop a scalable approach based on these properties to locate an efficient set of pricing controllers. For this chapter, we consider that pricing controller can only be located on links, such that they increase the cost of passing through this link for road users.

To develop such an approach, we first propose analyzing and defining some essential characteristics desirable for a set of controllers. Ideally, such controller set should be located so that a certain portion of trips between every origin-destination pair is controlled, such that every pair is at least covered by one controller. The chosen locations to be controlled should intercept as many flows as possible to maximize the number of flows that controllers can influence. Finally, controllers should be located such that the resulting set of controlled links are not linearly dependent, such as to avoid redundant controllers and capture the link flow dependencies emerging from route choice behavior. This set of intuitive rules are also guiding methodologies used for identifying sensor locations, such as the one proposed in Yang and Zhou [47].

This chapter aims to develop a method capable of generating a set of controllers that adheres as much as possible to these rules; however, we also seek to reduce the computation complexity and improve scalability. Therefore, we propose developing an approach based solely on the topological properties of transportation networks to reduce the associated computational load. Consequently, we do not consider any route flow information. Hence, we are unable to assess the exact capability of a controller set to capture flows and fully respect the flow capturing rule. Nonetheless, this work proposes a method to identify essential links to control without relying on flow information. Additionally, the experimental results section validates ex-post whether the produced controller set can capture and redirect flows and to which extent. This work also provides a solid basis to further develop methods and heuristics efficiently leveraging flow information on the identified topological-based optimal set of controller locations.

Chapter 2 provided a literature review of existing methods used for locating controllers and approaches used to identify sensor locations for observing a transportation network. We highlighted the lack of a scalable method for the problem of controller locations, but we also showed the similarities between this problem and the one of identifying counting sensor locations. Therefore we seek inspiration in approaches developed for locating counting sensors; more specifically, we focus on topological methods as they provide a scalable approach due to their overall low computational complexity. However, a substantial difference exists between sensors and controllers; the first ones are used only to observe a network and don't influence the current flow distribution. Controllers, however, are used for steering the flow distribution toward the desired state, thus changing the current state of the network. Therefore, while locations provided by approaches developed for observability can be used as locations for controllers, there is no guarantee that the resulting controller set can fully control the network. Thus applying observability methods for controllability is not a straightforward task.

For this purpose, we chose to apply and adapt the spanning tree methodology introduced by He [15] to the problem of identifying pricing controller locations. This approach meets our requirements, as it uses only topological information and exhibits significantly reduced computational complexity. In the next section, we begin by detailing the spanning tree method as originally described for the sensor location problem before developing and discussing which alterations were carried out to improve its performance at attaining full controllability.

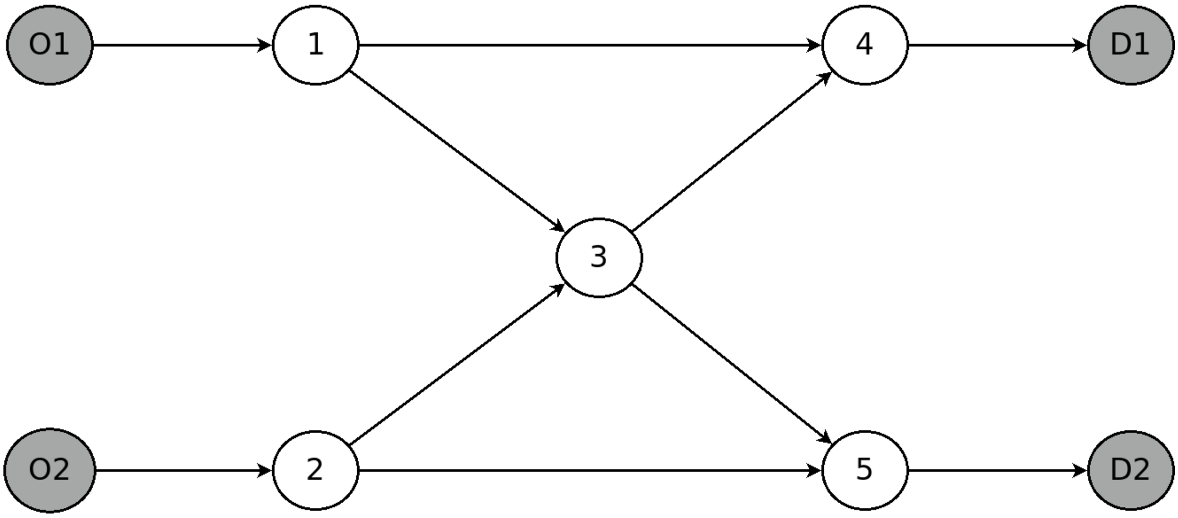
5.2 Methodology

Similarly as in previous chapters, a given transportation network is represented by a directed graph $G(N, L)$ comprising of a set N of nodes and a set $L : l \in L = (i, j)$, $i, j \in N$ of directed links connecting said nodes. Specific nodes are included to represent origin centroids, where traffic flows are produced, and destination centroids, where flows are attracted. Every link $l \in L$, is considered a potential location for a pricing controller. Note that the connector links required to map the centroids on the physical networks are not considered as part of L.

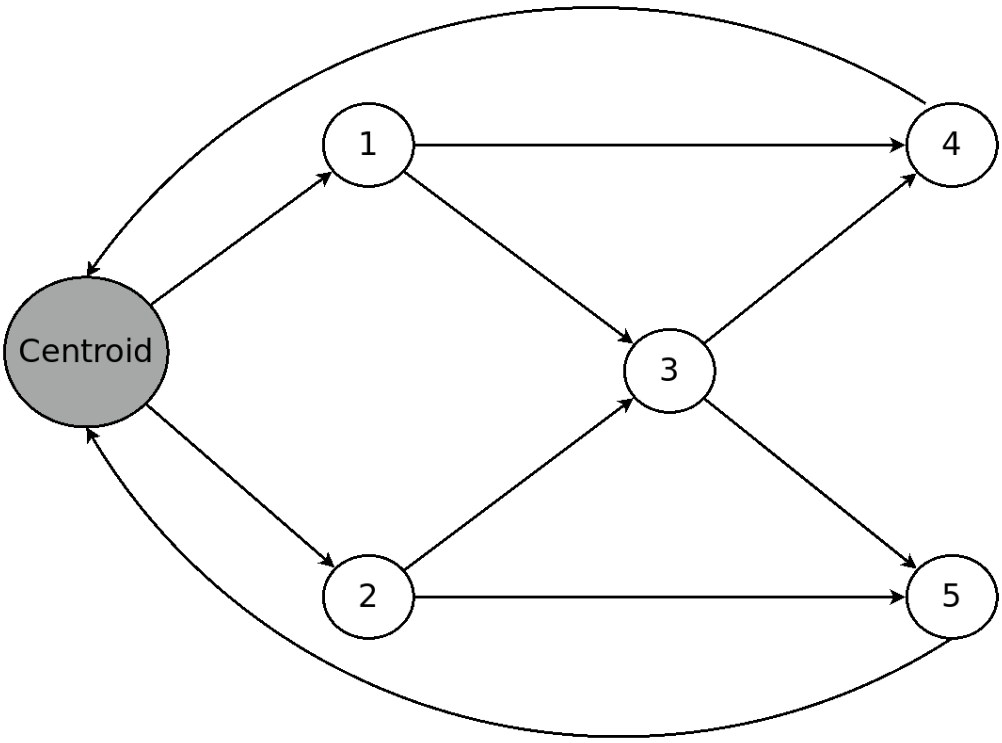
5.2.1 Spanning tree approach

To begin, we have to define the degree of a node as the number of links connected to the considered node. In this work, only origin and destination centroids have a degree of one on a transportation network. The first step of this approach consists of restructuring the network by replacing all origin and destination nodes with one unique virtual centroid. Thus, we can apply the flow conservation law on this node for produced and attracted flows the same way as for any other intermediate node. The resulting restructured network only differs in terms of origin-destination pairs. As this modified network is only employed for the sake of building spanning trees, and as we do not consider any flow information, the fundamental topological properties of the network can be regarded as invariant to this transformation. To illustrate this process, Figure 5.1a shows a simple network with two origins and two destination nodes, on which both origin and destination nodes are replaced by the same unique centroid as displayed in Figure 5.1b. Each outgoing link of the original origin nodes should change its starting point for the new virtual centroid without changing its direction. Likewise, each incoming link of the original destination nodes should change its ending point for the new virtual centroid.

The next step consists in searching for a spanning tree on the resulting transformed graph $\hat{G}$. First, we need to define what is a spanning tree and a tree on a graph. A tree can be defined as a connected acyclic undirected sub-graph of $\hat{G}$. A graph is considered acyclic if it doesn't contain any path in which a node is traversed multiple times. While considering a directed graph, we have to also consider a directed tree which is defined as a directed sub-graph of Ĝ whose underlying undirected graph is a tree. A directed tree T with links L_T and nodes N_T is called a spanning tree of graph $\hat{G}$ if $T \subset \hat{G}$, such that $N_T = \hat{N}$ and $L_T \subset \hat{L}$. Various algorithms have been developed to identify a spanning tree on the network; we choose to employ the one proposed by Kruskal et al. [21] for this work.



(a) Network with two origin and two destination nodes.



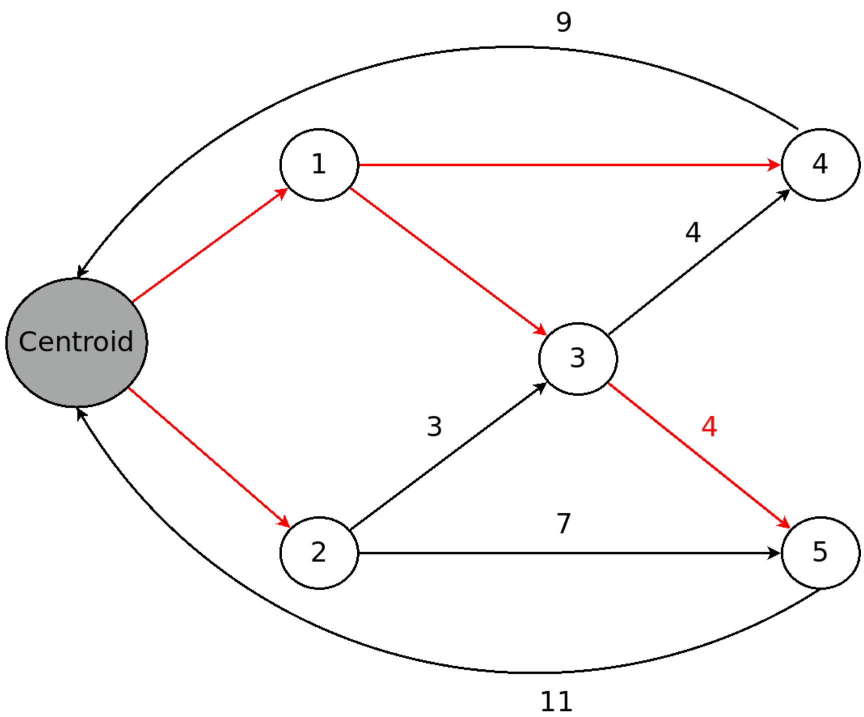
(b) Network with a unique virtual centroid.

Figure 5.1: Restructuring of a network with a unique centroid.

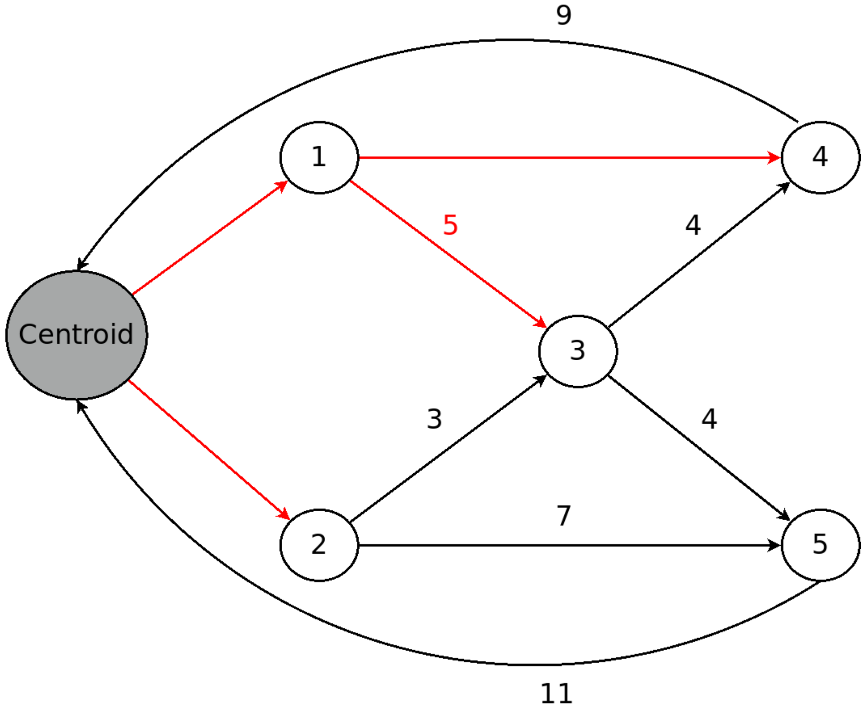
Based on the obtained spanning tree T, sensors are placed on every link that does not belong to the spanning tree T, such that $L_{obs} = \hat{L} \setminus \{L_T\}$. Therefore, this set of links is directly observed while the remaining unobserved links form a spanning tree, as presented in Figure 5.2. With this process, the network links are partitioned into two sets. One is composed of links directly equipped with a sensor, therefore directly observed and independent. One is composed of dependent links on which flows can be indirectly inferred based on the first set. The flows on the set on unobserved links can be inferred based on the flow conservation law on nodes, such that if at any node n, flow on all incident links are known except for one link $l^* \in L_T$, we can infer the flow on this link l^* based on the flows of the other observed links incident to this node. As the set of unobserved links form a spanning tree, we can exploit a fundamental property of spanning trees to identify a situation where link flows can be inferred. A tree always possesses at least one node that is a leaf of this tree, a leaf of a tree is defined as a node that has a degree of one; in our case, a leaf would be a node that has only one incident link that belongs to the considered spanning tree. Therefore, as the set of unobserved links form a spanning tree, we can always find a node n that is a leaf of the tree, thus corresponding to the previous situation where only one link l^* of the incident links to this node n is unobserved. The unknown flow on this link l^* can then be calculated based on the other incident link flows which are observed. Once the flow on link l^* is known, this link can be removed from the spanning tree, and we repeat this process on the newly formed tree. We illustrate this method on the same simple network on which we computed a spanning tree (Fig. 5.2); sensors are located onto all links that do not belong to the spanning tree. Therefore their flows are known (displayed by numbers in black in Figure 5.2a). If we consider the tree only, node 5 has a degree of one; thus, it is a leaf of this tree; consequently, the only incident link which is unobserved is the link (3,5). As shown on (Fig. 5.2a), the flow on link (3,5) can be deduced from the other observed incident links (2,5) and (5,C) as follows: $flow_{(3,5)} = flow_{(5,C)} - flow_{(2,5)} = 11 - 7 = 4$. Once the unobserved flow of a link is calculated, we can remove this link from the tree and start this process again by searching a new leaf on the new tree (Fig. 5.2b), and we repeat this process until all links flows are known.

This method only requires topological information in the form of a graph representing a transportation network and to compute a spanning tree; thus, it is easily applicable to any network type and size. Moreover, the computation complexity of determining a spanning tree is bound by $O(NlogL)$; thus, this approach does not require much computational time and is easily scalable. As this method can identify two sets of links with one dependent on the other, we propose applying this approach to the problem of controller locations. Based on the assumption that the set of not directly controlled links would be indirectly controlled by the set of links equipped with controllers. Similar to observability, a spanning tree leaf will provide an intersection where only one link will be uncontrolled. Thus, we assume that it can be controlled indirectly by the combined action of other controlled links and that this process can be applied consecutively to control all the spanning tree links that do not possess a controller. Additionally, as by definition, a spanning tree contains $|N| - 1$ links, we can infer that the number of controllers is always $|L| - (|N| - 1)$.

As discussed previously, domains of observability and controllability exhibit differences mainly due to controllers being used for impacting the flow distribution on the network, whereas sensors simply observe and do not modify the network state. Thus we need to evaluate the actual capability of the produced controller set to control the underlying network. For this purpose, we propose to use the controllability framework described in Chapter 2 and assess the capability of a pricing controller set to control a network by computing the associated level of controllability. However, this controllability framework, including the process of computing the level of controllability, is node-based, whereas the spanning tree approach provides link locations for pricing controllers. This implies that the set of controllers used should be located on nodes to be able to compute the level of controllability reached by the considered set of controllers. To resolve this incompatibility, we propose to use the principle of dual graph transformation introduced by Añez et al. [1] that we detailed in Chapter 2. The dual form of a graph still represents the same network but is a richer representation specifically concerning turning movements. In the dual graph form of a network, the



(a) Flow on link (3,5) can be inferred based on the observed flows of links (2,5) and (5,C).



(b) Flow on link (1,3) can be inferred based on the observed flows of links (2,3), (3,5) and (3,4).

Figure 5.2: A graph with a spanning tree displayed in red and observed flows.

dual nodes represent links of the primal graph, allowing us to locate pricing controllers on links of the primal graph to then transfer the controller locations over the dual graph nodes. Then we can compute the level of controllability reached by the set of controllers over the dual form of the network, the obtained level of controllability corresponds to the rank of the dual gramian matrix $rk(W_c(d))$ over the number of nodes in the dual network n_d, such that $LevelOfControllability = \frac{rk(W_c(d))}{n_d}$. If $rk(W_c(d)) = n_d$ then $LevelOfControllability = 1$ and full controllability is reached, we can also define full controllability as a level of controllability of 100%. According to Rinaldi and Viti [37] the level of controllability reached by a considered controller set represents the ratio of nodes in the dual network that is either directly or indirectly controlled by one or more controllers; therefore, the higher the level of controllability is, the more of the network is controllable. Thus, we should aim to maximize the level of controllability to guarantee that we can reach the maximum of the possible network states through the actions of the considered controller set.

To compare the efficiency of our approach to an existing one, we propose to use a method provided by previous work (Rinaldi and Viti, 2020 [37]) that was described in Chapter 2. In this work, the authors presented a node-based method relying on linear algebra to locate controllers on a network, which was adapted from the work of Yuan et al. (2013) [51]. They demonstrated that this approach was efficient but not exact. Based on the level of controllability principle, we can evaluate the spanning tree approach capability to produce a pricing controller set capable of fully controlling a transportation network. Additionally, we can compare the spanning tree approach efficiency with the Yuan's approach. For this purpose, we propose to apply the spanning tree method, as well as the modified Yuan's approach on a simple network (Fig. 6.1). Then we compare the level of controllability reached by each corresponding pricing controller set. In this network, links that connect origin and destination nodes are represented as grey dotted lines. These links are not considered potential locations for controllers since they are modeling artifacts and hence do not represent any real physical locations. Figure 6.1 displays the resulting controller sets produced by each approach; the chosen link locations for pricing controllers are displayed in blue in the figure.

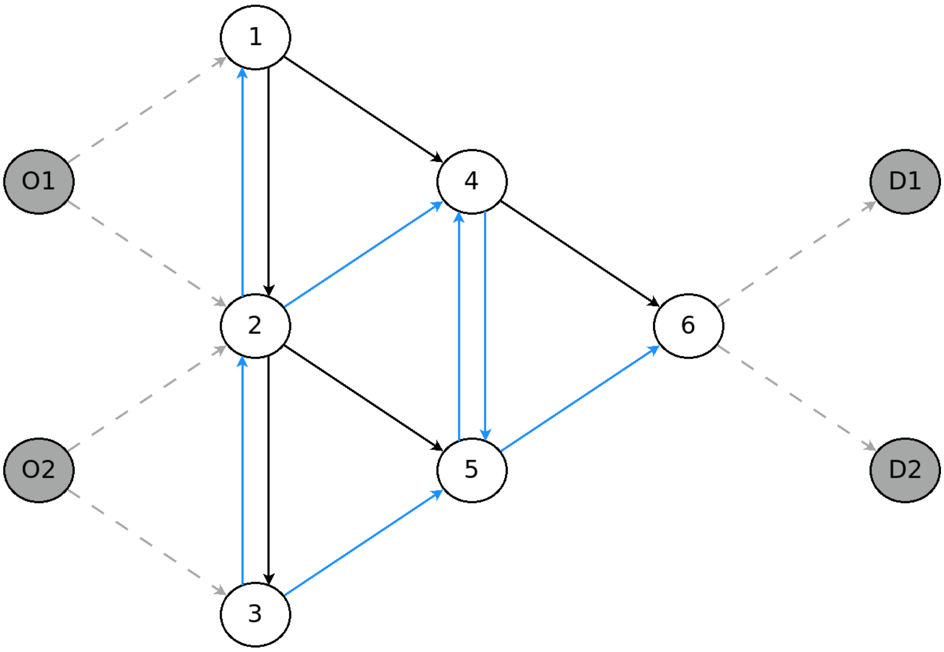
We can notice that the two approaches both produced a set of seven pricing controllers but at different locations. However, they reached a different level of controllability; the controller set produced by the modified Yuan's approach reached a level of controllability of 75 % while the spanning tree method was able to reach a higher level of controllability of 83 %. This demonstrates the importance of controller locations while highlighting that neither approach could reach full controllability even for a rather simplistic network. This implies that the direct application of the spanning tree method to the controller location problem does not guarantee full controllability over the network. Thus, leading to the necessity of adapting this approach to the specificities of the controller location problem to improve the resulting level of controllability.

5.2.2 Weighting schemes

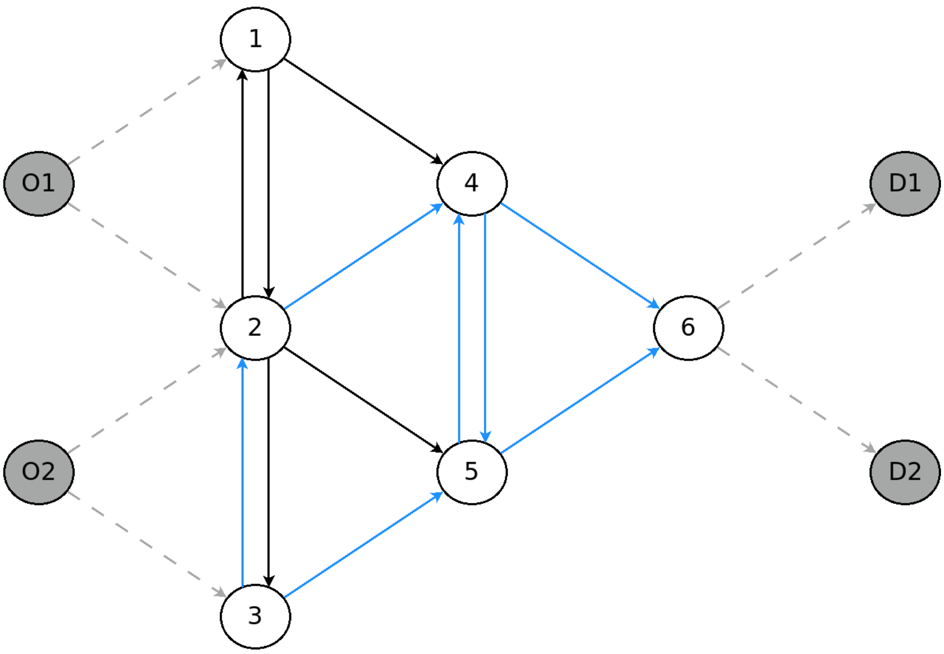
To adapt the spanning tree approach to the controller location problem, we propose to study if a specific spanning tree would produce a set of controllers capable of reaching higher levels of controllability than another. First, it is essential to notice that on a network that is not already a spanning tree, a spanning tree is not unique, and multiple different spanning trees can be found, as we propose to demonstrate in the following proof:

Given a connected graph $G(N, L)$, let $T(N, L_T)$ be a spanning tree in G such that $L - L_T \neq \emptyset$. Let a link $l' \in L - L_T$ such that adding l' to the spanning tree T will form a cycle C. Notice how a second spanning tree can be constructed by swapping a link $l'' \in L_T \cap C$ with l'. Therefore, T is not a unique spanning tree for G and multiple spanning trees can be found.

Additionally, as presented in Pieper (2008) [33], the number of possible spanning trees for a given network graph can be enumerated, with an upper bound of $|N|^{(|N|-2)}$ in the specific instance of a complete graph K_N. Therefore, we propose investigating the possibility of steering the selection of a spanning tree such that the selected controller locations favor desirable characteristics, like origin-destination pairs covering and flow interception. To steer the selection of a spanning tree, we propose to apply various weighting schemes on



(a) Controller locations chosen by the spanning tree approach (in blue).



(b) Controller locations chosen by the modified Yuan's approach (in blue).

Figure 5.3: Comparison of controller locations, blue links represent links equipped with a pricing controller.

links, thus seeking the minimum spanning tree of the resulting weighted graph. The proposed weighting schemes will reduce the amount of possible minimum spanning trees, thus reducing the number of possible controller sets. However, the uniqueness of the minimum spanning tree found cannot be generally guaranteed, even under weighing assumptions. In the following, we detail various heuristic approaches for applying weights on a network that each generates a different minimum spanning tree and thereby a different set of controller locations.

Alg. 5.1 For comparison, as a first algorithm, we apply the spanning tree approach as it is used for the sensor location problem, imposing the same unitary weight on every link:

$$l_{(weight)}^{Alg5.1} = 1$$

Alg. 5.2 For this second weighting scheme, we prioritize the direct control of links close to the origin nodes. As links contained in the minimum spanning tree are the set of indirectly controlled links, we want links close to origin nodes to have the highest weights to be more likely to be directly controlled. For this purpose, we employ the topological distance, defined as the distance, expressed in the number of links traversed, from a considered link to the closest origin node following the shortest path, then we apply the opposite of this distance as a weight to this link. Thus links close to origin nodes will bear the highest weights and will be less likely to be selected by the spanning tree. The intuition behind this approach is that by favoring the control of links close to origin nodes, we will cover all origin-destination pairs and capture route flows at their beginning before they start splitting onto different sub-routes downstream. Therefore, the resulting control locations are expected to imitate a pricing cordon around the origins. With nbO as the number of origin nodes and $dist(l, O_1)$ the distance between l and O_1 following the shortest path, we can define each link weight as follows:

$$l_{(weight)}^{Alg5.2} = -min(dist(l, O_1), dist(l, O_2), ..., dist(l, O_{nbO}))$$

Alg. 5.3 With a similar aim to the previous algorithm, we propose to employ the average distance between a considered link and every origin node instead of using the distance from a given link to the closest origin node. With this approach, we expect directly controlled links to cover all origin nodes and maximize the number of flows captured due to their positions being close to multiple origin nodes. Using the notation above, we can define each link weight as follows:

$$l_{(weight)}^{Alg5.3} = -\frac{dist(l, O_1) + dist(l, O_2) + ... + dist(l, O_{nbO})}{nbO}$$

Alg. 5.4 Inspired by work on weighting schemes for resilience in complex networks (Yang et al., 2009 [48]), we propose to use the sum of the degrees of both starting and ending nodes of a link as its weight. The degree of a node $n \in N$ is the number of links $l \in L$ that have for starting or ending node the node n. With this weighting rule, we will prioritize the control of links incident to nodes with a high number of connections. Thus, we expect those links to be often used and capture a more significant amount of flows than other links. This rule will intuitively lead to high-quality solutions in scale-free networks. Each link weight can be described as follows, where $l_{startNode}$ and $l_{endNode}$ being respectively the starting node and ending node of link l:

$$l_{(weight)}^{Alg5.4} = degree(l_{startNode}) + degree(l_{endNode})$$

Alg. 5.5 To exploit the topological properties of transportation networks, we propose to employ metrics typically used in network topology analysis to derive other weighting schemes. Specifically, we consider centrality measures such as betweenness centrality, as introduced by Freeman et al. (1977) [10], which is used to identify nodes that have the most important influence on other nodes in a network. To prioritize such central locations for links to be controlled in transportation networks, we propose to use the edge betweenness measure (Girvan et al., 2002 [11]). This measure for one link is defined as the number of shortest paths connecting each node pair that go through this link.

Controllers will therefore be preferentially placed on links that are likely to be the most often used by road users; thus, controllers are more likely to capture a high amount of flows. To compute edge betweenness we first need to compute the shortest path $sp_{i,j}$ between each node pair (i, j) and to define the variable $xl_{i,j}$ such that $xl_{i,j} = 1$ if link $l \in sp_{i,j}$ and $xl_{i,j} = 0$ otherwise. Therefore we can define each link weight as follows:

$$l^{Alg5.5}_{(weight)} = \sum_{i=1}^{|N|} \sum_{j=1}^{|N|} xl_{i,j}$$

Alg. 5.6 Based on the previous weighting scheme, we propose including additional information specific to transportation networks to have more accurate controller locations. Instead of using the number of shortest paths connecting each node pair that go through a link as a weight for this link, we use the number of routes connecting each origin-destination pair that go through the considered link. By following this weighting scheme, controllers will be located in priority on links considered central and that connect origin-destination pairs, thus being more likely to intercept flows. To be able to compute a route set on any type and size of networks, we decided to use the k-shortest paths algorithm (Yen, 1971 [49]) to compute a set of k routes between each origin-destination pair. For each route $r \in R$ we can define xl_r such that $xl_r = 1$ if link $l \in r$ and $xl_r = 0$ otherwise. Thus each link weight can be described as follows:

$$l^{Alg5.6}_{(weight)} = \sum_{r=1}^{|R|} xl_r$$

In order to assess if any of the proposed variations of the spanning tree approach is capable of providing a more efficient solution, the following section investigates how each approach influences the search for a spanning tree and the efficiency of the corresponding set of locations selected for controllers.

5.3 Experimental setup

To provide a relevant experimental setup, we propose to generate a large set of networks with various shapes. For this purpose, we employed the graph generator introduced in Mireles de Villafranca et al. (2019) [44] that was detailed in chapter 4. With it, we are able to generate arbitrarily randomized networks bearing sufficient resemblance to urban transportation networks. Each generated network possesses features related to transportation networks; specifically, each network contains a given amount of origin-destination nodes. In this work, we decided to introduce six origin and six destination nodes on each network; each origin node is after that connected to all destinations by a collection of routes. For this experimentation, the number of routes per origin-destination pair is set to $k = 3$, and each route set is computed following a k-shortest path algorithm (Yen [49]).

5.3.1 Topological validation

To compare the efficiency of the respective methods, we apply each proposed approach and evaluate the efficiency of the resulting controller sets by computing the level of controllability yielded by these controller sets. While considering the topological approach of the spanning tree, we compute the corresponding level of controllability while considering the complete network, such as n is the number of nodes in the entire network. However, the process of computing the level of controllability exhibits a severe space complexity, bounded by $O(N^4)$, which implies the necessity of storing increasingly large variables during the procedure as the size of networks increases. Therefore, with our current setup, we cannot compute the level of controllability yielded by a controller set on a network containing more than 50 nodes. Thus, we chose to generate networks ranging from 9 to 49 nodes for this first experimental phase. For each chosen network size, we generated 100 instances, each featuring a different random seed, thus producing networks with similar characteristics but different shapes. In each instance, we apply the different variations of the spanning tree approach and the modified method of Yuan to obtain the corresponding candidate controller sets. Then the efficiency of each respective controller set is assessed by computing the level of controllability reached by each controller set on the considered network.

To interpret the obtained results, we first propose to inspect the performance of the various approaches over 100 networks variations of 49 nodes. The obtained results are presented in the form of a boxplot (Fig. 5.4); each box represents a method, while the y-axis shows the level of controllability obtained over the 100 instances. As we can observe in the figure, the various spanning tree-based approaches managed to reach in general a level of controllability over 90%, which is statistically slightly superior to the average level of controllability obtained by the modified Yuan's method, that generally reached a level of controllability around 87%. These results indicate that the methods based on the spanning tree can reach a satisfying level of controllability and tend to perform better than the existing approach on small networks with less than 50 nodes. Additionally, we can observe that minimum spanning tree approaches based on weighting schemes (Alg. 5.2-5.6) generally reached a slightly higher level of controllability than the simple spanning tree method (Alg. 5.1). This indicates that all feasible spanning trees are not equal and that it is possible to identify a spanning tree that will produce a more efficient set of pricing controllers. Thus, showing the importance and necessity of adapting the spanning tree approach to the specificity of the controller location problem and the proposed weighting schemes provides a promising step in this direction. However, even if we can distinguish some approaches that performed better than others between the minimum spanning tree-based approaches, mainly the algorithms 5.2, 5.3 and 5.5, the difference in terms of the level of controllability reached is not significant enough to clearly establish that one weighting scheme should be preferably employed over the others. This trend can also be identified for other network sizes, as detailed in Table 5.1.

Let's consider the number of controllers employed by the different methods as an additional performance indicator. We can observe that the minimum spanning tree-based approaches used more controllers than the modified Yuan's approach. This can justify their slightly better performances in terms of the level of controllability obtained. We can also remark that all the spanning tree-based approaches produced the same number of controllers, which is

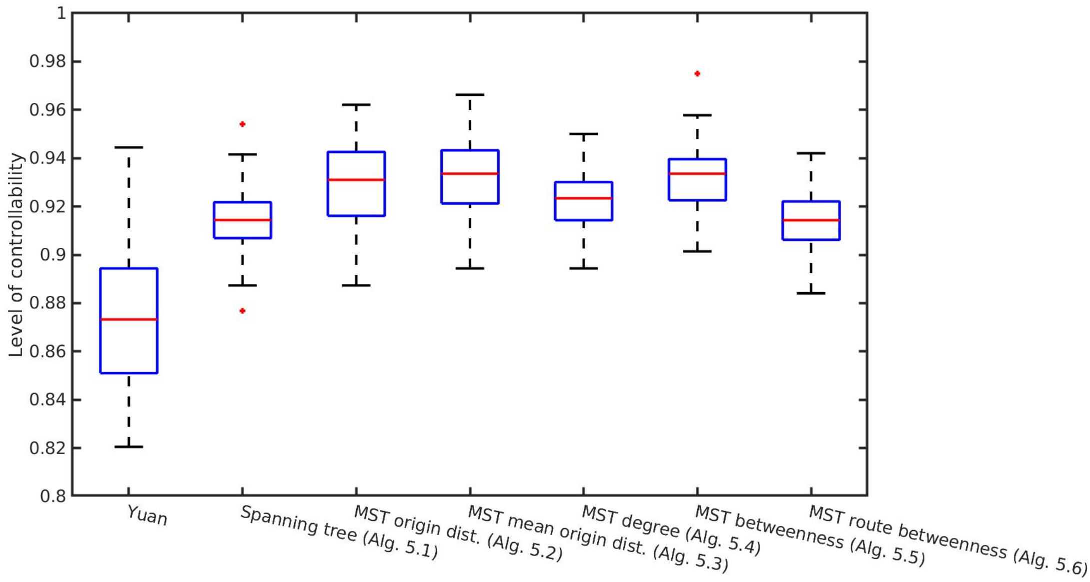


Figure 5.4: Level of controllability reached by various approaches on a set of 100 networks of 49 nodes.

expected as, by definition, the chosen number of locations depends entirely on the size of the spanning tree, which results in a constant number of controllers of $|L| - (|N| - 1)$. This value should be regarded as an upper bound for the minimum number of controllers needed to reach a satisfying controllability level. We can also observe that the experienced memory requirement is far more significant for the modified Yuan's approach, as expected, while the spanning tree approaches exhibit the desired scalability capabilities.

With this set of experiments, we showed the capability of spanning tree-based approaches to control small transportation networks efficiently. To complete this experimental study, we want to assess if the previously employed approaches can efficiently impact larger networks. For this purpose, we propose estimating the actual capability of produced controller sets to redirect flows such that the network infrastructure is used more efficiently. In what follows, we employ static assignment-based simulation to investigate whether controller sets produced are capable of actually reducing the total time spent by road users in the network.

Network size (in nodes)	Methods	Level of controllability	Average number of controllers	Average computation time	Average memory requirement
9	Yuan's	91.6 %	15.92	0.1727 s	11 MB
	Spanning tree (Alg. 5.1)	91.2 %	16.88	0.0021 s	0,0008 MB
	MST Origin dist. (Alg. 5.2)	94.6 %	16.88	0.0038 s	0,0008 MB
	MST Mean origin dist. (Alg. 5.3)	96.0 %	16.88	0.0063 s	0,0008 MB
	MST Degree (Alg. 5.4)	93.6 %	16.88	0.0022 s	0,0008 MB
	MST Betweenness (Alg. 5.5)	96.9 %	16.88	0.0106 s	0,0008 MB
	MST Route betweenness (Alg. 5.6)	93.6 %	16.88	0.0022 s	0,0008 MB
16	Yuan's	88.9 %	25.70	0.7882 s	91 MB
	Spanning tree (Alg. 5.1)	91.9 %	27.72	0.0026 s	0,0023 MB
	MST Origin dist. (Alg. 5.2)	94.5 %	27.72	0.0057 s	0,0023 MB
	MST Mean origin dist. (Alg. 5.3)	95.7 %	27.72	0.0091 s	0,0023 MB
	MST Degree (Alg. 5.4)	93.5 %	27.72	0.0028 s	0,0023 MB
	MST Betweenness (Alg. 5.5)	95.7 %	27.72	0.0189 s	0,0023 MB
	MST Route betweenness (Alg. 5.6)	93.3 %	27.72	0.0027 s	0,0023 MB
25	Yuan's	88.7 %	45.22	3.9528 s	418 MB
	Spanning tree (Alg. 5.1)	91.7 %	48.28	0.0044 s	0,0054 MB
	MST Origin dist. (Alg. 5.2)	93.6 %	48.28	0.0084 s	0,0054 MB
	MST Mean origin dist. (Alg. 5.3)	94.4 %	48.28	0.0154 s	0,0054 MB
	MST Degree (Alg. 5.4)	92.7 %	48.28	0.0046 s	0,0054 MB
	MST Betweenness (Alg. 5.5)	94.2 %	48.28	0.0441 s	0,0054 MB
	MST Route betweenness (Alg. 5.6)	92.5 %	48.28	0.0045 s	0,0054 MB
36	Yuan's	87.0 %	63.93	12.953 s	1534 MB
	Spanning tree (Alg. 5.1)	91.5 %	69.99	0.0061 s	0,0109 MB
	MST Origin dist. (Alg. 5.2)	93.3 %	69.99	0.0120 s	0,0109 MB
	MST Mean origin dist. (Alg. 5.3)	93.6 %	69.99	0.0219 s	0,0109 MB
	MST Degree (Alg. 5.4)	92.6 %	69.99	0.0066 s	0,0109 MB
	MST Betweenness (Alg. 5.5)	93.6 %	69.99	0.0801 s	0,0109 MB
	MST Route betweenness (Alg. 5.6)	91.8 %	69.99	0.0062 s	0,0109 MB
49	Yuan's	86.3 %	86.62	44.574 s	4748 MB
	Spanning tree (Alg. 5.1)	91.4 %	96.10	0.0110 s	0,02 MB
	MST Origin dist. (Alg. 5.2)	93.0 %	96.10	0.0193 s	0,02 MB
	MST Mean origin dist. (Alg. 5.3)	93.3 %	96.10	0.0353 s	0,02 MB
	MST Degree (Alg. 5.4)	92.2 %	96.10	0.0109 s	0,02 MB
	MST Betweenness (Alg. 5.5)	93.2 %	96.10	0.1633 s	0,02 MB
	MST Route betweenness (Alg. 5.6)	91.4 %	96.10	0.0105 s	0,02 MB

Table 5.1: Comparison of various approaches performance.

5.3.2 Static assignment validation

The objective of the experiment carried in this subsection is to assess the capability of the various controller sets produced by the previously presented approaches to improve the efficiency of a transportation network by reducing the total time spent by road users on the network. For this purpose, we employ static assignment-based simulation to generate flows on transportation networks and estimate the capability of a considered controller set to redirect flows. As in the work of Rinaldi and Viti [37], we employ BPR cost functions for each link, with link cost parameters dependent on the relative location of the given link. As every generated network is subdivided into three concentric zones (as shown in figure 4.1), the free-flow speeds $v_{f,l}$ and capacities c_l of all links are linearly interpolated in the interval ranging from $v_{f,l} = 90km/h$, $c_l = 1800$ representing the most distant zone and $v_{f,l} = 50km/h$, $c_l = 1500$ representing the most central zone, such as to reproduce the transition between outside areas to the city center. The length of a link $l_{i,j}$ is the direct result of the Euclidean distance between nodes i and j.

To generate flows, we employed the method of successive averages to reproduce the condition of static deterministic user equilibrium, which can be defined as an equilibrium that optimizes the time spent on the network for each individual road user. We chose this equilibrium for its simplicity and yet sufficient degree of representation for modeling the network-wide dynamics we are interested in. Moreover, as an objective to reach for controller sets, we compute an assignment of flows corresponding to the system optimum, which can be defined as the state where the total time spent by all road users is minimized. To compute it, we used the method of successive averages with an all-or-nothing assignment over a fixed set of routes per origin-destination couple determined through the K-Shortest Path algorithm, with explicit consideration of link marginal costs, under the assumption of cost function separability. In order to analyze the amount of resource wasted through users' selfish behavior that controller sets will attempt to minimize, we can define the price of anarchy as presented by Roughgarden [39] as the ratio of user equilibrium to system optimum such that for a network G, $PoA(G) = \frac{TTS_{UE}(G)}{TTS_{SO}(G)}$ with $TTS_{UE}(G)$ being the total time spent by road users on network G under user equilibrium and $TTS_{SO}(G)$ as the total time spent by road users under system optimum.

Similar to the previous experimental setup, we employed the previously presented graph generator to produce a set of randomly generated networks. For this experiment, we propose increasing the size of tested networks; thus, we generate graphs ranging from 64 nodes up to 256 nodes. For each network sizes selected, we produce 100 randomized networks. To produce situations in which controllers have the possibility to improve the current situation, we want to avoid situations where the demand is too low for the network, where no congestion is produced, but also cases where the complete network is congested and no control action can potentially modify the current network state. For this purpose, we aim at generating networks bearing a Price of Anarchy as high as possible. Thus, due to the inherent hysteresis of the Price of Anarchy with respect to network demand, we explored several combinations of randomized network demand levels, with the only constraint that every origin-destination pair in the network has the same demand level, to identify viable candidates for which a robust control action is required. This preconditioning is especially necessary under the assumption of steady-state static assignment; recent works have, however, shown how very large Price of Anarchy might, in fact, be quite common in dynamic settings (Belov et al. [2]). To assure that a certain Price of anarchy is present on networks, we selected only candidates that exhibited a Price of Anarchy superior or equal to 1.03, thus instances where User Equilibrium steady-state condition exhibits a Total Cost value at least 3% higher than that of System Optimum.

We apply the previously presented methods for each randomly generated network to obtain the corresponding candidate controller sets. Based on the topological validation subsection results, we chose not to apply minimum spanning tree approaches based on degree and route betweenness weighting schemes. The main reason is that as presented in Table 5.1, these two approaches constantly reached lower levels of controllability than other weighting schemes. The poor results of the minimum spanning tree based on route betweenness can be explained by the dependence of this method on the route set employed. In the case of this experiment, we used the k-shortest paths algorithm to generate routes between origin-destination pairs which tends to produce route sets lacking information. Therefore, we can expect that using a route set containing more information will provide a higher level of controllability; however, in this work, we will not focus on route set generation.

Similar to the process of computing the level of controllability, the modified Yuan's algorithm also exhibits such considerable space complexity so that we cannot directly apply this algorithm on the considered large networks. To be able to apply the modified Yuan's algorithm, we propose to use a network partitioning process to obtain multiple sub-networks, small enough to apply the method individually on each sub-network, such that we use the combination of all controller sets found on each sub-network for the complete network. To obtain such partitioning of the network, we propose to use a graph clustering method based on the k-means algorithm (Hartigan et al. [14]) to divide each network into a set of sub-networks. For this experiment, we chose a number of sub-networks defined as $K = \lceil |N|/50 \rceil$ such that each sub-network has a size inferior to 50 nodes, thus allowing the computation of the modified Yuan's method. For the sake of comparison, we add a randomly generated controller set, in which the number and the locations of controllers are randomly determined. Once controller sets corresponding to each method are computed for a considered network, we determine the toll level of all pricing controllers for each candidate controller set. For this purpose, we adopted the optimization framework as well as the objective function for total cost minimization employed in Rinaldi et al. [36], which is based on the Quasi-Newton Broyden-Fletcher-Goldfarb-Shanno (BFGS) method to perform non-linear optimization. Thus toll levels are determined with the aim of redirecting flows to minimize the total time spent by road users and steer the network state toward system optimum. However, due to the non-linear nature of the Total Cost objective function with general BPR cost functions, we cannot guarantee that the toll levels provided by the optimizer are optimal.

To evaluate and compare the results obtained, we define the variable ρ for a network G

and a controller set cs as follows:

$$\rho(G, cs) = \frac{(TTS(G, cs) - TTS_{SO}(G))}{(TTS_{UE}(G) - TTS_{SO}(G))} \tag{5.1}$$

With $TTS(G, cs)$ being the total time spent by road users resulting from the action of the considered controller set cs on network G. Therefore $\rho(G, cs) = 1$ when the total time spent by road users on the network G under the action of the controller set cs corresponds to the user equilibrium $TTS_{UE}(G)$ previously computed for this network. And $\rho(G, cs) = 0$ when the total time spent by road users corresponds to the system optimum $TTS_{SO}(G)$; thus, we aim at minimizing this value to steer the network toward system optimum.

To interpret the obtained results, we propose first to examine the results obtained by the various methods over 100 networks of 121 nodes. As displayed in Figure 5.5, all the proposed approaches managed to reach better performances than those obtained while using a randomly generated set of controllers. This result shows that using a guided approach to locate controllers on a transportation network will indeed produce a more efficient controller set. Interestingly, all approaches, either based on a minimum spanning tree or the modified Yuan's algorithm, reached similar ρ values. Table 5.2 displays the average ρ value reached by each method for each network size over all the instances generated, as well as the percentage of links equipped with a controller to reach these values. We can observe similar trends as previously reported, specifically for networks of 121 and 256 nodes.

We also propose considering the number of controllers employed as an additional criterion to compare the efficiency of the proposed approaches. Figure 5.6 displays each controller set result in terms of ρ value over the percentage of links in the whole network equipped with a controller to reach this value. Since all minimum spanning tree methods performed similarly and always use the same number of controllers, and to ensure clarity in the graphical representation, we chose to only display the results obtained by the spanning tree approach without weighting schemes (Alg. 5.1). Each circle represents a solution obtained using the spanning tree approach, whereas each triangle represents a solution obtained with the modified Yuan's method. Ideal solutions will attempt to minimize the number of controllers employed and minimize the resulting ρ value; thus, the most efficient solutions will be located on the bottom left corner in the figure. As we can observe in the figure, the average percentage of links equipped with a controller for the spanning tree approach is around 64 percent, whereas, for the modified Yuan's method, it is around 70 percent. Thus the spanning tree approach performs equally in terms of total time spent reduction but generally employs fewer controllers on the tested set of large networks. We can also observe that the standard deviation for the modified Yuan's approach is higher than for the spanning tree approach in terms of the percentage of links equipped with a controller, thus spanning tree-based methods are more consistent. This finding might appear in juxtaposition with the conclusions of the first experimental results; however, it's important to remark here that the items being compared are slightly different in this setting. Due to scalability concerns, the modified approach of Yuan becomes quickly unfeasible with growing network sizes, and we postulate that decomposition in smaller, tractable sub-networks might be at the root of this loss of quality.

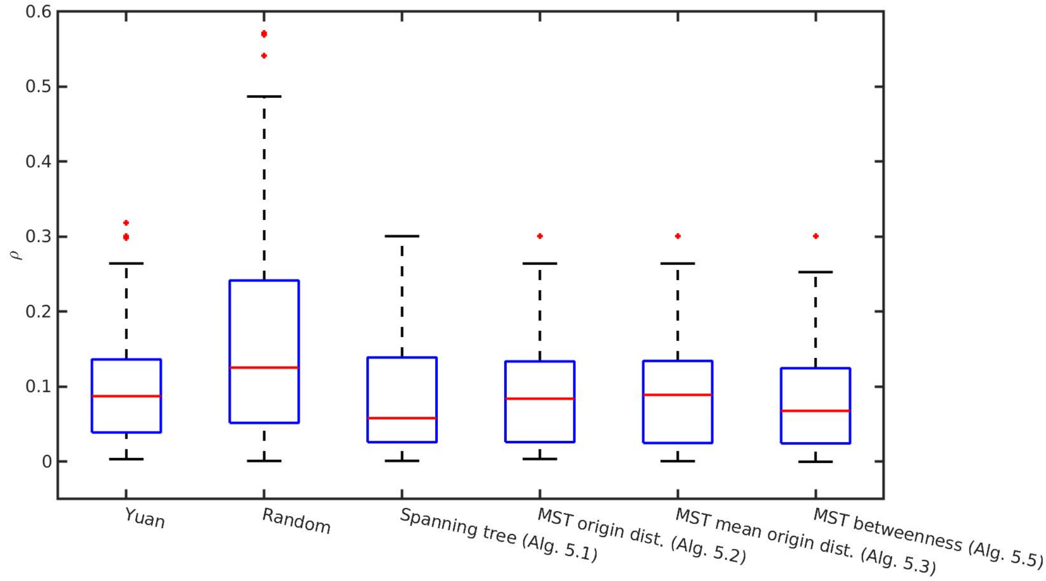


Figure 5.5: Reduction of total time spent on an ensemble of 100 networks of 121 nodes.

Network size (in nodes)	Methods	Average ρ	Average percentage of links with controllers
64	Yuan's	0.068	67.28 %
	Random	0.171	44.59 %
	Spanning tree (Alg. 5.1)	0.076	66.88 %
	MST Origin dist. (Alg. 5.2)	0.089	66.88 %
	MST Mean origin dist. (Alg. 5.3)	0.071	66.88 %
	MST Betweenness (Alg. 5.5)	0.072	66.88 %
121	Yuan's	0.097	71.84 %
	Random	0.187	48.28 %
	Spanning tree (Alg. 5.1)	0.089	66.79 %
	MST Origin dist. (Alg. 5.2)	0.090	66.79 %
	MST Mean origin dist. (Alg. 5.3)	0.090	66.79 %
	MST Betweenness (Alg. 5.5)	0.085	66.79 %
256	Yuan's	0.109	69.49 %
	Random	0.177	45.50 %
	Spanning tree (Alg. 5.1)	0.109	66.64 %
	MST Origin dist. (Alg. 5.2)	0.119	66.64 %
	MST Mean origin dist. (Alg. 5.3)	0.113	66.64 %
	MST Betweenness (Alg. 5.5)	0.116	66.64 %

Table 5.2: Comparison of various approaches capability to reduce total time spent.

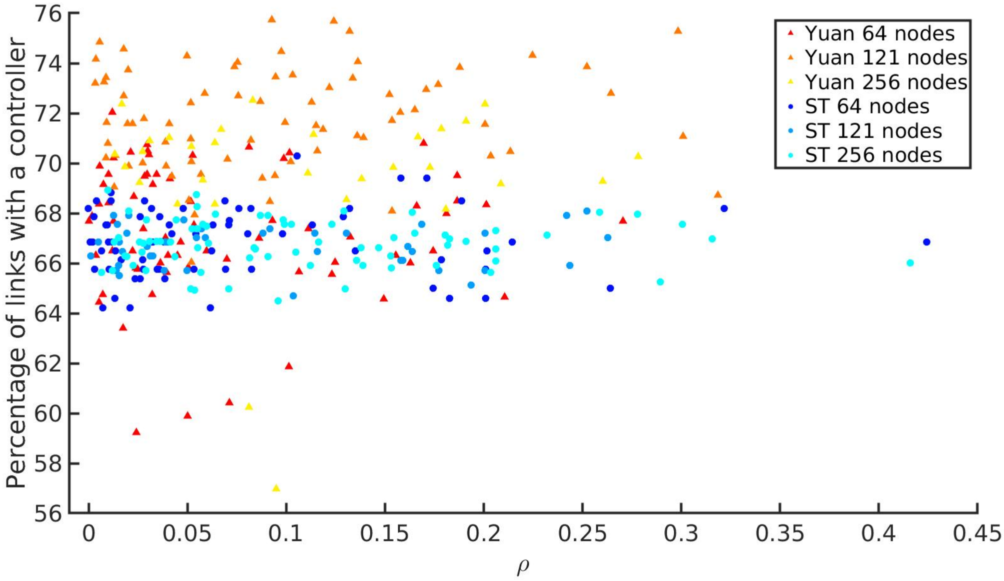


Figure 5.6: Reduction of total time spent over percentage of links equipped with a controller on 100 networks for each approach.

Finally, we assume that the efficiency of the spanning tree approach might effectively depend on how (dis)similar the considered network is to a spanning tree. Therefore, we propose to study the impact of node connection sparsity over the number of controllers provided by both spanning tree-based methods and the modified Yuan's approach. To modify the connectivity of networks produced with the previously presented β-skeleton graph generator, we altered the β parameter, which modifies the likelihood of a node being connected to surrounding nodes during the computation of the network, directly impacting the average node degree in the network. We propose investigating the impact of the β parameter and the corresponding average node degree on the solutions obtained over variably-sized networks, ranging from 100 nodes to 729 nodes. A set of one hundred randomly generated networks is computed for each network size. As we observe in Figure 5.7, the obtained results are relatively homogeneous over the different network sizes, and we can notice that the curves representing the two approaches intersect around a β value of 1.3, which corresponds to an average node degree around 6.5. In this experiment, each link direction is counted separately for the node degree, and only internal nodes are considered, that is, all nodes except origin and destination centroids. In general, we can deduce that the modified Yuan's approach requires fewer controllers than spanning tree-based approaches for a β value under 1.3. In contrast, the opposite holds with a β value higher than 1.3, indicating that spanning tree-based approaches will provide a solution for a lower cost, in terms of the number of controllers employed, in less densely connected networks. Previous experiments were done with a β of 1.5, which corresponds to a situation where minimum spanning tree-based approaches perform naturally better. In Osaragi et al. [31] the authors defined that to bear a maximal topological resemblance to existing street networks, the β value should be chosen between 1.1 and 1.5, therefore suggesting that the choice of algorithm to design optimal controller locations might be dependent on the specific network infrastructure.

With this set of experiments, we showcased that spanning tree-based approaches can provide a set of pricing controllers that can considerably reduce the total time spent by road users. More efficiently than with a randomly generated set of controllers and to an equal level than the proposed modified Yuan's algorithm. Additionally, spanning tree-based approaches are easier to apply and more scalable than existing approaches. The test cases also revealed that spanning tree-based approaches are impacted by networks configuration and will provide solutions at a lower cost, in terms of the number of controllers employed, on networks containing nodes with a low degree. Thus it provides an overall more efficient method to locate controllers on sparsely connected transportation networks. This property leads us to consider this approach as most desirable on larger, regional-scale networks, whose size and sparsity indeed fit the criteria for higher solution quality.

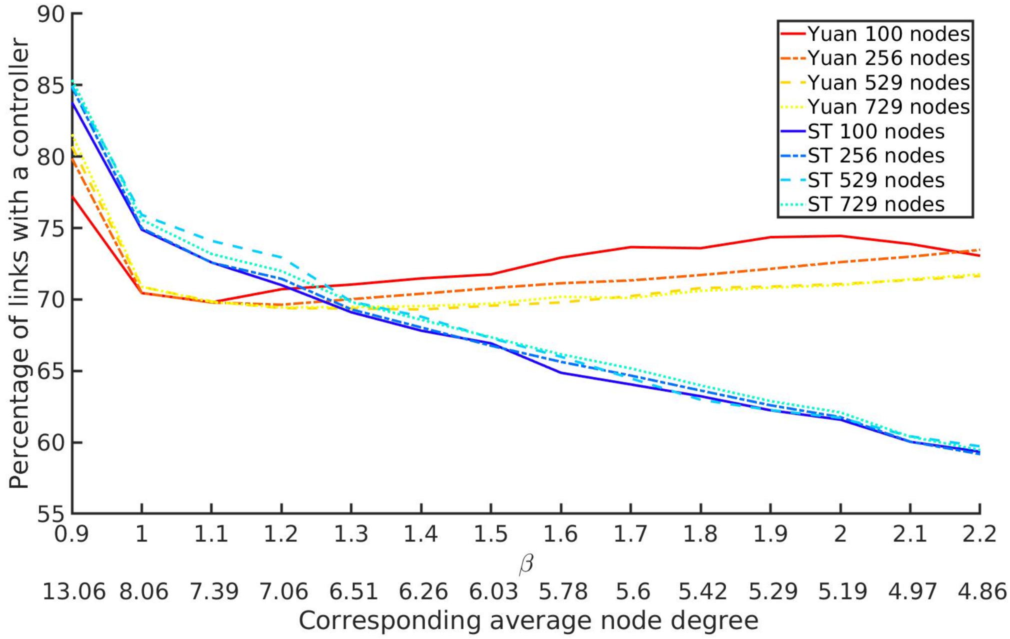


Figure 5.7: Change in number of controllers required as network connectivity reduces (i.e. β increases).

5.4 Conclusion

In this chapter, we aimed to provide a scalable approach capable of locating a set of controllers that can efficiently control the underlying transportation network while minimizing the number of controllers employed. To achieve this goal, we propose to employ the spanning tree method, which was originally developed for the sensor location problem. To adapt this method to the specificity of the controller location problem, we developed various weighting schemes to steer the selection of a controller set toward desirable characteristics such as origin-destination pair covering or flow capturing. We provided an extensive experimental analysis of the proposed approaches' capability to fully control transportation networks on small instances, as well as their efficiency at redirecting flows on larger networks. We compared their performance to existing methods and showed that spanning tree-based approaches are capable of reaching a satisfying level of controllability while using fewer controllers than existing approaches on networks containing nodes with a low average degree. Thus, we consider these approaches particularly suitable for sparsely connected networks such as regional networks. However, we make the assumption that the spanning tree approach can be further improved and adapted to our problem. For this purpose, the next chapter will study the possible integration of route information in the process and examine if it enhances the quality of the produced controller set, such as to still provide a scalable method but with better control of transportation networks.

Chapter 6

Spanning tree approach with additional flow information

The previous chapter introduced the spanning tree method as a scalable approach for the controller location problem. In this chapter, we propose to improve the efficiency of this method, in terms of capability to redirect flows over a transportation network, by introducing flow information in the process, based on the assumption that the additional information will provide more suitable locations for pricing controllers.

This chapter is based on the following paper that is currently being prepared for submission to the Transportation Research Record: Journal of the Transportation Research Board: X. Mazur, M. Rinaldi, R. Connors, and F. Viti, An Approach Associating Topological and Flow-based Information to Identify Pricing Controller Locations on a Transportation Network.

6.1 Introduction

In this dissertation, we aim to provide a scalable approach that is capable of locating a set of controllers such as the underlying transportation network is fully controllable. In the previous chapter, we adapted the spanning tree approach initially developed for the sensor location problem (He [15]) to the specificity of the controller location problem. This approach proved to be an easily scalable method that can be employed while considering large networks. However, as demonstrated in the previous chapter, the problem of locating sensors on a transportation network while being similar to the one of locating controllers possesses critical differences. Therefore, we showed that while the spanning tree approach can provide a set of pricing controllers that can satisfyingly control a network, it cannot guarantee that full controllability over the whole network is achieved.

Consequently, in this chapter, we aim at elaborating further on the spanning tree method by improving the capability of the produced controller set to control the considered network, aiming to be as close as possible to full controllability. A first step was provided in chapter 5 by developing various weighting schemes based on topological information, aiming at steering the selection of spanning trees to produce more suitable sets of controller locations. As we showed, these topology-based approaches could provide a small performance improvement compared to the original method. However, we make the assumption that it is possible to further improve the performance of the method, in terms of controllability, by combining the usage of topological information and route-based information. For this purpose, we propose to employ the concept of link ranking introduced by Verhoef [42], in which the authors used flow information to determine in which order links are the most critical to control. Our aim is to integrate the additional flow-based information provided by the link ranking process of Verhoef to the spanning tree approach such that the produced set of pricing controllers can more efficiently control the underlying transportation network.

The following section will detail the functioning of Verhoef's method used to provide a link ranking and how we combined the provided flow information with the topological spanning tree approach. Then, we present an experimental setup to evaluate the gain in controllability provided by the integration of additional flow information compared to the classical spanning tree approach.

6.2 Methodology

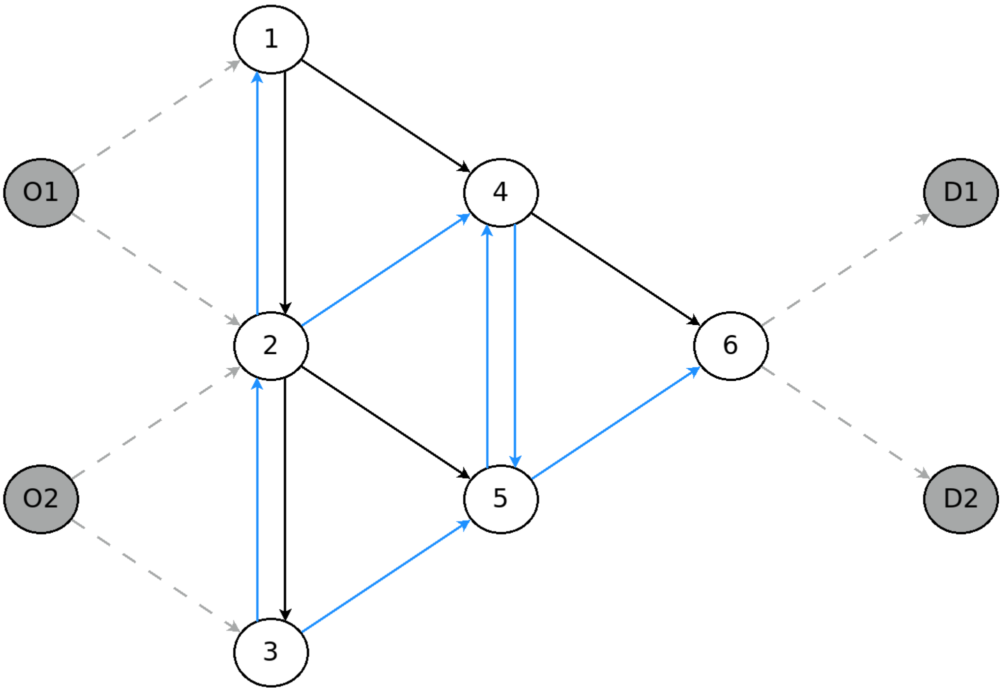
As presented in the previous chapter, the spanning tree approach only uses topological information. Consequently, it doesn't possess any knowledge of the current flow distribution on the network. We are thereby unable to a priori assess the exact capability of a produced controller set to capture flows. As presented in Pieper [33], on a given network, a spanning tree is not unique, and the number of possible spanning trees can be enumerated with an upper bound of $|N|^{(|N|-2)}$, in the specific instance of a complete graph K_N. Therefore, we desire to investigate the possibility of using flow-based information to influence the selection of a specific spanning tree, such that the resulting set of pricing controllers is more adapted to the current state of the network and capable of efficiently capturing and controlling flows.

In order to identify more accurately important link locations and prioritize them for controller locations, we propose to employ the link ranking system developed by Verhoef [42]. In this work, the authors introduced an indicator to predict the welfare gain from implementing a second-best toll on a specific link. Based on this process, they also proposed to rank each link in their order of priority to control by following the predicted welfare gain of every link, such that the most important links to control are those that provide the highest increase in welfare. To predict the potential welfare gain from using a pricing controller on a considered link, the authors employ exact knowledge regarding the considered link flow resulting from a no-toll equilibrium to determine the marginal cost of fitting a controller at this location. From this, each link of the network can be ranked in order of priority to control based on the predicted welfare gain. Therefore, we propose to compute a flow assignment corresponding to user equilibrium and to apply Verhoef's method to obtain the corresponding link ranking. In a similar way to the previous chapter, we steer the selection of a spanning tree by applying a weight to each link equal to its assigned rank. Then, we seek the minimum spanning tree on the resulting weighted graph, that is, the spanning tree that minimizes the sum of its link weights. The assumption behind this new approach is that the additional information provided by the vehicular flow, in the form of a link ranking, will allow the proposed method to identify which link or route is more important with respect to the equilibrium flow distribution and locate controllers accordingly.

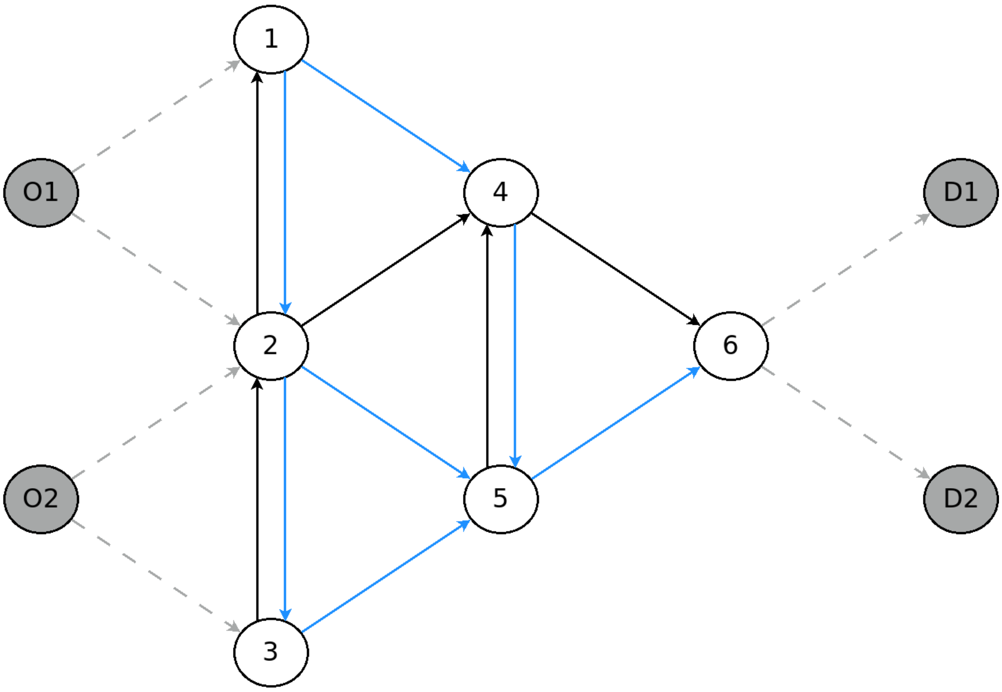
In addition, for comparison sake, we propose a second approach in which we simply locate pricing controllers by directly following the link ranking provided by Verhoef's method. This comparison will also help us determine the benefit of using both topological and flow information to assess if the additional computational cost is beneficial. However, this approach does not specify the required number of toll points to control the network efficiently. Thus, for the sake of fair comparison, we select a number of pricing controllers equal to that determined by the spanning tree approach. However, it is essential to notice that the number of controllers required for controllability is important additional information that enhances the classical Verhoef's approach. Thus, based on the spanning tree approach, the chosen locations will be the $|L| - (|N| - 1)$ first ranked links, with N being the number of nodes and L being the number of links.

First, we propose to study the difference between controller sets generated by the spanning tree approach and the minimum spanning tree approach based on the Verhoef link ranking. Figure 6.1 displays the obtained controller sets by each approach on a small network. Chosen locations for controllers are displayed in blue, and grey dotted links represent origin-destination connectors links that are not considered possible locations for controllers. We can see that while both methods provide the same number of pricing controllers, in this case, seven controllers, the chosen locations are different for each method. Therefore, we can deduce that using additional flow information will impact the selection of a controller set.

In this section, we proposed a novel approach that combined the usage of both topological and flow-based information. In the following, we will evaluate its efficiency at controlling transportation networks compared to purely topological or flow-based methods over a diverse range of networks.



(a) Controller locations selected by the spanning tree approach.



(b) Controller locations selected by the minimum spanning tree approach with Verhoef link ranking.

Figure 6.1: Comparison of the chosen locations between two approaches, links in blue represent locations equipped with a pricing controller.

6.3 Experimental setup

To test the proposed approaches, we generate a set of random networks using the graph generator introduced in Mireles de Villafranca et al. [44]; its functioning was detailed in Chapter 3. With this network generator, we can produce various arbitrarily randomized networks that resemble urban transportation networks. Each produced network is divided into three zones, representing the city center, suburban areas, and outside areas with associated characteristics, such as speed limits adapted on the area. On each network, nine origin nodes and nine destination nodes are integrated; thereafter, every possible origin-destination pair is connected by a collection of routes. To generate each route set, we employ a k-shortest path algorithm (Yen [49]) with a number of routes of $k = 3$.

First, we propose to assess the capability of the newly developed approaches to efficiently control transportation networks. To do so, we employed the principle of level of controllability that was detailed in Chapter 2. We propose to evaluate the level of controllability reached by the various proposed approaches over a set of 100 networks of 49 nodes produced by the previously mentioned graph generator. Therefore, each produced network will be of the same size but with a different shape. For every generated instance, we apply the previously presented approach that combines the spanning tree with the Verhoef link ranking and the method that solely uses the link ranking. We also apply the original spanning tree approach and a topology-based variation presented in chapter 5 (respectively Alg. 5.1 and Alg. 5.3) for comparison. As shown in figure 6.2, the method that combines topological information provided by the spanning tree with flow information provided by the Verhoef's ranking system (MST Verhoef) managed to provide a higher level of controllability on average than other approaches. This demonstrates that combining topology-based and flow-based information produces a more suitable set of controllers that provides better controllability over the considered network

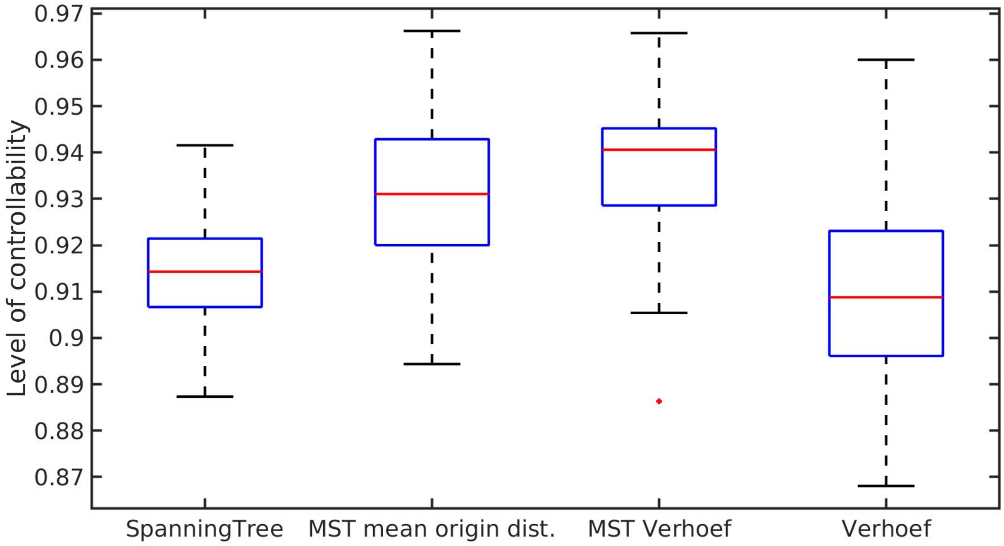


Figure 6.2: Level of controllability reached by different variants of the spanning tree approach over a set of 100 networks of 49 nodes.

To further assess the capability of the proposed approaches to improve the situation on transportation networks, we present a second experimental setup based on larger networks, in which we produced a set of 100 networks of 64 nodes based on the previously presented network generator. In a similar way to the previous experiment, we apply and compare the same four approaches in every instance. This experiment aims to evaluate the capability of controller sets produced by the proposed approaches to efficiently redirect flows on a network, such as to reduce the total time spent by road users. More precisely, we aim to assess if the additional flow-based information provided a controller set more adapted to the current network state. For this purpose, we follow the same process as in the previous chapter, and we employ static assignment simulation to generate flows on networks and compare the capability of the produced controller sets to redirect these flows. To generate a flow distribution on networks, we use the method of successive averages, intending to reproduce the condition of static deterministic user equilibrium, which can be defined as an equilibrium in which each road user minimizes its time spent on the network. Based on the work of Rinaldi and Viti [37], we use a BPR cost function for each link, in which the link cost parameters depend on the location of the given link with respect to the network's subdivision. As such, links belonging to a more central subdivision like the city center will possess a lower speed limit and lower capacity than links belonging to outside areas. We chose to reproduce this equilibrium for its simplicity and sufficiently realistic flow distribution. Additionally, for controller sets to have an objective to attain, we compute a flow assignment representing system optimum, that is, a flow assignment in which the total time spent by all road users is minimized. To generate such flow distribution, we employ the method of successive averages with an all-or-nothing assignment on a fixed set of routes per OD obtained with a K-Shortest Path algorithm, with explicit consideration of link marginal costs and under the assumption of cost function separability.

Like the previous chapter, we want to maximize the Price of Anarchy on each generated network to provide instances yielding a significant potential for improvement via control actions. For a network G, the Price of Anarchy is defined as $PoA(G) = \frac{TTS_{UE}(G)}{TTS_{SO}(G)}$ with $TTS_{UE}(G)$ as the total time spent by road users under user equilibrium and $TTS_{SO}(G)$ being the total time spent by road users under system optimum. For this purpose, the demand level on each network is configured such that the Price of Anarchy is maximized; for this first experiment, the demand levels of every origin-destination pair are equal.

Once every candidate controller set for a considered network is computed, we determine the optimal toll level for each individual control point using the same procedure as the one employed in Chapter 5. It is essential to notice that due to the non-linearity of the Total Cost objective function with general BPR link cost functions, we cannot guarantee that the obtained toll levels are globally optimal. To evaluate the quality of a produced solution for a network G and a controller set cs, we use the variable ρ as defined in the equation (5.1) presented in chapter 5. Therefore, we aim at producing a ρ value as close as possible to zero, such as to steer the network state toward system optimum.

We propose to examine results obtained on various instances of 64 nodes networks. Figure

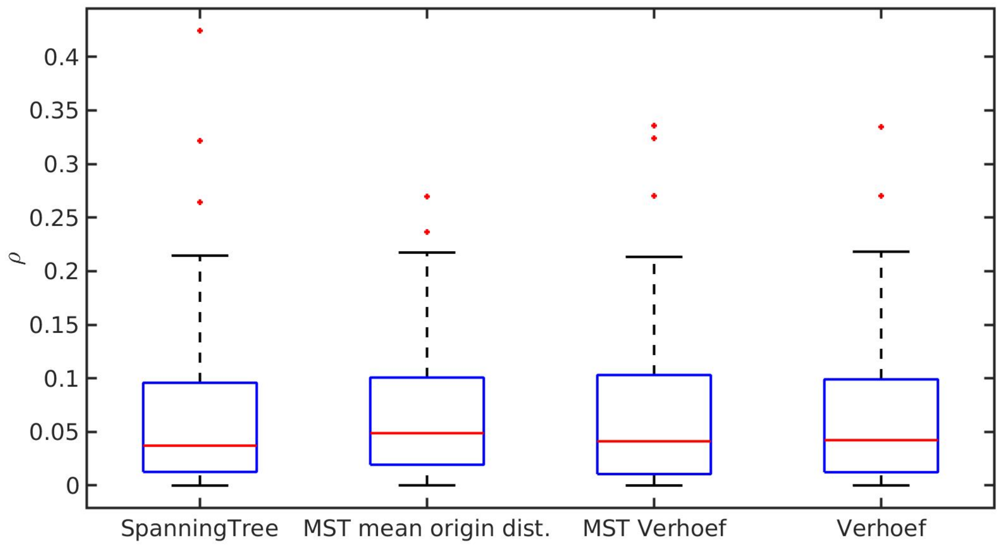


Figure 6.3: Reduction of total time spent on an ensemble of 100 networks of 64 nodes.

6.3 presents the results obtained in the form of a box plot; we can observe that the various proposed approaches reached similar results and that none of them is able to provide a significantly better solution. However, these results are obtained in the specific case where the demand levels between all OD pairs are the same. Due to the procedure we employ to generate random networks, they tend to possess a nearly symmetrical setup; thus, the resulting flow distribution tends to be rather regularly distributed over the network, a characteristic that can be agreeable for topological approaches.

Consequently, we propose modifying the presented experimental setup by incorporating some irregularity in the OD demand matrix to assess if some approaches are more capable of taking into account this irregularity. We randomly select a certain number of OD pairs for each generated network on which we transfer a certain percentage of demand from the other OD pairs. With this process, some OD pairs will have a higher demand; thus, they will be more important to control than previously, and flows are more likely to be irregularly distributed over the network. As an example, for a considered network, we potentially first randomly select three OD pairs, then for each of them, we increase their demand level by

5%. Then we reduce all non-selected OD pairs' demand levels equally, such that the total network demand is unchanged. The objective of this experimental setup is to evaluate if the methods that employ additional flow information can identify those more critical OD pairs and produce a controller set more adapted to the situation. We generate multiple new instances for each network, one per new demand configuration. To generate a new demand configuration, we first chose a number of randomly selected OD pairs that will receive an increase in demand. For this experiment, the number of selectable pairs is either 1 or 3 pairs. Once a set of OD pairs is selected for a network, we determine the percentage of demand that will be added to them; for each possible percentage of demand increase, a new instance is created. For this experiment, the possible percentage of demand increase is either 5%, 10%, 15%, or 20% of the current demand level for each selected OD pair. When multiple OD are selected, they all receive the same percentage of demand increase. Thus, we create eight new instances per generated network for this third experimental setup, one for each possible combination of the two previously presented parameters.

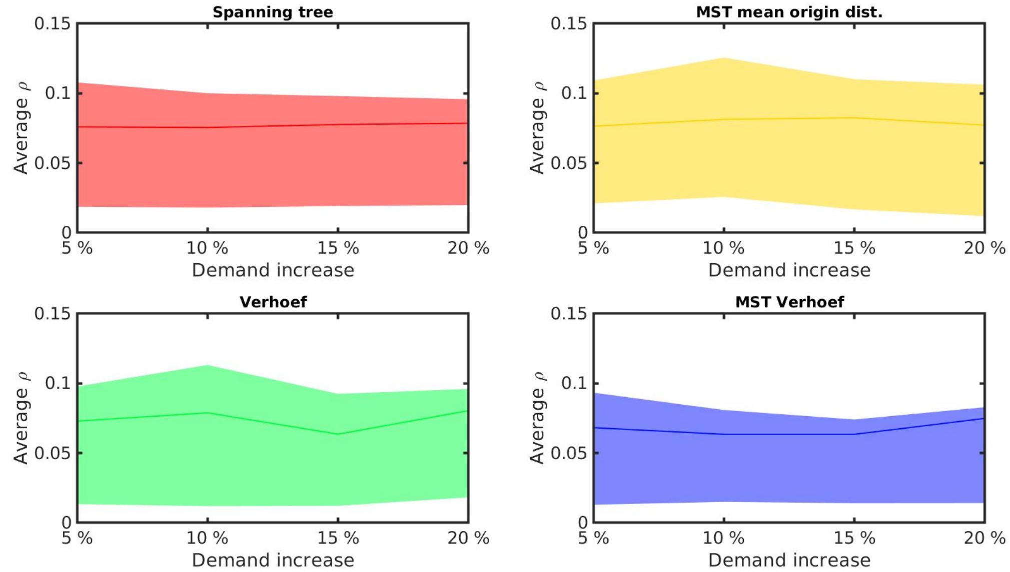
Therefore, we obtain one hundred different networks for each possible configuration of these parameters; for each of them, we compute the ρ value corresponding to the action of the controller sets produced by each previously presented method. To interpret the obtained results, we choose to display the evolution of the average ρ value per method as well as the 25th and the 75th percentile of ρ values obtained over each possible configuration (Fig. 6.4). In these pictures, the average value is represented by the middle line, the upper shade represents the 75th percentile, and the lower shade represents the 25th percentile. We can observe over the diverse setup that methods that rely on flow information, specifically, the Minimum Spanning Tree approach based on Verhoef link ranking (displayed in blue), tend to obtain a lower average ρ value. Also, the observed 25th and 75th percentile values for these approaches appear closer to the average, which indicates a lower variance. This showcases that, in situations where flows are irregularly distributed over a network, which is a more common setting in real-life applications, methods that rely on flow information tend to reach better performances. Additionally, the obtained results show a lower variance for these approaches, which suggests that they also tend to be more reliable.

We conduct a final experiment to provide instances with further irregularity in origin-destination demand matrices. Rather than increasing the demand of a fixed value, we will add a random percentage of demand for each selected origin-destination pair, ranging between 1% and either 5%, 10%, 15%, or 20% depending on the chosen configuration. This way, if multiple origin-destination pairs are selected, they will receive a different randomly determined increase in demand level. For this final case study, we can observe in Figure 6.5 that solutions corresponding to the Minimum Spanning Tree approach combined with the link ranking of Verhoef, displayed in blue, tend once again to exhibit a lower average ρ value. We can also observe that 25^{th} and 75^{th} percentile values also appear closer, which confirms that this approach tends to perform best and to be more resilient to changes in terms of demand configuration.

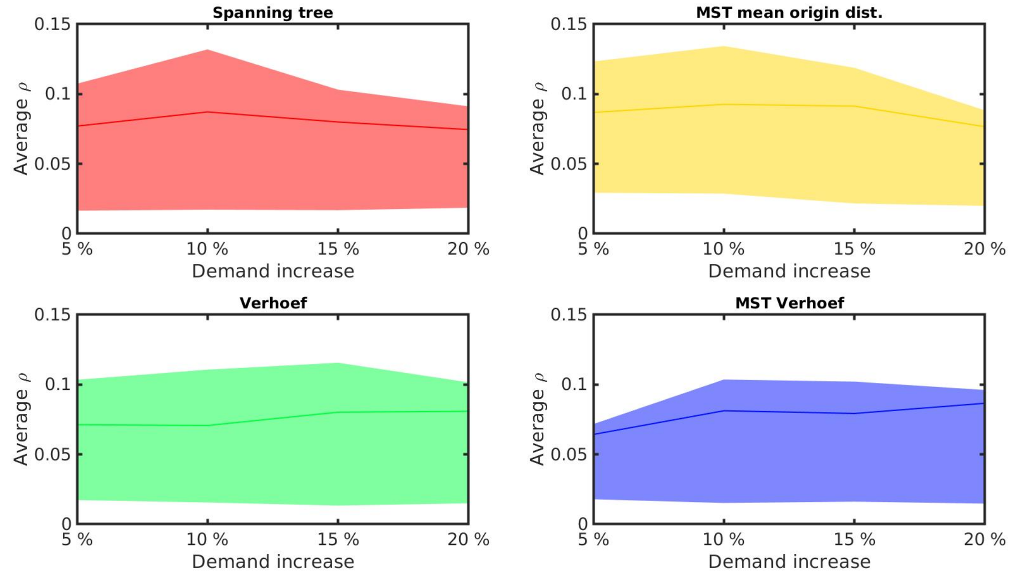
With this set of experiments, we first showed that all the compared approaches are indeed capable of providing a considerable reduction of the total time spent by road users on transportation networks compared to user equilibrium. It also revealed that approaches based on flow information would tend to perform more efficiently in situations where the demand is irregularly distributed. In general, the experiments highlighted that the combination of topological and flow information could provide a more efficient and reliable method, specifically while considering irregularly distributed demand over the network.

6.4 Case study over the network of Luxembourg

To provide a more complete set of experiments, as well as to demonstrated that the approaches developed in this dissertation are actually scalable, thus applicable on large-scale networks found in real networks, we propose to study the impact of these methods over the network of Luxembourg (Fig. 6.6). As the population of Luxembourg constantly grows, demand for road network infrastructures also increases and has already reached a critical

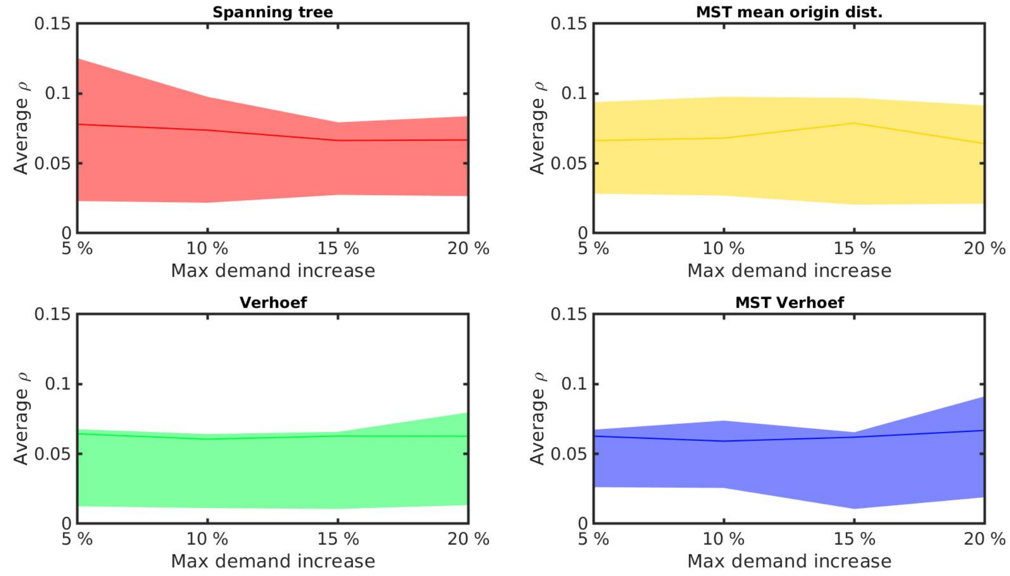


(a) 1 OD pair randomly selected per network.

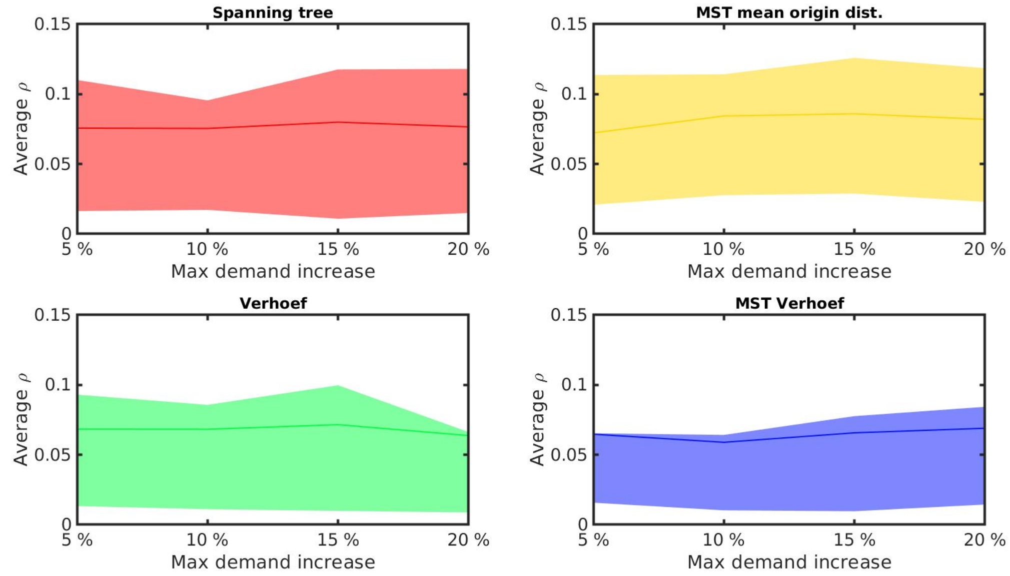


(b) 3 OD pairs randomly selected per network.

Figure 6.4: Average, 25th and 75th percentiles of ρ value in case of a demand increase on 1 or 3 OD pairs.



(a) 1 OD pair randomly selected per network.



(b) 3 OD pairs randomly selected per network.

Figure 6.5: Average, 25th and 75th percentiles of ρ value in case of a random demand increase over 1 or 3 OD pairs.

level leading to a high amount of congestion over the network, specifically during morning and evening peak hours. Thus we consider this network to be suitable for this study. This experiment aims to demonstrate that an adapted set of pricing controllers is capable of improving the current situation over the network of Luxembourg, and by extention realistic networks. For this purpose, we employ the approaches presented in this chapter to locate pricing controllers over the network of Luxembourg while searching for appropriate toll levels, such as to reduce the total time spent by road users on the network.

In order to obtain a representation of the network of Luxembourg, we followed the work of Cantelmo and Viti [5], in which the authors employed the road network of Luxembourg in their study. They obtained the network topology through the software OpenStreetMap, from which connections between nodes and links were extracted but also the link characteristics, such as length, free-flow speed, and capacity, for example. The resulting graph corresponds to a satisfying representation of the network of Luxembourg and is composed of 1405 nodes and 3871 links. Additionally, a realistic demand configuration over the network was generated using the software Visum by the PTV Group. For this purpose, the network is separated into multiple zones; for each zone, the flow production is generated based on existing population data of the considered zone. Similarly, the flow attraction of each zone is determined based on data representing the work density of each zone such that the total attraction is equal to the total production. Then the demand for each zone is distributed between multiple corresponding origin-destination pairs. Each pair is then connected by a set of routes generated using a k-shortest path algorithm (Yen [49]), following a similar fashion as in the previous experimental setup.

In order to provide a more representative experimental setup, we propose to generate a range of 40 different instances, where each has a different demand configuration over the network of Luxembourg. Based on the original demand configuration, we simply increase each origin-destination pair's demand by a different amount on each instance to generate

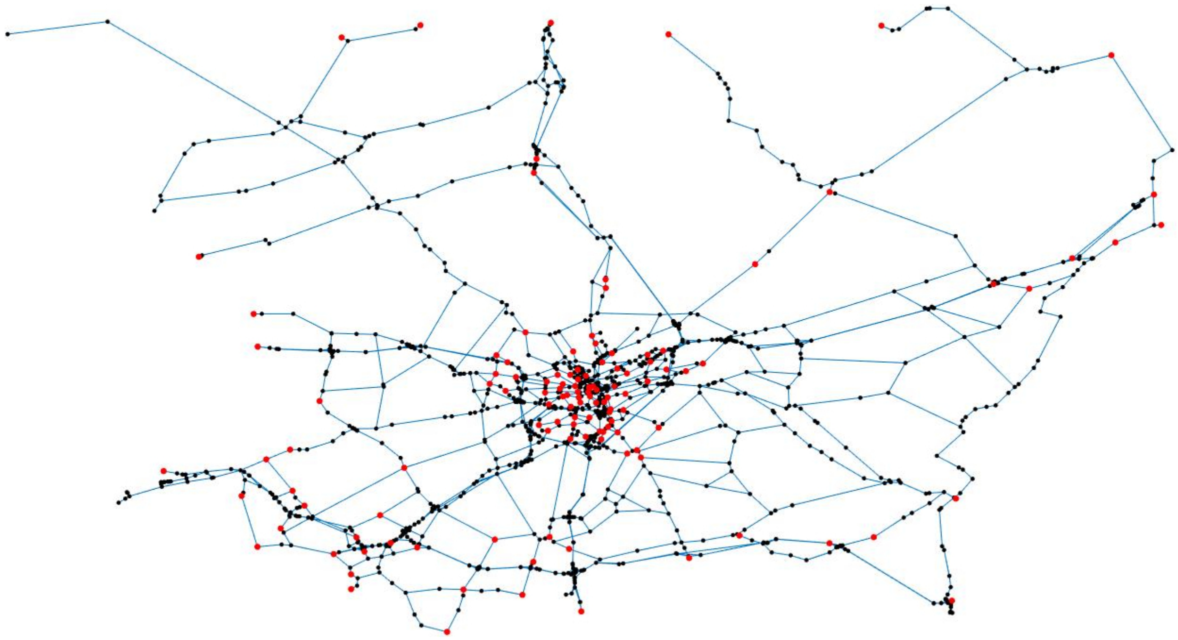


Figure 6.6: Network of Luxembourg. Each node displayed in red is both an origin and a destination node.

various demand configurations. Once the graph representing the network of Luxembourg and the corresponding demand distribution is obtained, we follow a similar experimental setup as in the previous section. We apply on each instance the same four approaches employed in the previous section. Therefore, we compare the two purely topological approaches, being the original spanning tree approach (Alg. 5.1) and a topology-based variation (Alg. 5.3), and two approaches relying on flow information, the approach combining the spanning tree with the Verhoef link ranking and the method that solely uses the link ranking.

Once each corresponding candidate set of pricing controller locations is obtained, the optimal toll level for each individual controller is determined using the same optimization framework as previously. However, due to the important size of the considered network, the required time to find toll values is greatly increased. To simplify the problem, we decided to replace the BPR function used to compute link cost on every link by a simpler linear cost function.This procedure will significantly reduce the computation time needed while computing successively new flow assignments resulting from the action of controllers such as to efficiently search for optimal toll values.

The figure 6.7 displays the results obtained in terms of reduction of the total time spent by road users expressed by the variable ρ as defined previously (equation 5.1) over every generated instance in the form of a box plot. As we can observe, the results obtained with methods that rely on flow information tend to provide solutions with a lower ρ value, meaning closer to system optimum than approaches that solely rely on topological information. Additionally, we can see that these approaches also possess a lower variance indicating that they provide a more reliable method. Overall, we can observe that both topology-based and flow information-based approaches improved the current situation by reducing the total time spent of road users over the network of Luxembourg. Therefore, this highlights the necessity to apply control actions over congested networks based on a set of controllers capable of efficiently controlling the entire network. Moreover, it confirms the additional benefit provided by flow information while considering the control of realistic networks.

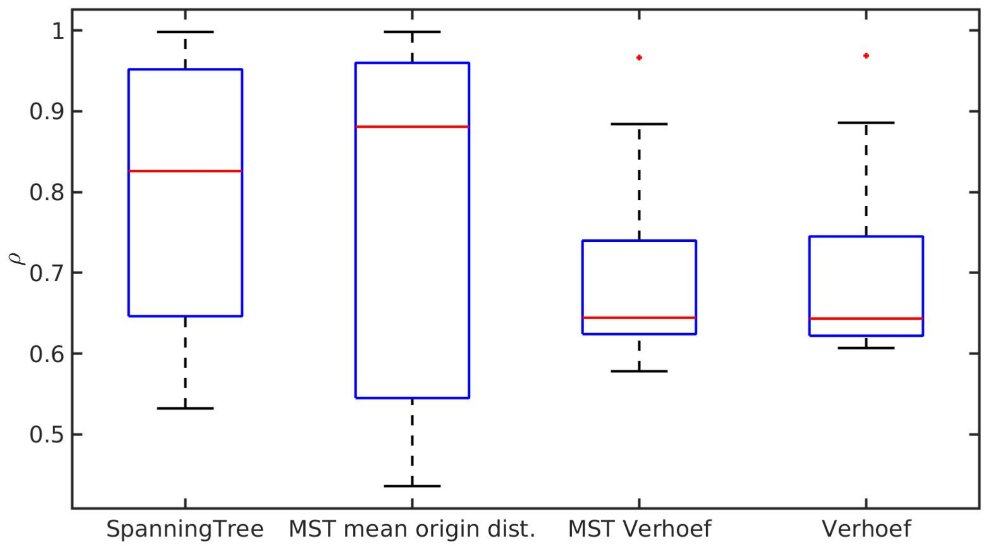


Figure 6.7: Reduction of total time spent on the network of Luxembourg over 40 different demand configurations.

6.5 Conclusion

In this chapter, our objective was to propose an approach that combines the advantages of topology-based methods and methods based on flow information. For this purpose, we proposed integrating flow information to the spanning tree approach by using the link ranking system provided by Verhoef as a weighting scheme, such as to steer the selection of the controller set provided by the spanning tree approach toward locations where flows are more critical to control. Consequently, we carried out an extensive experimental study of the proposed approach's capability to efficiently redirect flows to reduce the total time spent by road users on networks. These experiments showcased that our approach will perform better and tend to be more reliable than purely topological methods in the situation where the demand is irregularly distributed over the network due to the additional knowledge provided by the added flow information. Additionally, we studied the efficiency of both the classical spanning tree and the spanning tree with flow information approaches over the network of Luxembourg. We demonstrated that these approaches are applicable on real large networks subject to high levels of congestion and that the produce controller set is capable of improving the current situation on this network, but we also confirmed the advantage of integrating flow information when considering a realistic network.

Chapter 7

Conclusion

In this last chapter of the dissertation, we are drawing conclusions. The main research objectives that were defined at the start of this thesis are answered, and all findings are summarized. Then recommendations are proposed for future research works.

7.1 Research objectives

This dissertation focused on the controllability of transportation networks, specifically on the impact of the number, types, and locations of controllers on the capability to control the underlying network. The main research question of this dissertation is to define how to identify the minimal set of controllers needed to fully control any kind and size of transportation networks. The methodology and approaches developed, as well as their performance, were assessed in previous chapters. Based on the results obtained, the aims of this dissertation that were defined in the first chapter are addressed in the following:

A1 : The produced controller set should be capable of fully controlling the entire network.

We defined that to efficiently control a transportation network, the number, type, and locations of controllers should be chosen such that the whole network is fully controllable. The main difficulty was to be able to assess the actual capability of a controller set to control a considered network. For this purpose, we employed the controllability framework introduced in Rinaldi [34], based on which we are capable of computing the level of controllability yielded by a controller set. Based on this process, it was shown in chapter 3 that a set of pricing controllers could be replaced by traffic lights without any loss of controllability. Chapter 4 demonstrated that all the proposed heuristics were capable of reaching full controllability over all the tested networks. However, this process cannot be applied on large networks due to heavy computational complexity. Therefore, chapters 5 and 6 employed static assignment-based simulation to assess the capability of generated controllers sets to redirect flows on networks, such as to reduce the total time spent by road users over the network. We expect to obtain similar results under dynamic settings; however, further research works are necessary to confirm this assumption. It was shown that the developed spanning tree-based approaches, including those integrating additional flow information, could provide a satisfying reduction of the total time spent by road users and were more efficient than a randomly chosen set of pricing controllers. More specifically, the approach combining topological and flow-based information developed in chapter 6 was shown to be more efficient in the case where the demand is irregularly distributed over the network.

A2 : The controller set should contain the minimal number of controllers needed.

To minimize the corresponding installation cost, every developed approach aims at reducing the number of controllers employed. For this purpose, chapter 3 demonstrated that based on a set of pricing controllers, it is possible to find a set of traffic lights with an equivalent level of controllability using the same number of controllers. Chapter 4 presented a comparison of various heuristics in which it was shown that certain approaches could propose a solution reaching full controllability while reducing the number of controllers employed. The spanning tree approaches developed in chapter 5 were shown to use fewer controllers than existing approaches in the specific instance of sparsely connected networks such as regional networks. More specifically, the number of controllers used by this approach was shown to be strongly dependent on the average node degree such that the less connected a network is and the fewer controllers are required, thus providing a very suitable method for sparsely connected networks.

A3 : The developed method should be scalable.

The literature review proposed in chapter 2 showed the lack of a scalable approach to identify controller locations; therefore, developing a method applicable to networks of any size has been an essential aim during this dissertation. Chapter 5 specifically focused on this objective by proposing to apply the spanning tree approach, initially developed for the dual problem of sensor locations, and adapt it to the specificity of the controller location problem. As this approach is based solely on topological information, it only requires the knowledge of the network topology, such as node connections, in the form of a graph that is easy to obtain. Then the computational complexity of the method is bound by the search for a spanning tree that has a computational complexity of $O(NlogL)$, therefore spanning tree-based approaches require a very reasonable computational time. Additionally, in chapter 6, we applied spanning tree-based approaches to the real network of Luxembourg, which contains 1405 nodes and 3871 links, showcasing that this approach can be used even on real large networks.

7.2 Future research directions

This dissertation proposed multiple contributions to address the various research objectives defined in the first chapter. The methodology developed during this work can be the foundation for future research works to address research questions that were out of the scope of this dissertation.

A possible direction for future work would be to study the possibility of identifying redundant controllers in a previously obtained set of controllers. It would permit the improvement of existing methods, including those developed in this dissertation, aiming at reducing the installation cost. Additionally, reducing the number of controllers will reduce the number of variables to optimize while searching for the optimal control action, thus simplifying the necessary process to reach the desired network state.

Some transportation networks already employ controllers to reduce congestion; thus, another possible research direction would be to study the possibility of completing an existing set of controllers to improve the current control capability. This direction would be particularly suitable while considering the usage of traffic lights, as this type of controller is already commonly present in transportation networks and would allow identifying the missing traffic lights to reach full controllability of the network.

An important factor is the type of controller employed; in this dissertation, we focused solely on traffic lights and pricing controllers. Therefore, future research works could examine the possibility of fully controlling a transportation network based on other controller technologies.

Finally, a similar problem to the one of identifying controller locations would be to develop approaches to address the problem of partial controllability. That is to find the best set of controllers over a network while considering a constraint in installation cost, such that the number of controllers available is limited. One potential approach would be to identify the best subset of controllers based on a given optimal set.

References

[1] J. Añez, T. De La Barra, and B. Pérez. "Dual graph representation of transport networks". en. In: *Transportation Research Part B: Methodological* 30.3 (June 1996), pp. 209–216. ISSN: 01912615. DOI: 10.1016/0191-2615(95)00024-0. URL: https://linkinghub.elsevier.com/retrieve/pii/0191261595000240 (visited on 06/24/2020).

[2] Aleksandr Belov, Konstantinos Mattas, Michail Makridis, Monica Menendez, and Biagio Ciuffo. "A microsimulation based analysis of the price of anarchy in traffic routing: The enhanced Braess network case". In: *Journal of Intelligent Transportation Systems* (2021), pp. 1–16. DOI: 10.1080/15472450.2021.1904920.

[3] Lucio Bianco, Giuseppe Confessore, and Pierfrancesco Reverberi. "A Network Based Model for Traffic Sensor Location with Implications on O/D Matrix Estimates". In: *Transportation Science* 35.1 (2001), pp. 50–60. ISSN: 00411655, 15265447. URL: http://www.jstor.org/stable/25768939.

[4] Piet HL Bovy. "On modelling route choice sets in transportation networks: a synthesis". In: *Transport reviews* 29.1 (2009), pp. 43–68.

[5] Guido Cantelmo and Francesco Viti. "A Big Data Demand Estimation Model for Urban Congested Networks". en. In: *Transport and Telecommunication Journal* 21.4 (Dec. 2020), pp. 245–254. ISSN: 1407-6179. (Visited on 10/25/2021).

[6] Enrique Castillo, Aida Calviño, Hong K. Lo, José María Menéndez, and Zacarías Grande. "Non-planar hole-generated networks and link flow observability based on link counters". en. In: *Transportation Research Part B: Methodological* 68 (Oct. 2014),

pp. 239–261. ISSN: 01912615. DOI: 10.1016/j.trb.2014.06.015. URL: https://linkinghub.elsevier.com/retrieve/pii/S0191261514001222 (visited on 08/26/2019).

[7] Enrique Castillo, Pilar Jiménez, José María Menéndez, Ana Rivas, and Inmaculada Gallego. "A ternary-arithmetic topological based algebraic method for networks traffic observability". en. In: *Applied Mathematical Modelling* 35.11 (Nov. 2011), pp. 5338–5354. ISSN: 0307904X. DOI: 10.1016/j.apm.2011.04.044. URL: https://linkinghub.elsevier.com/retrieve/pii/S0307904X11002915 (visited on 03/02/2020).

[8] Anthony Chen, Surachet Pravinvongvuth, Piya Chootinan, Ming Lee, and Will Recker. "STRATEGIES FOR SELECTING ADDITIONAL TRAFFIC COUNTS FOR IMPROVING O-D TRIP TABLE ESTIMATION". In: *Transportmetrica* 3.3 (2007), pp. 191–211. DOI: 10.1080/18128600708685673.

[9] Stella Fiorenzo-Catalano, Sascha Hoogendoorn-Lanser, and Rob van Nes. "Choice set composition modelling in multi-modal travelling". In: *Multi-Modal Traveling. IATBR, 10Th International Conference on Travel Behaviour Research, Moving through Nets: The Physical and Social Dimension of Travel*. 2003.

[10] Linton C. Freeman. "A Set of Measures of Centrality Based on Betweenness". In: *Sociometry* 40.1 (1977), pp. 35–41. ISSN: 00380431. URL: http://www.jstor.org/stable/3033543.

[11] M. Girvan and M. E. J. Newman. "Community structure in social and biological networks". In: *Proceedings of the National Academy of Sciences* 99.12 (2002), pp. 7821–7826. ISSN: 1091-6490. DOI: 10.1073/pnas.122653799. URL: http://dx.doi.org/10.1073/pnas.122653799.

[12] Eric J Gonzales and Eleni Christofa. "Empirical assessment of bottleneck congestion with a constant and peak toll: San Francisco-Oakland Bay Bridge". en. In: *EURO Journal on Transportation and Logistics* (2015), p. 22.

[13] Pedro Henrique González, Glaubos Clímaco, Geraldo Regis Mauri, Bruno Salezze Vieira, Glaydston Mattos Ribeiro, Romulo Dante Orrico Filho, Luidi Simonetti, Leonardo Roberto Perim, and Ivone Catarina Simões Hoffmann. "New approaches for the traf-

fic counting location problem". en. In: *Expert Systems with Applications* 132 (Oct. 2019), pp. 189–198. ISSN: 09574174. DOI: 10.1016/j.eswa.2019.04.068. URL: https://linkinghub.elsevier.com/retrieve/pii/S0957417419303045 (visited on 09/28/2021).

[14] J. A. Hartigan and M. A. Wong. "Algorithm AS 136: A K-Means Clustering Algorithm". In: *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28.1 (1979), pp. 100–108. ISSN: 00359254, 14679876. URL: http://www.jstor.org/stable/2346830.

[15] Sheng-xue He. "A graphical approach to identify sensor locations for link flow inference". en. In: *Transportation Research Part B: Methodological* 51 (May 2013), pp. 65–76. ISSN: 01912615. DOI: 10.1016/j.trb.2013.02.006. URL: https://linkinghub.elsevier.com/retrieve/pii/S0191261513000325 (visited on 08/08/2019).

[16] S. P. Hoogendoorn, R. L. Landman, J. Van Kooten, M. Schreuder, and R. Adams. "Design and Implementation of Integrated Network Management Methodology in a Regional Network". In: *Transportation Research Record: Journal of the Transportation Research Board* 2489 (Jan. 2015), pp. 20–28. ISSN: 0361-1981. DOI: 10.3141/2489-03. (Visited on 02/11/2019).

[17] Shou-Ren Hu, Srinivas Peeta, and Chun-Hsiao Chu. "Identification of vehicle sensor locations for link-based network traffic applications". en. In: *Transportation Research Part B: Methodological* 43.8-9 (Sept. 2009), pp. 873–894. ISSN: 01912615. DOI: 10.1016/j.trb.2009.02.008. URL: http://linkinghub.elsevier.com/retrieve/pii/S0191261509000356 (visited on 09/05/2018).

[18] P. B. Hunt, D. I. Robertson, R. D. Bretherton, and R. I. Winton. "SCOOT - A Traffic Responsive Method Of Coordinating Signals". In: *Publication of: Transport and Road Research Laboratory* (1981). ISSN: 0266-7045. (Visited on 02/11/2019).

[19] Saif Eddin Jabari and Laura Wynter. "Sensor placement with time-to-detection guarantees". en. In: *EURO Journal on Transportation and Logistics* 5.4 (Dec. 2016), pp. 415–

433. ISSN: 21924376. DOI: 10.1007/s13676-015-0086-4. URL: https://linkinghub.elsevier.com/retrieve/pii/S2192437620300935 (visited on 05/26/2021).

[20] Rudolf Emil Kalman, Y. C. Ho, and S. K. Narendra. “Controllability of linear dynamical systems”. In: *Contrib. Differ. Equ.* 1.2 (1963), pp. 189–23.

[21] Joseph B Kruskal. “On the shortest spanning subtree of a graph and the traveling salesman problem”. In: *Proceedings of the American Mathematical society* 7.1 (1956), pp. 48–50.

[22] Y. Li, R. Mohajerpoor, and M. Ramezani. “Perimeter control with real-time location-varying cordon”. In: *Transportation Research Part B: Methodological* 150 (2021). cited By 1, pp. 101–120. DOI: 10.1016/j.trb.2021.05.016. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85109142296&doi=10.1016%2fj.trb.2021.05.016&partnerID=40&md5=a9fa248e1b1831eaf689793081ca3208.

[23] Yang-Yu Liu, Jean-Jacques Slotine, and Albert-László Barabási. “Controllability of complex networks”. en. In: *Nature* 473.7346 (May 2011), pp. 167–173. ISSN: 1476-4687. DOI: 10.1038/nature10011. (Visited on 02/11/2019).

[24] P. R. Lowrie. “SCATS, Sydney Co-Ordinated Adaptive Traffic System : A Traffic Responsive Method of Controlling Urban Traffic”. In: *Roads and Traffic Authority NSW, Darlinghurst* (1990). (Visited on 02/11/2019).

[25] Xavier Mazur, Marco Rinaldi, and Francesco Viti. “Heuristic methods for minimal controller location set problem in transportation networks”. en. In: *Transportation Research Procedia* 52 (2021), pp. 83–90. ISSN: 23521465. DOI: 10.1016/j.trpro.2021.01.011. URL: https://linkinghub.elsevier.com/retrieve/pii/S2352146521000168 (visited on 07/20/2021).

[26] David R. Morrison and Susan E. Martonosi. “Characteristics of Optimal Solutions to the Sensor Location Problem”. en. In: *arXiv:1106.0349 [cs, math]* (June 2011). URL: http://arxiv.org/abs/1106.0349 (visited on 02/20/2020).

[27] Se-il Mun, Ko-ji Konishi, and Kazuhiro Yoshikawa. “Optimal cordon pricing”. en. In: *Journal of Urban Economics* 54.1 (July 2003), pp. 21–38. ISSN: 00941190. DOI: 10.1016/S0094-1190(03)00022-6. URL: https://linkinghub.elsevier.com/retrieve/pii/S0094119003000226 (visited on 03/09/2020).

[28] ManWo Ng. “Synergistic sensor location for link flow inference without path enumeration: A node-based approach”. en. In: *Transportation Research Part B: Methodological* 46.6 (July 2012), pp. 781–788. ISSN: 01912615. DOI: 10.1016/j.trb.2012.02.001. URL: http://linkinghub.elsevier.com/retrieve/pii/S0191261512000161 (visited on 09/05/2018).

[29] Lucas Barcelos de Oliveira and Eduardo Camponogara. “Multi-agent model predictive control of signaling split in urban traffic networks”. In: *Transportation Research Part C: Emerging Technologies*. Information/Communication Technologies and Travel Behaviour 18.1 (Feb. 2010), pp. 120–139. ISSN: 0968-090X. DOI: 10.1016/j.trc.2009.04.022. (Visited on 02/11/2019).

[30] Javier Ortigosa, Monica Menendez, and Hector Tapia. “Study on the number and location of measurement points for an MFD perimeter control scheme: a case study of Zurich”. en. In: *EURO Journal on Transportation and Logistics* 3.3 (Oct. 2015), pp. 245–266. ISSN: 21924376. DOI: 10.1007/s13676-013-0034-0. URL: https://linkinghub.elsevier.com/retrieve/pii/S2192437620301412 (visited on 05/26/2021).

[31] Toshihiro Osaragi and Yuko Hiraga. “Street Network created by Proximity Graphs: Its Topological Structure and Travel Efficiency”. en. In: *Proceedings of the AGILE’2014 International Conference on Geographic Information Science, Castellon.* (2014).

[32] M. Papageorgiou, H. Hadj-Salem, and F. Middelham. “ALINEA local ramp metering: Summary of field results”. In: *Transportation Research Record* 1603 (1997), pp. 90–98. DOI: 10.3141/1603-12. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-3643094227&doi=10.3141%2f1603-12&partnerID=40&md5=3c6dbb249a8e3ece4b28fb2a2103f7c1.

[33] Konstantin Pieper. “The Number of Spanning Trees in a Graph”. en. In: *Course “Trees – The Ubiquitous Structure in Computer Science and Mathematics”, JASS’08* (2008).

[34] Marco Rinaldi. “Controllability of transportation networks”. In: *Transportation Research Part B: Methodological* 118 (Dec. 2018), pp. 381–406. ISSN: 0191-2615. DOI: 10.1016/j.trb.2018.11.005. (Visited on 02/11/2019).

[35] Marco Rinaldi, Willem Himpe, and Chris M. J. Tampère. “A sensitivity-based approach for adaptive decomposition of anticipatory network traffic control”. In: *Transportation Research Part C: Emerging Technologies*. Advanced Network Traffic Management: From dynamic state estimation to traffic control 66 (May 2016), pp. 150–175. ISSN: 0968-090X. DOI: 10.1016/j.trc.2016.01.005. (Visited on 02/11/2019).

[36] Marco Rinaldi, Chris M.J. Tampère, and Francesco Viti. “On characterizing the relationship between route choice behaviour and optimal traffic control solution space”. In: *Transportation Research Part B: Methodological* 117 (2018). TRB:ISTTT-22, pp. 892–906. ISSN: 0191-2615. DOI: https://doi.org/10.1016/j.trb.2017.08.014. URL: https://www.sciencedirect.com/science/article/pii/S019126151730694X.

[37] Marco Rinaldi and Francesco Viti. “Investigating the Relationship between Controller Locations and Dynamic Traffic Control in Generic Transportation Networks”. en. In: *Transportation Research Record: Journal of the Transportation Research Board* (Apr. 2020). ISSN: 0361-1981, 2169-4052. DOI: 10.1177/0361198120914302. URL: http://journals.sagepub.com/doi/10.1177/0361198120914302 (visited on 06/04/2020).

[38] Martin Rodriguez-Vega, Carlos Canudas-de-Wit, and Hassen Fourati. “Location of turning ratio and flow sensors for flow reconstruction in large traffic networks”. en. In: *Transportation Research Part B: Methodological* 121 (Mar. 2019), pp. 21–40. ISSN: 01912615. DOI: 10.1016/j.trb.2018.12.005. URL: https://linkinghub.elsevier.com/retrieve/pii/S0191261518307306 (visited on 02/19/2020).

[39] Tim Roughgarden. “Selfish Routing and the Price of Anarchy”. en. In: *MIT press Cambridge* (2005).

[40] Paul Rubin and Monica Gentili. “An exact method for locating counting sensors in flow observability problems”. en. In: *Transportation Research Part C: Emerging Technologies* 123 (Feb. 2021), p. 102855. ISSN: 0968090X. DOI: 10.1016/j.trc.2020.102855. URL: https://linkinghub.elsevier.com/retrieve/pii/S0968090X20307555 (visited on 09/28/2021).

[41] S. Shepherd, A. May, and A. Koh. “How to design effective road pricing cordons”. en. In: *Proceedings of the Institution of Civil Engineers - Transport* 161.3 (Aug. 2008), pp. 155–165. ISSN: 0965-092X, 1751-7710. DOI: 10.1680/tran.2008.161.3.155. URL: http://www.icevirtuallibrary.com/doi/10.1680/tran.2008.161.3.155 (visited on 03/09/2020).

[42] Erik T. Verhoef. “Second-best congestion pricing in general networks. Heuristic algorithms for finding second-best optimal toll levels and toll points”. In: *Transportation Research Part B: Methodological* 36.8 (Sept. 2002), pp. 707–729. ISSN: 0191-2615. DOI: 10.1016/S0191-2615(01)00025-X. URL: http://www.sciencedirect.com/science/article/pii/S019126150100025X (visited on 03/16/2018).

[43] Erik T. Verhoef, Richard H. M. Emmerink, Peter Nijkamp, and Piet Rietveld. “Information provision, flat and fine congestion tolling and the efficiency of road usage”. In: *Regional Science and Urban Economics* 26.5 (Aug. 1996), pp. 505–529. ISSN: 0166-0462. DOI: 10.1016/0166-0462(96)02130-8. (Visited on 02/11/2019).

[44] Alonso Espinosa Mireles de Villafranca, Richard D. Connors, and R. Eddie Wilson. “Static Traffic Assignment on Ensembles of Synthetic Road Networks”. en. In: *Traffic and Granular Flow ’17*. Ed. by Samer H. Hamdar. Cham: Springer International Publishing, 2019, pp. 29–36. ISBN: 978-3-030-11439-8 978-3-030-11440-4. DOI: 10.1007/978-3-030-11440-4_4. URL: http://link.springer.com/10.1007/978-3-030-11440-4_4 (visited on 01/30/2020).

[45] Hai Yang, Yasunori Iida, and Tsuna Sasaki. “An analysis of the reliability of an origin-destination trip matrix estimated from traffic counts”. en. In: *Transportation Research Part B: Methodological* 25.5 (Oct. 1991), pp. 351–363. ISSN: 01912615. DOI: 10.1016/

0191-2615(91)90028-H. URL: https://linkinghub.elsevier.com/retrieve/pii/019126159190028H (visited on 03/27/2020).

[46] Hai Yang, Chao Yang, and Liping Gan. "Models and algorithms for the screen line-based traffic-counting location problems". en. In: *Computers & Operations Research* 33.3 (Mar. 2006), pp. 836–858. ISSN: 03050548. DOI: 10.1016/j.cor.2004.08.011. URL: https://linkinghub.elsevier.com/retrieve/pii/S0305054804002011 (visited on 05/28/2019).

[47] Hai Yang and Jing Zhou. "Optimal traffic counting locations for origin-destination matrix estimation". en. In: *Transportation Research Part B: Methodological* 32.2 (Feb. 1998), pp. 109–126. ISSN: 01912615. DOI: 10.1016/S0191-2615(97)00016-7. URL: https://linkinghub.elsevier.com/retrieve/pii/S0191261597000167 (visited on 03/26/2020).

[48] Rui Yang, Wen-Xu Wang, Ying-Cheng Lai, and Guanrong Chen. "Optimal weighting scheme for suppressing cascades and traffic congestion in complex networks". en. In: *Physical Review E* 79.2 (Feb. 2009). ISSN: 1539-3755, 1550-2376. DOI: 10.1103/PhysRevE.79.026112. URL: https://link.aps.org/doi/10.1103/PhysRevE.79.026112 (visited on 11/30/2020).

[49] Jin Y. Yen. "Finding the K Shortest Loopless Paths in a Network". In: *Management Science* 17.11 (July 1971), pp. 712–716. ISSN: 0025-1909. DOI: 10.1287/mnsc.17.11.712. (Visited on 02/11/2019).

[50] Hwasoo Yeo and Alexander Skabardonis. "Understanding Stop-and-go Traffic in View of Asymmetric Traffic Theory". en. In: *Transportation and Traffic Theory 2009: Golden Jubilee*. Ed. by William H. K. Lam, S. C. Wong, and Hong K. Lo. Boston, MA: Springer US, 2009, pp. 99–115. ISBN: 978-1-4419-0819-3 978-1-4419-0820-9. DOI: 10.1007/978-1-4419-0820-9_6. URL: http://link.springer.com/10.1007/978-1-4419-0820-9_6 (visited on 01/04/2022).

[51] Zhengzhong Yuan, Chen Zhao, Zengru Di, Wen-Xu Wang, and Ying-Cheng Lai. "Exact controllability of complex networks". In: *Nature Communications* 4 (Sept. 2013).

[52] Xiaoning Zhang and Hai Yang. “The optimal cordon-based network congestion pricing problem”. en. In: *Transportation Research Part B: Methodological* 38.6 (July 2004), pp. 517–537. ISSN: 01912615. DOI: 10.1016/j.trb.2003.08.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S0191261503000900 (visited on 03/09/2020).