ABSTRACT

Rasool Mohebifard. Traffic Metering in Urban Street Networks. (Under the direction of Dr. Ali Hajbabaie).

Traffic metering or perimeter control is an effective approach to mitigate congestion in urban-street networks. Bimodal traffic signals (i.e., green and red indications only) can be placed at the borders of congested areas similar to on-ramps on freeway facilities to regulate the flow of vehicles. The objective of this study is to develop a methodology for traffic metering optimization and study its effects on traffic operations in urban street networks.

This dissertation developed an optimization program that aimed at increasing the number of completed trips in urban street networks by controlling traffic metering signals. The Benders decomposition technique was used to propose a solution technique that solved the program efficiently. The case study results for an urban street network of 20 intersections with 13 boundary gates showed that traffic metering significantly improved traffic operations by reducing the travel time of vehicles inside the network by 30.8% to 34.2% compared to a no-metering strategy using two demand profiles. Some vehicles were delayed at the metering signals; however, traffic metering reduced the system-level travel times by 2.7% to 5.4% by changing the used demand profiles.

While the solution technique was effective in providing metering strategies very close to the optimal solutions, it did not scale well with the size of the problem (network), and its runtime increased with the size of the network. In addition, traffic signal timings of intersections downstream of boundary gates were assumed to be predefined. Therefore, a cooperative distributed model predictive control approach was developed to optimize metering and intersection signals simultaneously. The methodology decomposed the network into several sub-networks,

allocated computational resources to each sub-network, and reduced the computational complexity of the problem. The results of applying the methodology in the network of 20 intersections showed that the cooperative approach optimized traffic metering and traffic signal timings in real-time. In other words, the time discretization in the analysis was 6 seconds and the runtime of solving each sub-network-level optimization problem was less than 6 seconds. Therefore, the updated solutions were available before getting to the next time step. Moreover, in the evaluated scenarios using two demand profiles, the cooperative approach increased the number of completed trips by 6.0 to 12.8% and 10.9 to 11.0% and reduced the total travel times by 8.1 to 9.0% and 23.6 to 24.2% compared to independent signal control and independent traffic metering, respectively.

In the proposed methodologies, the locations of gates were assumed to be predetermined while their locations were also an important factor in traffic metering. On one hand, installation and maintenance of metering signals were costly, and on the other hand, installing a higher number of metering signals provided greater flexibility in regulating the flow of vehicles. To address these concerns, an integrated formulation was developed that optimized static metering locations and dynamic metering levels within a limited budget. A solution technique based on the branch and bound technique was also suggested to iteratively reduce the feasible region of the problem to optimize the decision variables and solve the problem. The methodology was applied to an urban network of 49 intersections and lanes ranging from one to three in each direction. The network had 659 cells and was analyzed for two different demand profiles. The results showed that the integrated traffic metering location and level optimization resulted in similar or better network performance with lower costs compared to traffic metering with predefined locations.

The last chapter of this dissertation is dedicated to the development of a deep reinforcement learning methodology to train a traffic metering controller that can be implemented with minimal

computational resources. Despite the previous traffic metering methodologies that required dedicated optimization engines and high computational resources, the controller used neural networks to determine traffic metering decisions with low computational resources. The methodology captured traffic dynamics on a micro-level scale using a simulated model of an urban street network with the capability of continuous online improvement after implementation on an actual transportation network. The numerical results of using the methodology for a network of 20 intersections with 13 gates at the borders of the network showed that the methodology reduced the total travel time of vehicles by 3.4 and 15.5% compared to a No-Metering strategy using two demand profiles. Comparing the computational complexity of the methodology with the previous chapters' approaches also showed the potential of the methodology for online applications.

Traffic Metering in Urban Street Networks

by
Rasool Mohebifard

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Civil Engineering

Raleigh, North Carolina
2022

Approved by

Dr. Ali Hajbabaie Committee Chair	Dr. Emily Berglund
Dr. Leila Hajibabai	Dr. Nagui Rouphail
Dr. Billy Williams	

BIOGRAPHY

Rasool Mohebifard received his B.Sc. degree in civil engineering from Shahid Bahonar University, Kerman, Iran, in 2012, and his M.Sc. degree in civil engineering from Sharif University of Technology, Tehran, Iran in 2014. He started his Ph.D. studies at Washington State University, Pullman, WA in 2016 and moved to North Carolina State University in 2019.

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Ali Hajbabaie, for his support during my Ph.D. studies. I would also like to thank my committee members for their guidance to improve the quality of this dissertation.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1. INTRODUCTION

A study of 83 urban areas in the U.S. by the Harvard Center for Risk Analysis projected a $96 billion cost of wasted time due to congestion in 2030 (Levy et al., 2010). Traffic congestion is a major contributing factor to the excessive travel delay, fuel consumption, and air pollution in urban networks. Researchers have studied several traffic congestion management strategies such as traffic signal timing, variable speed limits and speed harmonization, and traffic assignment to mitigate traffic congestion in urban street networks. Traffic metering is another promising strategy that prevents queue spillovers and gridlocks that can happen in oversaturated flow conditions (Keyvan-Ekbatani et al., 2012).

1.1. Background

Daganzo (2007) utilized the idea of Macroscopic Fundamental Diagrams (MFDs) (Godfrey, 1969; Herman and Prigogine, 1979; Mahmassani et al., 1984) to explain the dynamics of urban networks with respect to vehicle accumulation inside a network and its throughput. According to Daganzo (2007), the dynamics of an urban network vary between the dynamics of two extreme systems of single-link and single-ring systems. In a single-link system, the number of vehicles in a link and the number of exiting vehicles have a relationship similar to Figure 1-1(a). Thus, the outflow increases with the density of the link (i.e., the number of vehicles in the link) up to the density D_C corresponding the link capacity F_{max}. The figure shows the declining part of the diagram with a dashed line since this part represents a congested region that happens due to restrictions on the outflows such as downstream queue spillbacks. In other words, the declining part does not happen due to the further increasing of the inflow. On the other hand, in the single-ring system of Figure 1-1(b), we can increase the inflow and accordingly the density of the system

more than D_C to observe the declining part. In the congested region of a single-ring system, the outflows can be reduced to zero by increasing the inflow and its density to the jam density D_J. Therefore, the declining part of the figure is now achievable due to the gridlock occurrence.

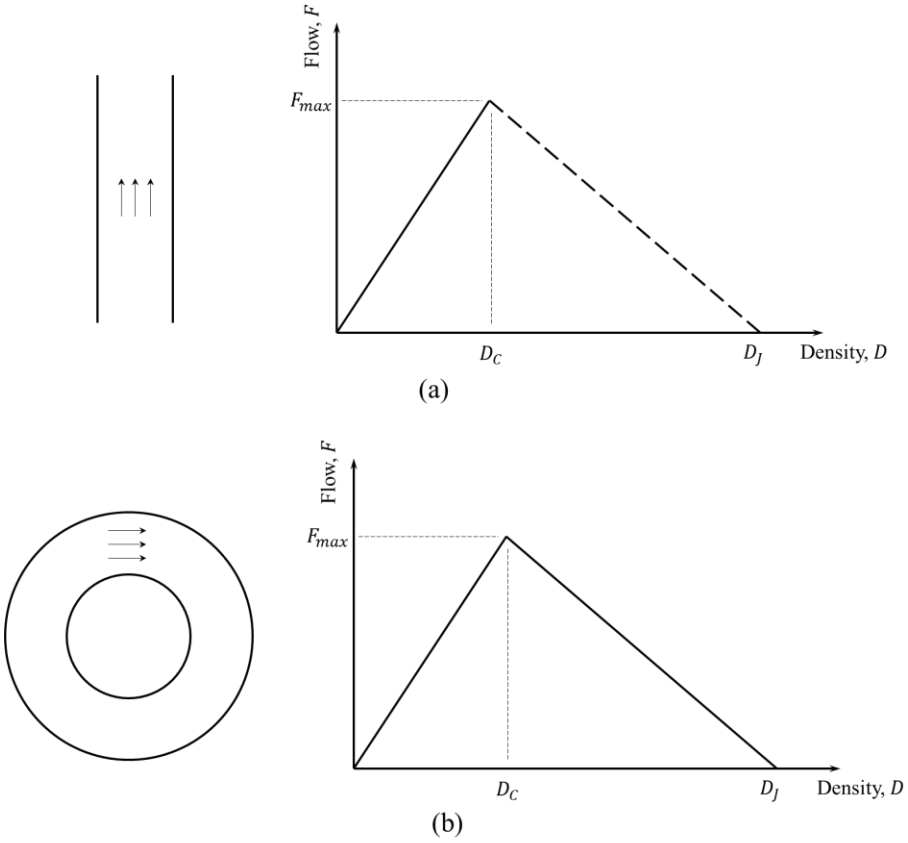


Figure 1-1 (a) single-link and (b) single-ring systems with their corresponding triangular fundamental diagrams (Daganzo, 2007)

The dynamics of real networks are between the dynamics of single-link and single-ring systems. Thus, the diagram of real networks may have a generic shape similar to Figure 1-2. In real networks, gridlock happens, but it does not last long due to the nature of transportation networks. Therefore, the maximum observed density D_v inside the network can be somewhere between D_C and D_J (Daganzo, 2007).

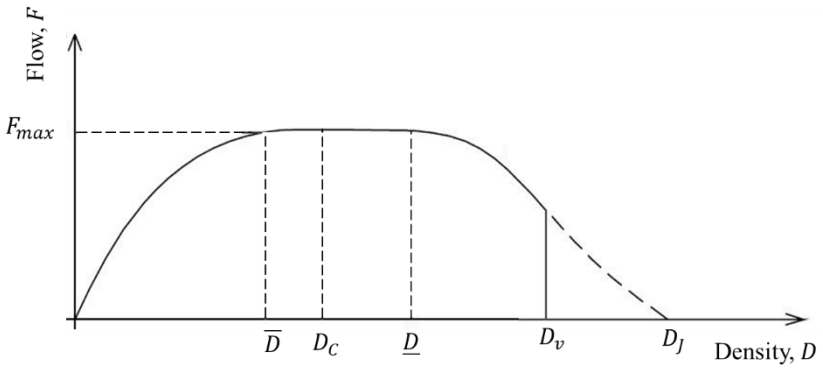


Figure 1-2 A generic fundamental diagram for real transportation networks (Daganzo, 2007)

Consequently, the density of a transportation network can be controlled to prevent outflow drops. For instance, keeping the density between $\underline{D}$ and $\overline{D}$ in a network with a generic fundamental diagram of Figure 1-2 ensures that the network can process the maximum number of vehicles without an outflow drop due to excessive congestion. Once the number of entry vehicles to a network surpasses the optimal accumulation interval $\left[\underline{D}, \overline{D}\right]$, "metering signals" will be activated to reduce the excessive number of entry vehicles.

Figure 1-3 shows a portion of an urban network that is equipped with metering signals at the borders of a region that needs to be protected from oversaturated flow conditions. Bimodal traffic signals (with green and red indications) can be placed at the borders of congested areas to regulate the inflow of vehicles. The metering signals can also be the available traffic signals at the intersections whose timings can be modified to accommodate the traffic metering application by prolonged red signal indications. Note that the inbound approaches of such intersections should have enough capacity for holding possible queues of vehicles. Otherwise, queue spillovers from the metering signals may deteriorate the network performance at their upstream.

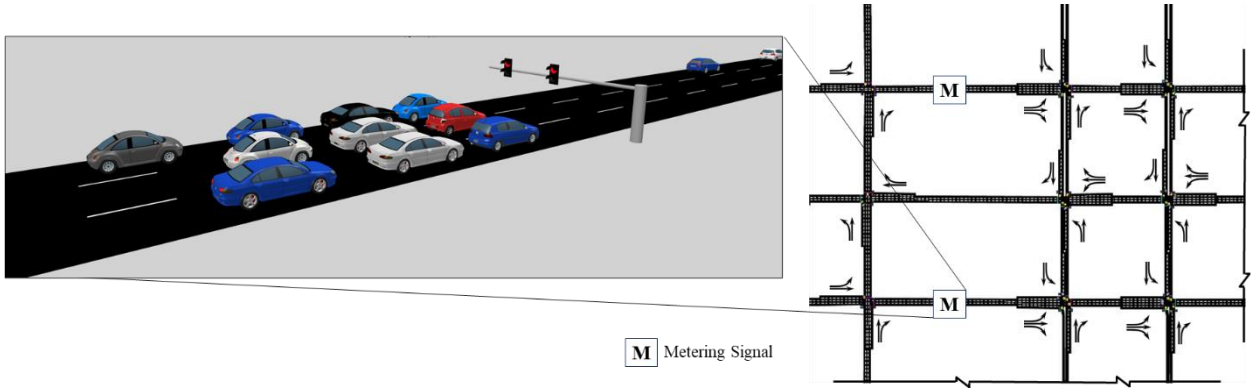


Figure 1-3 Metering gates that are placed at the borders of a region that needs to be protected from oversaturated flow conditions

1.2. Problem statement

As traffic congestion increases, the network capability for processing vehicles reduces due to queue spillovers and gridlocks. In this condition, controlling the number of entry vehicles to the congested areas reduces traffic congestion and allows the network to operate at its optimum level. Traffic metering is one of the congestion management strategies that can be used to protect congested areas of urban networks from getting oversaturated. Metering signals, similar to those implemented on on-ramps, can be placed at the borders of the congested areas to regulate the number of vehicles.

Traffic metering delays some vehicles at the metering locations to improve traffic operations in a broader region inside the network. The regulated flow of vehicles changes the number of vehicles that should be processed by signalized intersections downstream of metering locations. Hence, traffic metering also impacts the operations of those signalized intersections. Accordingly, it is critical to find the best traffic metering locations and rates to achieve the highest network performance.

Traffic dynamics in congested urban street networks are very complex and finding dynamic metering rates for each individual metering gate that yields the highest network performance is

very challenging. The available metering approaches directly used the discussed fundamental diagrams in control theory techniques to maintain the number of vehicles within a predefined accumulation interval (e.g., Haddad and Geroliminis, 2012; Keyvan-Ekbatani et al., 2012). While such approaches are easy to implement for large-scale networks, their efficiency relies on the accuracy of optimal accumulation estimation. In other words, the available approaches assume that a well-defined macroscopic fundamental diagram (MFD) that gives an accurate estimation for the optimal vehicle accumulation can be derived for each urban network (or its sub-networks). However, these diagrams might not exist for all networks. Moreover, urban networks might have significant heterogeneity in traffic demand, distribution of vehicles in the network, and traffic control settings and thus assuming a predefined optimal accumulation level is not realistic. Moreover, the interactions of traffic metering with other control systems such as signalized intersections or the effects of gate locations and their number on traffic metering cannot be identified.

Accordingly, it is essential to develop traffic metering methodologies that capture traffic flow more realistically. These methodologies should be able to adjust the metering rates based on the changes in the traffic demand, gate locations, and network control characteristics such as timings of signalized intersections. Adding such capabilities to the traffic metering strategies increases their complexity, and hence it is critical to develop efficient solution techniques to optimize traffic metering rates in real-time for implementation purposes.

1.3. Research objectives

The objective of this study is to develop methodologies for finding the best traffic metering locations and their corresponding traffic metering rates besides capturing the interactions of traffic metering and signal timing control. This dissertation aimed at finding answers to the following

fundamental research questions:

1. What is the mathematical representation of the traffic metering problem for an urban street network?
2. How can we find the optimal solutions to the traffic metering problem?
3. How can we formulate and capture the interactions of traffic metering and signal timing control?
4. How can we find the best traffic metering locations and their corresponding metering rates?
5. How can deep reinforcement learning help reduce the complexity of traffic metering optimization?

1.4. Research contributions

The available traffic metering approaches rely on well-defined macroscopic fundamental diagrams (MFD). Besides the fact that such diagrams might not exist for all urban networks, the networks might have significant heterogeneity in traffic demand, distribution of vehicles in the network, and traffic control settings. Therefore, assuming that the network dynamics follow a well-defined MFD is not realistic. Moreover, the MFD-based techniques cannot directly capture the interactions of traffic metering with other control strategies such as traffic signal timing, the effects of gate locations on traffic metering, and the effects of queued vehicles on traffic operations upstream of the gates. This dissertation addresses these gaps by developing new methodologies to optimize traffic metering rates for urban street networks.

In this dissertation, the traffic metering problem is formulated as a mixed-integer non-linear optimization program (MINLP). The program employs the cell transmission model (Daganzo, 1995, 1994) to capture traffic dynamics and maximizes the network throughput by optimizing

traffic metering rates at each network gate. A Benders decomposition-based approach (Benders, 1962; Geoffrion, 1972) is also developed to solve the program and tackle its computational complexity. The results of this approach are optimal traffic metering rates for all metering gates over the entire analysis period, regardless of whether they can be found in real-time or not.

Both traffic signal and traffic metering improve traffic operations in congested urban-street networks. Signal control optimizes the timings of signalized intersections to improve traffic operations by avoiding long queues and preventing gridlocks. Traffic metering, on the other hand, regulates the flow of incoming vehicles to a protected area of a congested network with a similar objective. In addition, traffic metering rates depend on the timings of signalized intersections inside a congested area. Accordingly, cooperative traffic signal control and metering at network gates can lead to significantly more efficient network-level traffic operations compared to independent signal control and independent perimeter control. This cooperative problem is formulated in the present dissertation and a distributed solution technique is proposed to optimize both traffic metering rates and timings in signalized intersections in real-time.

Furthermore, the available traffic metering methodologies meter traffic flow at predefined locations at the boundary of a congested area. Selecting proper metering locations can yield further improvements in traffic operations. Moreover, installation and maintenance of metering signals are costly, and thus metering signals should be installed at proper locations to achieve the highest performance while meeting budget restrictions. The other contribution of this dissertation is the incorporation of traffic metering location decisions in the metering level optimization by considering a fixed budget for installation and operational costs of traffic metering locations. An integrated MINLP that optimizes static metering locations and dynamic metering levels within a limited budget is proposed. A solution technique is also suggested to divide the feasible region of

the problem into several sub-regions based on the metering locations. Then, two MINLPs with less complexity for each sub-region that only optimize traffic metering levels for predetermined fixed locations were constructed. The MINLPs are later decomposed into primal and master problems that were solved iteratively to find upper and lower bounds for the solutions of the sub-regions. The solution technique iteratively reduces the feasible region to optimize decision variables.

A major part of the discussed methodologies relies on complex optimization techniques that require dedicated optimization engines and high computational resources. Therefore, the last section of this dissertation uses deep neural networks and reinforcement learning techniques to train a traffic metering controller. The controller determines traffic metering rates using a neural network with low computational resources.

1.5. Dissertation layout

The exposition of this doctoral proposal is as follows.

Chapter 2 reviews the available studies in the traffic metering area and highlights the research gaps and the contributions of this dissertation to address those gaps.

Chapter 3 presents the traffic metering formulation and elaborates on its objective and constraints. The limitations and assumptions of the formulation are also discussed.

Chapter 4 includes the proposed decomposition-based solution technique to find the optimal solutions to the traffic metering problem. This chapter also includes numerical results of applying the solution technique to a case study network and evaluates the effects of traffic metering on traffic operations.

Chapter 5 develops the formulation of cooperative traffic metering and traffic signal control. A modified distributed optimization and coordination algorithm is also proposed to solve this problem in real-time. The effects of this cooperation on a case study are detailed in this chapter.

Chapter 6 builds on the developed methodologies in the previous chapters to create an integrated formulation that optimizes the traffic metering locations within a limited budget for installation, operation, and maintenance of metering locations. A new solution technique that combines the ideas of the branch-and-bound and Benders decomposition techniques is proposed to solve the problem. This methodology is applied to a case study and its numerical analyses are discussed.

Chapter 7 presents the deep reinforcement application for traffic metering. In this chapter, the ideas that will be pursued to train a deep neural network for traffic metering and the expected timeline will be discussed.

CHAPTER 2. LITERATURE REVIEW

This chapter presents a review of the traffic metering literature. These approaches are categorized into Scenario- and MFD-based studies. The details of each category follow.

2.1. Scenario-based traffic metering

Scenario-based approaches evaluate improvements in the network performance that can be achieved using predefined metering scenarios. These scenarios can be fixed reduction percentages in the traffic flows passing through the metering signals. Rathi and Lieberman (1989) simulated a portion of Manhattan, New York, and tested different entry volumes. The simulation results indicated that traffic metering had positive effects on improving network performance. Rathi and Lieberman (1989) did not consider the effects of traffic metering on the delayed vehicles at the metering signals and suggested further studies for better evaluation of traffic metering benefits.

Hajbabaie and Benekohal (2011) evaluated several metering scenarios. They optimized traffic signals for several traffic metering scenarios so that the operational benefits of traffic metering can be better evaluated. They did not find the optimal metering levels, but their results indicated that there existed a metering level at which the network had the best performance. A similar analysis by Medina et al. (2013) suggested that dynamic metering rates offered more efficient network performance compared to fixed metering levels.

2.2. MFD-based traffic metering

The majority of traffic metering studies are based on macroscopic fundamental diagrams (MFDs) for urban networks (Daganzo, 2007; Geroliminis and Daganzo, 2008). The MFDs provide a relationship between the number of vehicles inside a network and its throughput (see 0 for more

details). Therefore, traffic metering controllers can be designed to maintain the number of vehicles in the network within the range that corresponds to the highest throughout. Moreover, accurate estimation of the optimal range requires a low scattered and well-defined MFD that can be expected for networks with homogenous distribution of traffic. Hence, the primary step in designing the controllers is to divide a network into homogenous sub-networks and estimate a well-defined MFD for each sub-network.

Geroliminis and Sun (2011) stated that despite several empirical studies on the existence of MFDs, these diagrams might not exist for all networks. Moreover, they stated that the shape and scatteredness of such diagrams were a function of a network and its control. Furthermore, they empirically and analytically showed that the spatial distribution of vehicle density was a key component that affected the shape and scatteredness of an MFD.

Zhang et al. (2013) used the stochastic cellular automaton model (De Gier et al., 2011) to compare the MFDs of a network under various conditions of adaptive signal settings and demand rates. Their simulation-based analysis showed that the shape of MFDs was influenced by the demand rate such that their case study network reached its capacity at higher densities when demand was uniformly distributed in the network compared to a non-uniform demand distribution case. They also observed that traffic signal settings that distributed density in the network uniformly resulted in an MFD with higher capacity and outflows compared to an adaptive signal control system.

Ji and Geroliminis (2012) proposed a clustering algorithm so that a heterogeneous network could be divided into several sub-networks each with smaller link density variances compared to the original network. This approach allowed finding sub-networks with well-defined MFDs such that each sub-network could be controlled with its boundary gates.

Keyvan-Ekbatani et al. (2012) designed a feedback-based controller that maintained the number of vehicles in a network with the optimal range that could be found from the network MFD. They represented the network dynamics with a first-order non-linear system that was linearized around the optimal accumulation range. The controller determined the total inflows (the number of vehicles that should be allowed to enter the network), and each gate processed a fraction of the total inflow proportional to its saturation flow rate.

Geroliminis et al. (2013) used MFDs to represent the dynamics of a two-region urban network. They formulated an optimal control problem for this system such that the number of completed trips were maximized by regulating the flow of vehicles at the shared boundaries of regions. Hence, the decision variables of the problem were the total number of vehicles that should be allowed to move between the regions. They solved the problem by model predictive control (MPC) so that the errors in the MFD and demand estimation could be considered in the perimeter control decisions. This approach was a high-level control problem that found the overall metered flows rather than optimizing metering rates for each gate. Moreover, the application of this formulation to a real or simulated network required a more detailed representation of flow dynamics such that the effect of gate locations, metering rates at each gate, and the queue of vehicles at the gates could be captured on the network performance.

Haddad et al. (2013) considered perimeter and ramp metering problems in a mixed urban and freeway network. They considered a two-region urban network and one freeway facility that received or sent vehicles to each of these regions. They captured traffic dynamics in the urban regions with MFDs and used an asymmetric cell transmission model (Gomes and Horowitz, 2006) for the freeway facility. Although this study provided invaluable insights into controlling a large-scale and mixed traffic system, the proposed formulation and its MPC-based solution technique

could not capture detailed traffic dynamics and the interactions between the systems. For instance, the effect of queues at the freeway on-ramps was not captured in the dynamics of urban regions. Moreover, the perimeter control between the regions was represented by the total flow that should be transferred between the regions, and thus the effects of locations of gates, queues, and traffic metering rates at each gate were not considered in this modeling approach.

Haddad and Shraiber (2014) stated that the previously discussed MFD-based controllers could regulate the flow of vehicles around a predefined accumulation setpoint, which was the accumulation that corresponded to the highest network throughput in a network's MFD. However, this study linearized the non-linear MFD-based system dynamics around a stochastic setpoint such that uncertainties in the MFD-based dynamics could be considered in the controller design. The numerical analysis of this study showed that the designed controller had a superior performance in different congestion levels, unlike previous controllers that were effective once the congestion level was around the setpoint. This study stated that translating the optimized metered flows to signal indications at metering signals required further studies.

Keyvan-Ekbatani et al. (2015) stated that different parts of a network might experience congestion at different times. Hence, they designed MFD-based controllers for concentric urban networks where congestion spread from the innermost to outermost regions. Accordingly, different perimeter controllers protect each region according to their own MFD and desired congestion level. They also proposed a policy based on queue lengths at the gates so that the total gated flows could be divided between gates by converting flows to signal indications. However, the proposed flow distribution approach might not be optimal because the controllers did not optimize the gate flows for each gate individually. Moreover, the gates could not enforce the same gated flows as were optimized by the controllers due to the conversion of flows to signal indications.

Haddad and Mirkin (2017) stated that central controllers for multi-region networks were not robust to data collection and communication failures. Thus, they designed local adaptive perimeter controllers such that upper-level information about the desired setpoints of regions was shared among them. This information could coordinate the regions and improve the overall system performance. However, this coordination did not ensure the capacity restriction of neighboring regions. Accordingly, congestion might spread from a downstream region to its upstream region and their controllers fail to maintain the vehicle accumulation around their desired setpoints.

Kouvelas et al. (2017) mentioned that the dynamics of multi-region networks were complex and defining setpoints according to the MFD of each region might not be optimal for the entire network. In other words, some regions might need to keep vehicle accumulation more or less than their desired set-point to favor other regions and to improve the overall operations of the network. Accordingly, they proposed an adaptive approach to find controller parameters such as setpoints according to network conditions rather than using fixed values for the entire study period. Although their numerical analysis in a microsimulation environment showed the capability of this approach in improving the network performance, the approach did not ensure the optimality of the parameters. Moreover, the optimized metering rates were distributed between the gates proportional to their saturation flow rates.

Fu et al. (2017) stated that applying perimeter control to networks with high demands that created heterogeneity in a network might not improve network performance. Thus, they proposed a hierarchical perimeter control approach for a two-region network such that upper and lower bounds could be found for the flow control variables in the first level such that system stability can be ensured. Moreover, each region was further divided into sub-regions and sub-regional-level perimeter controllers regulated the flow of vehicles to minimize the heterogeneity among the sub-

regions. Moreover, the sub-regions were defined by a clustering algorithm and used a genetic algorithm to solve the non-linear control problem. However, the proposed bounds might not be ensured because the control problem optimized the total metered flows, but the flows had to be divided between gates and converted to traffic signals so that they could be implemented in a network. These steps created approximations that might violate the stability conditions.

The numerical analysis of Haddad (2017) on a two-region network with MFD-based dynamics showed that the capacity of gates for holding queued vehicles had a significant effect on perimeter control policies. He proposed a control model with explicit constraints on the aggregated gate capacities at the boundaries between regions. Accordingly, the limited capacity of boundary gates could be captured explicitly in the perimeter control policies. However, the capacity restrictions were considered as an aggregated measure in the model. Therefore, the model did not enforce the capacity restrictions of individual gates. In other words, gates could have different capacities, vehicle flows, and metering rates. Therefore, the aggregated capacity limits did not account for the locations and capacity of gates.

Ding et al. (2017) proposed an approach to consider perimeter control and route guidance simultaneously in multi-region networks using MFD-based dynamics. They captured the effect of route guidance by estimating the compliance rate of drivers to the suggested alternative routes that passed through less congested regions. The proposed algorithm had a hierarchical structure such that route guidance and perimeter control minimized the average delay of vehicles sequentially. The proposed approach required predefined homogenous sub-regions and gate locations, while traffic guidance changed the distribution of traffic in a network, and hence the shape of MFDs, number of sub-regions, and the gate locations.

Sirmatel and Geroliminis (2017) developed an MPC for perimeter control and route guidance

using MFD-based network dynamics. They argued that keeping vehicle accumulation around predefined set-points in multi-region networks might be infeasible for perimeter controllers. Moreover, the accumulation-based objective did not necessarily result in the best network performance. Hence, they minimized the total travel time in the MPC formulation. Moreover, the proposed formulation in this study was a mixed-integer non-linear program that was solved with an optimization engine. However, the quality of the solutions in terms of optimality bounds was not discussed in the study.

Yang et al. (2019) considered social welfare besides operational performance as the objectives of perimeter control. They considered priority lanes that could be assigned dynamically to different traffic modes. The tolls for the priority lanes were optimized according to the lane choice information of connected vehicles. The analysis results showed that the proposed approach could distribute the total costs of delays and tolls more uniformly than the case without priority which showed improvements in social welfare.

2.3. Summary

The reviewed literature shows that scenario-based techniques do not provide dynamic metering rates and their efficiency depends on the quality of the pre-defined scenarios. On the other hand, MFD-based techniques regulate the flow of vehicles dynamically and are suitable for large-scale transportation networks. However, MFDs might not exist for all networks. In addition, the MFD-based techniques find aggregated metering rates for all gates and do not consider the effects of gate locations and queued vehicles on the network performance upstream of the gates. The aggregated rates require a heuristic post-processing step to be converted into the metering rates for each individual gate. The MFD-based techniques do not also consider the cooperation of traffic metering and signal control at signalized intersections downstream of the gates.

CHAPTER 3. TRAFFIC METERING PROBLEM FORMULATION

The available traffic metering approaches for urban street networks either fail to provide dynamic optimal metering levels or rely on well-defined MFD, which cannot be derived easily for heterogeneous networks without decomposing them into homogeneous subnetworks (Geroliminis and Sun, 2011; Mazloumian et al., 2010). Besides, such diagrams are prone to change if network characteristics, such as signal control parameters, change (Laval, 2010).

Hajbabaie and Benekohal (2011) explored the operational effects of metering traffic at the perimeter of an urban street network. They tested several metering rates and optimized the timing of signalized intersections for each rate to eliminate the impacts of sub-optimal signal timing parameters on network performance. They did not propose an optimal metering strategy; however, showed that an optimal metering threshold exists that yields more efficient network performance than a no-metering strategy, by keeping flows just below the saturation level and avoiding gridlocks. Keyvan-Ekbatani et al. (2012) exploited the MFD concept and designed a control model and a feedback control structure for metering vehicles at network gates. They found optimal vehicle accumulation for a sample network by simulating the network in a microscopic simulation environment. By setting the optimal range in the proposed feedback controller, they showed sensible improvements in the network performance. Further development of the feedback-based gating model is discussed in (Keyvan-Ekbatani et al., 2015a). This study enhanced the previous model to account for gate signals placed upstream of the protected network area and not exactly at its borders. Remote signals added extra travel time to vehicles traveling between the gate signals and the border of the study region. This travel time was considered as the time delay in the feedback controller. Haddad and Shraiber (2014) argued that a simple feedback controller could regulate traffic just near the optimal vehicle accumulation setpoint; however, network behavior

varied significantly across multiple urban regions, thus traffic scenarios were too diverse to depend on a reliable optimal vehicle accumulation setpoint.

While scenario-based approaches do not provide optimal dynamic metering levels, MFD-based approaches can control metering rates dynamically. However, such approaches need to have a well-defined MFD for a network (or divided regions) and find metering rates to maintain network density within a certain range (Kouvelas et al., 2017). This chapter fills this knowledge gap by finding dynamic optimal metering rates at individual network gates using a mathematical programming approach.

The developed optimization program for optimizing traffic metering rates is discussed in this chapter. The program is based on the cell transmission model (CTM) (Daganzo, 1995, 1994). The CTM discretizes both time and space and provides numerical solutions to the hydrodynamic traffic flow model proposed by Lighthill and Whitham (1955) and Richards (1956). The discretization makes finding the numerical solutions easy and provides the capabilities of considering different traffic flow regimes ranging from undersaturated to oversaturated and gridlock conditions in urban street networks. CTM divides each network link into homogenous segments which are called "cells." Figure 3-1 shows a simple network of four intersections with twelve one-way links that are represented by 52 cells according to the CTM (Hajbabaie et al., 2020).

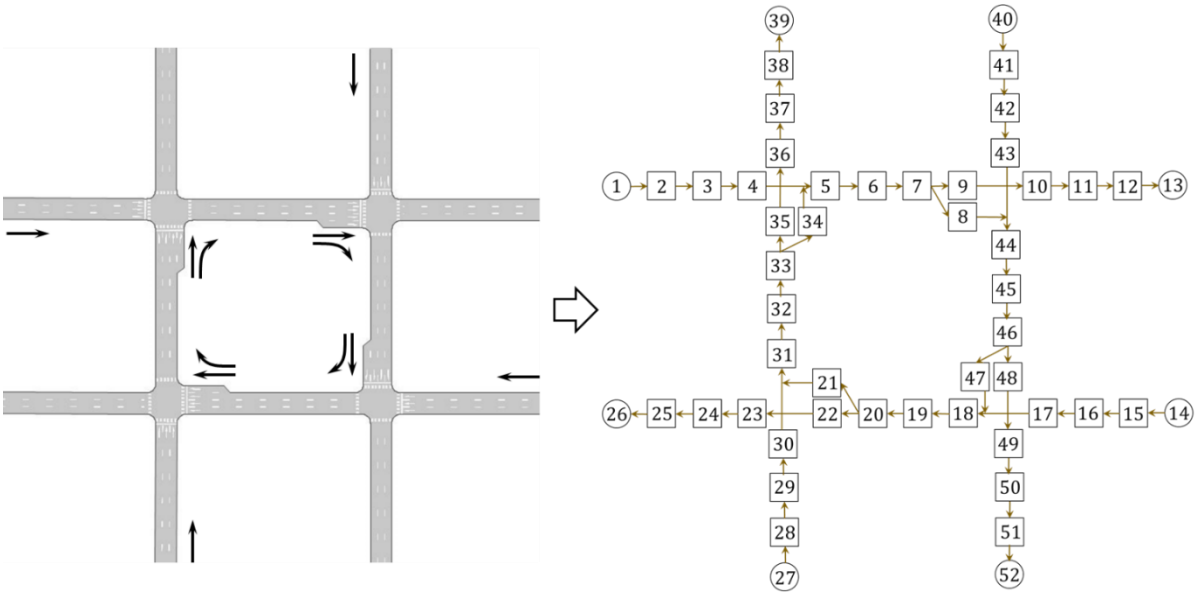


Figure 3-1 Types of cells according to the CTM

A mixed-integer non-linear program (MINLP) is proposed to optimize traffic metering rates at each metering location. The decision variables of the program are the number of vehicles y_{ij}^t that should be allowed to leave gate cell $i \in C_G$ to its successor cell $j \in S(i)$ at time step $t \in T$. Constraints (3-1)-(3-6) ensure that the gate flows do not violate the flow feasibility conditions. Constraints (3-1) ensure that gate flows y_{ij}^t do not exceed the available number of vehicles x_i^t in gate cell $i \in C_G$ at each time step $t \in T$. Moreover, the gate flows should not be more than the available capacity $\delta\left(N_j - x_j^t\right)$ of receiving cell $j \in S(i)$, as shown in Constraints (3-2). In these constraints, N_j is the capacity of a cell in terms of the number of vehicles it can hold, and δ is the ratio of backward to forward shockwave speeds (Daganzo, 1994). Constraints (3-3) and (3-4) guarantee that the gate flows are less than or equal to the saturate flow rates Q_i^t and Q_j^t of the sending and receiving cells, respectively. The number of vehicles x_i^t in gate cell $i \in C_G$ needs to be less than gate capacity N_i for holding the queue of vehicles, see Constraints (3-5). Constraints (3-6) show the non-negativity requirements of the gate flows.

$$y_{ij}^t \le x_i^t \qquad \forall i \in C_G, j \in S(i), t \in T \qquad (3\text{-}1)$$

$$y_{ij}^t \le \delta\left(N_j - x_j^t\right) \qquad \forall i \in C_G, j \in S(i), t \in T \qquad (3\text{-}2)$$

$$y_{ij}^t \le Q_i^t \qquad \forall i \in C_G\ , j \in S(i), t \in T \qquad (3\text{-}3)$$

$$y_{ij}^t \le Q_j^t \qquad \forall i \in C_G\ , j \in S(i), t \in T \qquad (3\text{-}4)$$

$$x_i^t \le N_i \qquad \forall i \in C_G, t \in T \qquad (3\text{-}5)$$

$$y_{ij}^t \ge 0 \qquad \forall i \in C_G, j \in S(i), t \in T \qquad (3\text{-}6)$$

Furthermore, the flow of vehicles between all other cells except the gate cells should follow the flow-density diagram of the CTM that is shown by flow-feasibility Constraints (3-7)-(3-10). Constraints (3-7) ensure that the flow of vehicles in ordinary cells $i \in C\backslash\{C_G, C_S, C_D, C_I\}$ be equal to the minimum of x_i^t, Q_i^t, Q_j^t, and $\delta\left(N_j - x_j^t\right)$. Note that cell $j \in S(i)$ is successor of cell i. The $\min(.)$ function guarantees that the flows follow the flow-density diagram of CTM (Daganzo, 1995), and the flow holing-back problem is eliminated (Lo, 1999). Constraints (3-8) use the variable saturation flow rates $g_i^t Q_i^t$ instead of Q_i^t to account for the effect of signal indications g_i^t on the saturation flow rate of intersection cells $i \in C_I$. The signal indication g_i^t is a binary parameter that is one for green signals and zero otherwise. Note that the signal indications are input to this constraint and should be defined based on the signal settings of a case study network prior to solving the traffic metering problem.

$$y_{ij}^t = \min\left\{x_i^t\ , Q_i^t, Q_j^t, \delta\left(N_j - x_j^t\right)\right\} \qquad \begin{array}{l}\forall i \in C\backslash\{C_G, C_S, C_D, C_I\},\\ j \in S(i), t \in T\end{array} \qquad (3\text{-}7)$$

$$y_{ij}^t = \min\left\{x_i^t\ , g_i^t Q_i^t, Q_j^t, \delta\left(N_j - x_j^t\right)\right\} \qquad \forall i \in C_I, j \in S(i), t \in T \qquad (3\text{-}8)$$

$$y_{ij}^t = \beta_j^t \min\left\{ x_i^t, Q_i^t, \frac{Q_i^t}{\beta_j^t}, \frac{\delta\left(N_j - x_j^t\right)}{\beta_j^t}, \frac{Q_k^t}{\beta_k^t}, \frac{\delta(N_k - x_k^t)}{\beta_k^t}, \frac{Q_m^t}{\beta_m^t}, \frac{\delta(N_m - x_m^t)}{\beta_m^t} \right\} \quad \begin{array}{l} \forall i \in C_D, j \in S(i), k \in S(i), \\ m \in S(i), j \neq k \neq m, t \in T \end{array} \tag{3-9}$$

$$y_{ij}^t = \min\{x_i^t, Q_i^t\} \min\left\{1, \frac{\min\left\{Q_j^t, \delta\left(N_j - x_j^t\right)\right\}}{\sum_{k \in P(j)} \min\{x_k^t, Q_k^t\}}\right\} \quad \forall j \in C_M, i \in P(j), t \in T \tag{3-10}$$

Constraints (3-9) show the flow-feasibility conditions for diverge cells $i \in C_D$ based on dynamic turning ratios β_j^t, β_k^t, and β_m^t for distinct successor cells $j, k, m \in S(i)$. For diverge cells with two successor cells, the m index should be removed from these constraints. Figure 3-3 illustrates constraints for a diverge cell with two and three downstream cells. Dynamic turning ratios β_j^t, β_k^t, and β_m^t are time-dependent and input to the constraint and show the fraction of flow that enters each of the downstream cells. Hence, the summation of the turning ratios should add up to one, i.e., $\beta_j^t + \beta_k^t + \beta_m^t = 1$. Furthermore, Constraints (3-9) ensure the fist-in-first-out conditions for diverge cells. In other words, if cell $j \in S(i)$ is full and cannot accommodate any vehicles, the flow of other downstream cells will be zero. This condition captures the capacity reductions at intersections due to queue spillovers or spillbacks.

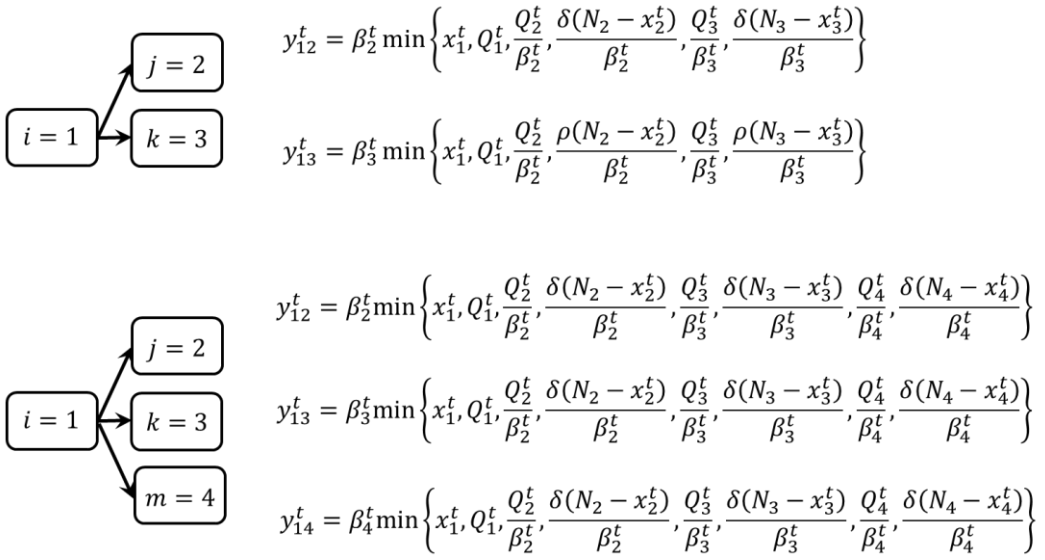


Figure 3-2 Cell representation of a sample network

We define five cell types as shown in Figure 3-1. Gate cells represent the entry points to a network. The inflow to gates follows a predefined demand profile, while the outflow of the gates will be defined based on the optimized traffic metering rates. Sink cells absorb the flow of vehicles leaving the network. Merge cells receive vehicles from multiple immediate upstream cells and diverge cells send vehicles to multiple immediate downstream cells. Ordinary cells are the cells that receive vehicles from and send vehicles to only one immediate cell. We also used intersection cells which are like ordinary cells but with a variable saturation flow rate that represents the effect of signals. The set of all network, gate, sink, merge, diverge, and intersection cells are denoted by respectively C, C_G, C_S, C_M, C_D, and C_I, Therefore, the set of ordinary cells are all other cells that is $C \backslash \{C_G, C_S, C_M, C_D, C_I\}$. Moreover, the set of all successor and predecessor cells of cell $i \in C$ are shown respectively by $S(i)$ and $P(i)$. The predecessor and successor cells of cell $i \in C$ are those that are immediately upstream or downstream of the cell, respectively.

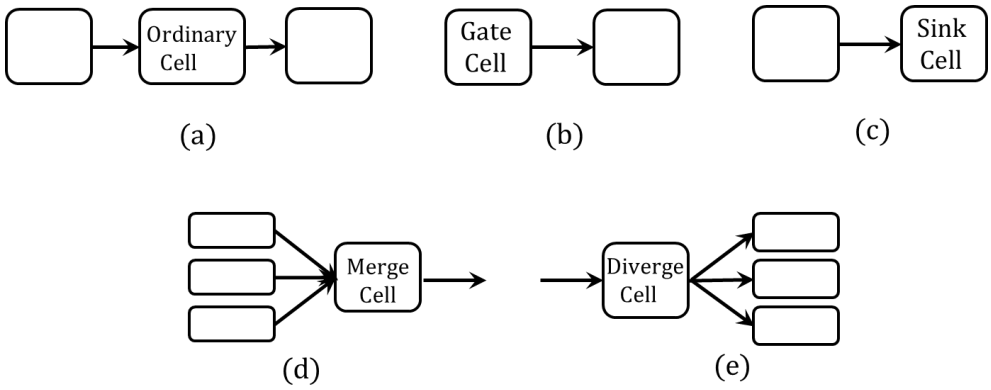
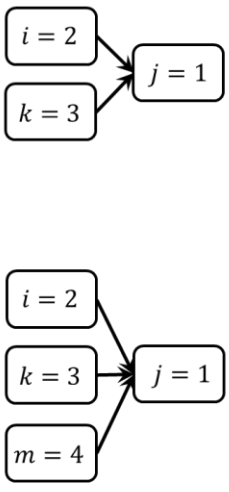


Figure 3-3 Illustration of the flow-feasibility constraints for two diverge cells with two and three downstream cells

Flow feasibility conditions of merge cells $j \in C_M$ is shown by Constraints (3-10): the available capacity of the merge cell $\min\{Q_j^t, \delta(N_j - x_j^t)\}$ will be divided between its upstream cells $k \in P(j)$ proportional to their flows $\min\{x_k^t, Q_k^t\}$. This constraint is further illustrated in Figure (3-4).



$$y_{21}^t = \min\{x_2^t, Q_2^t\} \min\left\{1, \frac{\min\{Q_1^t, \delta(N_1 - x_1^t)\}}{\min\{Q_2^t, x_2^t\} + \min\{Q_3^t, x_3^t\}}\right\}$$

$$y_{31}^t = \min\{x_3^t, Q_3^t\} \min\left\{1, \frac{\min\{Q_1^t, \delta(N_1 - x_1^t)\}}{\min\{Q_2^t, x_2^t\} + \min\{Q_3^t, x_3^t\}}\right\}$$

$$y_{21}^t = \min\{x_2^t, Q_2^t\} \min\left\{1, \frac{\min\{Q_1^t, \delta(N_1 - x_1^t)\}}{\min\{Q_2^t, x_2^t\} + \min\{Q_3^t, x_3^t\} + \min\{Q_4^t, x_4^t\}}\right\}$$

$$y_{31}^t = \min\{x_3^t, Q_3^t\} \min\left\{1, \frac{\min\{Q_1^t, \delta(N_1 - x_1^t)\}}{\min\{Q_2^t, x_2^t\} + \min\{Q_3^t, x_3^t\} + \min\{Q_4^t, x_4^t\}}\right\}$$

$$y_{41}^t = \min\{x_4^t, Q_4^t\} \min\left\{1, \frac{\min\{Q_1^t, \delta(N_1 - x_1^t)\}}{\min\{Q_2^t, x_2^t\} + \min\{Q_3^t, x_3^t\} + \min\{Q_4^t, x_4^t\}}\right\}$$

Figure 3-4 Illustration of the flow-feasibility conditions for merge cells with two and three upstream cells

The flow conservation concept is shown by Constraints (3-11)-(3-13) for different cell types. The number of vehicles x_i^{t+1} in gate cell $i \in C_G$ at time step $t + 1 \in T$ is equal to the number of vehicles x_i^t that were available in the cell in the previous time step t plus the network demand D_i^t

minus the number of vehicle $\sum_{j\in S(i)} y_{ij}^t$ that leave the gate cell to its downstream cells $j \in S(i)$. In these constraints, the number of vehicles entering that gate cells is determined based on a predefined and time-dependent demand profile D_i^t, and the number of vehicles that leave the gate cells are indeed the optimized gate flows. Constraints (3-12) show the flow conservation for sink cell $j \in C_S$. The sink cells are assumed to have infinite capacity and absorb all the vehicles that leave the network. With this representation, the summation of vehicles in all sink cells at each time step shows the accumulated number of completed trips up to that time step. Constraints (3-13) show the flow conservation requirements for ordinary cells.

$$x_i^{t+1} = x_i^t + D_i^t - \sum_{j\in S(i)} y_{ij}^t \qquad \forall i \in C_G, t \in T \tag{3-11}$$

$$x_j^{t+1} = x_j^t + \sum_{i\in P(j)} y_{ij}^t \qquad \forall j \in C_S, t \in T \tag{3-12}$$

$$x_i^{t+1} = x_i^t + \sum_{k\in P(i)} y_{ki}^t - \sum_{j\in S(i)} y_{ij}^t \qquad \forall i \in C\backslash\{C_S, C_G\}, t \in T \tag{3-13}$$

The objective function of the problem is shown in (3-14): it maximizes the cumulative number of vehicles in all sink cells. This objective function maximizes the network throughput and is shown to be a suitable objective function for oversaturated conditions (Hajbabaie et al., 2011, 2010; Medina et al., 2010).

$$\text{maximize } Z = \sum_{\forall i\in C_S} \sum_{\forall t\in T} x_i^t \tag{3-14}$$

In addition, it can be shown that the objective function (3-14) is equivalent to the minimization of the total travel time but with fewer decision variables compared to travel time minimization.

Proposition 3-1. The objective function (3-14) is equivalent to the total travel time minimization.

Proof. We can write equation (3-15) for each time step $\tau \in T$ based on vehicle flow conservation principle.

$$\sum_{t=1}^{\tau}\sum_{i\in C_r} D_i^t + \sum_{i\in C\setminus C_s} x_i^0 = \sum_{i\in C\setminus C_s} x_i^\tau + \sum_{i\in C_s} x_i^\tau \tag{3-15}$$

In equation (3-15), the total number of vehicles that have entered the network from the beginning of analysis $t = 1$ up to time step $\tau \in T$ ($\sum_{t=1}^{\tau}\sum_{i\in C_G} D_i^t$) plus the number of vehicles that were present at the network before the start of the analysis at time $t = 0$ ($\sum_{i\in C\setminus C_s} x_i^0$) is equal to the total number of vehicles that are now in the network ($\sum_{\forall i\in C\setminus C_s} x_i^\tau$) at time $\tau \in T$ plus the total number of vehicles that have left the network ($\sum_{\forall i\in C_s} x_i^\tau$). If we write this equation for each $\tau \in T$ ranging from $\tau = 1$ to $\tau = T$ and sum the equations, we will have equation (3-16).

$$\sum_{\tau=1}^{T}\sum_{t=1}^{\tau}\sum_{i\in C_r} D_i^t + \sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^0 = \sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^\tau + \sum_{\tau=1}^{T}\sum_{i\in C_s} x_i^\tau \tag{3-16}$$

In the latter equation, $\sum_{\tau=1}^{T}\sum_{t=1}^{\tau}\sum_{i\in C_r} D_i^t$ and $\sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^0$ are constant values and $\sum_{\tau=1}^{T}\sum_{i\in C_s} x_i^\tau$ is the cumulative number of vehicles in the sink cells. Furthermore, travel time can be found as, $\sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^\tau$ multiplied by the duration τ of each time step (Beard and Ziliaskopoulos, 2006). Suppose that the summation of constant values is Θ. Hence, the cumulative number of vehicles in the sink cells is equal to (3-17):

$$\sum_{\tau=1}^{T}\sum_{i\in C_s} x_i^\tau = \Theta - \sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^\tau \tag{3-17}$$

Consequently, maximization of the cumulative number of vehicles in the sink cells $\sum_{\tau=1}^{T}\sum_{i\in C_s} x_i^\tau$ is equal to the minimization of total travel time, $\sum_{\tau=1}^{T}\sum_{i\in C\setminus C_s} x_i^\tau$ (since the duration τ of each time step is a fixed number). ■

Based on the discussed constraints and objective function, the optimization program for optimizing the gate flows can be summarized as program (P1).

(P1) maximize $Z = \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t$, subject to (3-1)-(3-13) (3-18)

CHAPTER 4. OPTIMAL SOLUTIONS TO THE TRAFFIC METERING PROBLEM

This section presents an approach that finds the optimal solutions for program (P1), see Chapter 3, within an optimality gap. Note that (P1) is a complex optimization program due to the excessive number of mixed-integer decision variables and nonlinear constraints. The decision variables of the program are continuous, but the representation of the min(.) operators in the flow-feasibility constraints (3-7)-(3-10) requires adding auxiliary dummy variables to the program. For instance, the standard representation of Constraint (3-7) is equivalent to Constraints (4-1)-(4-10) using the big-M technique. Constraints (4-1)-(4-4) ensure the flow feasibility conditions while Constraints (4-4)-(4-10) ensure that flow of vehicles y_{ij}^t from each cell $i \in C\backslash\{C_G, C_S, C_D, C_I\}$ to cell $j \in S(i)$ at each time step $t \in T$ is exactly equal to either x_i^t, Q_i^t, Q_j^t, or $\delta(N_j - x_j^t)$. In other words, the addition of Constraints (4-4)-(4-10) with auxiliary binary variables θ_{ij}^t, ϑ_{ij}^t, ψ_{ij}^t, and χ_{ij}^t sets the flow of vehicles equal to the one of the right-hand side values of Constraints (4-1)-(4-4). In the latter constraints, M is an arbitrarily large number.

$$y_{ij}^t \leq x_i^t \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}1)$$

$$y_{ij}^t \leq Q_i^t \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}2)$$

$$y_{ij}^t \leq Q_j^t \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}3)$$

$$y_{ij}^t \leq \delta(N_j - x_j^t) \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}4)$$

$$y_{ij}^t \geq x_i^t - M(1 - \theta_{ij}^t) \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}5)$$

$$y_{ij}^t \geq Q_i^t - M(1 - \vartheta_{ij}^t) \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}6)$$

$$y_{ij}^t \geq Q_j^t - M(1 - \psi_{ij}^t) \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}7)$$

$$y_{ij}^t \geq \delta(N_j - x_j^t) - M(1 - \chi_{ij}^t) \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}8)$$

$$\theta_{ij}^t + \vartheta_{ij}^t + \psi_{ij}^t + \chi_{ij}^t = 1 \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}9)$$

$$\theta_{ij}^t, \vartheta_{ij}^t, \psi_{ij}^t, \chi_{ij}^t \in \{0,1\} \qquad \forall i \in C\backslash\{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \qquad (4\text{-}10)$$

Accordingly, once the flow-feasibility constraints are converted to standard constraints with the big-M technique, many auxiliary binary variables will be added to the program and will significantly increase the problem complexity. Hence, the conventional optimization techniques in commercial solvers such as CPLEX (CPLEX, 2009) cannot solve the program for medium-sized networks in a reasonable amount of time. We tackled this issue by developing a novel decomposition solution technique to solve (P1) more efficiently. Decomposition techniques have been used widely to solve complex transportation problems (e.g., Hajibabai et al., 2014; Hajibabai and Ouyang, 2013; Hajibabai and Saha, 2019).

The general idea of the solution technique is that once the gate flows are known, all other variables of the program can be determined with a CTM simulation without solving any optimization programs. Figure 4-1 shows the idea of the solution technique for a link with one gate. According to the figure, the gate can be separated from the rest of the link by relaxing the constraints that connect gate cell i to its successor cell $j \in S(i)$. Then, an optimization program for the gate can be solved to find gate flows y_{ij}^t and use them as input for the rest of the link to find all other variables. Then, by connecting the two steps and iterating between them, the optimal solutions can be found.

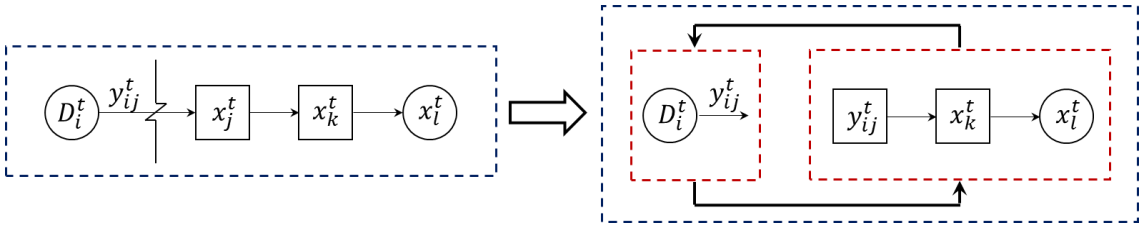


Figure 4-1 General decomposition idea for one link

We utilized the generalized Benders decomposition technique (Benders, 1962; Geoffrion,

1972) to implement the discussed ideas such that the convergence and optimality of the solution technique can be shown. Benders decomposition technique is developed for solving a class of complex optimization programs that can be solved efficiently once a set of decision variables are temporarily set to predefined values. In other words, by fixing the values of the complicating decision variables, the remaining optimization program can be solved easier than the original problem.

Accordingly, the Benders decomposition technique decomposes the original problem into a primal and a mater problem. The primal problem is equivalent to the original problem in which the values of the complicating variables are temporarily fixed. The solutions of the primal problem are the optimal values of the non-complicating variables, the optimized value of the objective function, and the dual values of the constraints that include both complicating and non-complicating variables. These values will be used to construct the master problem. The master problem is the relaxed dual to the original problem whose decision variables are the complicating variables that were fixed in the primal problem.

In other words, the primal problem solves the original problem with temporarily fixed values of gate flows ($y_{ij}^{t}: \forall i \in C_G, j \in S(i), t \in T$) and relaxing their associated constraints. Optimal solutions to the primal problem are lower bounds to the original problem. Then, the master problem is constructed by including constraints on just the complicating variables and the Lagrange function, which is also called a Benders cut. Optimal solutions to the master problem are upper bounds to the original problem and find new values for complicating decision variables. At each iteration, a new Lagrange function will be added to the master problem and iterating between the two sub-problems continues until an appropriate stopping criterion is met.

Based on this algorithm, the Primal problem with known gate flows ($\hat{y}_{ij}^{t}: \forall i \in C_G, j \in S(i), t \in$

T) is shown below. In the Primal problem, the set of cells immediately after the gate cells is denoted by C_A.

Primal problem

$$\text{maximize } Z = \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t \tag{4-11}$$

Subject to:

$$\hat{y}_{ij}^t \le x_i^t \qquad \forall i \in C_G, j \in S(i), t \in T \tag{4-12}$$

$$\hat{y}_{ij}^t \le \delta\left(N_j - x_j^t\right) \qquad \forall i \in C_G, j \in S(i), t \in T \tag{4-13}$$

$$x_i^t \le N_i \qquad \forall i \in C_G, t \in T \tag{4-14}$$

$$y_{ij}^t = \min\left\{x_i^t, Q_i^t, Q_j^t, \delta\left(N_j - x_j^t\right)\right\} \qquad \forall i \in C \backslash \{C_G, C_S, C_D, C_I\}, j \in S(i), t \in T \tag{4-15}$$

$$y_{ij}^t = \min\left\{x_i^t, g_i^t Q_i^t, Q_j^t, \delta\left(N_j - x_j^t\right)\right\} \qquad \forall i \in C_I, j \in S(i), t \in T \tag{4-16}$$

$$y_{ij}^t = \beta_j^t \min\left\{\begin{matrix} x_i^t, Q_i^t, \frac{Q_i^t}{\beta_j^t}, \frac{\delta\left(N_j - x_j^t\right)}{\beta_j^t}, \frac{Q_k^t}{\beta_k^t}, \frac{\delta(N_k - x_k^t)}{\beta_k^t}, \\ \frac{Q_m^t}{\beta_m^t}, \frac{\delta(N_m - x_m^t)}{\beta_m^t} \end{matrix}\right\} \qquad \begin{matrix} \forall i \in C_D, j \in S(i), k \in S(i), \\ m \in S(i), j \ne k \ne m, t \in T \end{matrix} \tag{4-17}$$

$$y_{ij}^t = \min\{x_i^t, Q_i^t\} \min\left\{1, \frac{\min\left\{Q_j^t, \delta\left(N_j - x_j^t\right)\right\}}{\sum_{k \in P(j)} \min\{x_k^t, Q_k^t\}}\right\} \qquad \forall j \in C_M, i \in P(j), t \in T \tag{4-18}$$

$$x_i^{t+1} = x_i^t + \hat{y}_{ji}^t - \sum_{k \in S(i)} y_{ik}^t \qquad \forall i \in C_A, j \in P(i), t \in T \tag{4-19}$$

$$x_j^{t+1} = x_j^t + \sum_{i \in P(j)} y_{ij}^t \qquad \forall j \in C_S, t \in T \tag{4-20}$$

$$x_i^{t+1} = x_i^t + \sum_{k \in P(i)} y_{ki}^t - \sum_{j \in S(i)} y_{ij}^t \qquad \forall i \in C \backslash \{C_S, C_G, C_A\}, t \in T \tag{4-21}$$

The decision variables of the Primal problem are the number of vehicles that should be in cells

immediately downstream of gate cells (C_A) to maximize the objective function. We can simplify the Primal problem by taking into account that traffic metering is expected to be effective in oversaturated flow conditions. In other words, in oversaturated conditions, network gates meter some vehicles ($y_{ij}^t \leq x_i^t \ , \forall i \in C_G$) if metering improves the objective function. On other hand, if the maximum network throughput can be achieved by sending more vehicles than the available vehicles at the gates ($y_{ij}^t > x_i^t \ , \forall i \in C_G$), traffic metering is not necessary. Therefore, Constraints (4-12) can be automatically satisfied in oversaturated flow conditions and will be removed from the Primal problem. We can also assume that the gates are located at locations with enough capacity to hold queued vehicles for the sake of simplicity. Thus, Constraints (4-14) are also relaxed from the Primal problem.

Using Propositions 1 and 2, it is shown that the simplified Primal problem has only one feasible solution that can be found by a CTM simulation. For this purpose, we need to first show that Constraints (4-13) are not binding.

Proposition 4-1. If the number of vehicles in cell $j \in C_A$ at time step $t \in T$ is less than its saturation flow rate Q_j, or in other words $x_j^t \leq Q_j$, Constraints (4-13) in the Primal problem (4-11), (4-13), (4-15)-(4-21) are not binding.

Proof. Before proving the statement, we should obtain the relationship between the saturation flow rate Q_j and capacity N_j of cells. Consider the flow-density relationship in the shape of a trapezoid for cell $j \in C$, as shown in Figure 4-2.

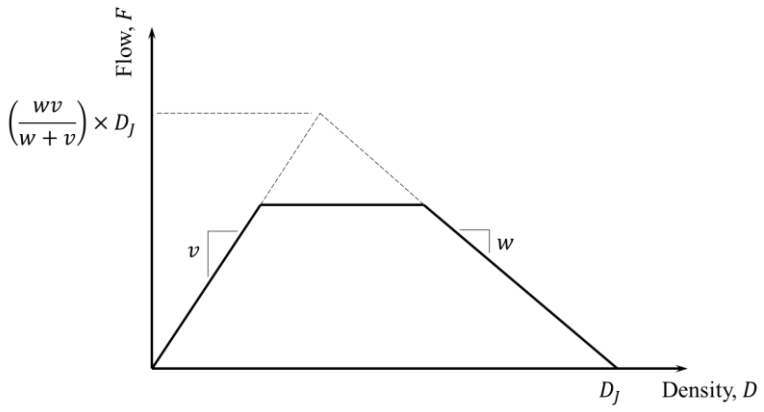


Figure 4-2 Flow-density relationship in cell transmission model

Based on the relationship of Figure 4-2, flow is less than or equal to the intersection point of extending the trapezoid legs, or:

$$F \leq \left(\frac{vw}{w+v}\right) D_J \tag{4-22}$$

For the length of one cell and one time step t, we have:

$$F = Q_j/t \qquad \forall j \in C \tag{4-23}$$

$$D_J = N_j/vt \qquad \forall j \in C \tag{4-24}$$

As such, after substitution of (4-23) and (4-24) in (4-22), we will find:

$$Q_j \leq \left(\frac{\delta}{1+\delta}\right) N_j. \qquad \forall j \in C \tag{4-25}$$

Feasible gate flows (solution to the master problem) ensure that $y_{ij}^t \leq Q_j$. Furthermore, for Constraints (4-13) not to be binding, we should have $Q_j \leq \delta\left(N_j - x_j^t\right)$. Assuming that $x_j^t \leq Q_j$, the lowest value that $\delta\left(N_j - x_j^t\right)$ can take will be $\delta\left(N_j - Q_j\right)$. Therefore, we will have $Q_j \leq \delta\left(N_j - Q_j\right)$ that is equivalent to equation (4-25), proving that Constraints (4-13) are not binding, which concludes the proof. ■

Since Constraints (4-13) are not binding, we can eliminate them from the Primal problem without any changes in the optimal solution. However, to ensure that the constraint is not binding we need to have $x_j^t \leq Q_j$. To achieve this goal, we add $\left(Q_j - x_j^t\right)$ term with a penalty of $\rho > 0$ to the objective function of the Primal problem. While this is not added as a constraint, the penalty term in the objective function will help find solutions with fewer vehicles in cells immediately downstream of gate cells than their saturation flow rate Q_j. Therefore, the updated objective function of the Primal problem is as follows:

$$Z = \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t - \rho \sum_{\forall j \in C_A} \sum_{\forall t \in T} \left(x_j^t - Q_j\right) \tag{4-26}$$

Now, without Constraints (4-13), we can show that the simplified Primal problem is indeed a cell transmission simulation rather than an optimization problem. In other words, if we know gate flows, we can find all other variables and the network throughput by running a CTM simulation rather than solving an MILP problem.

Proposition 4-2. With known gate flows, the Primal problem (4-15)-(4-21), (4-26) has just one feasible solution, which can be found by a single CTM simulation run.

Proof. If we separate the constraints of problem (4-15)-(4-21), (4-26) over t, starting from $t = 0$, y_{ij}^0 for all cells $i \in C \backslash C_G$ and $j \in S(i)$ can be calculated from Constraints (4-15)-(4-18). Besides, their values are unique due to the equality constraints. Constraints (4-19)-(4-21) show how cell occupancies can be found at $t = 1$, x_i^1, for all $i \in C \backslash C_G$, based on known initial cell occupancies and flows x_i^0 and y_{ij}^0 ($j \in S(i)$ for $i \in C \backslash C_G$). Continuing the same procedure, all other variables

in the next time steps can be calculated from the previous steps with unique values. As such, Primal problem (4-15)-(4-21), (4-26) has only one feasible solution that can be found by a simulation run.

■

The previous discussions showed how the Primal problem can be simplified a reduced to a CTM simulation. The second step of the solution technique is deriving the Lagrange function. This function is constructed by the summation of Primal problem's objective function and penalty factors for the relaxed constraints as shown in (4-27). Note that λ^{*t}_i are the dual values of Constraints (4-19) for cell $i \in C_A$ at time $t \in T$, and their values are found by solving the Primal problem. Besides, values of variables marked with an asterisk are found from the solutions of the Primal problem.

$$L^*(y^t_{ij}, \lambda^{*t}_i) = \sum_{\forall i \in C_S} \sum_{\forall t \in T} x^{*t}_i + \sum_{\forall i \in C_A} \sum_{\forall j \in P(j)} \sum_{\forall k \in S(i)} \sum_{\forall t \in T} \lambda^{*t}_i \left(x^{*t+1}_i - x^{*t}_i + y^t_{ji} - y^{*t}_{ik}\right) \quad (4\text{-}27)$$

Finally, the objective function of the Master problem is to maximize μ, where the value of the Lagrange function has to be more than or equal to μ, see Constraints (4-32). The Master problem is an LP in which k shows the Benders algorithm iteration count. Note that at each Benders iteration, one optimality cut is added to the Master problem.

Master problem:

maximize μ (4-28)

Subject to:

$y^t_{ij} \le Q^t_i \qquad \forall i \in C_G, j \in S(i), t \in T$ (4-29)

$y^t_{ij} \le Q^t_j \qquad \forall i \in C_G, j \in S(i), t \in T$ (4-30)

$$x_i^{t+1} = x_i^t + D_i^t - \sum_{j \in S(i)} y_{ij}^t \qquad \forall i \in C_G, t \in T \tag{4-31}$$

$$L^{*k}\left(y_{ij}^t, \lambda^{*t}_i\right) \geq \mu \qquad k = 1, \dots, K \tag{4-32}$$

$$y_{ij}^t \geq 0 \qquad \forall i \in C_G, j \in S(i), t \in T \tag{4-33}$$

It should be noted that as we are not using any optimization algorithm for solving the Primal problem and thus, we cannot find dual values for Constraints (4-32) automatically from the optimization algorithm. Therefore, dual values are found by increasing the right-hand side of Constraints (4-32) by one unit and obtaining the change in the objective function. Accordingly, we need one simulation run to find all decision variables of the Primal problem for fixed gate flows, and several other simulation runs to find dual values. Finally, the steps of the proposed algorithm to find optimum gate flows are as follows:

Step 0: Set $k \leftarrow 0$ and find an initial feasible solution $\hat{y}_{ij}^t$ for each $i \in C_G$, $j \in S(i)$, and $t \in T$. Setting all $\hat{y}_{ij}^t$ equal to zero is always a feasible solution to the problem.

Step 1: Set $k \leftarrow k + 1$ and run a CTM simulation with the known gate flows, $\hat{y}_{ij}^t$, and find Z^*, x^{*t}_i, and y^{*t}_{ij} for each $i \in C \backslash C_G$, $j \in S(i)$, and $t \in T$.

Step 2: For each gate $i \in C_G$ and each time step t, increase $\hat{y}_{ij}^t$ by one unit, run a CTM simulation and find Z_i^t. Set $\lambda^{*t}_i = Z_i^t - Z^*$.

Step 3: Construct the Lagrange function, $L^{*k}\left(y_{ij}^t, \lambda^{*t}_i\right)$.

Step 4: Add $L^{*k}\left(y_{ij}^t, \lambda^{*t}_i\right)$ to the Master problem and solve program (4-28)-(4-33) to find μ^* and a new set of $\hat{y}_{ij}^t$ for all $i \in C_G$, $j \in S(i)$, and $t \in T$.

Step 5: If stopping criterion $(\mu^* - Z^*)/\mu^* \leq \varepsilon$ is met, $\hat{y}_{ij}^t$ are the optimal solutions. Otherwise, go to Step 1.

The above steps are represented in Figure 4-3.

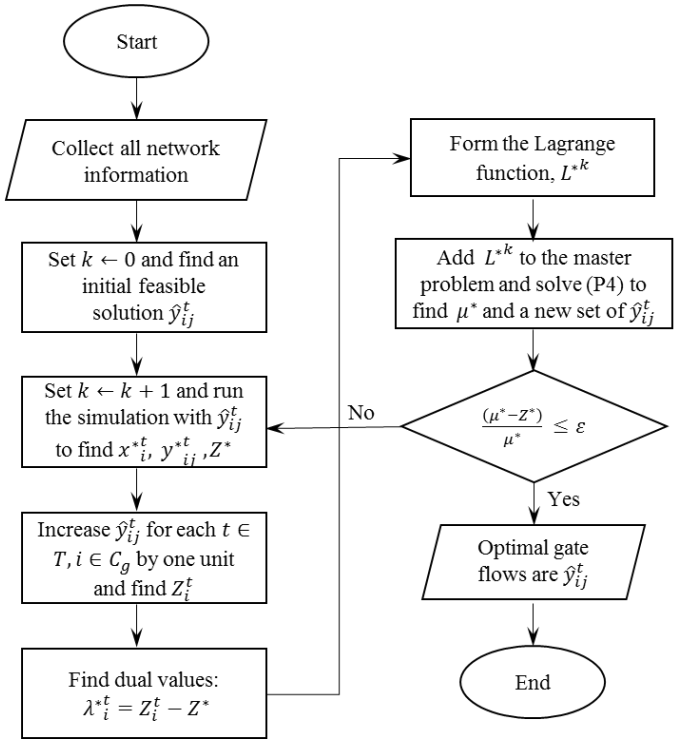


Figure 4-3 The proposed algorithm for finding optimal gate flows

4.2. Case study

We used a portion of the urban street network in downtown Springfield, Illinois to evaluate the proposed traffic metering methodologies. The case study network is shown in Figure 4-4. This network consists of 20 intersections with a combination of one-way and two-way streets. The number of lanes in the streets varies between 1 to 3 as is shown in Figure 4-4. We placed 13 gates at the boundary of this network to regulate the flow of vehicles entering the network according to

the optimized metering rates. The network is analyzed with two demand profiles that are shown in Figure 4-5. The total analysis period is 750 time steps (75 minutes) with 150 time steps for network loading. In addition, we optimized signal timings of the intersections inside the network with the approach presented in (Hajbabaie, 2012; Hajbabaie and Benekohal, 2013, 2015) for each demand profile so that the effects of traffic metering on the network performance can be better evaluated. Other information on the case study network such as cell characteristics is summarized in Table 4-1.

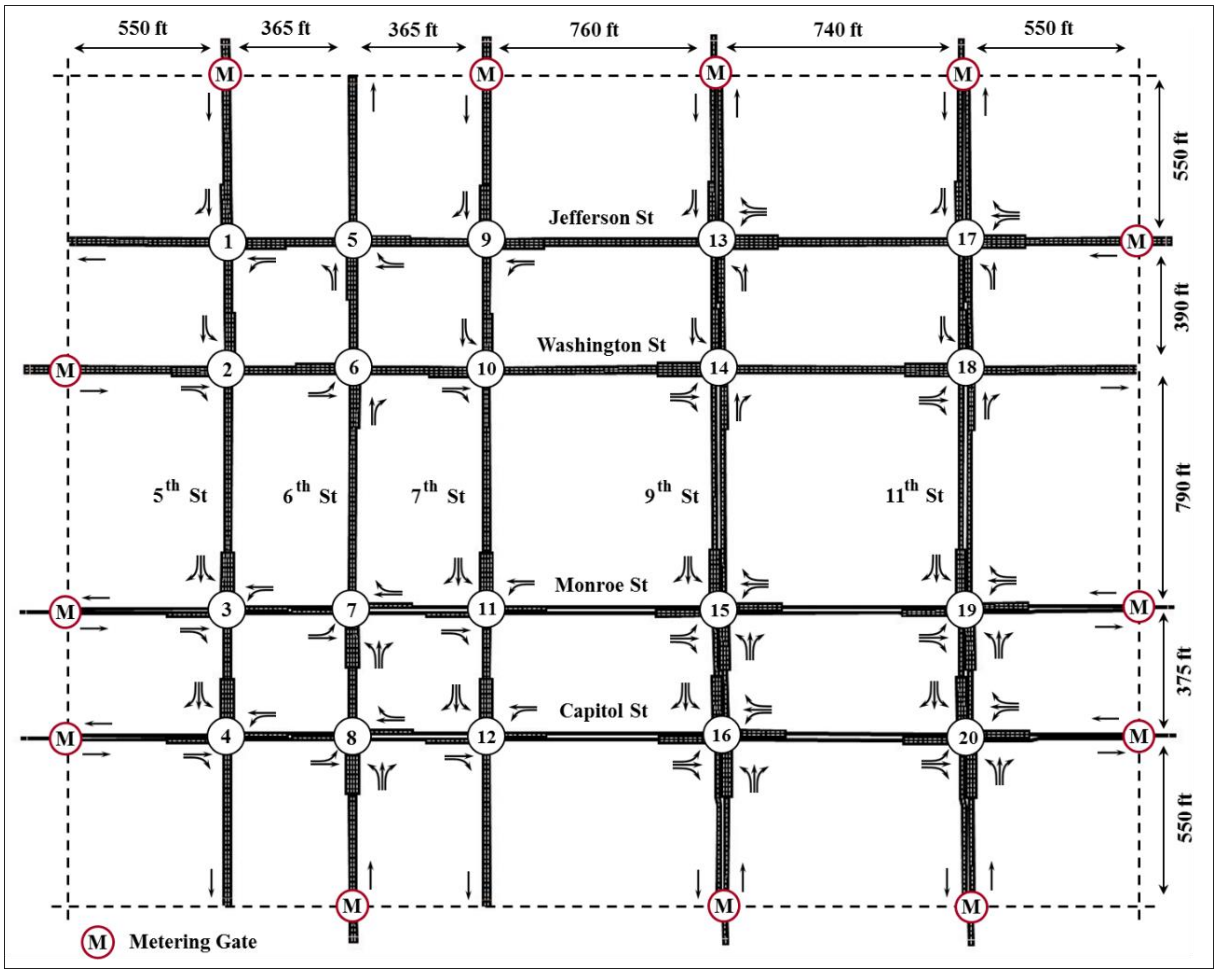


Figure 4-4 Downtown Springfield, Illinois that is used as the case study network

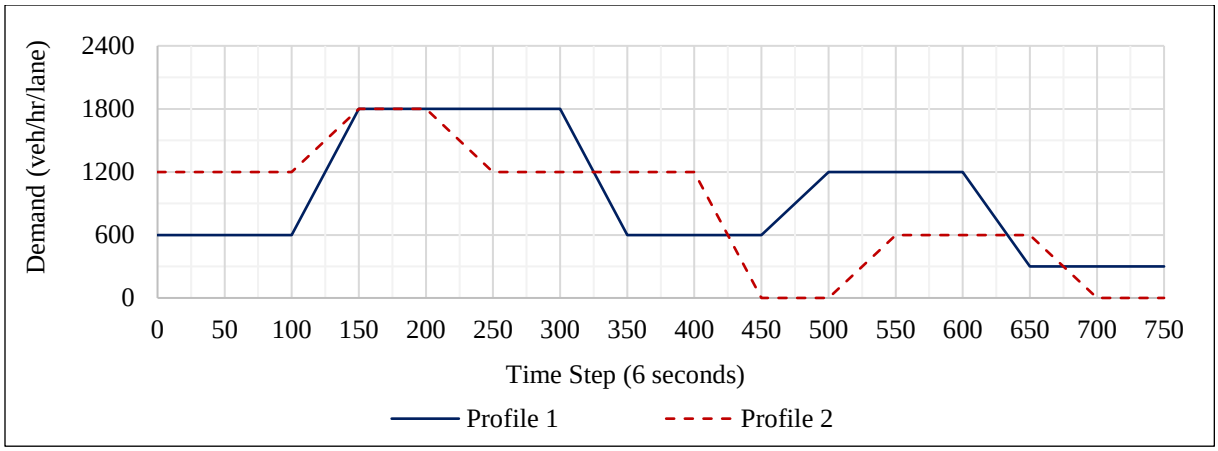


Figure 4-5 The demand profiles that are used for the analysis of the case study network

Table 4-1 Characteristics of the case study network

Element	Value
Free-flow speed (mph)	25
Saturation flow rate of links (veh/hr/lane)	1800
Time step duration (s)	6
Prediction horizon (min)	15
Total number of cells	316
Length of cells (ft)	225
Saturation flow rate of cells (veh/lane/time step)	3
Jam density of cells (veh/lane/cell)	12

4.3. Numerical results

We evaluated the performance of the case study network in the simulation (SIM) and optimal metering (OPT) scenarios for two demand profiles. The results in Table 4-2 show that traffic metering in the OPT scenario increased the network throughput by 5.5% and 3.4%, respectively in Demand Profiles 1 and 2. In addition, the travel time of vehicles inside the network (excluding delayed vehicles at the gates) was reduced by 34.2% in Demand Profile 1 and 30.8% in Demand

Profile 2. Delay of vehicles inside the network also shows a similar trend of 42.3% and 37.2% decrease using traffic metering compared to the simulation scenarios in the two demand profiles. These results indicate that traffic metering could significantly improve the operations of the case study inside the protected region by gates.

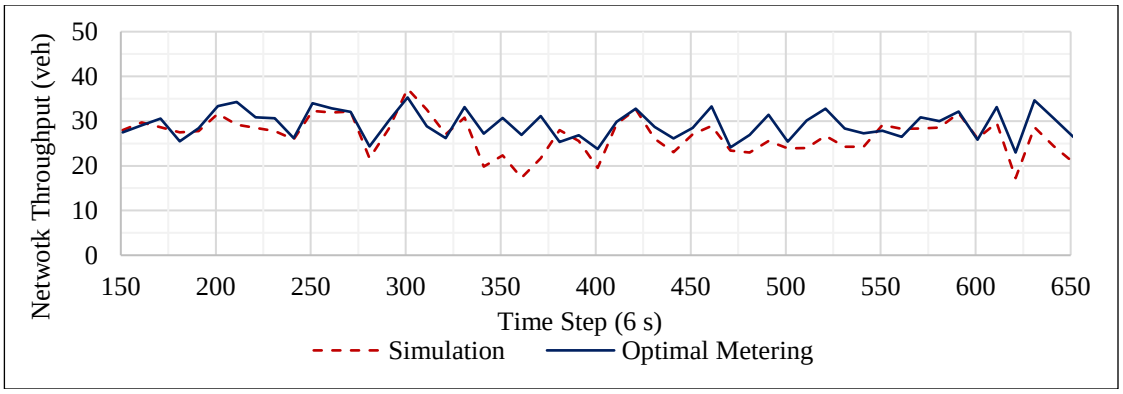
The discussed improvements were achieved by delaying some vehicles at the gates. Therefore, the delay of vehicles at the gates was increased by 2.3% and 5.2% in OPT compared to the SIM scenario in respectively Demand Profiles 1 and 2. However, if we consider all vehicles including vehicles at the gates and inside the network, the overall system-level delay was reduced by 5.7% and 2.9% due to traffic metering, see Table 4-2.

Note that the benefits of traffic metering in Table 6-2 are in fact the improvements that could be achieved in addition to signal timing optimization. In other words, the traffic signal timings of the network were optimized before the application of traffic metering. Hence, we could achieve further improvement in traffic operations with another level of control using traffic metering.

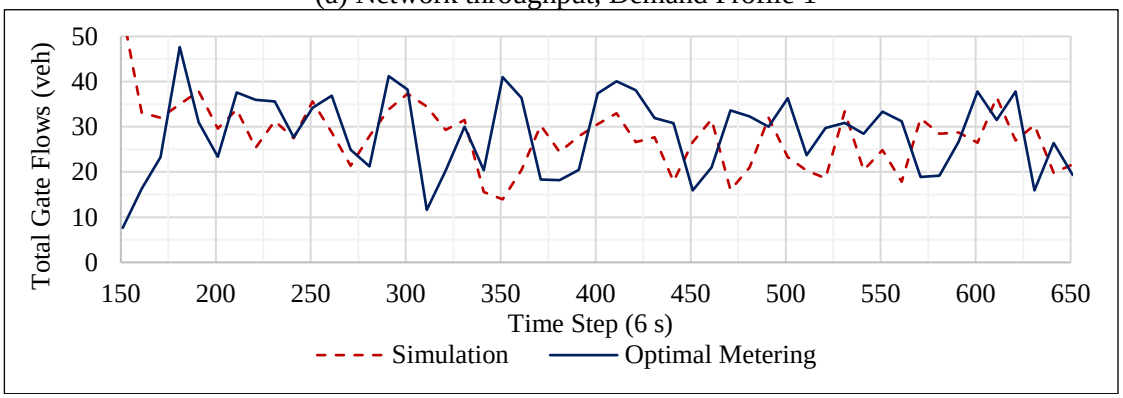
Table 4-2 Network performance measures for different scenarios

Criterion	Simulation (SIM)	Optimal Metering (OPT)	% Difference
	Demand Profile 1		
Completed trips (veh)	18,681	19,708	5.5
Travel time of vehicles inside the network (min)	141,088	92,861	-34.2
Travel time of vehicles at the gates (min)	530,493	542,715	2.3
System-level travel time (min)	671,581	635,576	-5.4
Delay of vehicles inside the network (veh-min)	116,508	67,252	-42.3
Delay of vehicles at the gates (veh-min)	527,672	539,894	2.3
System-level delay (veh-min)	644,180	607,146	-5.7
System-level average speed (mph)	1.4	1.5	10.3
	Demand Profile 2		
Number of completed trips (veh)	19,265	19,926	3.4
Travel time of vehicles inside the network (min)	147,183	101,917	-30.8
Travel time of vehicles at the gates (min)	518,380	545,426	5.2
System-level travel time (min)	665,563	647,343	-2.7
Delay of vehicles inside the network (veh-min)	123,193	77,406	-37.2
Delay of vehicles at the gates (veh-min)	516,062	543,107	5.2
System-level delay (veh-min)	639,255	620,513	-2.9
System-level average speed (mph)	1.3	1.4	5.2

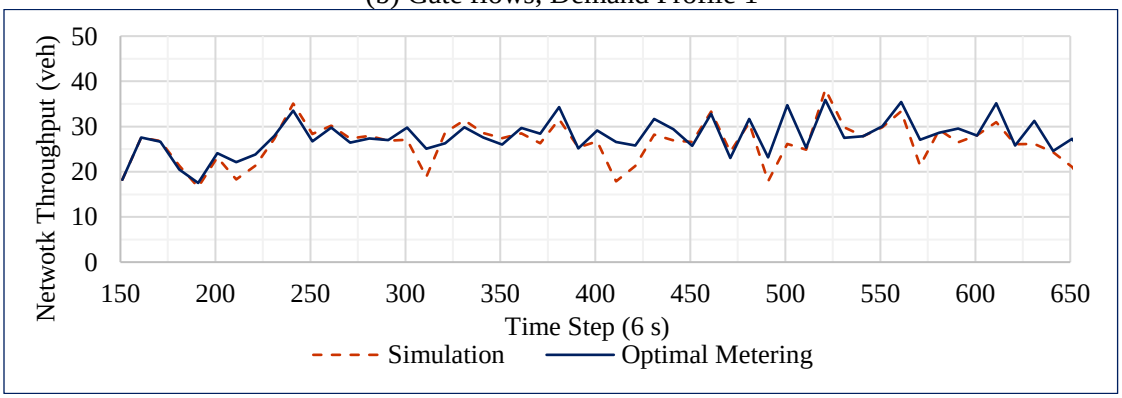
Figure 4-6 (a) and (c) show the network throughout and Figure 4-6 (b) and (d) show the gate flows for Demand Profiles 1 and 2 over time. Figure 4-6 (a) and (c) show that the Optimal Metering scenario could keep the network throughput higher than the Simulation scenario over time. This improvement was achieved by reducing the number of vehicles inside the network once traffic metering was applied at the beginning of the analysis at time step 150, see Figure 4-6 (b) and (d). Note that the network was loaded from time step 0 to 150 without traffic metering and the metering gates were activated from time step 150 until the end of the analysis period. After a significant reduction in the gate flows at time step 150, the gate flows in the Optimal Metering scenario fluctuate over time that representing the traffic metering application.



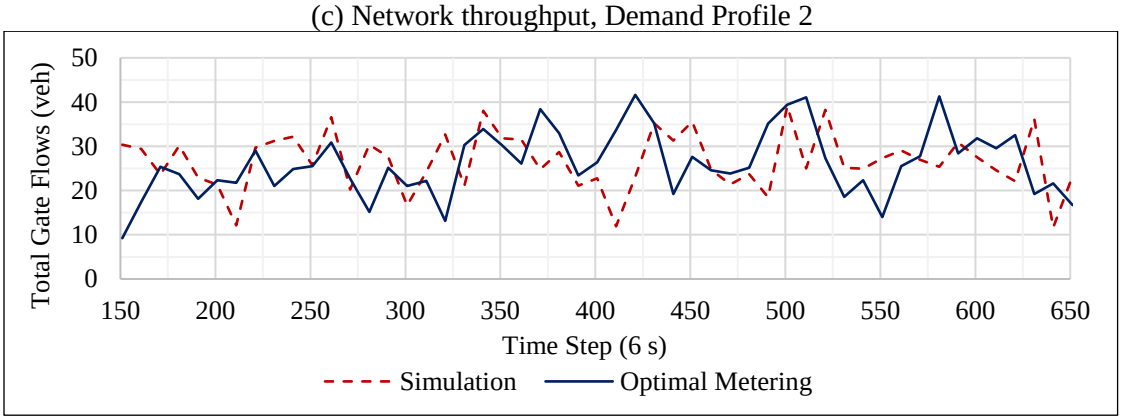
(a) Network throughput, Demand Profile 1



(b) Gate flows, Demand Profile 1



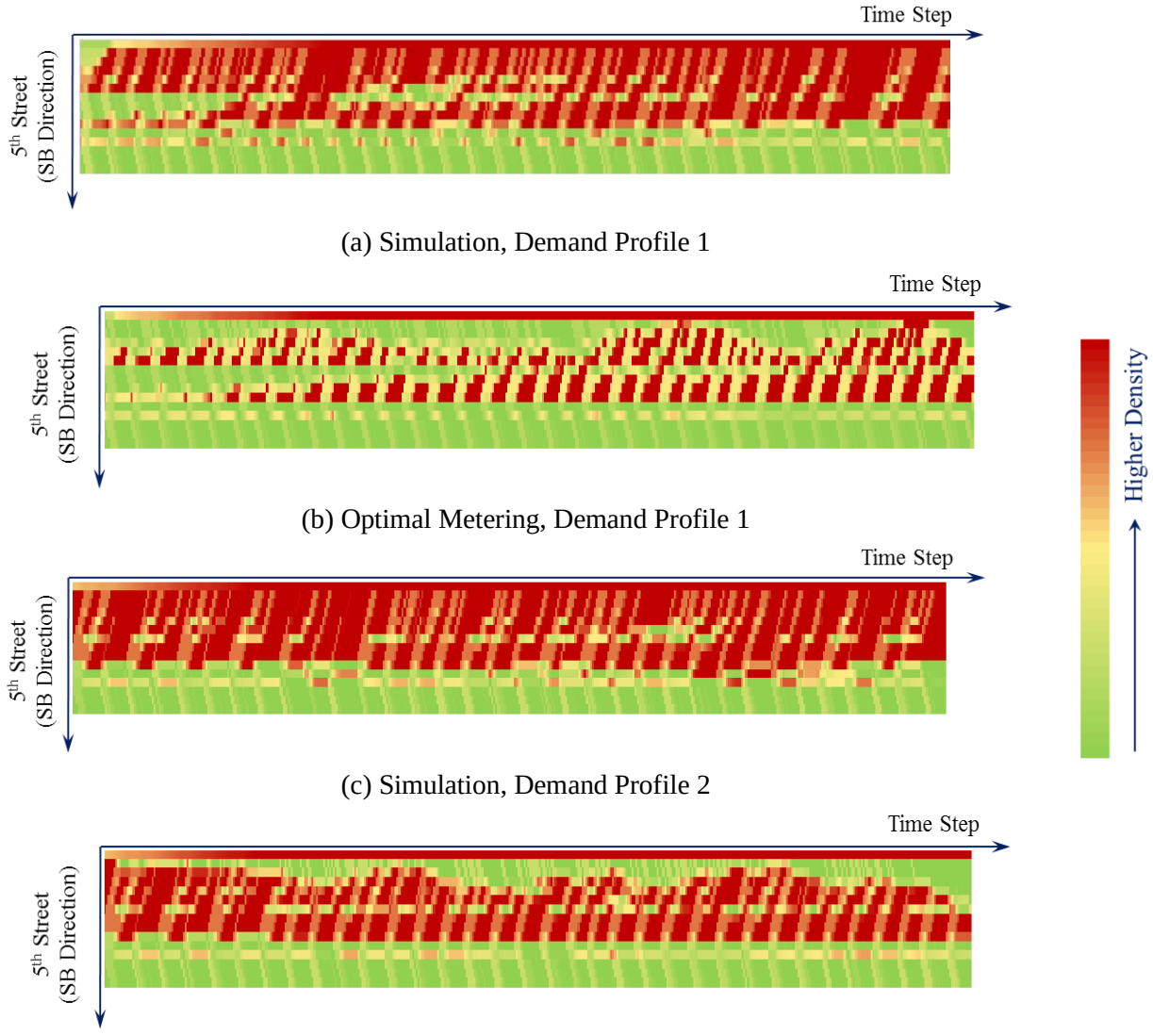
(c) Network throughput, Demand Profile 2



(d) Gate flows, Demand Profile 2

Figure 4-6 Network throughput and gate flows for two demand profiles

The flow reductions and traffic metering can be better observed by comparing the density profiles of a sample arterial street in the Simulation and Optimal Metering scenarios. Figure 4-7 shows the density profiles on 5th Street (Figure 4-4) in the mentioned scenarios. The profiles in Figure 4-7 show that the Simulation scenario resulted in high densities over an extended section of 5th Street. However, traffic metering reduced both spatial and temporal density profiles.



(a) Simulation, Demand Profile 1

(b) Optimal Metering, Demand Profile 1

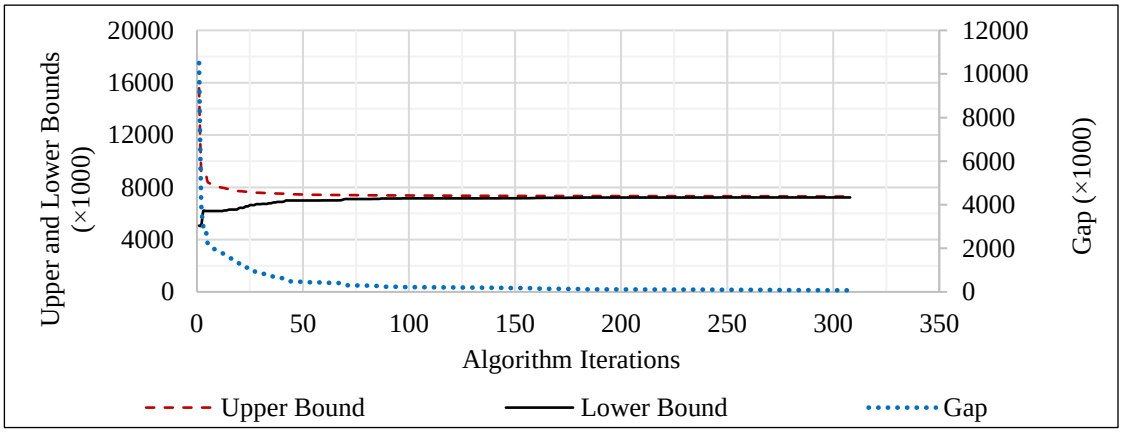
(c) Simulation, Demand Profile 2

(d) Optimal Metering, Demand Profile 2

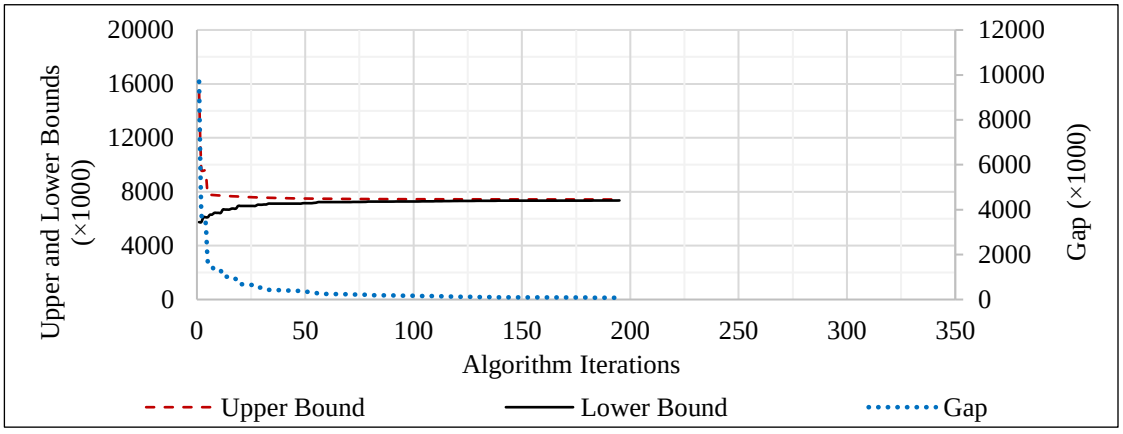
Figure 4-7 Density diagram for one arterial street

We set the convergence criterion of the optimal traffic metering algorithm to a 1% gap between the upper bound and lower bounds, and the algorithm was terminated once the criterion was met.

Figure 4-8 shows the bounds and their gap over the iterations. In the beginning, the gap between the bounds was relatively large, but the gap was reduced iteratively. The algorithm required 310 and 190 iterations to converge for respectively Demand Profiles 1 and 2. Although the criterion was met after these iterations, the algorithm reduced the gap significantly over the first 50 iterations. Therefore, we could terminate the algorithm after 50 iterations and find good solutions relatively fast. The runtimes of finding solutions for Demand Profile 1 and 2 were 6.8 and 4.7 hours on a quad-core PC with 18GB of memory.



(a) Demand Profile 1



(b) Demand Profile 2

Figure 4-8 Convergence diagram of the algorithm

CHAPTER 5. COOPERATIVE TRAFFIC METERING AND SIGNAL CONTROL

Both traffic signal and traffic metering (i.e., perimeter control) improve traffic operations in congested urban-street networks. Signal control optimizes the timing of signalized intersections with the objective of minimizing travel time while avoiding long queues. Traffic metering, on the other hand, regulates the flow of incoming vehicles to a protected area of a congested network with similar objectives to prevent gridlocks (Daganzo, 2007). The main hypothesis of this chapter is that cooperative traffic signal control and metering at network gates can lead to significantly more efficient network-level traffic operations compared to independent signal control and independent perimeter control. While there is a significant number of published researches on traffic signal or perimeter control, not enough attention has been given to the cooperative optimization of them. This study fills this knowledge gap and formulates cooperative traffic signal and perimeter control and develops an efficient distributed methodology to solve it in real-time.

The idea of cooperative traffic metering and signal timing was first applied to freeway facilities by creating coordination among ramp metering signals and adjacent intersections. Without the coordination, high traffic volumes coming from upstream intersections can exceed the available storage of the downstream on-ramps, resulting in queue spillbacks and unnecessary delays. As such, Lu et al. (2013), Kwon et al. (2003), Su et al. (2014), and Kan et al. (2017) developed integrated control algorithms to adjust the signal settings of upstream intersections with respect to the available storage of downstream on-ramps. They showed that the integrated control improved the operations on freeway facilities. However, the application of the integrated control on a network-level scale raises several computational complexities that need to be further addressed. The complexities are due to a large number of decision variables and the difficulties of capturing

the network dynamics including shockwaves and gridlocks that are crucial for dynamic traffic operations in congested urban street networks.

Keyvan Ekbatani et al. (2016) and Kouvelas et al. (2018) integrated the MFD-based perimeter control with adaptive signal timing. They proposed a hierarchical approach, whose upper level determined traffic metering levels based on an MFD, and the lower level found the signal timing parameters. They showed that the integration improved traffic operations; however, they did not optimize signal timings and traffic metering levels simultaneously, and the gates maintained the vehicle accumulation level within a fixed range throughout the analysis period. Note that MFDs are prone to change when the signal timing parameters change (Ji et al., 2010; Laval, 2010). Thus, considering the interaction of MFD-based traffic metering and signal control is not trivial.

The available literature shows that the cooperative traffic signal and perimeter control offers great potential for further improvements in traffic operations. However, the existing cooperative methods are either MFD-based or computationally complex. The properties of an MFD may change as signal timing parameters vary inside the network; however, the existing methods assume a fixed or stochastic MFD (Haddad and Shraiber, 2014), which may influence the quality of their solutions. This chapter fills this gap by developing an optimization program and a solution technique for cooperative traffic signal and perimeter control in semi-connected urban-street networks. In other words, this chapter formulates cooperative control of traffic signals and metering rates in an urban street network by modifying the traffic metering problem formulation. The program jointly optimizes the timing of signalized intersections and metering rates at network gates. The gates can be either the existing traffic signals at the intersections upstream to a protected region or bimodal traffic signals, similar to ramp metering signals, at the entry links to the region. In either case, the boundary signals regulate the number of vehicles entering the protected region

according to the optimized traffic metering levels, while accounting for the added delay experienced at the gates.

A Distributed Optimization and Coordination Algorithm (DOCA) for Cooperative traffic Signal and Perimeter Control (CSPC) algorithm is also developed to solve the problem. The proposed algorithm distributes the network-level problem into several intersection-level stand-alone sub-problems that can be optimized in real-time. At each time step, the algorithm samples the network state by collecting the location of CVs and loop detector data (from the system model that can be either a real or simulated network), estimates the network state (density across the network links), optimizes the signal parameters and metering rates in a CTM-based optimization program, implements the optimized control variables in the system model, and proceeds to the next time step to repeat the procedure until the study period is over.

In this chapter, the formulation of the problem is first presented and then the solution technique is detailed.

5.1. Problem formulation

In Chapter 3, the formulation of the traffic metering problem with flow feasibility and flow conservation constraints are presented. Here, we first modified the objective function of the problem to account for the performance of each individual intersection as well as network-level performance. Then, the decision variables and constraints related to signal timings are added to the traffic metering formulation.

The objective function (5-1) maximizes both cumulative network throughput and individual intersections with respectively the first ($M \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t$) and second expressions ($\sum_{\forall i \in C_{is}} \sum_{\forall t \in T} w_i x_i^t$) using weight parameters M and w_i. A large weight $M \gg w_i$ is assigned for

the network sink cells to prioritize the completion of network-level trips while the assigned w_i to the intersection sink cells $i \in C_{is}$ considers the throughput of each intersection. Generally, the value of M and w_i can be determined based on the preference of a system manager and are exogenous information to the program (the following sections of this chapter provide more information about selecting these weights).

$$\text{maximize } Z_1 = M \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t + \sum_{\forall i \in C_{is}} \sum_{\forall t \in T} w_i x_i^t \tag{5-1}$$

Constraints (5-2)-(5-4) capture the signal timing settings in the formulation. The variables of Constraints (5-2)-(5-4) are signal indications g_i^t for each intersection cell $i \in C_I$ and time step $t \in T$. Moreover, the signal-related sets that are used in these constraints are illustrated in Figure 5-1.

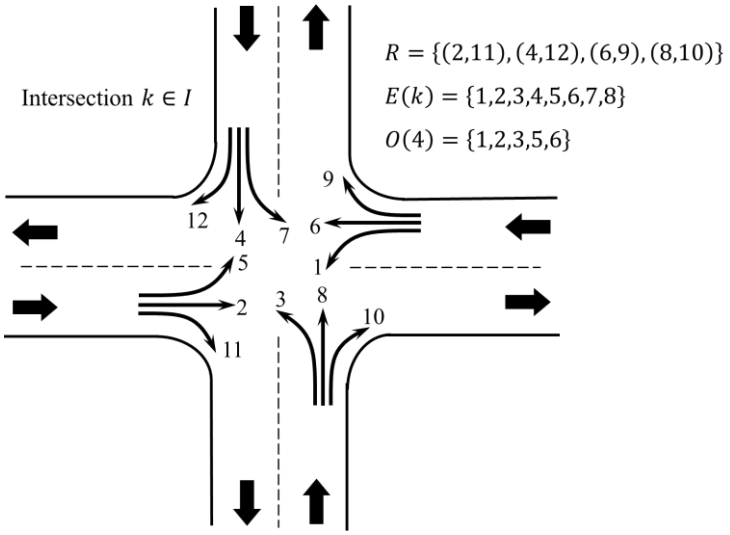


Figure 5-1 Illustration of movements at an intersection and their corresponding sets

Constraint (5-2) ensures that at most two movements from the set of through and left-turning movements $E(k)$ of intersection $k \in I$ receive a green signal indication at time step $t \in T$ (S. M.A.Bin Al Islam et al., 2021; Islam and Hajbabaie, 2021, 2017). Furthermore, only one of the

conflicting movements $i \in E(k)$ and $j \in O(i)$ can have the green indication at each time step $t \in T$, see Constraint (5-4). Constraint (5-3) sets the signal indication of the concurrent through and right turning movements $(i, j) \in R$ equal. Hence, the variables of Constraints (5-2)-(5-4) are signal indications g_i^t for each intersection cell $i \in C_I$ and time step $t \in T$.

$$\sum_{j \in E(k)} g_j^t \leq 2 \qquad \forall k \in I, t \in T \tag{5-2}$$

$$g_i^t + g_j^t \leq 1 \qquad \forall k \in I, i \in E(k), j \in O(i), t \in T \tag{5-3}$$

$$g_i^t = g_j^t \qquad \forall (i, j) \in R, t \in T \tag{5-4}$$

Constraints (5-5) and (5-6) allocate minimum and maximum values to green times of through and left-turning movement $i \in E(k)$ in intersection $k \in I$ (Beard and Ziliaskopoulos, 2006; Mehrabipour, 2018).

$$\sum_{\tau=t+1}^{t+G_{min}} g_i^\tau \geq (g_i^{t+1} - g_i^t) G_{min} \qquad \forall k \in I, i \in E(k), t \in T, t \leq |T| - G_{min} \tag{5-5}$$

$$\sum_{\tau=t}^{t+G_{max}+1} g_i^\tau \leq G_{max} \qquad \forall k \in I, i \in E(k), t \in T, t \leq |T| - G_{max} \tag{5-6}$$

In the formulation of Chapter 3, g_i^t was considered as a parameter, but it is a binary variable here. Constraint (5-7) shows this requirement.

$$g_i^t \in \{0,1\} \qquad \forall i \in C_I, t \in T \tag{5-7}$$

Overall, the proposed formulation for the cooperative traffic metering and signal timing is as follows:

maximize $Z_1 = M \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t + \sum_{\forall i \in C_{is}} \sum_{\forall t \in T} w_i x_i^t$, subject to (3-1)-(3-13), (5-2)-(5-7) (5-8)

5.2. DOCA-CSPC

This section presents the development of a Distributed Optimization and Coordination Algorithm (DOCA) for Cooperative traffic Signal and Perimeter Control (CSPC) following an MPC framework. We used Vissim (PTV Group, 2013) as the system (plant) model to implement and evaluate the proposed algorithm using the designed platform shown in Figure 5-2.

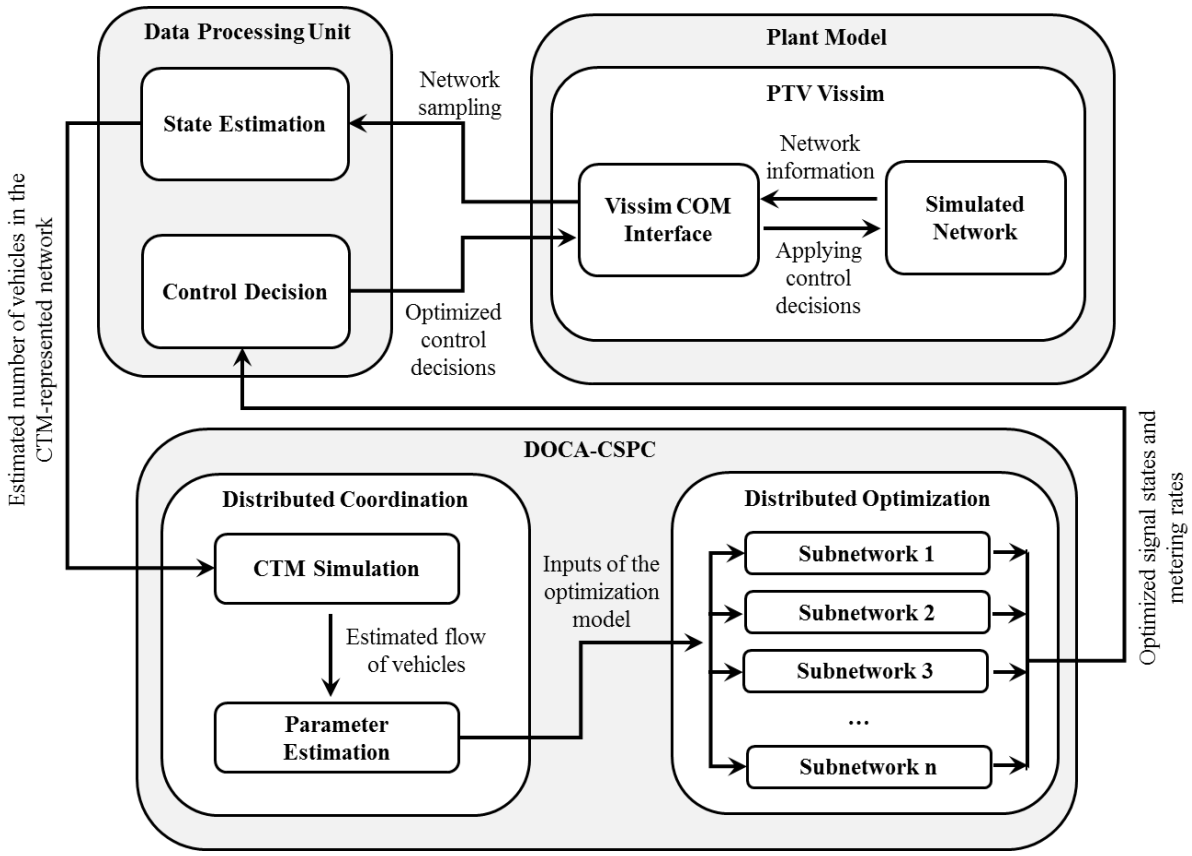


Figure 5-2 The developed implementation and evaluation platform of DOCA-CSPC in Vissim

The platform in Figure 5-2 has three main components: Plant Model, Data Processing Unit

(DPU), and DOCA-CSPC. Plant Model includes the simulated network in Vissim and Vissim's Component Object Model (COM) interface. The COM interface allows reading data from the simulated network or applying the control decisions to the network in real-time. DPU estimates the total number of vehicles in the cells of the CTM-represented network once the location information of CVs and vehicle counts of loop detectors are collected from the network through the COM interface at each time step (Section 5.2.1).

The state estimations provide inputs for a CTM-based simulation to find the required parameters of the optimization programs (Section 5.2.5). Then, the sub-networks solve their optimization programs, and the optimized control variables are sent back to the DPU to be translated to intersection and gate actions for the COM interface. Finally, the COM interface implements the signal indications and metering rates in the simulated network. Furthermore, the platform in Figure 5-2, except for the Plant Model component, was coded in Java, and IBM CPLEX (CPLEX, 2009) was used as the solver. In the following sections, the details of each component of DOCA-CSPS are explained.

5.2.1. System state estimation

The initial system state in terms of cell occupancies x_i^0 for all $i \in C$ at time zero should be given as input to the optimization program. Hence, the location of vehicles in the network should be converted into cell occupancies. Therefore, we used the location information of connected vehicles (CV) and vehicle counts from loop detectors to estimate the cell occupancies (Mohebifard and Hajbabaie, 2018a).

Consider the link in Figure 5-3. The link includes several CVs that are equipped with onboard units that can transmit different information such as the location of CVs to roadside units in a network. If the penetration rate (PR) of connected vehicles is 100%, the state estimation is

straightforward. The location of vehicles will be mapped to their corresponding cells and the cell occupancies will be the summation of vehicles in the cells. However, the location of unequipped vehicles should be estimated once the penetration rate is less than 100% (Islam et al., 2020; S M A Bin Al Islam et al., 2021).

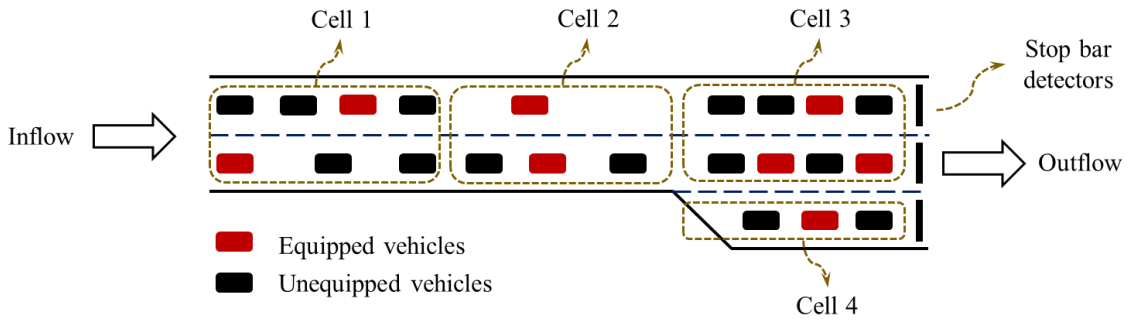


Figure 5-3 A link with the corresponding cells including both equipped (CV) and unequipped (non-CV) vehicles

The proposed approach for system state estimation is the integration of two different estimation techniques. The first technique approximates the density distribution of vehicles in each network link with a vehicle sample that includes CVs. The second technique uses flow feasibility and conservation of CTM to estimate cell occupancies. These two estimations are averaged based on the market penetration rate of CVs.

5.2.1.1. State estimation using location information of CVs

The distribution of vehicles in each network link can be estimated using Equation (5-9). In the equation, xe_i^t is the number of CVs in cell $i \in C$ at time step $t \in T$. By dividing xe_i^t over the total number of CVs in the link $l \in L$ that contains cell i, distribution of CVs in the link can be found. In this equation, L represents the set of all network links. By multiplying the estimated distribution of CVs by the total number of vehicles in the link V_l^t the distribution of both CVs and non-CVs can be found. The total number of vehicles in each link can be estimated by tracking the inflow

and outflow of each link such that V_l^t will be equal to the cumulative outflow minus the cumulative inflow of each link up to time step $t \in T$, assuming the all links are equipped with stop-bar detectors.

$$x_i^{t,CV} = V_l^t \frac{xe_i^t}{\sum_{i \in l} xe_i^t} \qquad \forall i \in C, l \in L, t \in T \qquad (5\text{-}9)$$

Note that as the penetration rate of CVs increases, the distribution estimation error decreases because the sample size of CVs increases and better represent the distribution of vehicles in a link. However, the estimation error increases for low penetration rates. To address this issue, we used the CTM flow conservation and feasibility equations to adjust the estimations in low penetration rates.

5.2.1.2. State estimation using CTM flow feasibility and conservation principles

In this technique, we used the flow feasibility and conservation equations (3-7)-(3-13) to track cell occupancies in each link. For more clarification, consider Figure 5-3 again. The cell occupancies will be initialized to zero at the beginning of the study when the network is empty. As vehicles enter the network, the loop detectors can track the entry vehicles to and exiting vehicles from each link. This information can be utilized to simulate each link with the demand rate equal to the total vehicle counts of its upstream stop-bar detectors. If we assume the flow of vehicles follows the flow conservation and feasibility equations (3-7)-(3-13), we can estimate the cell occupancies $x_i^{t,CTM}$ for each cell $i \in C$ and time step $t \in T$. Note that this approach does not require any information from CVs and only relies on vehicle counts of loop detectors, assuming that all links are equipped with stop-bar detectors.

5.2.1.3. Combined CV and CTM state estimations

Once the estimations of $x_i^{t,CV}$ and $x_i^{t,CTM}$ are found with the proposed approaches in Sections 5.2.1.1 and 5.2.1.2, the estimations are combined using Equations (5-10). In this equation, the state estimations are adjusted by the penetrations rate $0 \leq PR \leq 1$ of CVs. This adjustment gives higher weights to the estimations based on CVs because the estimations are more accurate, but in low penetration rates, the estimations of the CTM-based approach will have higher weights. Moreover, the estimated cell occupancies cannot increase the capacity of the cell N_i.

$$x_i^t = \min\left\{PR x_i^{t,CV} + (1 - PR) x_i^{t,CTM}, N_i\right\} \qquad \forall i \in C, l \in L, t \in T \tag{5-10}$$

5.2.2. Distributed optimization

The distributed optimization decomposes the network-level CSPC problem into several intersection-level sub-problems and consequently reduces the problem complexity. Details will follow:

5.2.3. Model decomposition

A common spatial network decomposition technique in traffic operations is the intersection-level decomposition (Goodall et al., 2013; Priemer and Friedrich, 2009; Tajalli and Hajbabaie, 2018a; Timotheou et al., 2015). In the context of this paper, the intersection-level decomposition generates two types of sub-networks:

Type I: Sub-networks that do not have a network gate as such, the only decision variables are the signal timing parameters.

Type II: Sub-networks that include a network gate; therefore, the decision variables are both signal indications and traffic metering levels.

Figure 5-4 shows the two types of sub-networks for a sample network, which is decomposed

into nine intersection-level sub-networks. Sub-networks 2 and 5 are Type I, as they only have signal timing decision variables while all other sub-networks are Type II.

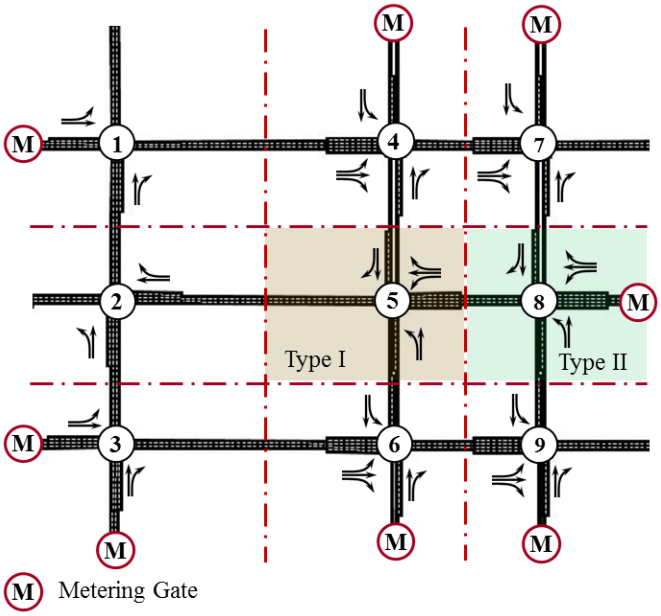


Figure 5-4 Type I and Type II sub-networks

This decomposition is achieved by relaxing the flow-feasibility Constraints (3-7)-(3-10) for only the links that connect boundary cells of a sub-network to the cells in neighboring sub-networks. Then, an optimization problem will be solved for each sub-network. Thus, each sub-network optimizes the control variables that are within its defined geometry. The sub-networks need to be a stand-alone system for the optimization purpose, as such, dummy source and sink cells are added to them when needed. However, each sub-network needs to have information on the number of incoming vehicles from the upstream sub-networks and the available capacity of receiving links at the downstream sub-networks. This information will be exchanged between sub-networks by distributed coordination, see Section 5.2.5.

5.2.4. Optimization complexity reduction

Each sub-network level MINLP has several min(.) functions, which are required to avoid the

flow holding back problem. However, unlike the original network, the sub-networks are cycle free (Ahuja et al., 1993) as such, the flow holding back problem can be prevented following the approach proposed by Zhu and Ukkusuri (2013). This approach assigns a weight to each cell and modifies the objective function and some constraints of the problem. The weights increase with the direction of traffic flow. The weights can be found based on topological ordering algorithms (Ahuja et al., 1993; Mohebifard and Hajbabaie, 2019a) as follows:

Topological ordering is a numbering pattern where all downstream cells will be assigned a weight that is greater than the weight of all upstream cells.

Two types of weight exist in the context of CSPC:

1. Weights M and w_i for $i \in C_{is}$ are defined in the objective function (5-1) to maximize the cumulative number of vehicles in the network-level and intersection-level sinks (C_S and C_{iS}, respectively) in the central problem.

2. The second type of weights w_i are for all cells except the sink cells $i \in C \backslash C_S$ in the objective function (5-1) for the sub-problems. These weights should follow the topological ordering concept to eliminate the flow-holding back problem.

Note that the second type of weights in the objective function (5-1) is added to the sub-problems to eliminate the flow-holding back. If we ensure that $M \gg w_i$, then the sub-problems cooperatively optimize the same objective as the central problem. In other words, the summation of the sub-network-level optimization programs is approximately equal to the objective function of the central problem if $M \gg w_i$.

Accordingly, we proposed the algorithm in Figure 5-5 for generating weights according to the discussed properties. In this algorithm, $\mathrm{Ind}(i)$ shows the number of immediate upstream cells for

a cell $i \in C$.

Algorithm *sub-network cell weights;*

Begin

set $w_i = M$ for $i \in C_s$ and $w_i = W$ for $i \in C_{is}$

set LIST: $= \emptyset$ and next $= 0$;

for all $i \in C$

if $\text{Ind}(i) = 0$ **then** LIST $\coloneqq$ LIST $\cup\, i$;

end

while LIST $\neq \emptyset$

select a cell i in LIST and delete it;

next: $=$ next $+ 1$;

$w_i =$ next;

for all $j \in S(i)$

$\text{Ind}(j) \coloneqq \text{Ind}(j) - 1$;

if $\text{Ind}(j) = 0$ **then** LIST $\coloneqq$ LIST $\cup\, j$

end

end

for all $i \in C_g$

$w_i \coloneqq$ next $+ 1$;

end

end

Figure 5-5 Pseudo code for topological ordering of cells in each sub-network

Note that topological ordering of cells is possible when sub-networks are cycle-free (e.g., intersection-level sub-networks). If sub-networks happen to have cycles, the min(.) operators in the flow-feasibility constraints need to be included in the formulation.

With the modified objective function, we can eliminate the min(.) operator in Constraints (3-

7)-(3-10) and ensure the feasibility conditions with linear Constraints (5-12)-(5-16) which are computationally less complex than the original problem and still finds non-holding back solutions. The simplified mathematical program follows:

$$\text{maximize } Z_2 = M \sum_{\forall i \in C_S} \sum_{\forall t \in T} x_i^t + \sum_{\forall i \in C \setminus C_S} \sum_{\forall t \in T} w_i x_i^t \tag{5-11}$$

Subject to:

(3-1)-(3-6), (3-11)-(3-13), (5-2)-(5-7) and the following constraints:

$$\sum_{j \in S(i)} y_{ij}^t \leq x_i^t \qquad \forall i \in C \setminus C_S, t \in T \tag{5-12}$$

$$\sum_{j \in S(i)} y_{ij}^t \leq Q_i^t \qquad \forall i \in C \setminus C_S, t \in T \tag{5-13}$$

$$\sum_{i \in P(j)} y_{ij}^t \leq Q_j^t \qquad \forall j \in C \setminus C_G, t \in T \tag{5-14}$$

$$\sum_{i \in P(j)} y_{ij}^t \leq \rho\left(N_j - x_j^t\right) \qquad \forall j \in C \setminus C_G, t \in T \tag{5-15}$$

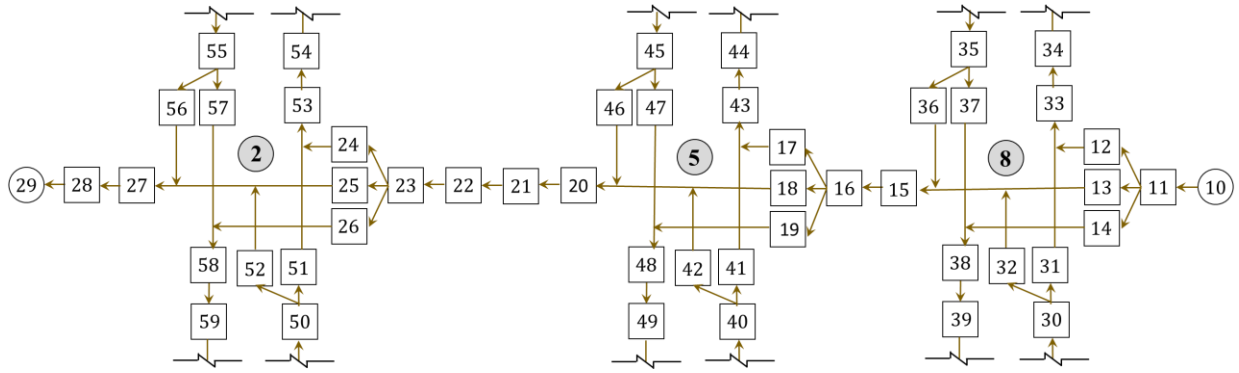
$$y_{ij}^t = \beta_j^t \sum_{k \in S(i)} y_{ik}^t \qquad \forall j \in C_I, i \in P(j), t \in T \tag{5-16}$$

5.2.5. Distributed coordination

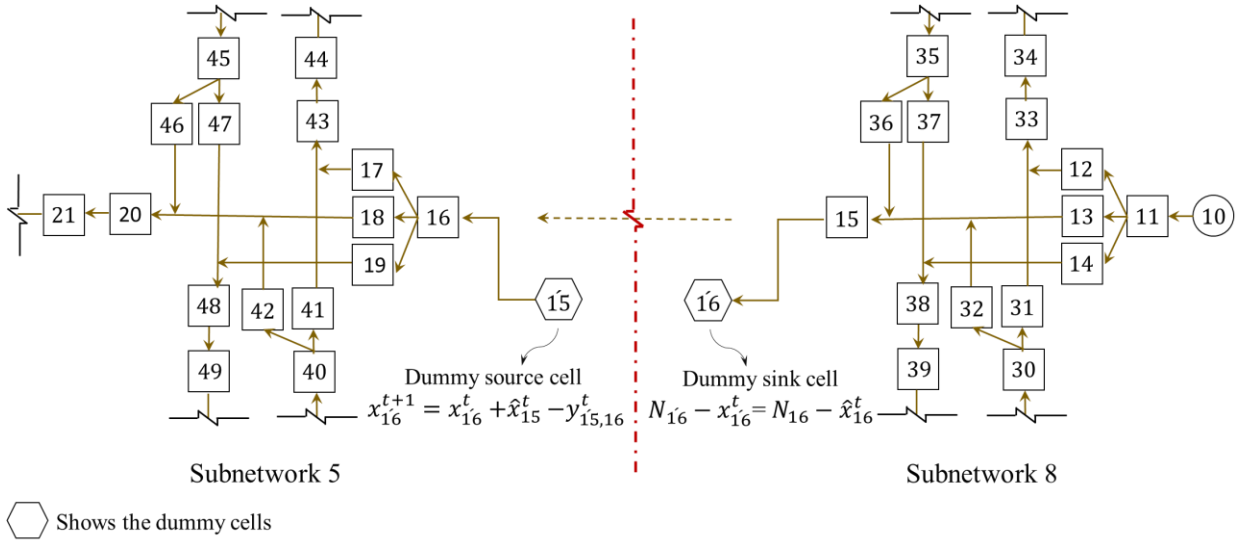
5.2.5.1. Subnetwork coordination and parameter estimation

Constraints (3-7)-(3-10) were relaxed for the links that connect boundary cells in neighboring sub-networks to decompose the network-level problem into several intersection-level sub-problems. As a result, the solutions found by the distributed optimization may be infeasible or sub-optimal. Hence, we will exchange information among sub-problems that share a relaxed constraint and re-introduce and re-enforce the relaxed constraints to ensure that the solutions are feasible and

improve their quality. Consider the street that is shown in Figure 5-6 (a). The number of vehicles that sub-network 8 can send to sub-network 5 depends on the available capacity of cell 16 (in sub-network 5) at each time step. Similarly, the number of vehicles that sub-network 5 receives depends on the number of vehicles in cell 15 (in sub-network 8).



(a) cell representation of the arterial street including sub-networks 2, 5 and 8 in Figure 5-4



(b) dummy cell representation of sub-networks 5 and 8

Figure 5-6 Illustration of dummy source and dummy sink cells of the sub-networks in the middle westbound arterial street in the network of Figure 5-4

In Figure 5-6 (b), dummy sink cell $1\acute{6}$ represents the available capacity of the receiving cell at the downstream sub-network, and dummy source cell $1\acute{5}$ represents the number of vehicles that enter sub-network 5 from the upstream sub-network. Hence, Constraint (3-11)-(3-13) are

compacted and modified as shown in Constraint (5-17) to consider the number of vehicles entering a sub-network from dummy source cells, and Constraint (5-15) is modified as Constraint (5-18) for dummy sink cells. Note that the Kronecker delta ($\delta_{ij} = 1$ when $i = j$; otherwise $\delta_{ij} = 0$) is used in Constraint (5-17) to facilitate the presentation of the Constraint (3-11)-(3-13) for different cell types.

$$(\delta_{io} + \delta_{is}) \sum_{k \in P(i)} y_{ki}^t - (\delta_{io} + \delta_{ir} + \delta_{ik}) \sum_{j \in S(i)} y_{ij}^t + \delta_{ir} D_i^t = (\delta_{io} + \delta_{ir} + \delta_{is})(x_i^{t+1} - x_i^t) + \delta_{ik}(x_i^{t+1} - \hat{x}_i^t) \quad (5\text{-}17)$$

$$\forall i \in C, o \in C \backslash \{C_s, C_g, C_{dg}, C_{ds}\}, r \in \{C_g\}, k \in C_{dg}, s \in \{C_s, C_{ds}\}, t \in T$$

$$\sum_{i \in P(j)} y_{ij}^t \leq (1 - \delta_{js}) \rho(N_j - x_j^t) + \delta_{js} \rho(N_j - \hat{x}_j^t) \quad (5\text{-}18)$$

$$\forall j \in C \backslash \{C_g, C_{dg}, C_{ds}\}, s \in C_{ds}, t \in T$$

In Constraints (5-17) and (5-18), several expressions are added to consider the role of the dummy cells in the sub-network-level optimization programs. The parameters with a "hat" sign ($\hat{x}_i^t$ and $\hat{x}_j^t$) need to be predicted and exchanged between adjacent sub-networks. Accordingly, each sub-network can consider the effect of other sub-networks' decisions on the optimization process and coordinate their decisions.

For predicting $\hat{x}_i^t$ and $\hat{x}_j^t$ in Constraints (5-17) and (5-18), we use a CTM simulation. In the absence of any prediction, we can set the number of vehicles coming to each sub-network to zero ($\hat{x}_i^t = 0$) and the available capacity of receiving cells at downstream sub-networks to Infinity ($\hat{x}_j^t = \infty$). Then, each sub-network optimizes the signal timings and the metering rates to find

initial optimized values. We can find the flow of vehicles inside the network in the subsequent time steps from the CTM simulation given the updated optimized control variables, and hence predict $\hat{x}_i^t$ and $\hat{x}_j^t$.

5.2.5.2. Enhanced sub-network coordination

The existing intersection-level coordination is mostly effective in coordinating adjacent sub-network decisions. However, traffic metering requires coordination among more intersections over an arterial street, especially because many sub-networks are Type I and do not optimize metering rates. Consider sub-network 8 in Figure 5-6, for instance. Sub-network 8 is Type II and optimizes both signals at intersection 8 and the metering rates at gate cell 10. The role of gate cell 10 is to prevent queue spillovers and gridlocks in the sub-networks that might be farther than its neighboring sub-networks. Accordingly, Type II sub-networks are required to have network state data from a broader region of the network to be able to optimize the metering rates more effectively.

Consequently, we propose to extend the Type II sub-networks in the direction of incoming traffic from their gates through the network sink cell so that the sub-network optimization program can explicitly account for the traffic state in the rest of the arterial street. The control variables of the Type II sub-networks do not change with this modification, and only the state variables of the extended segment will be added to the optimizing program. Extended sub-network 8 for sub-network 8 is depicted in Figure 5-7.

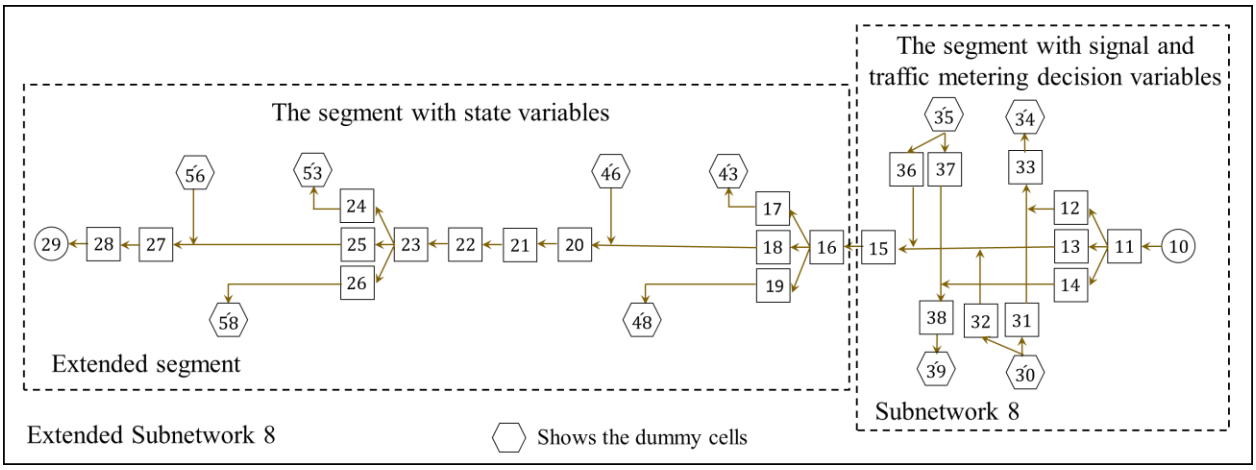


Figure 5-7 Extended sub-network 8 with the corresponding dummy cells

Note that the role of boundary metering gates is to prevent queue spillbacks and gridlocks that might start from the innermost parts of a protected region. The extended sub-networks will allow the gates to include traffic states in such areas of the network in the optimization, as such they can predict queue propagations and prevent them in coordination with signal timing control decisions.

Note that the control variables in the extended and original sub-network 8 are the same, and the only difference is that the arterial street downstream of gate 10 is added to the sub-network. Moreover, the signal indications of intersection cells 17 to 19 and 24 to 26 get optimized in their own sub-networks (sub-networks 5 and 2, see Figure 5-6), and only their estimated values enter the optimization of extended sub-network 8. Thus, $g_i^t Q_i^t$ expression in (3-8) is replaced with q_i^t in Constraint (5-19) to account for the intersection cells in the extension of Type II sub-networks, see Figure 5-7.

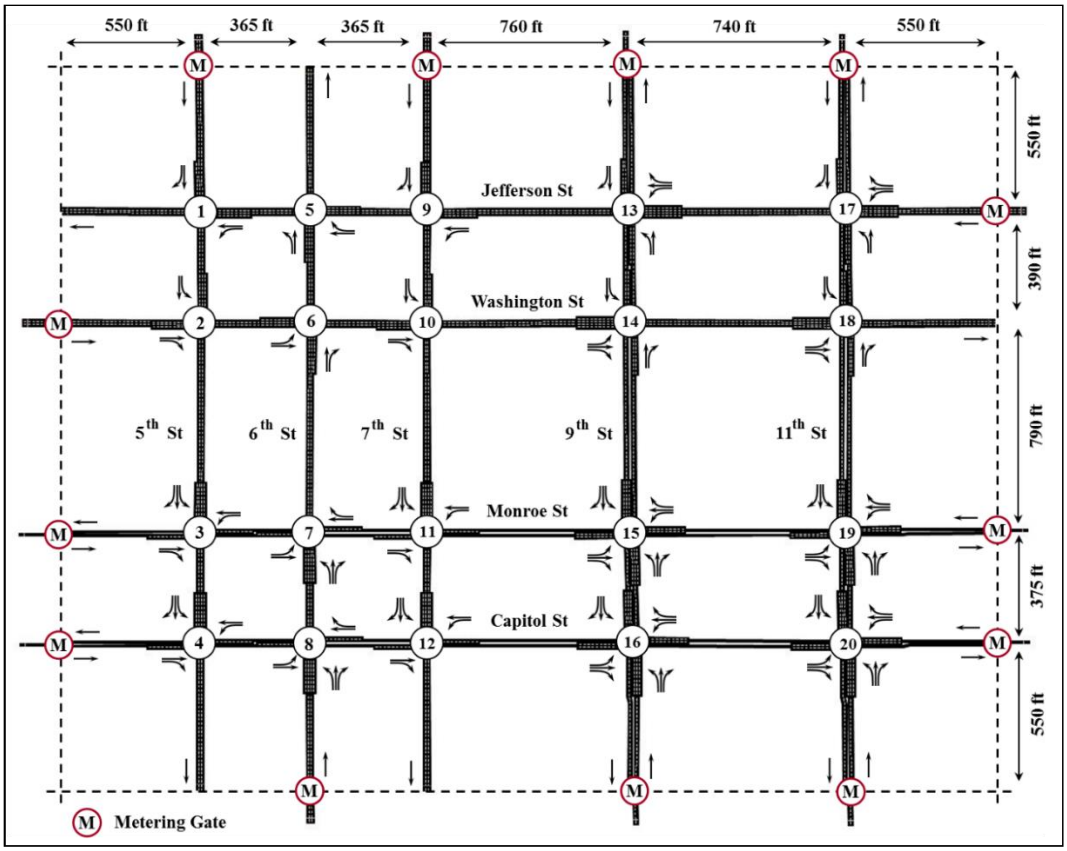
$$q_i^t = (1 - \delta_{ij}) g_i^t Q_i^t + \delta_{ij} \hat{g}_i^t Q_i^t \qquad \forall i \in C_i, j \in C_{ie}, t \in T \tag{5-19}$$

The saturation flow rate of intersection cells in the extended segment is equal to $\hat{g}_i^t Q_i^t$ in

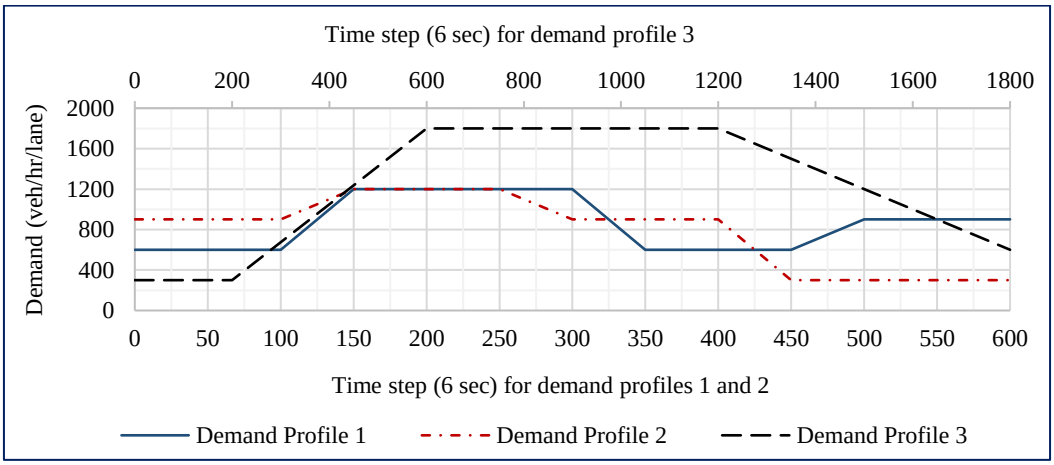
Constraint (5-17), and $\hat{g}_i^t$ is the estimated signal indication that will be obtained from the optimized values of other sub-networks. Note that $\hat{g}_i^t$ are the estimated signal indications whose values are optimized in the sub-networks where they belong to. For instance, cell 18 belongs to sub-network 5 in Figure 5-6 and its indication is optimized in sub-network 5. Then, we use its optimized values as $\hat{g}_{18}^t$ in the signal and metering optimization of sub-network 8. Moreover, once the sub-networks update their signal plan, the values of $\hat{g}_i^t$ are updated accordingly.

5.3. Case study

We used the network in Figure 5-8 (a) as the case study. The network consists of 20 intersections, and 13 gates are placed at the boundary of the network. The case study is analyzed for one hour with the demand profiles 1 and 2 and for three hours with demand profile 3 that are shown in Figure 5-8 (b). Moreover, the characteristics of the network are shown in Table 5-1.



(a) The Springfield Downtown network



(b) The demand profiles

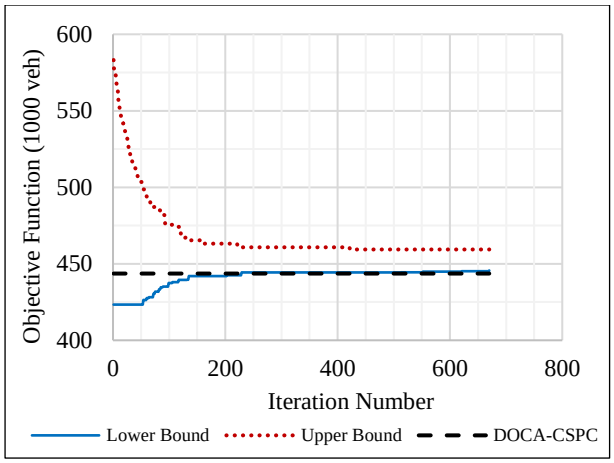
Figure 5-8 The case study information

Table 5-1 The characteristics of the case study network

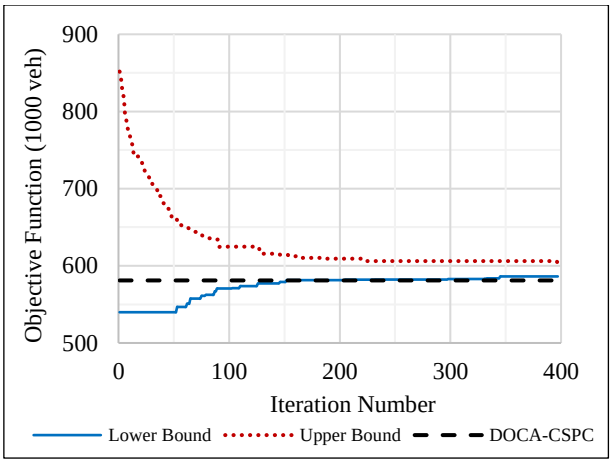
Element	Value	Element	Value
Free flow speed (mph)	25	Jam density of cells (veh/cell/lane)	12
Saturation flow rate of links (vphpl)	1800	Turning percentages (%)	10
Time interval (sec)	6	Minimum green time for through movements (sec)	18
Total number of cells	316	Minimum green time for left-turning movements (sec)	6
Length of cells (ft)	225	Maximum green time for through movements (sec)	60
Saturation flow rate of cells (veh/time step/lane)	3	Maximum green time for left-turning movements (sec)	24

5.4. Numerical results

We start this section by comparing the results of DOCA with the benchmark solutions for 200 time steps (20 minutes). The benchmark solutions are those that were found with the modified version of the proposed algorithm in Chapter 4 to account for the cooperative optimization problem. The benchmark algorithm's runtime limited the duration of the study period. The termination criterion for the benchmark algorithm was reaching an optimality gap of 3%. Figure 5-9 shows the convergence of the benchmark algorithm and DOCA's solution. The benchmark algorithm found solutions after 670 and 397 iterations for demand profiles 1 and 2, respectively. DOCA's solutions were real-time and close to the best feasible solutions that were found by the benchmark algorithm.



(a) Demand profile 1



(b) Demand profile 2

Figure 5-9 Convergence of the customized solution technique for solving (P1) along with the objective function value of DOCA-CSPC

Table 5-2 shows the objective function (cumulative number of vehicles in the sink cells) values of DOCA's solution and the upper and lower bounds found by the benchmark algorithm. The results showed that the objective function of DOCA was at most 0.88% different from the best feasible solutions (lower bound) that could be found. Moreover, the solutions of DOCA were at most 3.72% different from the theoretical upper bounds. These trends indicate that DOCA could find near-optimal solutions for the case study network under demand profiles 1 and 2. We observed the same trends in our other tests.

Table 5-2 Comparing the objective function values of DOCA with the optimal solutions to the central problem

Demand Profile	Benchmark (veh)		% Diff.	DOCA (veh)	% Diff. DOCA vs.	% Diff. DOCA vs.
	Upper Bound (UB)	Lower Bound (LB)	UB vs. LB		UB	LB
Demand profile 1	459,355	445,635	2.99	443,630	-3.42	-0.45
Demand profile 2	602,838	586,204	2.76	581,036	-3.62	-0.88

We compared different network performance measures that were found by DOCA and the benchmark solutions in Table 5-3. The network-level performance measures found by the DOCA

solution were at most 2.6% different than those found by the benchmark solution, indicating a near-optimal network-level performance. Note that the benchmark algorithm runtimes were 23.4 hours on average.

Table 5-3 Network performance measures of DOCA and the benchmark solutions

Criteria	Demand Profile 1			Demand Profile 2		
	Benchmark	DOCA	% Diff.	Benchmark	DOCA	% Diff.
Network throughput (veh)	5,516	5,449	-1.2	6,660	6,548	-1.7
Total travel time (min)	17,041.0	17,241.5	1.2	25,911.9	26,428.7	2.0
Total delay (min)	7,139.0	6,968.9	-2.4	14,014.7	14,200.4	1.3
Average speed (mph)	20.2	20.7	2.6	15.9	16.0	0.8

In the rest of this section, the results of implementing the proposed algorithm in Vissim will be presented. We analyzed the case study network under the following four scenarios:

1. Vistro: We optimized the signal timing parameters in the case study network using Vistro as one of the state-of-the-practice traffic signal optimizers.
2. Signal-Only: This scenario shows the effect of traffic signal timing on a network without restricting the entry demand. We removed the traffic metering decision parameters and optimized the signals using the proposed distributed approach. This approach is equivalent to the approach in (Mehrabipour and Hajbabaie, 2017).
3. Metering-Only: This scenario shows the effect of traffic metering on a network with predefined signal settings. We used the signal timing parameters that were found by Vistro and only optimized the metering rates. This approach is equivalent to the approach in (Mohebifard and Hajbabaie, 2019a).
4. CSPC: This scenario performs cooperative traffic signal and perimeter control.

Furthermore, we used the market penetration rate of 100% in all the above scenarios because the benchmark solutions (Signal-Only and Metering-Only) are only for fully connected networks. Table 5-4 shows network performance measures in the case study network for each scenario. The Metering-Only scenario increased the network throughput by 27.7% - 29.1% and decreased the average delay by 5.0% - 5.7% compared to Vistro solutions. Although the network throughput shows a significant improvement, the average delay or total travel time did not improve as significantly as the network throughput. The reason is that in the Metering-Only scenario, vehicles were delayed at the network gates, and thus the system-level delay (vehicles both at the gates and inside the network) was influenced by the excessive delay of vehicles at the gates. On the other hand, the Signal-Only scenario increased the network throughput by 26.9% - 33.8% and reduced the average delay by 22% - 23.2%. Therefore, optimizing the signal timing parameters increased the network capacity for processing vehicles such that both network throughput and average delay of vehicles show significant improvements.

Furthermore, Table 5-4 shows that CSPC outperforms all other scenarios. CSPC increased the network throughput by 11.0% and 10.9% compared to Metering-Only and by 6.0% and 12.8% compared to Signal-Only for demand profiles 1 and 2, respectively. CSPC reduced the total travel time by 8.1% to 24.2% compared to Metering-Only and Signal-Only scenarios. The trends shown in Table 5-4 indicate that optimizing both signal indications and metering rates result in a more efficient network performance compared to each of the benchmark scenarios.

Table 5-4 Network performance measures of DOCA-CSPC compared to other benchmark solutions with 100% penetration rate

Performance Measure	Vistro	Metering-Only	% Diff. Metering-Only VS. Vistro	Signal-Only	% Diff. Signal-Only VS. Vistro	CSPC	% Diff. CSPC VS. Metering-Only	% Diff. CSPC VS. Signal-Only
	Demand Profile 1							
Average delay (sec)	540.0	509.1	-5.7	414.7	-23.2	368.8	-27.6	-11.1
Average speed (mph)	2.7	3.7	35.4	4.4	63.8	5.2	40.5	18.2
Total travel time (hr)	3,748.9	3,628.7	-3.2	3,023.0	-19.4	2,750.2	-24.2	-9.0
Total delay (hr)	3,486.8	3,287.2	-5.7	2,678.2	-23.2	2,381.3	-27.6	-11.1
Total number of stops	740,158	117,736	-84.1	648,165	-12.4	324,321	175.5	-50.0
Throughput (veh)	13,554	17,310	27.7	18,142	33.8	19,222	11.0	6.0
	Demand Profile 2							
Average delay (sec)	681.4	647.4	-5.0	531.7	-22.0	476.6	-26.4	-10.4
Average speed (mph)	2.3	3.2	37.2	3.6	55.6	4.5	40.6	25.0
Total travel time (hr)	4,344.0	4,227.1	-2.7	3,514.2	-19.1	3,230.7	-23.6	-8.1
Total delay (hr)	4,082.4	3,878.6	-5.0	3,185.5	-22.0	2,855.6	-26.4	-10.4
Total number of stops	773,477	124,068	-84.0	620,764	-19.7	362,851	192.5	-41.5
Throughput (veh)	13,785	17,792	29.1	17,498	26.9	19,736	10.9	12.8

Figure 5-10 shows the gate flows and network throughput of each scenario over the simulation period for demand profile 1. The metering only scenario kept the gate flows at a lower level compared to all other scenarios from time 900 to 1800 seconds although the network demand was at the highest level at this time interval. These metering levels resulted in steady network throughput and gate flows from time 900 to 3600 seconds. In the Signal-Only scenario, the gate flows were not decreased during the peak period of the demand profile. Accordingly, in Figure 5-10 (a), there is still a decreasing trend in the gate flows especially from time 1900 to 2300 indicating that the performance of the signals is affected by the high demand level. However, with

the dynamic signals in the Signal-Only scenario, the network capacity is maintained higher than the throughput of the Vistro scenario in the study period.

The CSPC resulted in gate flows that were between the follows of the Metering-Only and Signal-Only scenarios between time 900 to 1800 seconds. The flows were higher than the Metering-Only flows because the CSPC increased network capacity by optimizing the signals, which allowed the processing of more vehicles. The flows are less than the Signal-Only flows due to the metering that not only maintained the network throughput at the same level as the Signal-Only scenario in time 0 to 1800 seconds but also resulted in higher throughput from time 1800 to 2400 seconds, see Figure 5-10 (b).

Overall, the gate flows and the network throughout in Figure 5-10 show that the CSPC improved network operations by maximizing the network capacity and regulating incoming flows.

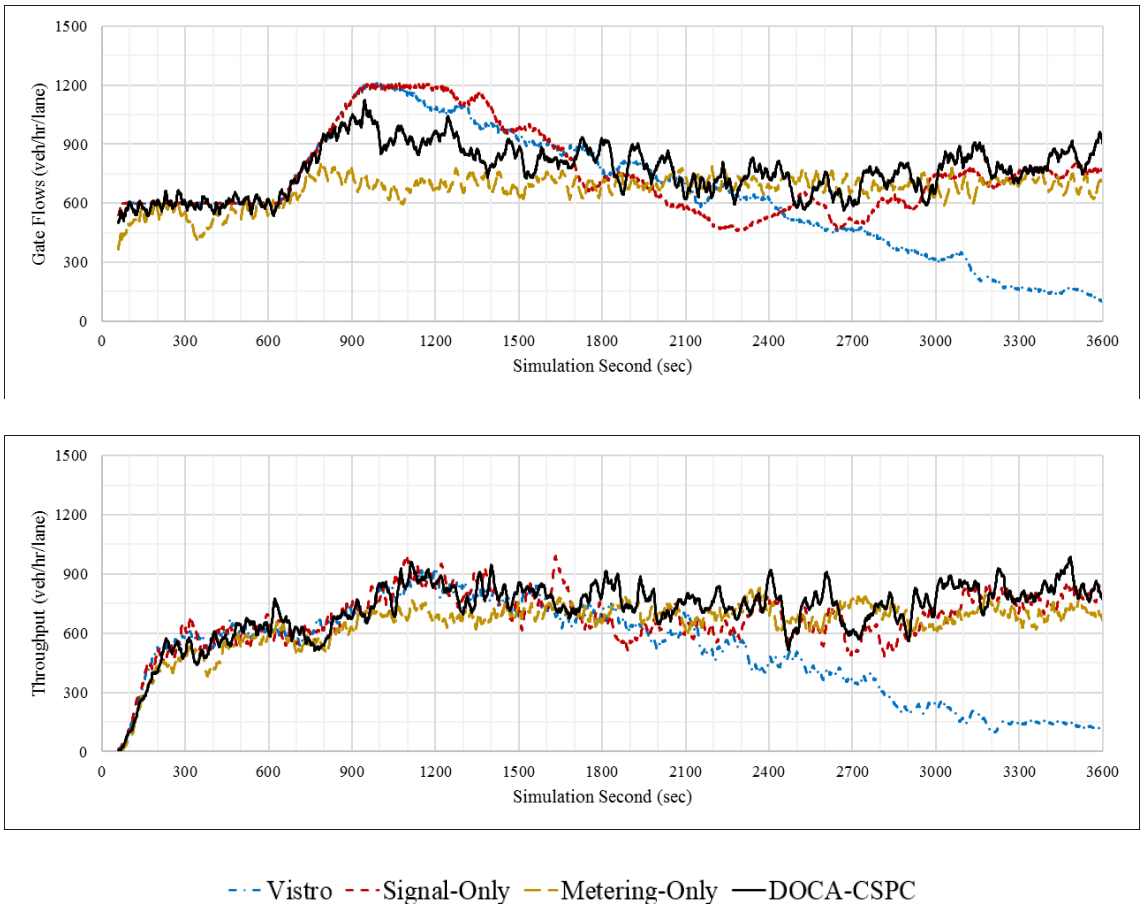


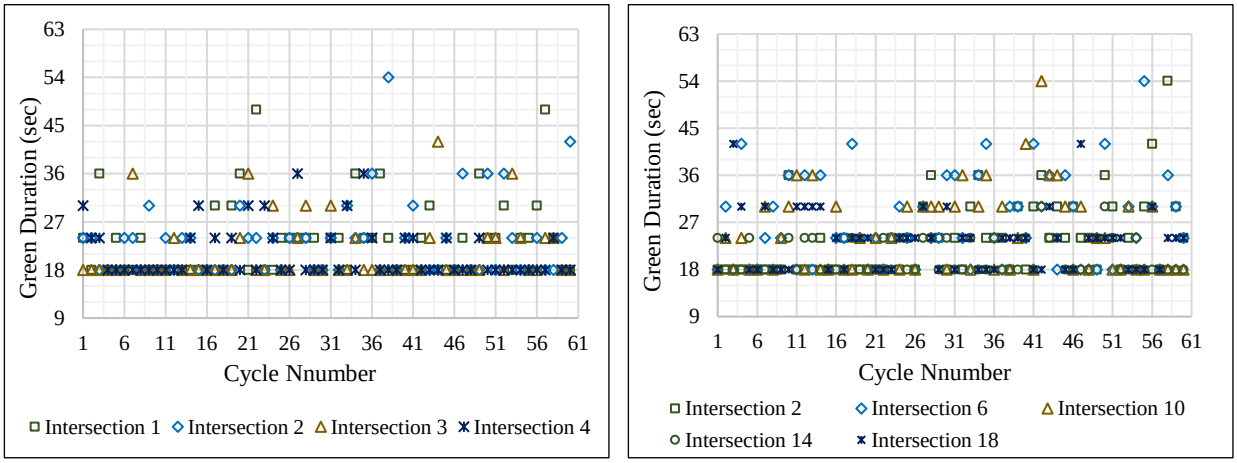
Figure 5-10 Gate flows and network throughput of different scenarios for Demand Profile 1

Moreover, we analyzed the performance of each scenario (Metering-Only, Signal-Only, and DOCA-CSPC) under additional demand profiles. The first analysis is on demand profile 1 with three scaling factors of 0.8, 1.0, and 1.2. Table 5-5 shows that DOCA-CSPC reduced average delay by 26.1% and 18.3% and total travel time by 20.6% and 16.5% compared to Metering-Only and Signal-Only scenarios, respectively. Table 5-5 also shows similar trends for demand profile 3. The network is analyzed for three hours (1,800 time steps) under this demand profile. DOCA-CSPS reduced average delay and total travel time of vehicles between 22.9% to 26.3% compared to the Signal-Only and Metering-Only scenarios. The results of this table show the capability of DOCA-CSPC in improving traffic operations under various demand conditions that were considered in this paper.

Table 5-5 Network performance measures for demand profile 1 with different scaling factors and demand profile 3

Performance Measure	Metering-Only	Signal-Only	CSPC	% Diff. CSPC VS. Metering-Only	% Diff. CSPC VS. Signal-Only
	Average measures of Demand Profile 1 with scaling factors 0.8, 1.0, and 1.2				
Average delay (sec)	482.2	435.7	356.1	-26.1	-18.3
Total travel time (hr)	3,571.4	3,397.5	2,836.1	-20.6	-16.5
	Demand Profile 3				
Average delay (sec)	2,488.8	2,400.1	1,834.0	-26.3	-23.6
Total travel time (hr)	72,085.6	69,600.8	53,651.0	-25.6	-22.9

Figure 5-11 shows the green duration of the through movement signals in two sample arterial streets: 5th in the southbound and Washington St. in the eastbound direction. The figure shows that the green times are maintained within the defined minimum and maximum green times in Figure 5-8 (b) and are changed dynamically throughout the study period.



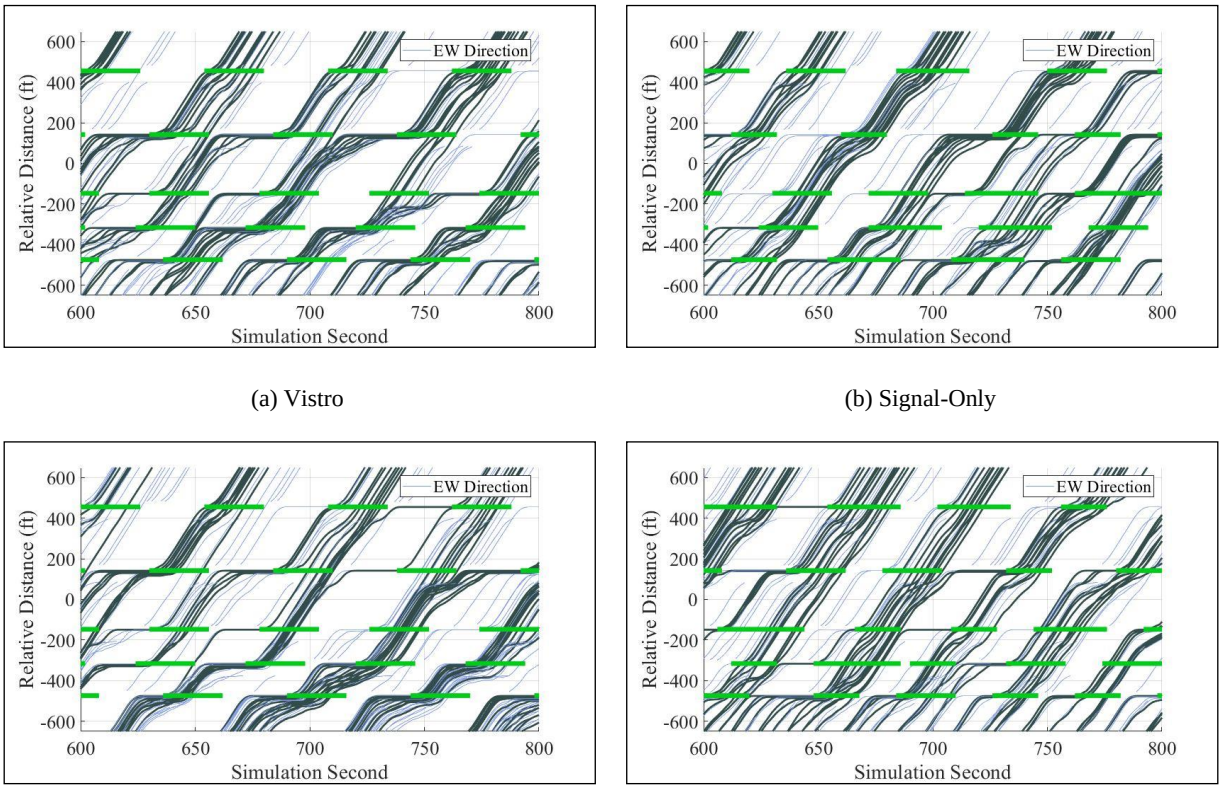
(a) North-South direction

(b) East-West direction

Figure 5-11 Green durations of through movements in 5th and Washington St. for Demand Profile 1 in DOCA-CSPC

Figure 5-12 shows vehicle trajectories and green times in Washington St. with five intersections. The proposed CSPC coordinated the signals along the arterial street while there was

no explicit constraint for signal coordination. Note that Vistro and Metering-Only scenarios have coordinated the signals as well since Vistro is designed to do so.



(a) Vistro

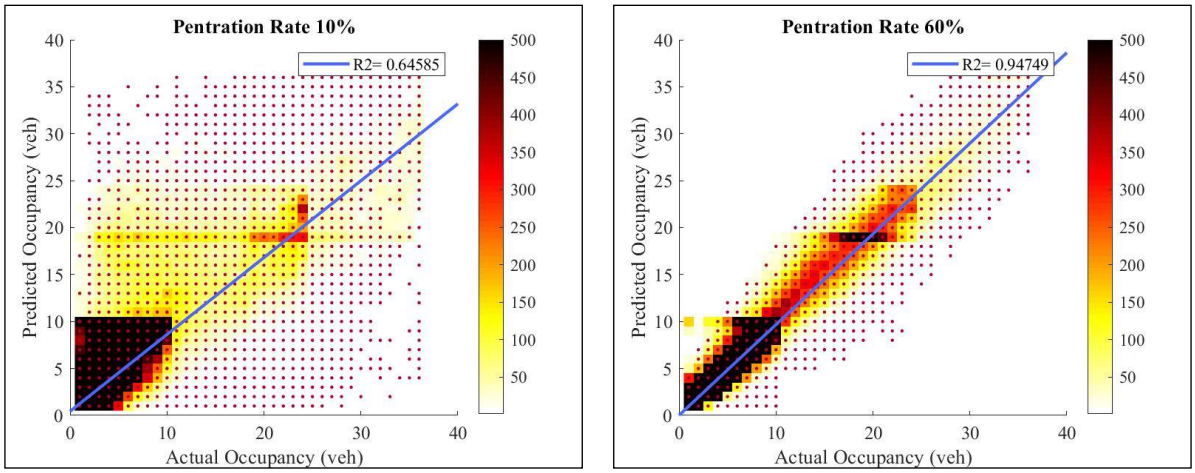
(b) Signal-Only

(c) Metering-Only

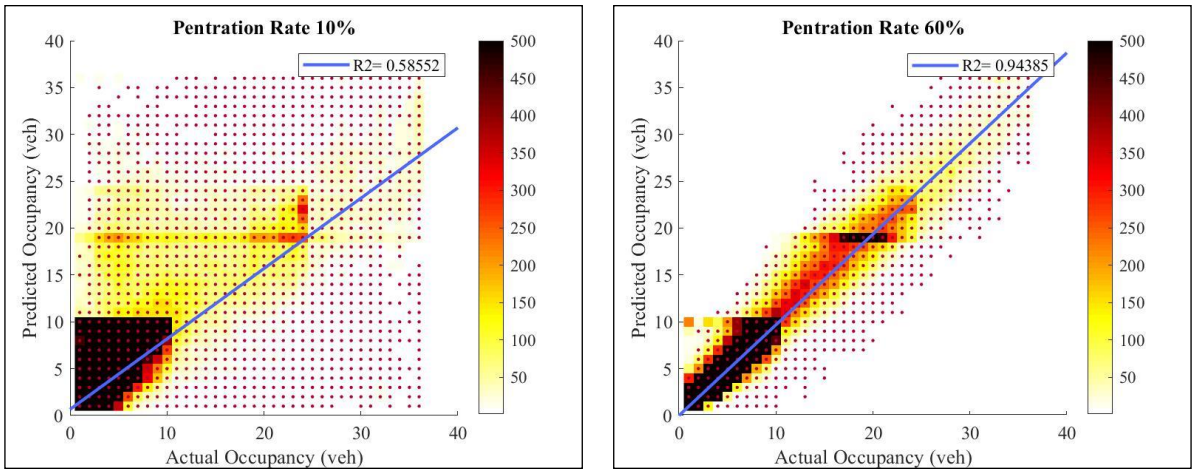
(d) CSPC

Figure 5-12 Green times and trajectory of vehicles for Washington St. in Demand Profile 1

In addition, we evaluated the sensitivity of the proposed algorithm to the market penetration rate of connected vehicles. Figure 5-13 shows the predicted vs. observed cell occupancies with a fitted regression line for penetration rates of 10% and 60%. The colors in these figures show the density of points on different parts of the figures. Figure 5-13 shows that increasing the market penetration rate reduced the scatteredness of the data and improved the adjusted R-squared values.



(a) Demand Profile 1



(b) Demand Profile 2

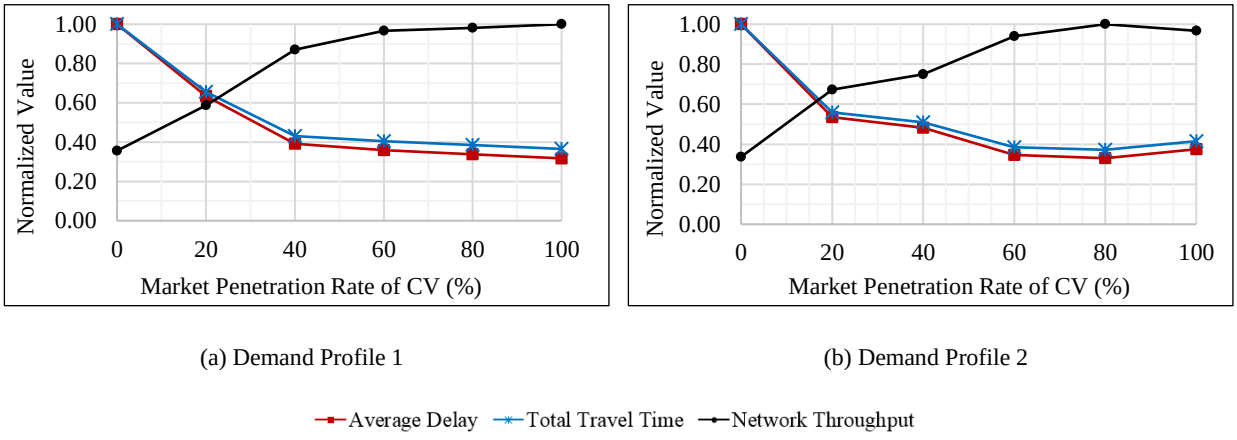
Figure 5-13 The simulation data of actual versus estimated network state (cell occupancy) and the fitted linear regression lines for two sample market penetration rates of connected vehicles

Table 5-6 shows the R-squared values for penetration rates of 0% to 100% with 10% increments. The results indicate that without any location information of connected vehicles (0% penetration rate), the state estimation algorithm predicts around 39.8% of the cell occupancies. The estimation accuracy increases with an increase in the market penetration rate. Moreover, the accuracy of the prediction is more than 82% when the penetration rate is 30%. The estimation accuracy is more than 90% when the penetration rate is more than 50%.

Table 5-6 The R-squared values (%) of the fitted regression line on the simulation data of actual versus estimated network state

Penetration Rate	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Demand Profile 1	39.8	64.6	74.5	83.4	88.5	91.5	94.7	96.5	98.1	99.1	100.0
Demand Profile 2	43.4	58.6	71.1	82.7	88.2	91.3	94.4	96.2	98.1	99.1	100.0

Figure 5-14 shows network performance measures obtained with various connected vehicle market penetration rates. The values of the performance measures are normalized between zero and one so that they can be demonstrated in the same figure. Increasing the penetration rate improved the performance of DOCA-CSPC in terms of total travel time, average delay, average speed, and network throughput in both demand profiles.



(a) Demand Profile 1

(b) Demand Profile 2

Figure 5-14 The normalized values of performance measures of DOCA-CSPC with different market penetration rates of CVs

Finally, the average runtimes of solving the sub-network-level optimization programs for the case study network is shown in Figure 5-15. We solved the programs on a Core i7 PC with 24 gigabytes of memory. The figure shows the runtimes fluctuate between 1.0 to 3.0 seconds while the time step is 6 seconds, see Figure 5-8 (b). In other words, in six-second time intervals, the network state is estimated, the optimization programs are solved, and before getting to the next

time interval, the optimized solutions are found. Thus, the algorithm can be implemented in real-time.

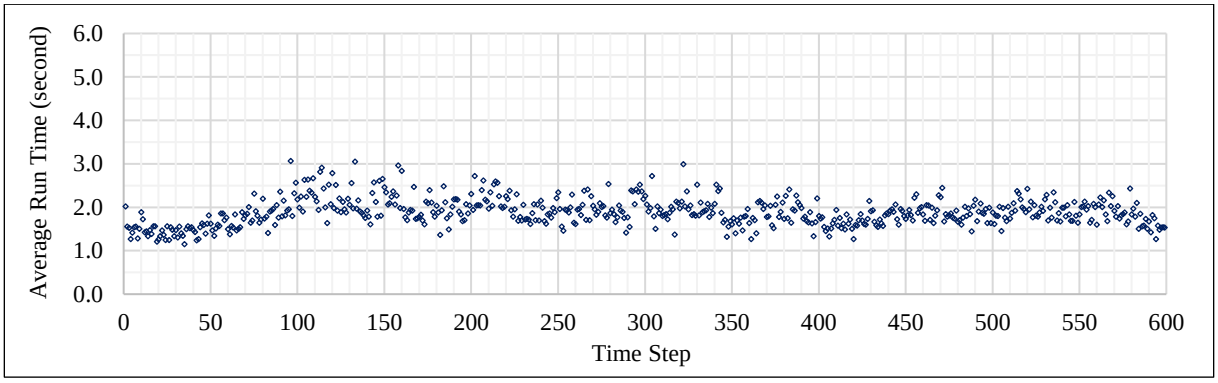


Figure 5-15 Average runtime of solving the sub-network-level optimization programs

CHAPTER 6. TRAFFIC METERING LOCATION AND LEVEL OPTIMIZATION

Installation and maintenance of metering signals can be costly. These costs are associated with geometric construction, signal displays and support, detection devices, controllers, signing, and pavement marking. For instance, Atkins (2013) estimated that construction and equipment costs of a typical ramp metering location are between $530,000 and $772,000 plus $31,000 per year for its maintenance in the State of North Carolina. Accordingly, it is critical to use the fewest metering locations possible to meet budget restrictions. At the same time, the number of metering locations should be enough to be effective in reducing traffic congestion and improving network performance.

Keyvan-Ekbatani et al. (2012) designed a feedback-based controller that maintained the number of vehicles in a network within the optimal range obtained by MFD. The controller determined the total number of vehicles that should be allowed to enter the network, and each metering location processed a fraction of the total optimized inflow proportional to its saturation flow rate. They stated that the metering locations were selected in an "ad hoc" way. Haddad and Geroliminis (2012) studied the stability of a two-region network where perimeter control was applied to the border between the regions. They identified the network conditions that would lead to a gridlock regardless of the perimeter control strategy. These conditions assumed that metering locations were at the borders between the two regions, whereas distributing metering locations both at the border and inside the network might result in a control strategy that prevented gridlocks. Geroliminis et al. (2013) used MFDs to represent the dynamics of a two-region urban network. They formulated an optimal control problem whose decision variables were the total number of vehicles that should be allowed to move between the regions. The application of this formulation

on a real or simulated network required a more detailed representation of flow dynamics such that the effect of metering locations and the queue of vehicles at those locations could be captured on the network performance.

Haddad and Shraiber (2014) stated that the previously discussed MFD-based controllers could regulate the flow of vehicles around a predefined accumulation setpoint that corresponded to the highest network throughput on a network's MFD. Thus, they linearized the non-linear MFD-based system dynamics around a stochastic setpoint such that uncertainties in the MFD-based dynamics could be considered in the controller design. The study suggested further studies for converting the total metered flow that was found by the designed controller into dynamic signal indications at the metering locations. Keyvan-Ekbatani et al. (2015) designed MFD-based controllers for concentric urban networks where congestion spread from the innermost region to the outer regions. They proposed a policy based on queue lengths at the metering locations so that the total metered flow that was found from the controllers could be divided between the locations by converting flows into signal indications. However, the metering signals could not enforce the same metered flows as was optimized by the controllers due to the conversion of flows to signal indications, or due to the congestion that might existed downstream of the metering locations and did not allow processing the determined flows. Keyvan-Ekbatani et al. (2016) proposed an MFD-based perimeter control approach to maximize the network throughput besides a queue management strategy at the gates. They proposed a quadratic knapsack formulation to balance the queue of vehicles at the gates. The simulation results showed positive impacts of this approach to reducing the queue lengths at gates and improving network performance. The numerical analysis of Haddad (2017) on a two-region network with MFD-based dynamics showed that the capacity of metering locations for holding queued vehicles had a significant effect on perimeter control policies. He

proposed a control model with explicit constraints on the aggregated capacities of metering locations at the boundaries between regions. Accordingly, the limited capacity of metering locations was captured explicitly in the perimeter control policies. However, the capacity restrictions were considered as an aggregated measure in the model. Therefore, the model did not enforce the capacity restrictions of individual locations.

The available traffic metering approaches assumed that metering locations were predefined, and the only decision variables were traffic metering levels. Moreover, they did not consider any budget restrictions on the installation and operational costs of metering locations. The present study fills this gap by proposing an integrated formulation and a solution technique for optimizing the traffic metering locations and levels with a limited budget for installation and operational costs of traffic metering locations. Accordingly, the main hypothesis of this chapter is that integrated traffic metering "location" and "level" optimization yields better or similar network performance with lower costs compared to sole metering level optimization for predefined metering locations at the boundaries of congested areas. In this hypothesis, the metering locations have the flexibility of being at the boundary or inside congested areas, and metering levels are their signal indications.

The decision variables of the integrated traffic metering location and level optimization problem are metering locations and their corresponding dynamic metering levels with the objective of network throughput maximization. The proposed optimization program has nonlinear constraints and large spatial and temporal dimensions. Thus, we propose a solution technique that divides the feasible region of the problem into smaller sub-regions over the metering locations. Then, we show that two new MINLPs can be constructed for each of the sub-regions that only optimize traffic metering levels for fixed metering locations. The MINLPs are further decomposed into primal and master problems whose solutions provide respectively lower and upper bounds for the solutions

of the sub-regions. By removing the sub-regions from the search space that are identified as those without an improved solution, the size of the search space reduces at each iteration and results in the convergence of the algorithm.

6.1. Problem formulation

This section presents the proposed formulation for simultaneous traffic metering location and level optimization. The formulation selects a certain number of metering locations among a larger set of candidate locations for traffic metering. Traffic metering levels are defined as dynamic green or red signal indications for the metering signals.

We define G and T as the sets of all candidate locations for traffic metering and analysis time steps, respectively. The decision variables are metering locations ϕ_i for each candidate location $i \in G$, and traffic metering level R_i^t for each location $i \in G$ at time step $t \in T$. Metering locations ϕ_i are binary variables that take on the value of one if location $i \in G$ is selected for traffic metering and zero otherwise. Traffic metering levels R_i^t are also binary variables that show the signal indication of metering location $i \in G$ at time step $t \in T$ (one for green, zero for red).

We considered several constraints on metering locations and levels. Let γ_i be the installation and operational costs of metering location $i \in G$. Constraint (6-1) restricts the total costs of the selected metering locations, $\sum_{i \in G} \gamma_i \phi_i$, to the available budget B. Constraint (6-2) ensures that the metering levels can take any values greater than or equal to zero if a location is selected for traffic metering ($\phi_i = 1$). On the other hand, this constraint sets the traffic metering levels to one when a location is not selected ($\phi_i = 0$) that is the baseline scenario representing a no-metering condition.

$$\sum_{i \in G} \gamma_i \phi_i \leq B \tag{6-1}$$

$$R_i^t \geq 1 - \phi_i \qquad \forall i \in G, t \in T \tag{6-2}$$

The maximum outflow ζ_i^t at candidate location $i \in G$ at time step $t \in T$ is defined by Constraint (6-3). If location $i \in G$ is not selected for metering ($\phi_i = 0$), the expression $\phi_i R_i^t Q_i^t$ will be zero, and the maximum outflow ζ_i^t will be equal to Q_i^t that is the saturation flow rate of location $i \in G$. On the other hand, if location $i \in G$ is selected for traffic metering ($\phi_i = 1$), the maximum outflow ζ_i^t will be equal to $R_i^t Q_i^t$. Therefore, the maximum outflow of the location can take either zero or Q_i^t values depending on the metering level R_i^t. Constraint (6-3) is defined by the multiplication of two decision variables ϕ_i and R_i^t that makes it non-linear. Note that this constraint can be linearized by introducing auxiliary binary variables and several linear constraints.

$$\zeta_i^t = \phi_i R_i^t Q_i^t + (1 - \phi_i) Q_i^t \qquad \forall i \in G, t \in T \tag{6-3}$$

Constraints (6-4) and (6-5) ensure the integrality condition of the traffic metering levels and locations, respectively.

$$R_i^t \in \{0,1\} \qquad \forall i \in G, t \in T \tag{6-4}$$

$$\phi_i \in \{0,1\} \qquad \forall i \in G \tag{6-5}$$

Constraint (6-6) shows the flow feasibility conditions for metering location at cell $i \in G$. Flow of vehicles y_{ij}^t advancing to the cell $j \in S(i)$ at time step $t \in T$ depends on x_i^t, Q_j^t, $\rho\left(N_j - x_j^t\right)$, and ζ_i^t. The maximum outflow ζ_i^t adjusts the flow of vehicles leaving the metering location $i \in G$ according to the optimized traffic metering levels, see Constraints (6-3).

$$y_{ij}^t = \min\left\{x_i^t, Q_j^t, \rho\left(N_j - x_j^t\right), \zeta_i^t\right\} \qquad \forall i \in G, j \in S(i), t \in T \tag{6-6}$$

If we add the discussed constraints to the flow feasibility and conservation constraints that are discussed in Chapter 3, the integrated traffic metering location and level optimization can be

expressed as follows:

$$\text{maximize } Z = \sum_{t \in T} \sum_{i \in C \setminus C_S} x_i^t \tag{6-7}$$

subject to Constraints (3-7)-(3-13), and (6-1)-(6-6)

6.2. Solution technique

The main ideas of the solution technique are as follows:

(1) The feasible region of the optimization problem (6-7) can be divided into several smaller sub-regions with respect to the metering location decision variables.

(2) Two easier MINLPs compared to the original problem can be constructed for each sub-region that only optimize traffic metering levels for fixed metering locations. These MILNPs can be solved by decomposing them into primal and master problems that need to be solved iteratively to optimize traffic metering levels.

(3) The solutions to the MINLPs provide upper and lower bounds for the solutions of the original problem within each sub-region.

(4) The bounds can verify whether a sub-region includes the optimized solutions for the original problem or not. A sub-region that does not include such solutions will be removed from the set of sub-regions, which results in a smaller feasible region for the original problem.

Dividing the feasible region of a problem into smaller sub-regions and finding bounds for them is the idea of the branch-and-bound technique (Bianchessi and Irnich, 2019; Dell'Amico et al., 2006; He et al., 2019; Lawler and Wood, 1966), but the technique is modified to solve the integrated problem of the present study as shown in Figure 6-1. The solution technique has five

components of Branching (Section 6.2.1), Upper Bound Calculation (Section 6.2.2), Lower Bound Calculation (Section 6.2.3), Metering Level Optimization (Section 6.2.4), and Pruning (Section 6.2.5) that are discussed in the following sections.

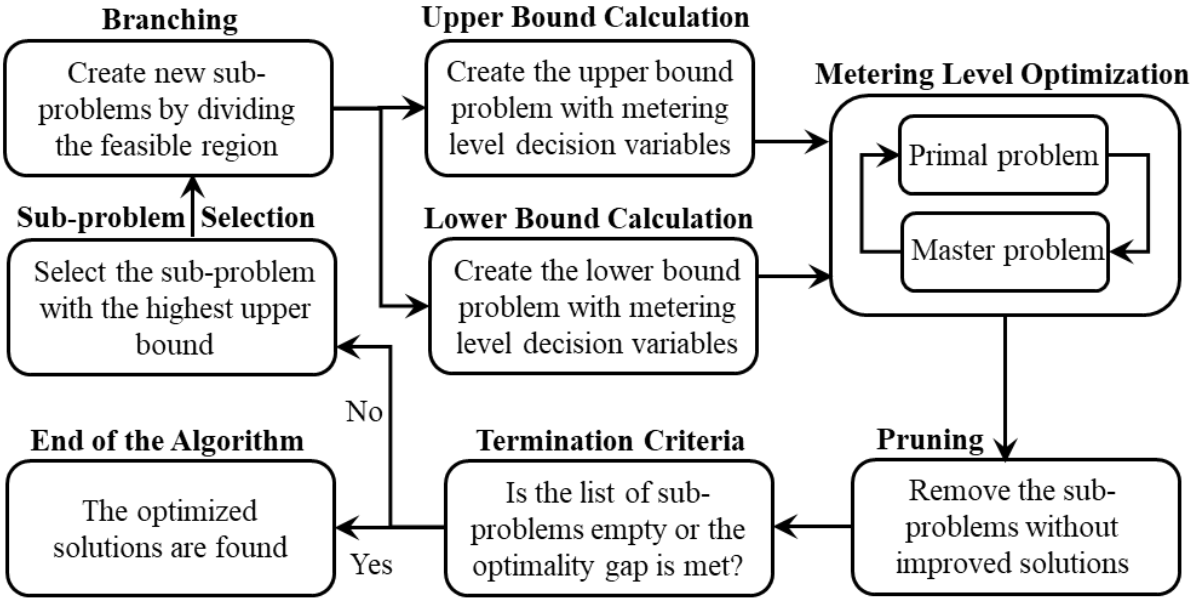


Figure 6-1 Overview of the solution technique to solve the optimization program

6.2.1. Branching

Branching divides the feasible region of problem (6-7) into smaller sub-regions with respect to metering location $i \in G$. Metering location decision variables can take values of either zero or one, see Constraint (6-5). Therefore, branching over location $i \in G$ creates two mutually exclusive feasible regions where $\phi_i = 0$ in one of the regions and $\phi_i = 1$ in the other one. Accordingly, new optimization programs or sub-problems can be constructed for each of the feasible regions. The sub-problems are identical to the original problem but with the addition of constraints $\phi_i = 0$ or $\phi_i = 1$. Definition 1 presents the definition of a sub-problem.

Definition 1. Sub-problem $SP(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1)$ is an optimization program that will be created after branching over gates $\{i_1, .., i_m, j_1, .., j_n\} \in G$ where $m + n \leq$

$|G|$, and it is equal to program (6-7) with the addition of $m+n$ constraints $\phi_{i_1} = \cdots = \phi_{i_m} = 0$ and $\phi_{j_1} = \cdots = \phi_{j_n} = 1$. Note that $|.|$ shows the cardinality of a set and all other metering locations $k \in G \backslash \{i_1, \ldots, i_m, j_1, \ldots, j_n\}$ are un-branched locations for this sub-problem.

It can be verified that the created sub-problems after branching are still joint optimization of metering location and level with smaller feasible areas compared to the original problem (because of the added constraints to each sub-problem). Besides, it is shown in Proposition 6-1 that rather than solving one optimization program (6-7) several sub-problems should be solved to find the solutions to the original problem. However, if we find tight upper and lower bounds for each of the sub-problems by solving an *easier problem*, the sub-problems that will not contain the solutions of the original problem can be identified and removed from the analysis.

Proposition 6-1. The optimal solution Z^* of (6-7) is equal to (6-8) provided that $SP(\phi_i = 0)$ and $SP(\phi_i = 1)$ are feasible.

$$Z^* = \max\{Z^*(SP(\phi_i = 0)), Z^*(SP(\phi_i = 1))\} \quad (6\text{-}8)$$

Proof. Let $\mathcal{L}$, $\mathcal{L}_0$, and $\mathcal{L}_1$ be the feasible regions of (6-7), $SP(\phi_i = 0)$, and $SP(\phi_i = 1)$, respectively. Branching ensures that $\mathcal{L} = \mathcal{L}_0 \cup \mathcal{L}_1$. Thus, the optimal solutions will be either in $\mathcal{L}_0$ or $\mathcal{L}_1$. Hence, Z^* that is the optimal objective value of (6-7) is equal to the maximum objective value of the sub-problems. Note that if $SP(\phi_i = 0)$ is infeasible or mathematically $\mathcal{L}_0 = \emptyset$, then $\mathcal{L} = \mathcal{L}_1$ and $Z^* = Z^*(SP(\phi_i = 1))$. Similarly, if $\mathcal{L}_1 = \emptyset$, $Z^* = Z^*(SP(\phi_i = 0))$. The results of this proposition can be extended to branching the created sub-problems multiple times and finding the optimal solutions by solving more sub-problems each with a smaller feasible area. □

6.2.2. Upper bound calculation

This section introduces an optimization program whose solutions are upper bound Z_u for the sub-problems that are created by several branching over metering location variables. Upper bound Z_u for the objective value of sub-problem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1\right)$ with metering locations $\{i_1, .., i_m, j_1, .., j_n\} \in G$ and $m + n \leq |G|$ can be found by solving the upper bound problem (UBP), see Proposition 6-2 for more details.

(UBP) $$Z_u = Z^*\left(SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1, \phi_{k_1} = \cdots = \phi_{k_p} = 1\right)\right)$$

where $\{k_1, \dots, k_p\} = G \backslash \{i_1, \dots, i_m, j_1, \dots, j_n\}$, and Constraint (6-1) is relaxed.

Proposition 6-2. Upper bound Z_u for the objective value of sub-problem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1\right)$ with gates $\{i_1, .., i_m, j_1, .., j_n\} \in G$ and $m + n \leq |G|$ can be found by solving the upper bound problem (UBP).

(UBP) $$Z_u = Z^*\left(SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1, \phi_{k_1} = \cdots = \phi_{k_p} = 1\right)\right)$$

where $\{k_1, \dots, k_p\} = G \backslash \{i_1, \dots, i_m, j_1, \dots, j_n\}$, and Constraint (6-1) is relaxed.

Proof. Because Constraint (6-1) that limits the total costs to B is relaxed in (UBP), adding more metering locations to the set of selected locations increases the feasible area and hence does not deteriorate the objective value. Therefore, the best objective value can be found once all unbranched gates $\{k_1, \dots, k_p\} = G \backslash \{i_1, \dots, i_m, j_1, \dots, j_n\}$ are set to one with constraints $\phi_{k_1} = \cdots = \phi_{k_p} = 1$ that results in the upper bound problem (UBP). □

There exists one constraint in (UBP) for each metering location $i \in G$ that sets decision variables ϕ_i equal to one or zero. Accordingly, *all* metering location decision variables are fixed

and the upper bound problem (UBP) only optimizes traffic metering levels for the metering locations whose values are set to one. Hence, (UBP) can be solved efficiently with the available techniques in the literature. We will discuss the employed approach to optimize traffic metering levels for predefined metering locations in Section 6.2.4.

6.2.3. Lower bound calculation

The lower bound Z_l for the objective value of the sub-problem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1\right)$ with metering locations $\{i_1, .., i_m, j_1, .., j_n\} \in G,\ m + n \leq |G|$ and $\sum_{r=1}^{m} \gamma_{j_r} + \sum_{r=1}^{q} \gamma_{l_r} \leq B$ can be found by solving the lower bound problem (LBP) and is feasible to (6-7), see Proposition 6-3 for more details.

(LBP) $Z_l = Z^* \left(SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_m} = 1, \phi_{k_1} = \cdots = \phi_{k_p} = 0, \phi_{l_1} = \cdots = \phi_{l_q} = 1\right)\right)$

where $\{k_1, \ldots, k_p, l_1, \ldots, l_q\} \in G$, $n + p = B$, and

$\{i_1, .., i_m, j_1, .., j_n, k_1, .., k_p, l_1, .., l_q\} = G$

Proposition 6-3. The lower bound Z_l for the objective value of the sub-problem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_n} = 1\right)$ with metering locations $\{i_1, .., i_m, j_1, .., j_n\} \in G$, $m + n \leq |G|$ and $\sum_{r=1}^{m} \gamma_{j_r} + \sum_{r=1}^{q} \gamma_{l_r} \leq B$ can be found by solving the lower bound problem (LBP)) and is feasible to (6-7).

(LBP) $Z_l = Z^* \left(SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 0, \phi_{j_1} = \cdots = \phi_{j_m} = 1, \phi_{k_1} = \cdots = \phi_{k_p} = 0, \phi_{l_1} = \cdots = \phi_{l_q} = 1\right)\right)$

where $\{k_1, \ldots, k_p, l_1, \ldots, l_q\} \in G$, $n + p = B$, and

$\{i_1, .., i_m, j_1, .., j_n, k_1, .., k_p, l_1, .., l_q\} = G$

Proof. According to Definition 1, $SP(.)$ has all constraints of (6-7) with the addition of several

constraints on the metering locations. Thus, (LBP) is feasible to (6-7) if and only if Constraint (6-1) on the budget limit is not violated. This condition will be satisfied in (LBP) because the total costs$\sum_{r=1}^{m}\gamma_{j_r}+\sum_{r=1}^{q}\gamma_{l_r}$ is less than or equal to B, ensuring that it satisfies Constraint (6-1) and is feasible to (6-7). Moreover, any feasible solution to (6-7) is a lower bound for Z^* as (6-7) is a maximization problem. □

Some of the metering locations in (LBP) are already set to one by branching (locations $j_1 \dots j_m$), but the rest should be selected from the un-branched metering locations ($l_1 \dots l_q$). Note that any selection strategy that satisfies $\sum_{r=1}^{m}\gamma_{j_r}+\sum_{r=1}^{q}\gamma_{l_r}\leq B$ in (LBP) is a valid strategy. Accordingly, we propose assigning a value to each metering location that shows its individual contribution to improving network performance over its cost. Then, we select metering locations with the highest contribution value and use them in (LBP). Definition 2 presents the definition of metering location contribution value.

Definition 2. Metering location contribution value is defined as $\mathcal{H}_i = Z^*\left(SP\left(\phi_i = 1, \phi_j = 0 : j \in G\backslash\{i\}\right)\right)/\gamma_i$ for each location $i \in G$. In other words, $\mathcal{H}_i$ is equal to the objective value of (6-7) over its cost if location $i \in G$ is the only location for traffic metering.

Accordingly, we propose to use un-branched locations with the highest contribution values to solve (LBP). With the notations of (LBP), the subset $\{l_1, \dots, l_q\} \in G$ and $\{k_1, \dots, k_p\} \in G$ will be the locations that satisfy $\mathcal{H}_{l_1} \geq \cdots \geq \mathcal{H}_{l_q} \geq \mathcal{H}_{k_1} \geq \cdots \geq \mathcal{H}_{k_p}$ and $\sum_{r=1}^{m}\gamma_{j_r}+\sum_{r=1}^{q}\gamma_{l_r}\leq B$. Note that lower bound problem (LBP) is a problem whose metering location decision variables (ϕ_i: $i \in$

G) are all set to either one or zero.

6.2.4. Traffic metering level optimization

The previous discussions show that the proposed solution technique converts (6-7) with metering location and level decision variables into several optimization programs, upper and lower bound problems (UBP) and (LBP), that only optimize traffic metering levels. Accordingly, (UBP) and (LBP) are optimization programs whose decision variables are traffic metering levels, and metering locations are inputs to them. Accordingly, we define set $\tilde{G}$ as the set of metering locations that are set to one in the upper or lower bound problems. Definition 3 shows the mathematical representation of $\tilde{G}$.

Definition 3. Set $\tilde{G}$ for upper and lower bound problems (UBP) and (LBP), $SP(.)$, is defined as (6-9). According to (6-9), $\tilde{G}$ is the set of all metering locations whose values are set to one by the constraints of $SP(.)$. These metering locations must be used for traffic metering.

$$\tilde{G} = \{i: i \in G, \phi_i = 1 \text{ in } SP(.)\} \tag{6-9}$$

Now, the sub-problem that should be solved for finding upper and lower bounds for (UBP) and (LBP) can be stated as (P). Note that (UBP) and (LBP) have a different set $\tilde{G}$ and hence their traffic metering decision variables are different, but the problems have a similar mathematical representation as (P).

(P) maximize Z

subject to (3-7)-(3-13), and the following constraints:

$$R_i^t \in \{0,1\} \qquad \forall i \in \tilde{G}, t \in T \tag{6-10}$$

$$\zeta_i^t = R_i^t Q_i^t \qquad \forall i \in \tilde{G}, t \in T \tag{6-11}$$

$$y_{ij}^t = \min\{x_i^t, Q_j^t, \rho(N_j - x_j^t), \zeta_i^t\} \qquad \forall i \in \tilde{G}, j \in S(i), t \in T \tag{6-12}$$

The decision variables of (P) are traffic metering levels R_i^t for locations $i \in \tilde{G}$ at time step $t \in T$. Moreover, Constraints (6-1), (6-2), and (6-5) of the original problem that was for the metering location variables are relaxed in (P), and Constraints (6-3), (6-4), and (6-6) of the original problem are modified for gate set $\tilde{G}$ in (P).

Program (P) is an MINLP that is still hard to solve with commercial optimization software. Thus, we employed the approach developed in Chapter 4 to solve (P). One of the main properties of (P) is that its objective value Z can be found with a simple CTM simulation once traffic metering levels are temporality set to fixed values. In other words, if we set traffic metering variables R_i^t equal to $\hat{R}_i^t$ for each $i \in \tilde{G}$ and $t \in T$, all other variables of (P2) such as $\zeta_i^t : i \in \tilde{G}, t \in T$, $x_i^t : i \in C, t \in T$, and $y_{ij}^t : i \in C, j \in S(i), t \in T$ can be found with a CTM simulation because all constraints are equality constraints and represent flow conservation and flow feasibility conditions. This property is utilized to decompose (P) into primal and master problems.

The primal problem (PR) is equal to (P) with fixed values of $R_i^t = \hat{R}_i^t$ for each $i \in \tilde{G}$ and $t \in T$ and relaxation of Constraint (6-10). Solutions to the primal problem will be used in the master problem (MR). The master problem is constructed by relaxing the dual of the original problem and including Constraint (6-10), which was relaxed in the primal problem. Solutions to the master problem are updated values of $\hat{R}_i^t$ for the primal problem. Hence, the iterations between the primal and master problems continue until the optimal solutions are found. The primal and master problems are shown as follows:

(PR) maximize Z

subject to (3-7)-(3-13), (6-12), and the following constraints:

$$\zeta_i^t = \hat{R}_i^t Q_i^t \qquad \forall i \in \tilde{G}, t \in T \tag{6-13}$$

(MR) maximize μ (6-14)

subject to (6-10), and the following constraints:

$$L^k\left(Z^*, \lambda^{*t}_i, \zeta^{*t}_i\right) = Z^* + \sum_{i \in \tilde{G}} \sum_{t \in T} \lambda^{*t}_i \left(\zeta^{*t}_i - R_i^t Q_i^t\right) \geq \mu \qquad k = 1, \dots, K \tag{6-15}$$

At iteration k, the primal problem is solved with a simple CTM simulation, and its solutions Z^*, λ^{*t}_i, and ζ^{*t}_i are used to construct $L^k\left(Z^*, \lambda^{*t}_i, \zeta^{*t}_i\right)$ in the master problem. Note that λ^{*t}_i is the dual value of Constraint (6-12) in the primal problem. Once $L^k\left(Z^*, \lambda^{*t}_i, \zeta^{*t}_i\right)$ is constructed, it will be added as a new constraint (i.e., cut) to the master problem in Constraint (6-15). The decision variables of the master problem are $R_i^t: i \in \tilde{G}, t \in T$, and its solutions are updated traffic metering rates that will be used in the primal problem. Accordingly, one constraint will be added to the master problem at each iteration resulting in a smaller feasible region for the traffic metering rates. This process continues until the optimal solutions are found within an ε-optimality gap where $\varepsilon \geq 0$.

6.2.5. Pruning

Suppose that branching creates a subproblem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 1, \phi_{j_1} = \cdots = \phi_{i_n} = 0\right)$ with m being the number of selected metering locations and Z_u being its upper bound. Moreover, if the best (highest) lower bound of all created sub-problems is shown by Z_L, the following conditions can occur:

(1) $\sum_{r=1}^{m} \gamma_{i_r} > B$: This condition represents an infeasible sub-problem because the total costs of selected metering locations cannot exceed B according to Constraint (6-1). Hence, this sub-problem and all the sub-problems that can be created from it will be infeasible. Hence, this sub-problem will be removed, i.e., pruned, from the set of sub-problems for further branching.

(2) $Z_u \leq Z_L$: This condition states that the best possible objective value (Z_u) of the sub-problem is not better than the objective value of the best-known feasible solution (Z_L). Accordingly, this sub-problem and its further branches do not contain the optimized solutions of the problem, and thus will be pruned.

Note that branching creates a set of sub-problems that each can be further branched. However, once a sub-problem is pruned, it will be removed from the set sub-problems for branching. Accordingly, pruning reduces the search space iteratively.

6.2.6. Solution technique steps

The following steps combine all the discussed components in a procedure for solving the integrated optimization program:

Step 1: Let $\mathcal{F} = \{SP(\emptyset)\}$. $\mathcal{F}$ is a list that includes all created sub-problems by branching. At the first iteration, there is only one sub-problem that is equal to the original problem, i.e., $SP(\emptyset) =$ (6-7).

Step 2: Find upper bound $Z_u^{\emptyset}$ and lower bound $Z_l^{\emptyset}$ for $SP(\emptyset)$ according to the discussions in Sections 6.2.2 and 6.2.3. Set the best upper bound $Z_U = Z_u^{\emptyset}$ and the best lower bound $Z_L = Z_l^{\emptyset}$, and SP^L equal to the lower bound problem, see section 6.2.3.

Step 3: Select a sub-problem $SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 1, \phi_{j_1} = \cdots = \phi_{j_n} = 0\right)$ in $\mathcal{F}$ with the highest

upper bound and a location $k \in G\backslash\{i_1, \dots, i_m, j_1, \dots, j_n\}$ such that $\mathcal{H}_k \geq \mathcal{H}_p$ for each $p \in G\backslash\{i_1, \dots, i_m, j_1, \dots, j_n\}$ and $p \neq k$. This step selects a sub-problem and one of its un-branched gates that has the highest contribution value.

Step 4: Remove the selected $SP(.)$ in Step 2 from $\mathcal{F}$, i.e., $\mathcal{F} \coloneqq \mathcal{F}\backslash SP(.)$.

Step 5: Branch the selected $SP(.)$ over location k and create two new sub-problems $SP^0(.) = SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = 1, \phi_{j_1} = \cdots = \phi_{j_1} = \phi_k = 0\right)$ and $SP^1(.) = SP\left(\phi_{i_1} = \cdots = \phi_{i_m} = \phi_k = 1, \phi_{j_1} = \cdots = \phi_{j_n} = 0\right)$.

Step 6: Find upper and lower bounds Z_u^0 and Z_l^0 for $SP^0(.)$ and Z_u^1 and Z_l^1 for $SP^1(.)$. If any of the sub-problems are infeasible, set their lower bounds to $-\infty$ and their upper bounds to Z_U.

Step 7: Add $SP^0(.)$ and $SP^1(.)$ to $\mathcal{F}$, i.e., $\mathcal{F} \coloneqq \{\mathcal{F}, SP^0(.), SP^1(.)\}$.

Step 8: Set $Z_L \coloneqq \max\left\{Z_L, \max_{SP(.)\in\mathcal{F}}\left\{Z_l^{SP(.)}\right\}\right\}$ and SP^L equal to the lower bound problem that corresponds Z_L.

Step 9: Prune (remove) all $SP(.) \in \mathcal{F}$ with $Z_u^{SP(.)} \leq Z_L$.

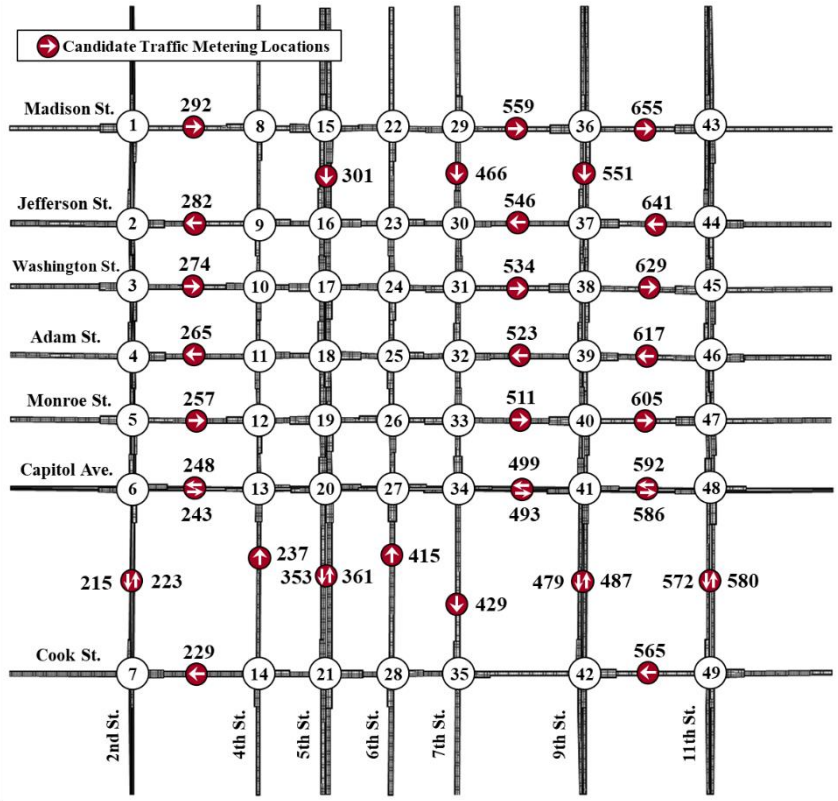
Step 10: If $\frac{(Z_U - Z_L)}{Z_U} \leq \varepsilon$, stop; the optimized solutions of (P1) are the solutions of the lower bound problem SP^L. Otherwise, set $Z_U \coloneqq \max_{SP(.)\in\mathcal{F}}\left\{Z_u^{SP(.)}\right\}$ and return to Step 3. Note that ε is the gap tolerance and is a predefined parameter.

Note that there are several strategies for selecting a sub-problem for branching in Step 3 such as selecting based on the highest lower bound. In this approach, we selected the sub-problems with the highest upper bound as was suggested by (Zhou and Zhong, 2007). Moreover, Step 9 ensures

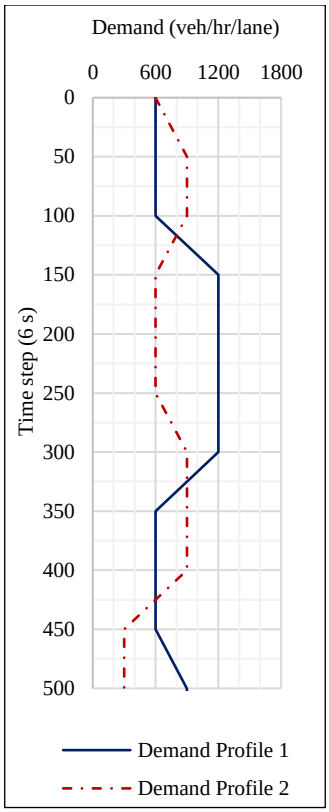
that $\mathcal{F}$ always includes a sub-problem with the best solutions. Since $Z^* = \max_{SP(.)\in\mathcal{F}}\{Z^*(SP(.))\}$, Z_L and Z_U in Steps 8 and 10 are respectively defined as $Z_L = \max\left\{\max_{SP(.)\in\mathcal{F}}\left\{Z_l^{SP(.)}\right\}\right\}$ and $Z_U = \max_{SP(.)\in\mathcal{F}}\left\{Z_u^{SP(.)}\right\}$, inequality $Z_L \leq Z^* \leq Z_U$ holds. Moreover, the algorithm terminates in Step 10 once $\frac{(Z_U - Z_L)}{Z_U} \leq \varepsilon$ indicating that a solution within an ε-optimality gap is found.

6.3. Case study

We implemented the formulation and solution technique in a case study network adapted from downtown Springfield, Illinois, see Figure 5-8(a). The network has 49 intersections with a combination of one-way and two-way streets and lanes ranging from one to three in each direction. The network has 659 cells and is analyzed for 500 time steps with the initial loading of 100 time steps for two different demand profiles that are shown in Figure 5-8(b). Moreover, cell characteristics and other parameters are summarized in Figure 5-8(c).



(a) The modified network of Downtown Springfield, IL



(b) Demand profiles

Parameter	Value	Parameter	Value
Time interval duration (s)	6	Jam density (veh/mile/lane)	288
Length of cells (ft)	225	Turning percentages (%)	10
Free flow speed (mile/hr)	25	Gap (ε, %)	3
Saturation flow rate (veh/hr/lane)	1800	Installation and operational costs of each metering location ($)	650,000

(c) The characteristics of the case study network

Figure 6-2 The case study network, characteristics, and demand profiles

6.4. Numerical results

We considered the following scenarios in our analyses:

1. Metering Location and Level Optimization (MO): In this scenario, we used the proposed formulation and solution technique in the present study to find the best metering locations and their corresponding metering levels for different budgets ranging from $1,300,000 to $19,500,000. The candidate metering locations are those that are shown in see Figure 6-2(a).

2. Perimeter Control (PC): We used 13 boundary metering locations at the entry links to the inner region of the case study network, and only optimized their metering levels. These locations are shown with the labels of 301, 466, 551, 641, 617, 592, 487, 415, 361, 237, 243, 257, and 274 in Figure 6-2(a). This scenario shows a conventional perimeter control strategy where there is no budget limitation for traffic metering and predefined metering locations are at the boundary of a congested area.

3. Simulation (SM): The case study is simulated using its prevailing condition without any traffic metering for this scenario.

In our first analysis, we limited the budget to $3,900,000 and $6,050,000 in Metering Location and Level Optimization (MO) and compared the results with the Simulation (SM) and Perimeter Control (PC) scenarios in Table 6-1. The results of this table for Demand Profile 1 show that network throughput, total travel time, and total delay in MO with a $3,900,000 budget limit were less than 1.28% different from the PC scenario indicating that the two cases resulted in relatively similar network performance. However, the total traffic metering costs in MO with a budget of $3,900,000 were 53.85% lower than the costs of PC. Once the budget was increased to $6,050,000 in MO, the network throughput increased by 7.58%, and total travel time and total delay were decreased by respectively 2.58% and 3.21% in Demand Profile 1. Still, the cost of MO with a budget of $6,050,00 was 23.08% lower than the costs of PC. In Demand Profile 2, similar trends to Demand Profile 1 were observed. The network throughput was improved between 0.92% and 7.50%, and total travel time and total delay were reduced between 1.98% and 3.42% compared to PC for the two discussed budget limits. In both MO and PC scenarios, network performance was significantly higher than the Simulation scenario showing that traffic metering had positive effects

on traffic operations of the case study.

The results of Table 6-1 indicate that Metering Location and Level Optimization lead to similar or better network performance compared to Perimeter Control but with significantly lower costs. Therefore, the approach found metering locations that had great impacts on the network performance within a limited budget.

Table 6-1 Comparison of network performance measures for Simulation, Perimeter Control, and Metering Location and Level Optimization scenarios

Performance measure	Simulation (SM)	Perimeter Control (PC)	Metering Location and Level Optimization (MO)			
			$3,900,000 Budget	% Diff. (MO vs. PC)	$6,050,000 Budget	% Diff. (MO vs. PC)
Demand Profile 1						
Throughput (veh)	10,488	11,199	11,304	0.94	12,048	7.58
Total travel time (min)	454,680.2	438,604.9	433,526.9	-1.16	427,290.4	-2.58
Total delay (min)	431,797.5	414,328.6	409,033.7	-1.28	401,017.8	-3.21
Total traffic metering cost ($)	0	8,500,000	3,900,000	-53.85	6,050,000	-23.08
Demand Profile 2						
Throughput (veh)	10,184	10,802	10,901	0.92	11,613	7.50
Total travel time (min)	342,802.1	328,752.7	322,258.6	-1.98	319,760.9	-2.74
Total delay (min)	322,706.2	307,012.4	300,702.1	-2.06	296,518.9	-3.42
Total traffic metering cost ($)	0	8,500,000	3,900,000	-53.85	6,050,000	-23.08

Figure 6-3 shows the optimized metering locations in MO in both Demand Profiles 1 and 2. The figure shows that the optimized locations were a combination of metering locations in different parts of the network. Hence, the cooperation of different metering locations improved traffic operations to meet the defined budget limitations (see Table 6-1).

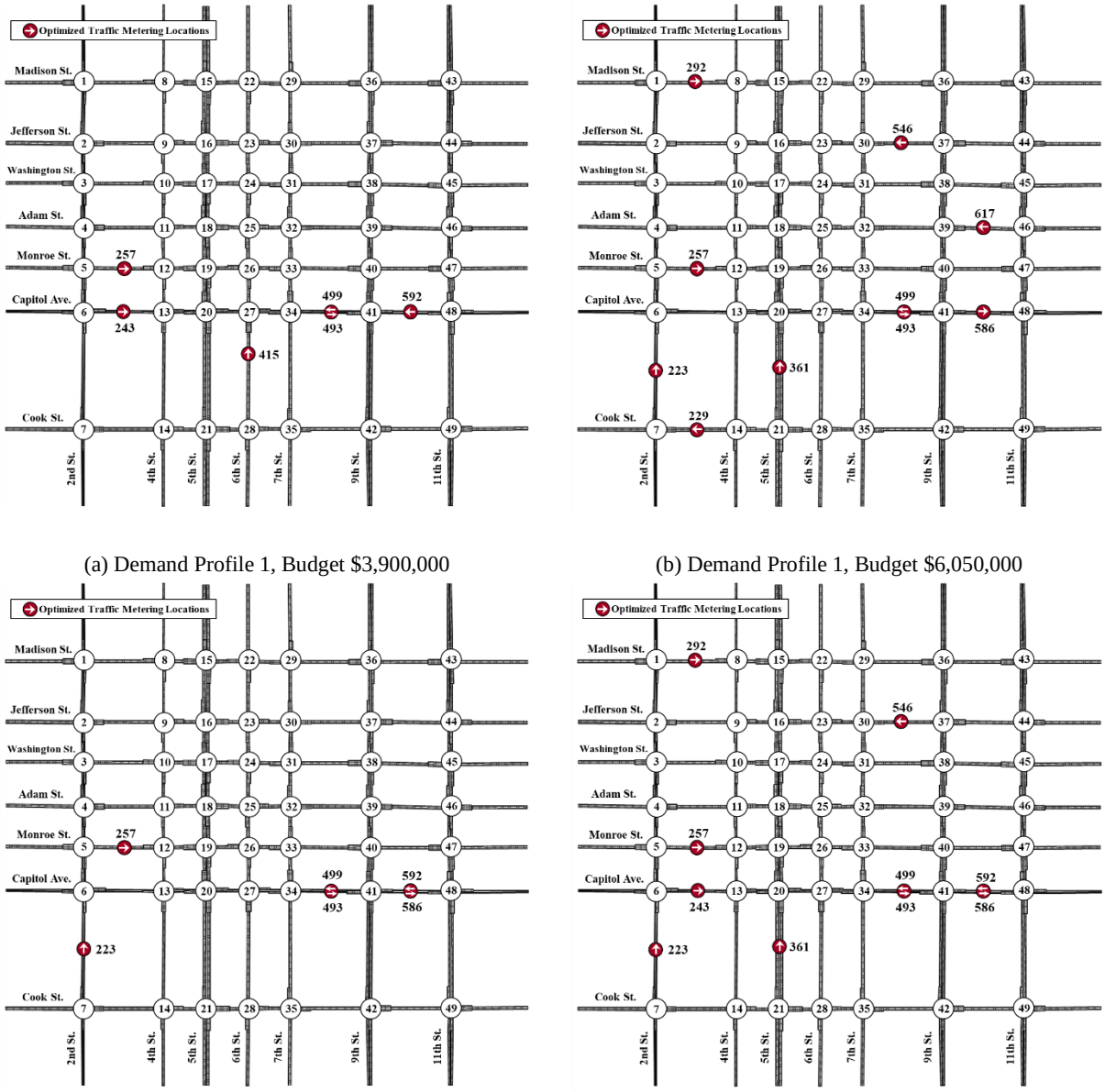


Figure 6-3 Optimized metering locations in Demand Profiles 1 and 2 and budget limits $3,900,000 and $6,050,000

Optimized traffic metering levels of MO in Demand Profile 1 and a budget limit of $3,900,000 are shown in Figure 6-4. The figure shows that traffic metering levels switched between zero (red signal indication) and one (green signal indication) dynamically. The duration of each time step is 6 seconds. Hence, the duration of each traffic metering decision was at least 6 seconds. Moreover, we observed similar dynamic trends in other evaluated cases.

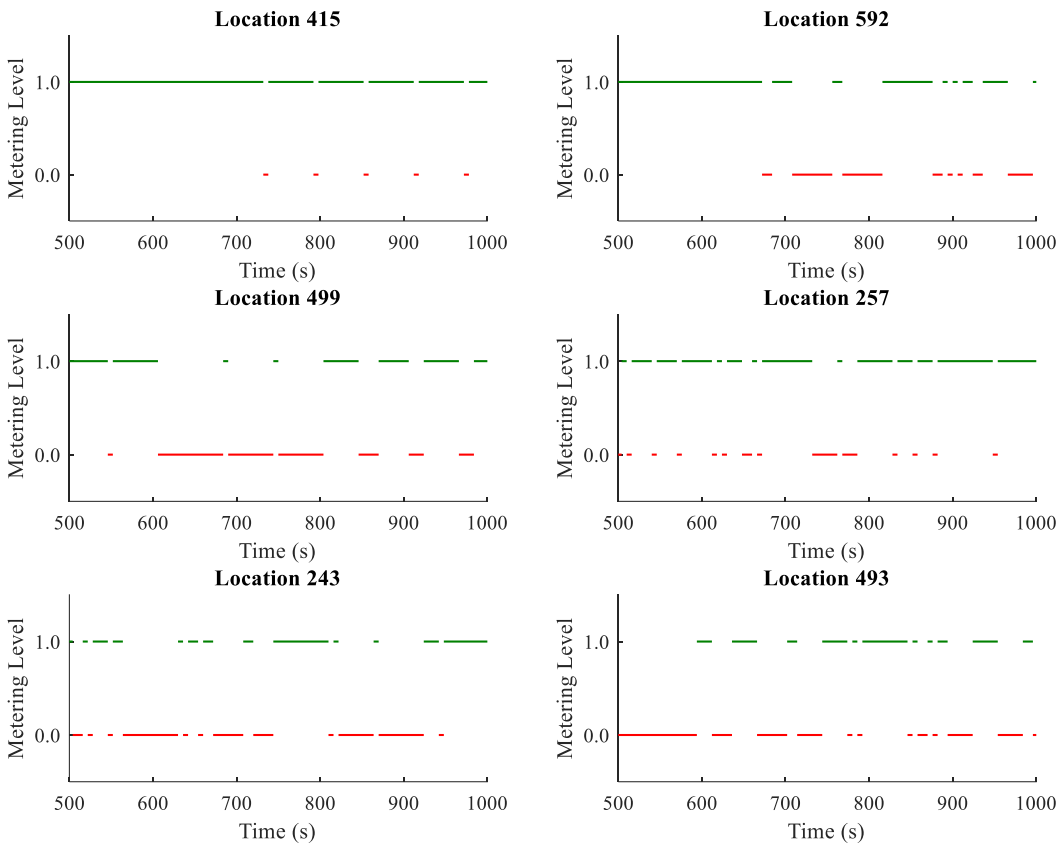
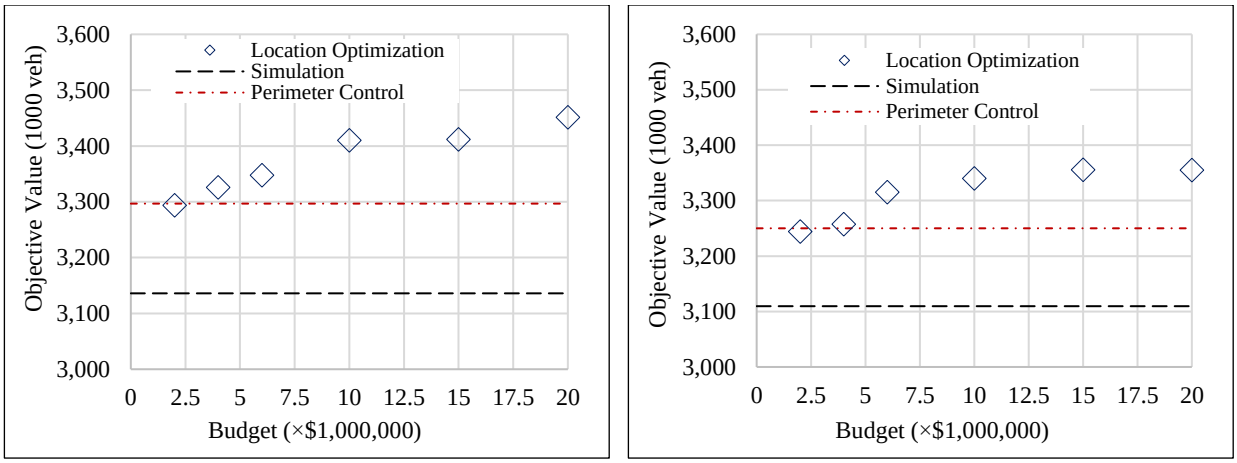


Figure 6-4 Optimized traffic metering levels for the metering locations in Demand Profile 1 and a budget limit of $3,900,000

The previous analysis showed different performance measures and metering levels when the budget was limited to $3,900,000 and $6,050,000. We performed a sensitivity analysis on the objective value of (P1) for budget limits of $1,210,000, $2,420,000, $9,075,000, $12,100,000, and $18,150,000. We also compared the results with the objective values of Perimeter Control and Simulation scenarios. Figure 6-5 shows that the objective value of (P1) increased with the number of metering locations. This observation was expected because increasing the number of metering locations should not deteriorate the objective value. The sensitivity analysis also shows that the improvements by increasing the budget by more than $2,420,000 were negligible. Thus, the operational costs were increased without considerable improvements in traffic operations with budget limits beyond $2,420,000.



(a) Demand Profile 1

(b) Demand Profile 2

Figure 6-5 The sensitivity of objective function to different budgets and their comparison with the objective values of Perimeter Control, and Simulation scenarios

The efficiency of the solution technique is evaluated by comparing its run-time with CPLEX (CPLEX, 2009) as a commercial optimization solver. Since CPLEX cannot consider the closed-form of min(.) operators in the flow feasibility constraint of (P1), we linearized the min(.) operators with the big-M technique discussed in Chapter 2. Moreover, we set the maximum run-time of CPLEX to 24 hours and used a computer with 16 cores and 24 gigabytes of memory to report the run-times. The reported results in Table 6-2 show that the proposed solution technique found solutions in 0.78 to 3.86 hours with an average of 1.85 hours. On the other hand, CPLEX did not find any solutions for the evaluated scenarios within 24 hours because (P1) included many binary variables that made the problem complex. The binary variables were metering locations, levels, and the variables that were required to linearize the min (.) operators with the big-M technique.

Table 6-2 Runtime comparison between the proposed solution technique and CPLEX

Budget ($)	4		6		10		20	
Parameter	Objective value (veh)	Run time (hr)	Objective value (veh)	Run time (hr)	Objective value (veh)	Run time (hr)	Objective value (veh)	Run time (hr)
Metering Location and Level Optimization Solution Technique								
Demand Profile 1	3,325,604	3.86	3,347,496	2.73	3,409,862	1.47	3,451,470	0.78
Demand Profile 2	3,257,693	2.15	3,315,086	1.57	3,340,063	1.17	3,354,830	1.19
CPLEX								
Demand Profile 1	*NFI**	24	*NFI**	24	*NFI**	24	*NFI**	24
Demand Profile 2	*NFI**	24	*NFI**	24	*NFI**	24	*NFI**	24

NFI: No feasible integer solution was found within the 24-hour runtime period

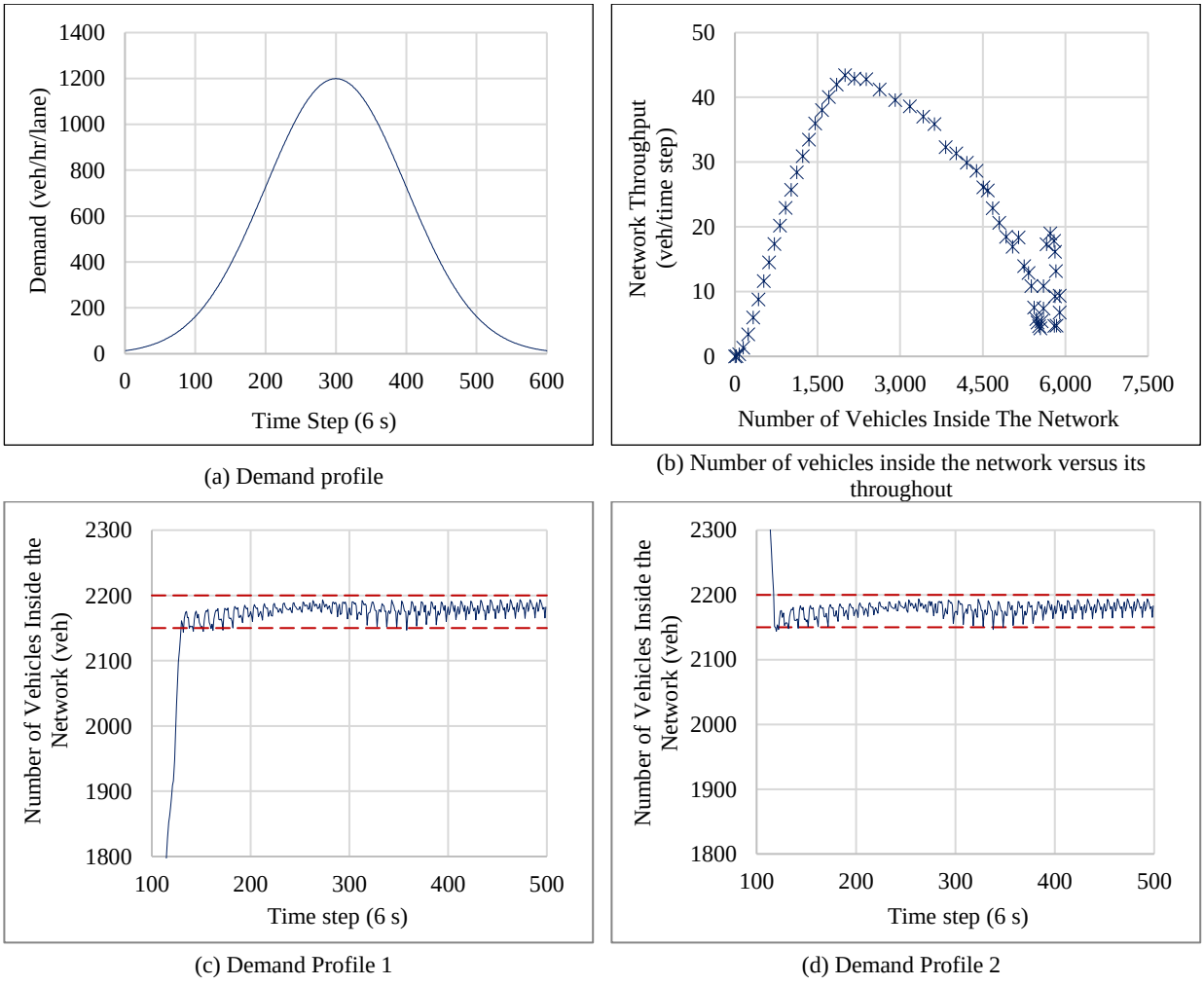
We used additional scenarios in our final benchmark comparison:

1. LP Relaxation (LR): We relaxed all integrality constraints of (P1) and converted (P1) into a linear program. In other words, we converted binary metering location and level decision variables into continuous variables and replaced the min(.) operators of the flow-feasibility constraints with a set of linear inequalities similar to the feasibility constraints proposed in (Lo, 1999). Accordingly, the LP Relaxation provided a "theoretical" non-achievable upper bound for (P1) where the budget limitation is not a restricting constraint in the optimization program. We used CPLEX to solve the linearized program.

2. Metering Location and Level Optimization with Relaxed Budget (MB): The budget constraint in (P1) is relaxed and the solution technique in the present paper is used to solve the modified program. The budget constraint is relaxed to have a fair comparison between this scenario and the LP Relaxation scenario.

3. MFD-based Perimeter Control (MFC): We used the MFD-based perimeter control approach

proposed by Daganzo (2007). We simulated the case study using the demand profile shown in Figure 6-6(a) and derived its MFD in Figure 6-6(b). According to Figure 6-6(b), the critical vehicle accumulation is within the [2150,2200] interval. Hence, the metering levels at the boundary metering locations, see Figure 6-2(a), were set to one once the vehicle accumulation level inside the network was less than 2,150 vehicles and was set to zero for accumulation levels higher than 2,200 vehicles. Note that there are many advanced perimeter controllers in the literature, yet this simple approach provides a benchmark for the performance comparison.



(a) Demand profile

(b) Number of vehicles inside the network versus its throughout

(c) Demand Profile 1

(d) Demand Profile 2

Figure 6-6 (a) Demand profile and (b) diagram of vehicle accumulation and network throughput for the MFD-based Perimeter Control scenario

Table 6-3 shows that the objective values of Metering Location and Level Optimization with

Relaxed Budget (MB) were 4.49% and 6.17% lower than the LP Relaxation solutions in respectively Demand Profiles 1 and 2. This comparison indicates that solutions of MB were within a reasonable gap from the upper bound solutions of LP Relaxation. The number of vehicles inside the network in the MFD-based Perimeter Control is shown in Figure 6-6(c) and (d). According to the figure, the latter scenario could maintain the number of vehicles around the setpoint below the critical accumulation of 2,200 vehicles. However, the MB solutions outperformed the MFD-based Perimeter Control solutions by 24.48% and 25.35% in Demand Profiles 1 and 2. These results show that maintaining the number of vehicles around the setpoint improved traffic operations compared to the current state of the network (see the objective value of the Simulation scenario in Figure 6-5), but the metering location and level optimization provided higher potential for improvements in the network performance.

Table 6-3 Comparison of objective values (veh) for LP Relaxation (LR), MFD-based Perimeter Control (MFC), and Metering Location and Level Optimization with Relaxed Budget (MB)

Scenario	LR	MFC	MB	% Diff. (MB vs. LR)	% Diff. (MB vs. MFC)
Demand Profile 1	4,346,962	3,256,908	4,154,901	-4.49	24.48
Demand Profile 2	4,383,163	3,281,035	4,112,792	-6.17	25.35

CHAPTER 7. A DEEP REINFORCEMENT LEARNING TECHNIQUE FOR TRAFFIC METERING

Traffic metering is a network-level strategy, and it needs to capture traffic dynamics on a network-level scale. This scale creates significant modeling and computational challenges for optimizing traffic metering signals. The available traffic metering strategies addressed these challenges by simplifying traffic dynamics using macroscopic fundamental diagrams (MFDs) (Aboudolas and Geroliminis, 2013; Geroliminis et al., 2013; Keyvan-Ekbatani et al., 2015b; Kouvelas et al., 2017; Ramezani et al., 2015) or with the time-space discretization of the hydrodynamic traffic flow model (Mohebifard and Hajbabaie, 2019a, 2018b). In addition, the latter set of techniques requires high computational resources to optimize traffic metering signals.

In the present study, a deep reinforcement learning (RL) methodology (Huo et al., 2020; Pandey et al., 2020; Tan et al., 2019; Yan and Xu, 2020) is proposed to address the limitations of the existing studies. Deep RL has been widely used in traffic signal timing and some of these studies are reviewed here. This review focused on the scope of their problem and their employed methodologies.

Arel et al. (2010) stated that most of the learning methodologies fail to scale with the size of a transportation network due to the lack of a long-term reward policy. This long-term reward policy ensures that intersections can cooperate with each other to improve operations over a transportation network. Therefore, they employed a Q-learning methodology to approximate a decision-making policy. They considered a special type of network configuration where a central intersection is surrounded by four outer intersections. Their methodology used a longest-queue-first algorithm to control the outer intersections while the innermost intersection used traffic estimations of its upstream intersections to learn the optimal control policy. They defined the system state as the

total delay of vehicles in a lane divided by the average delay of all lanes in the intersection and the action set included all possible non-conflicting green signal indications. Bakker et al. (2010) developed a multi-agent reinforcement learning methodology where each individual intersection operates locally using the traffic state estimations of its surroundings. They also added the capability of receiving congestion information from other intersections and using a coordination graph to connect neighboring intersections to improve the local solutions for a network-level application. Abdoos et al. (2014) suggested that control of signalized intersections is a challenging problem, and thus they integrated Q-learning and tile coding (Sutton, 1995) to develop a hierarchical control methodology. The methodology aimed at reducing the complexity of defining a set of states for a transportation network as inputs in a learning-based methodology. Therefore, the set of states was defined for each individual intersection in the lower level of the hierarchy and tile coding is used as a linear function approximation method to coordinate individual intersections. L. Li et al. (2016) considered using a Deep Q Network (DQN) for traffic signal control to maximize the traffic outflow or throughput of a single intersection. In their case, the Q-function was approximated effectively by using a deep-stacked autoencoder neural network (Lange and Riedmiller, 2010). Genders and Razavi (2016) proposed a discrete traffic state representation and applied DQN to a single isolated intersection control with multiple lanes. However, training and maintaining one deep neural network per intersection was computationally intractable for large-scale transportation networks. Besides, the absence of coordination between intersections prevented achieving near-optimal solutions. To coordinate intersections, der Pol and Oliehoek (2016) extended the solution of a single-agent DQN to multi-agent settings using transfer planning and the max-plus coordination algorithm (Kok and Vlassis, 2005). They first trained a DQN agent on a base problem with two intersections and used transfer planning to extend the

solution of the base problem to multi-agent settings with up to four intersections. The max-plus coordination algorithm was used for action optimization. Casas (2017) formulated the problem as a continuous control problem and applied the deep deterministic policy gradient (DDPG) (Lillicrap et al., 2015) to centralized traffic signal control over the entire traffic system. However, this centralized approach only achieved slightly better performance than a basic Q-learning on a traffic grid with six intersections in their experiments. Mousavi et al. (2017) developed separate reinforcement learning-based models, one policy-based and one value-based controller that receives an image of an intersection as inputs. The actions were defined as whether north-south or east-west approaches receive green at each time step. They stated that combination of policy and value networks prevented instability in the training process.

Aslani et al. (2018) compared several reinforcement learning methodologies including discrete and continuous state Q-learning, actor-critic, and residual actor-critic. The results of a case study in Tehran, Iran in a microsimulation environment showed that the continuous actor-critic method had superior performance compared to the other techniques. The study by Chen et al. (2020) used deep reinforcement learning to develop a methodology that can be applied to large-scale transportation networks. Similar to previous studies, they also used local agents for intersections and each agent decides to whether switch a signal phase or continue with its current phase. They implicitly captured signal coordination using a max-pressure concept (Varaiya, 2013) into the reward function to implicitly coordinate traffic signals. Accordingly, queue lengths at intersections were the main state variables. Garg et al. (2018) designed a customized 3D traffic simulator to employ a deep reinforcement learning approach for traffic signal control. While the car-following and lane-changing models of the simulator were not discussed, the simulator allowed taking 3D pictures of vehicles at an intersection. The pictures represented traffic states and were used as

inputs to convolutional layers of a deep neural network. Their results indicate that the developed methodology could have similar performance compared to a fixed-time signal control setting.

Ge et al. (2019) stated that while reinforcement learning is a promising approach to controlling dynamic systems like traffic signals in transportation networks, coordination of multiple intersections remains a challenge. They developed a cooperative deep Q-network with Q-value transfer between neighboring intersections. In this framework, each intersection controls its own signal indications by receiving the most updated optimal Q-values of its neighboring intersection. Ge et al. (2019) represented traffic state with vehicle position and speed matrices. They divided the entry lanes of each intersection into several cells and represented traffic state as a matrix of zeros and ones depending on whether a cell is occupied by a vehicle or not. Vehicle speeds were also normalized with respect to free-flow speed to construct a matrix with a similar size to the vehicle positions matrix. Gong et al. (2019) argued that the efficiency of most of the learning-based methodologies is evaluated in ideal simulation environments. They used one of the existing reinforcement learning approaches, a double dueling deep Q network, to control a corridor of signalized intersections and tested their controller on a simulated model of a real-world corridor with coordinated actuated traffic signals in Seminole County, Florida. Their analyses results showed a 10.3% reduction in total travel time and 6.5% in total delay.

Despite most of the previously discussed studies optimized traffic signals in a cycle-free setting, Liang et al. (2019) proposed a reinforcement learning technique that determined traffic signal durations in a cycle. Therefore, the actions were the duration of every phase in the next cycle, and they defined the reward as the difference in cumulative waiting times between two cycles. This action and reward definition may result in large variations in phase durations in two consecutive phases, and thus they limited the changes to five seconds between two consecutive phases.

However, this definition limits the capability of the model since the timings of one phase can be adjusted at a cycle.

The developed methodology in the present chapter captures traffic dynamics on a micro-level scale using a simulated model of an urban street network with the capability of continuous online improvement after implementation on an actual transportation network. Besides, the methodology has low complexity and does not require high computational or dedicated optimization resources.

The proposed deep reinforcement learning (DRL) methodology employs deep neural networks to directly optimize the control actions (i.e., traffic metering signals) based on the response of a transportation network to the control actions using an actor-critic training framework. The methodology collects the location information of connected vehicles and feeds them as inputs to two neural networks. The first neural network (actor) maps the inputs to traffic metering signal indications, and the second network (critic) estimates the value of an objective function for the traffic metering optimization problem (Ying et al., 2020; Zhong et al., 2018). The results of the neural networks are used to construct a non-convex loss function whose optimization provides the optimal parameters for the neural networks. Due to the complexity of the loss function, an iterative optimization process based on gradient descent with Monte-Carlo simulation is employed to optimize the loss function.

The following sections provide more background on the traffic metering approaches in the literature and further elaborate on the present study's contributions. Then, the developed methodology and the deep RL components are presented. After introducing the case study network, detailed numerical results are discussed. The final section of the paper provides concluding remarks and future research directions.

In summary, the MFD-based techniques represent traffic dynamics with a macroscopic

diagram. The inherent assumptions on the MFDs for a network, such as homogeneity of traffic distribution, limit their application to a general transportation network. Besides, a microscopic traffic flow model can capture more details on the interaction of vehicles, queue formation, and congestion propagation throughout a transportation network. On the other hand, the CTM-based techniques employed a more detailed traffic flow model in an optimization framework. However, they require high computational and optimization resources. This study fills this gap by:

1. Developing a methodology that optimizes traffic metering signals using a microscopic traffic flow model to represent traffic dynamics more accurately.

2. Reducing the computational complexity of traffic metering for real-time and online applications by developing a deep RL methodology that employs neural networks to determine traffic metering signal indications based on the network state.

7.1. Methodology

In this section, the proposed deep RL methodology is discussed. The *state, action,* and *reward* and their related notations are defined below:

State: The state of a network s^t at time $t \in T$ is a vector that shows the distribution of vehicles (i.e., the ratio of the number of vehicles in a link to its capacity) in a transportation network, and T is the set of all analysis time steps. Since the distribution of vehicles over network links (i.e., roadways) is not uniform, the links are divided into smaller sections and the number of vehicles in each section over the total number of vehicles that can fit in the section is estimated. Accordingly, $s^t = \{s_i^t : i \in C\}$ where s_i^t is the number of vehicles in each section $i \in C$ over its capacity for holding vehicles and C is the set of all sections. More discussions on the sections and using the

information of connected vehicles to estimate the number of vehicles in each section are presented in the next section.

Action: Signal indication g_i^t at traffic metering location $i \in G$ is defined as the action at each time step $t \in T$. The decision variable g_i^t takes on the value of one for a green and zero for a red signal indication, and G is the set of all traffic metering signals. Moreover, g^t is defined as the set of all traffic metering signal indications, i.e., $g^t = \{g_i^t : i \in G\}$.

Reward: The reward r^t is defined as the cumulative number of completed trips from the beginning of the analysis time $\tau = 0$ up to time $\tau = t$ for $\tau, t \in T$. This reward definition is consistent with the desired traffic metering objective that aims to maximize the number of completed in a transportation network (Daganzo, 2007; Mohebifard and Hajbabaie, 2018b).

The list of all notations in the present study is presented in Table 7-1. Moreover, the following section elaborates further on the network state definition and is followed by the employed deep RL model architecture details.

Table 7-1 Notations and their Definition

Sets	
T	Set of all time steps
C	Set of all sections
G	Set of all traffic metering signals
S	Set of all network states
A	Set of all actions
Decision variables	
θ, θ'	Parameters of the policy and value neural networks, respectively
Variables	
g_i^t	Signal indication of metering signal $i \in G$ at time step $t \in T$
g^t	Vector of gate signal indications, i.e., $g^t = \{g_i^t : i \in G\}$
z_{s^t}	Expected rewards for state s^t
$V_{\theta'}$	The output of the value neural network
x_i	Number of vehicles in section $i \in C$
d_i	Number of vehicles x_i over N_i for section $i \in C$
s_i^t	State of section $i \in C$ at time step $t \in T$
s^t	Vector of states, i.e., $s^t = \{s_i^t : i \in C\}$
Parameters	
ε	Step size, $\varepsilon > 0$
N_i	Total number of vehicles that can fit in section $i \in C$

7.1.1. Network State

The network state is defined as the number of vehicles in a road section over the total number

of vehicles that can fit in that section. This idea is based on the successful application of macroscopic traffic flow models such as the cell transmission model for solving different traffic control problems. Therefore, we used the same concept to divide each roadway link into several sections $i \in C$ and defined d_i as the number of vehicles x_i within the boundaries of that section over the total number of vehicles N_i that can fit in that section. Figure 7-1 illustrates the defined sections for a sample link. In this figure, the maximum number of vehicles that can fit in each lane is four vehicles which results in $d_1 = 0.875$ and $d_2 = 0.500$, for instance. Note that mapping the location of vehicles to their corresponding sections and estimating the number of vehicles in each section is straightforward when the connected vehicle market penetration rate is 100%. However, for penetration rates lower than 100%, we need to use a separate state estimation algorithm and approximate the location of unconnected vehicles (Islam et al., 2020; Mohebifard and Hajbabaie, 2018a). In the present study, we assumed a penetration rate of 100% rate for simplicity.

In addition, the input dimension of the neural networks depends on the defined lengths for each section and hence the total number of sections in a network. One strategy to set the length of each section is to consider a constant time step duration and define the section lengths as the distance that a vehicle can travel with a free-flow speed. In the present study, we used a 6 s time step duration (Mohebifard and Hajbabaie, 2019b) that results in sections with 220 ft lengths with a speed limit of 25 mph.

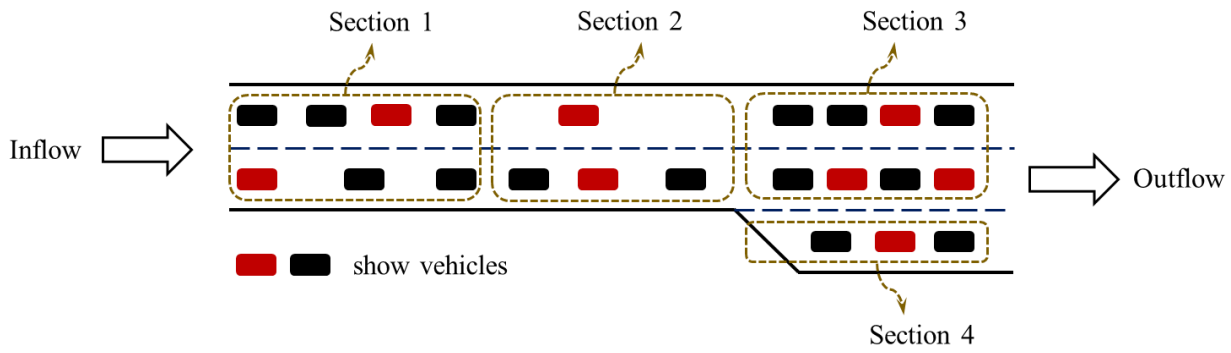


Figure 7-1 Illustration of a transportation link that is divided into four sections.

Once the number of vehicles in each section is estimated, the network state will be defined as a vector including all estimations d_i that is $s^t = [d_i : i \in C]$.

7.1.2. Deep RL Model Architecture

Figure 7-2 shows a general overview of the proposed deep RL model architecture. This architecture is known as actor-critic (Haarnoja et al., 2018) and is adapted in the present study due to its capability of solving large-scale control problems (Tan et al., 2019). In this architecture, traffic state will be used as inputs into two neural networks. The first neural network is a *policy network* with the parameter set of θ and maps the inputs (i.e., network state s^t) to the probability distribution $\pi_\theta(g^t|s^t)$ in the action space. The second neural network is the *value network* $V_{\theta'}(s^t)$ parametrized by θ' and maps state s^t to the expected rewards z_{s^t} over the analysis period $t, t+1, \dots, |T|$. Note that $|.|$ shows the cardinality of a set.

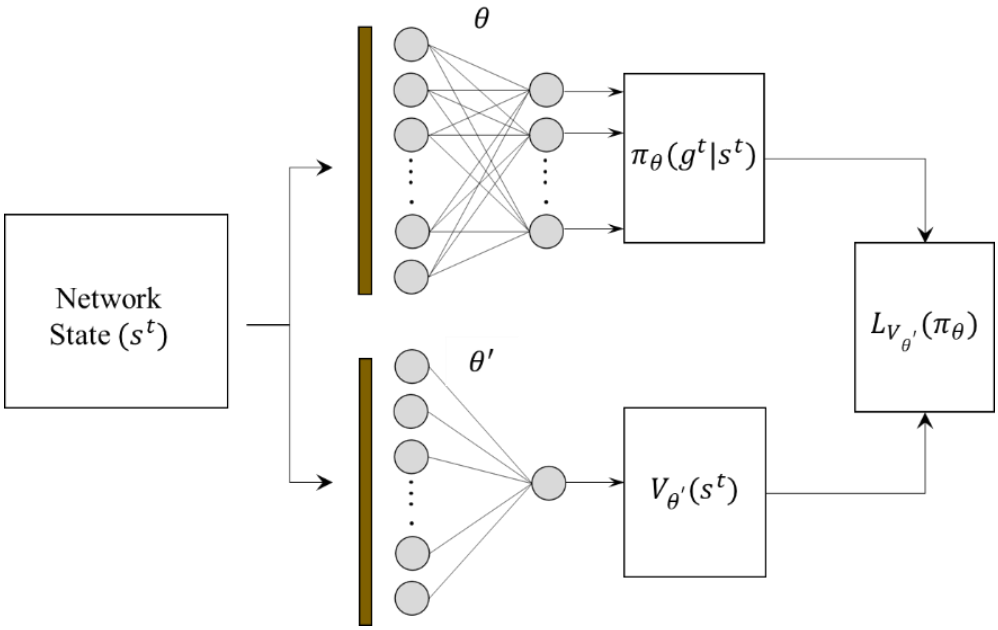


Figure 7-2 General overview of the proposed deep RL model architecture.

Furthermore, $L_{V_{\theta'}}(\pi_\theta)$ in Figure 7-2 is the objective function (i.e., loss function) of the deep RL architecture, which is defined by Equation (7-1). This objective function includes two expressions for each of the neural networks. The first expression is for the policy network and is the summation of average rewards that can be achieved over all states $s^t \in S$ where S is the set of all possible transportation network states. The average reward $\sum_{s^t \in S} p(s^t) \sum_{g^t \in A} \pi_\theta(g^t|s^t) z_{s^t}$ in each state s^t is the multiplication of the probability $p(s^t)$ of being in state s^t and the expected total rewards over all the actions $\sum_{g^t \in A} \pi_\theta(g^t|s^t) z_{s^t}$. In the latter expression, A shows the set of all actions. The second expression of the objective function minimizes the squared difference of value network output $V_{\theta'}$ and the expected rewards z_{s^t}. Therefore, the minimization of $\sum_{s^t \in S} \left(V_{V_{\theta'}} - z_{s^t}\right)^2$ increases the estimation accuracy of the value neural network.

$$\text{maximize } L_{V_{\theta'}}(\pi_\theta) = \sum_{s^t \in S} p(s^t) \sum_{g^t \in A} \pi_\theta(g^t|s^t) z_{s^t} - \sum_{s^t \in S} (V_{\theta'} - z_{s^t})^2 \tag{7-1}$$

The maximization of $L_{V_{\theta'}}(\pi_\theta)$ in (7-1) results in optimal parameter sets θ and θ' of the neural networks and the proposed deep RL architecture. However, the maximization of $L_{V_{\bar{\theta}}}(\pi_\theta)$ (i.e., model training) has several complexities such as non-convexity of $L_{V_{\theta'}}(\pi_\theta)$, large possible state space of S and the unknown state probabilities $p(s^t)$ for a transportation network. Therefore, we employed an iterative gradient descent optimization technique to tackle the non-convexity of $L_{V_{\theta'}}(\pi_\theta)$ and used Monte-Carlo simulation to address the scale of the decision space and the stochasticity of network states for optimizing $L_{V_{\theta'}}(\pi_\theta)$.

7.1.3. Model Training

7.1.3.1. Gradient descent optimization

Optimizing the model parameters θ and θ' requires several steps. In the first step, we use a gradient descent technique to optimize $L_{V_{\theta'}}(\pi_\theta)$ iteratively. Let $\hat{\theta}_k$ and $\widehat{\theta'}$ be the parameter values at iteration k. Then, the new values for the parameters at iteration $k+1$, $\hat{\theta}_{k+1}$ and $\widehat{\theta'}_{k+1}$Can be found with Equations (7-2) and (7-3). According to these equations, the new parameter values will be moved towards the negative direction of the objective function gradients $-\nabla_\theta L_{V_{\theta'}}(\pi_{\hat{\theta}_k})$ and $-\nabla_{\theta'} L_{V_{\theta'}}(\pi_{\widehat{\theta'}_k})$ (the greatest ascents) with respect to θ and θ' at $\hat{\theta}_k$ and $\widehat{\theta'}_k$, respectively, and the step size of $\varepsilon > 0$.

$$\hat{\theta}_{k+1} = \hat{\theta}_k - \varepsilon \nabla_\theta L_{V_{\theta'}}(\pi_{\hat{\theta}_k}) \qquad k = 1,..K \tag{7-2}$$

$$\widehat{\theta'}_{k+1} = \widehat{\theta'}_k - \varepsilon \nabla_{\theta'} L_{V_{\theta'}}(\pi_{\widehat{\theta'}_k}) \qquad k = 1,..K \tag{7-3}$$

The gradients $\nabla_\theta L_{V_{\theta'}}(\pi_{\hat{\theta}_k})$ and $\nabla_{\theta'} L_{V_{\theta'}}(\pi_{\hat{\theta}'_k})$ in Equations (7-2) and (7-3) are vectors whose elements are the derivatives with respect to each individual parameter. Since π_θ will be produced by applying a soft-max function on the outputs of the policy neural network, and the calculation of derivatives will be easier by applying a $\log(.)$ function to $\pi_\theta(g^t|s^t)$ and calculate the derivates based on Equation (7-4). Equation (7-5) also shows the how $\nabla_{\theta'} L_{V_{\theta'}}(\pi_\theta)$ can be calculated.

$$\nabla_\theta L_{V_{\theta'}}(\pi_{\hat{\theta}_k}) = \sum_{s^t \in S} p(s^t) \sum_{g^t \in A} \pi_{\hat{\theta}_k}(g^t|s^t) z_{s^t} \nabla_\theta \log \pi_{\hat{\theta}_k}(g^t|s^t) \tag{7-4}$$

$$\nabla_{\theta'} L_{V_{\theta'}}(\pi_\theta) = 2 \sum_{s^t \in S} \left(V_{\hat{\theta}'_k} - z_{s^t}\right) \nabla_{\theta'} V_{\hat{\theta}'_k} \tag{7-5}$$

7.1.3.2. Monte-Carlo simulation

Although the discussed gradient descent algorithm can optimize $L_{V_{\theta'}}(\pi_\theta)$ despite its non-convexity, the large set of network states S and unknown probabilities $p(s^t)$ in Equations (7-4) and (7-5) prevent calculating the derivatives. Therefore, we use the Monte-Carlo simulation technique to sample the state of a transportation network, take actions according to a sample distribution, and observe the next state of the network and the achieved rewards. Since this approach requires simulating a transportation network, we used Vissim (PTV Group, 2013) traffic simulator to simulate a case study network and implemented the Monte-Carlo simulation technique. Suppose that the policy and value neural networks are initialized with random initial parameters $\hat{\theta}$ and $\hat{\theta}'$ and the state of the simulated network in Vissim is s^t at time $t = 0$. Now, we can perform the following steps:

1. Find action $\hat{g}^t$ from the policy neural network $\pi_{\hat{\theta}}$ such that $\hat{g}^t = \underset{g^t \in A}{\operatorname{argmax}}\{\pi_{\hat{\theta}}(g^t|s^t)\}$,

2. Execute action $\hat{g}^t$ in the simulated network in Vissim and observe the reward r^t (i.e., network

throughput),

3. Find the output of the value neural network $V_{\hat{\theta}'}$, and

4. Advance the current time step of the simulated network; $t := t + 1$.

By executing steps 1 to 4, we can store s^t, $\hat{g}^t$, r^t, $\hat{\pi}_{\hat{\theta}}(g^t|s^t)$, and $\hat{V}_{\hat{\theta}'}(s^t)$ over the entire analysis period $t = 0,1, \dots, |T|$ and estimate the expected total rewards z_{s^t} by $\sum_{\tau=t,t+1,\dots,|T|} r^\tau$. Note that it is conventional to use discounted rewards with a recursive Bellman equation to estimate the expected rewards and assign higher weights to the short-term rewards compared to long-term rewards. However, traffic metering signals should be proactive in reducing traffic congestion before its propagation throughout the network. Therefore, long-term rewards are as important as short-term rewards. As such, rewards will not be discounted.

Using the results of the Monte-Carlo simulation, Equations (7-4) and (7-5) will be modified as Equations (7-6) and (7-7), respectively. Now, the discussed gradient descent approach can use Equations (7-6) and (7-7) to optimize the parameters of the neural networks. Note that the Monte-Carlo simulation starts from a sample initial state and executes actions in the Vissim microsimulation environment. Since the possible states of a transportation network are very large, we need to iterate the optimization on a relatively large number of randomly generated initial states $s^{t=0}$ so that the estimations can result in a good network performance.

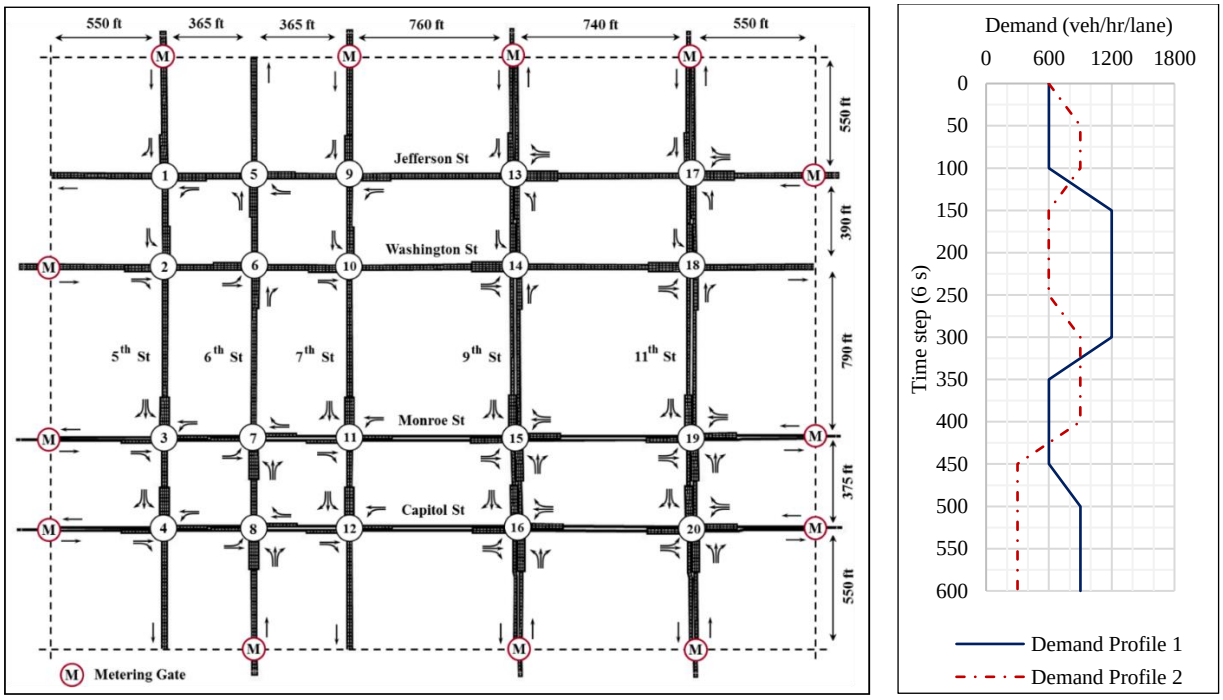
$$\nabla_\theta L_{V_{\theta'}}(\pi_{\hat{\theta}}) = \sum_{t \in T} \sum_{g^t \in A} z_{s^t} \nabla_\theta \log \pi_{\hat{\theta}}(g^t|s^t) \tag{7-6}$$

$$\nabla_{\theta'} L_{V_{\theta'}}(\pi_\theta) = 2 \sum_{t \in T} (V_{\hat{\theta}'} - z_{s^t}) \nabla_{\theta'} V_{\hat{\theta}'} \tag{7-7}$$

Once the optimization iterations are over, the policy neural network with the optimized parameters θ^* will be used to determine the traffic metering signal indications for each network state.

7.2. Case study

The case study network is shown in Figure 7-3 (a). The network is adapted from Downtown Springfield, IL, and is equipped with 13 metering gates at its border.



(a) Adapted network of Downtown Springfield, IL (b) Demand profiles

Figure 7-3 Case study network and demand profiles.

The network signal timings of intersections (inside the network) are optimized in Vistro (America, 2014) before the traffic metering application. The network is analyzed with two demand profiles that are shown in Figure 7-3 (b) for 75 minutes and the first 15 minutes were for network loading. In addition, the input layer of the policy and value neural networks was designed to have

the size of 316 nodes (the total number of defined sections) that feeds into a dense layer with the size of 400 nodes with Relu activation functions. The final layer of the policy neural network employs a softmax activation function to output the probability distribution over the action space. The learning rate of the subgradient approach was also set to 0.01.

7.2.1. Benchmarking scenarios

The performance of the proposed methodology is evaluated by considering the following scenarios for each of the demand profiles:

1. No-Metering: In this scenario, the case study network is evaluated without any traffic metering. The results of this scenario show the network performance under its current conditions.
2. CTM-Based Metering: The proposed methodology in (Mohebifard and Hajbabaie, 2019a) is used to optimize traffic metering signals. This approach collects the location of connected vehicles from the simulated network in Vissim at each time step and optimizes traffic metering signals using a mixed-integer non-linear program and a CTM network loading model.
3. Deep RL: The proposed deep reinforcement learning strategy that is proposed in the present study.

7.3. Numerical results

Different network performance measures calculated for each of the discussed scenarios are shown in Table 7-2. The Deep RL strategy increased the number of completed trips by 41.2% and 21.3% compared to the No-Metering scenario for demand profiles 1 and 2, respectively. The average delay of vehicles also decreased by 18.3% and 5.0%. Total travel time and the average number of stops were also improved using the Deep RL methodology. These results indicate that the proposed methodology improved the overall network performance. Comparing the Deep RL

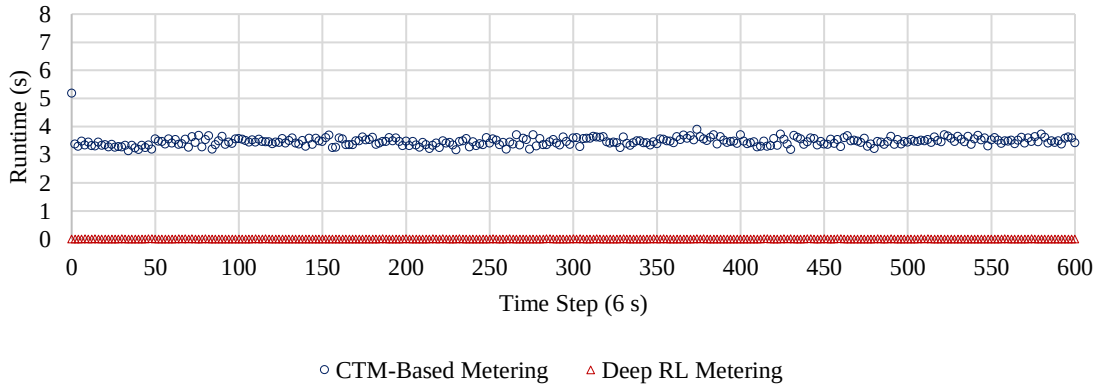
results with the CTM-Based Metering strategy, we observed that throughput, average delay, and total travel time of vehicles were less than 2.1% different from the CTM-based Metering. The latter comparison shows that the Deep RL methodology had comparable performance despite having lower computational complexity compared to the CTM-based methodology.

Table 7-2 Comparison of network performance measures for different scenarios

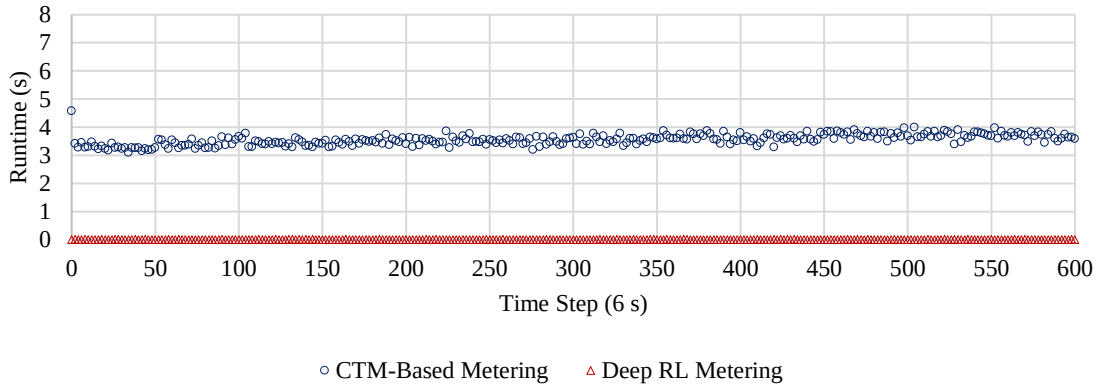
Performance Measure	No-Metering	CTM-Based Metering	Deep RL	% Diff. compared to No-Metering	% Diff. compared to CTM-Based Metering
Demand Profile 1					
Throughput	14,705	20,504	20,770	41.2	1.3
Average Delay (s)	768.7	634.9	628.1	-18.3	-1.1
Average Number of Stops	25.1	5.4	7.1	-71.6	32.8
Total Travel Time (hr)	6,259.9	5,402.5	5,289.0	-15.5	-2.1
Total Delay (hr)	5,984.3	4,932.9	4,889.8	-18.3	-0.9
Demand Profile 2					
Throughput	17,100	20,559	20,745	21.3	0.9
Average Delay (s)	749.6	709.2	712.2	-5.0	0.4
Average Number of Stops	36.2	5.6	8.9	-75.3	58.6
Total Travel Time (hr)	6,149.0	5,976.5	5,938.2	-3.4	-0.6
Total Delay (hr)	5,830.0	5,505.3	5,539.2	-5.0	0.6

The computational complexity of the Deep RL and CTM-based Metering can be better compared by looking at the runtime of each approach in Figure 7-4. The CTM-Based Metering required the CPLEX optimization solver (CPLEX, 2009) with 64 G of memory to optimize traffic metering levels with a total run-time of 2600 s. However, the total run-time of the Deep RL was

less than 1 s without the need for any commercial solver. As Figure 7-4 shows, the Deep RL runtimes at each time step were significantly lower than those in the CTM-Based metering scenario.



(a) Demand Profile 1

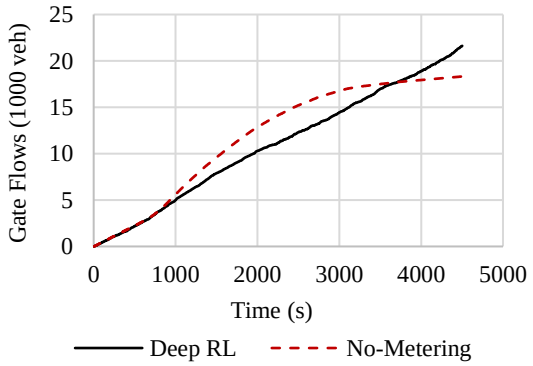


(b) Demand Profile 2

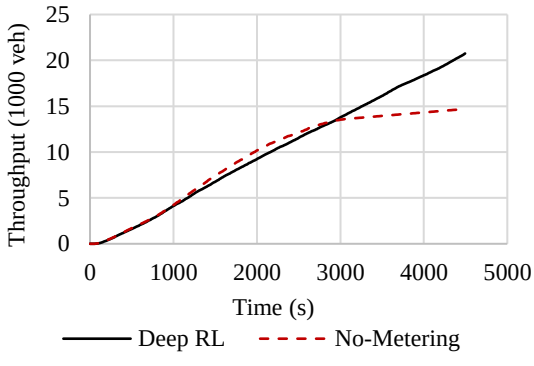
Figure 7-4 Runtimes of CTM-Based and Deep RL metering at each time step

The cumulative gate flows and network throughput over time are shown in Figure 7-5. In the Deep RL scenario, the cumulate gate flows were lower than the No-Metering strategy at simulation times 0 to 3700 and 0 to 4100 respectively for demand profiles 1 and 2 due to reducing the number of vehicles entering the network by traffic metering, see Figure 7-5 (a) and (c). However, queue spillback inside the network that reached the boundary gates reduced the gate flows significantly

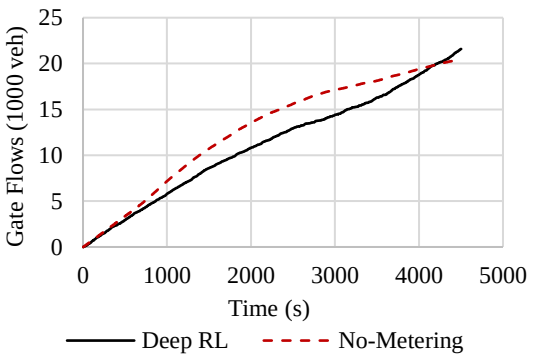
after time 3700 and 4100 in the demand profiles 1 and 2 in No-Metering, while the flows in the Deep RL scenario show an increasing linear trend. According to Figure 7-5 (b) and (d), the reductions in the gate flows also created a linear increasing cumulative network throughput in the Deep RL scenario, whereas the network throughput decreases in No-Metering.



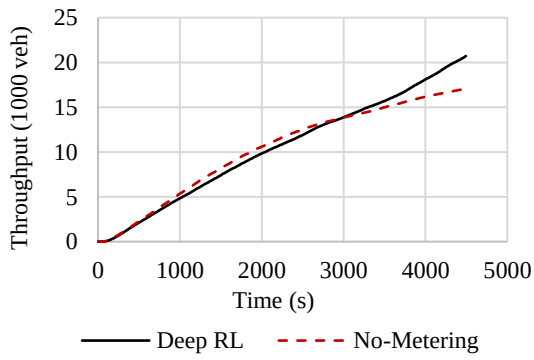
(a) Cumulative gate flows, Demand Profile 1



(b) Cumulative throughput, Demand Profile 1



(c) Cumulative gate flows, Demand Profile 2

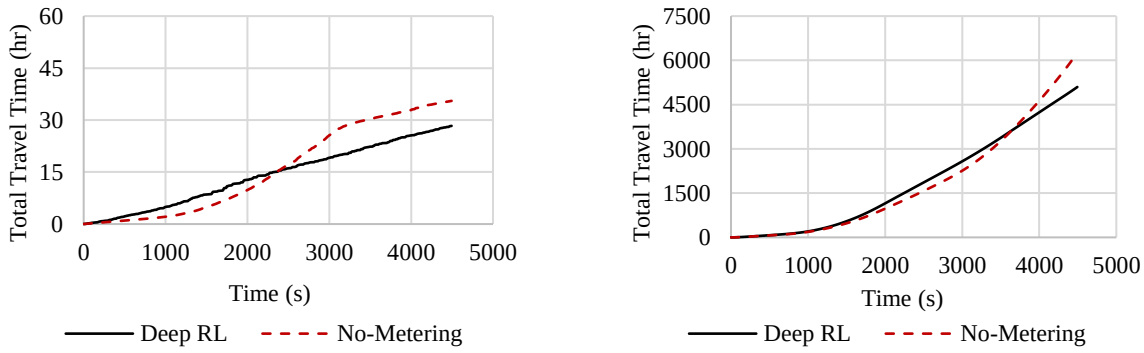


(d) Cumulative throughput, Demand Profile 2

Figure 7-5 Cumulative gate flows and network throughput for two demand profiles

Further analysis is performed on the travel time of vehicles at the boundary gates in Figure 7-6. Figure 7-6(a) shows that the travel time of vehicles at the boundary gates was higher in Deep RL compared to No-Metering from times 0 to 2280 seconds. However, the trends changed from times 2280 to the end of the analysis period. At the end of the analysis, total travel times were reduced by 20.3% at the gates. By comparing the total travel times in Figure 7-6(a) and (b), it can

be concluded that the total travel time of vehicles inside the network (excluding the boundary gates) was reduced by 15.4% for demand profile 1. This improvement is similar to the travel time improvements discussed in Table 7-2 since the number of vehicles inside the network and the time they spent inside the network was higher than the borders.

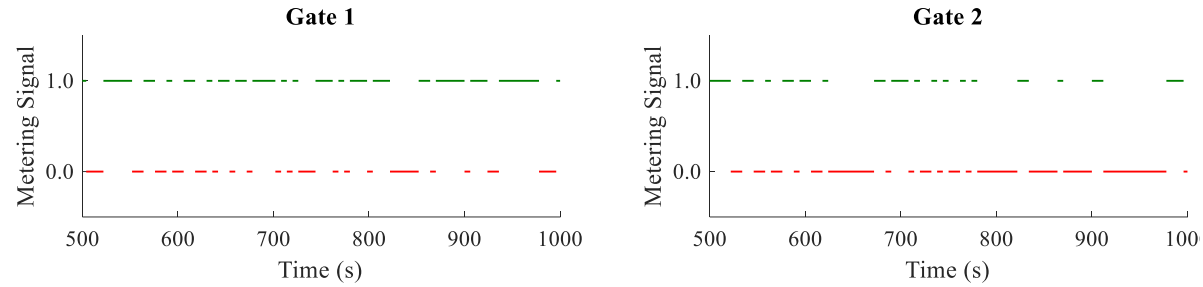


(a) Cumulative travel times at boundary gates

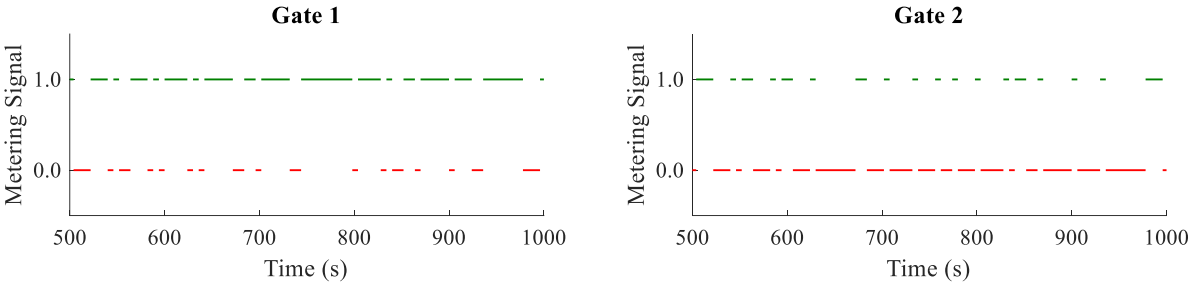
(b) Cumulative total travel times (both inside network and boundary gates)

Figure 7-6 Cumulative gate and total (both gate and inside the network) travel times for vehicles for Demand Profile 1

Figure 7-7 shows metering signal indications for two sample gate locations. In the figure, green indications are shown with level 1 and red signals with level 0. The figure shows that the signal indications changed dynamically over time and resulted in the discussed improvements in Table 7-2.



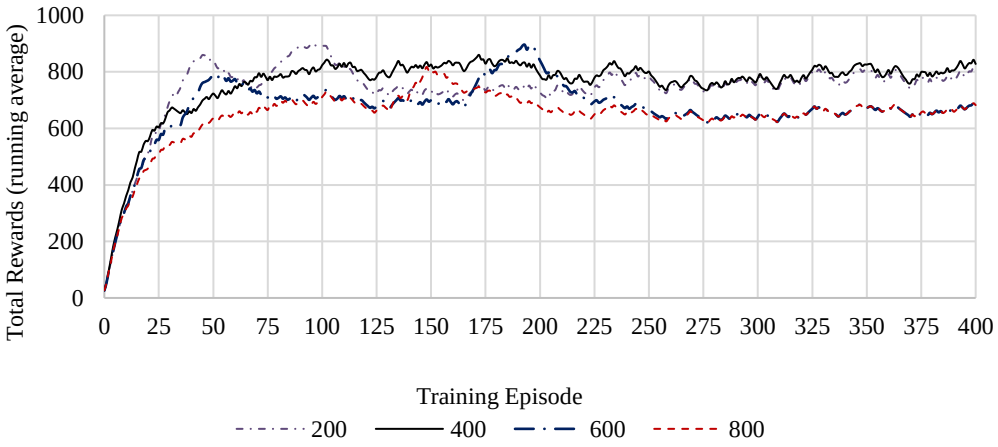
(a) Demand Profile 1



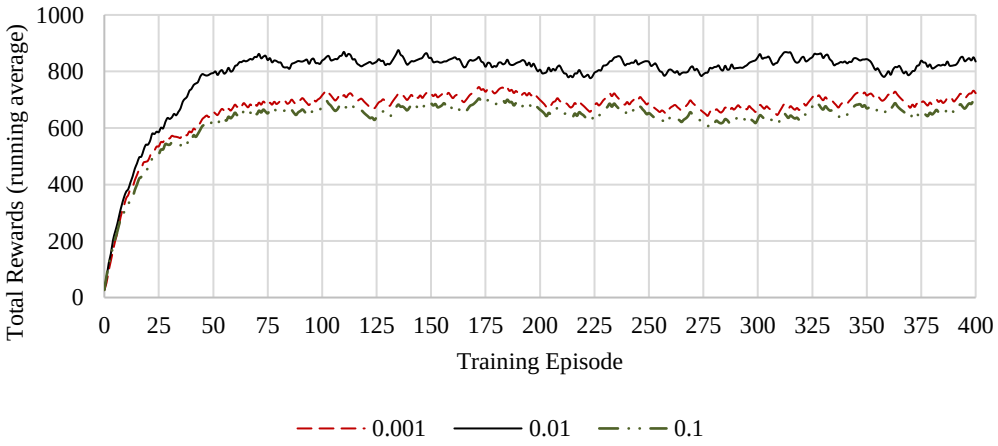
(b) Demand Profile 2

Figure 7-7 Gate signal indications for two sample gates.

We performed a sensitivity analysis on the number of hidden layers and learning rates in the observed training rewards. The results of Figure 7-8(a) show the running rewards for hidden layer sizes of 200, 400, 600, and 800. The results show that layer sizes of 400 and 200 performed relatively better in terms of final observed rewards. Since the hidden layer size of 400 had slightly better running rewards with lower fluctuations over the training episodes, the hidden layer size of 400 was used for the final model (please see Case Study section). Moreover, the effects of learning rates of 0.001, 0.01, and 0.1 on the training rewards are shown in Figure 7-8(b). According to the figure, the learning rate of 0.01 had superior performance compared to the other evaluated cases.



(a) Total observed rewards for different neural network hidden sizes

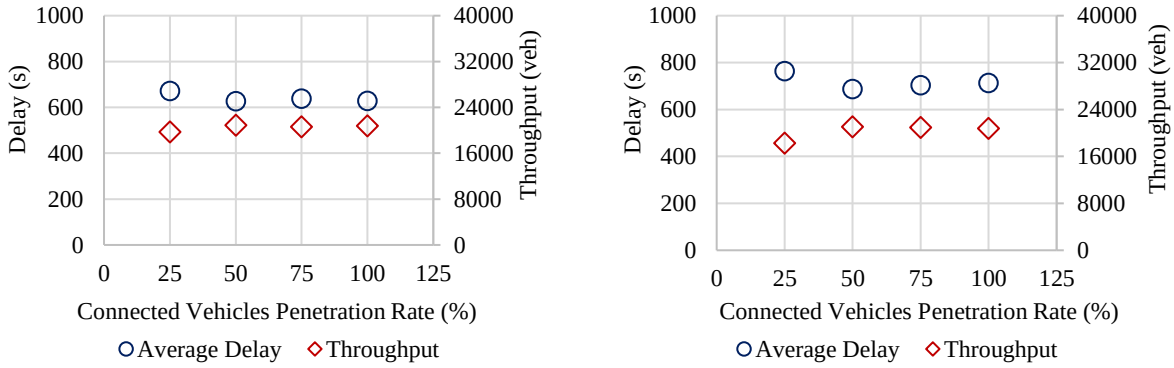


(b) Total observed rewards for different learning rates

Figure 7-8 Comparison of the total observed rewards for different neural network hidden sizes and learning rates

The discussed results were based on the assumption of a 100% connected vehicle penetration rate. In other words, the inputs of the Deep RL methodology were the number of vehicles in each link section over its capacity that was calculated from the location data of connected vehicles. We integrated an estimation algorithm (Mohebifard et al., 2019) into our analysis to test the sensitivity of the results to this assumption. The estimation algorithm received the stop bar loop detector and location data of connected vehicles and estimated the number of vehicles in each link section for

penetration rates less than 100%. Average delay and network throughput for connected vehicle penetration rates of 25, 50, 75, and 100% are shown for demand profiles 1 and 2 in Figure 7-9. Note that the results of the 100% penetration rate are similar to the results in Table 7-2. Figure 7-9 shows that the algorithm resulted in a relatively similar performance for penetration rates of 50, 75, and 100%. However, delays were higher, and throughputs were lower in the penetration rate of 25% than their corresponding values in penetration rates of 50, 75, and 100%. These results show that the developed methodology has the capability of improving network performance once the penetration rate of connected vehicles was at least 50%.



(a) Demand Profile 1

(b) Demand Profile 2

Figure 7-9 Average delay and throughputs for connected vehicles penetration rates 25, 50, 75, and 100%

In the developed methodology, we assumed that traffic metering signals are placed at locations that can accommodate the queue of vehicles. The analysis of the results showed that the maximum observed queue lengths were at most 50 veh/lane. However, traffic metering locations might have lower capacities for holding queues of vehicles. Therefore, we added the maximum queue length parameter to the Deep RL implementation. Once the maximum queue length condition was met, the traffic metering signal downstream of the target location turned green. Table 7-3 shows the results for maximum queue lengths of 40, 20, and 10 veh/lane. According to the results, reducing

the maximum queue length from 50 to 40 changed average delay, total travel time, and throughput by less than 1%. However, lower maximum queue lengths of 20 and 10 increased average delays by 16.4 and 25.3%, and reduced throughput by 23.4 and 29.19%, respectively. Therefore, the maximum queue length was a critical factor in the efficiency of the traffic metering methodology, and thus traffic metering signals should be placed in locations that have enough capacity to accommodate the queue of vehicles to achieve the maximum improvements in the network performance.

Table 7-3 Effects of queue length limit on the traffic metering solutions

Queue Length (veh/lane)	No Limit (50)	40	% Diff (compared to No Limit)	20	% Diff (compared to No Limit)	10	% Diff (compared to No Limit)
Average Delay (s)	712.2	709.42	-0.39	829.17	16.42	892.51	25.32
Total Travel Time (hr)	5,938.2	5,920.2	-0.30	6,749.4	13.66	7,220.3	21.59
Throughput (veh)	2,0745	2,0930	0.89	1,5891	-23.40	1,4696	-29.16

CHAPTER 8. CONCLUSIONS

8.1. Summary

This dissertation explores the traffic metering or perimeter control problem and develops several optimization formulations and solution techniques to solve this problem for urban street networks. The formulations aim at increasing the number of completed trips by controlling traffic metering signals.

In this dissertation, a mixed-integer non-linear optimization program is developed that maximized the number of completed trips by optimizing traffic metering rates. The program used a cell transmission model (CTM) to capture traffic flow dynamics. The non-linear flow conservation and feasibility constraints of the program made solving the program very challenging such that commercial optimization solvers could not solve the program for a network of 20 intersections. Therefore, the Benders decomposition technique was employed to utilize the special structure of the problem and proposed a solution technique that solved the program within an optimality bound. The objective of this solution technique was to provide the capability of solving the program to a certain optimality level regardless of its run-time efficiency. The results of applying this approach to a case study network of 20 intersections with 13 boundary gates using two demand profiles showed that traffic metering improved traffic operations by reducing the travel time of vehicles inside the protected area by gates by 30.8% and 34.2% compared to a no-metering strategy (the current state of the network before applying traffic metering). Some vehicles were delayed at the network gates due to traffic metering, but traffic metering reduced the system-level travel times including vehicles both inside and at the gates of the network by 2.7% and 5.4%. These results indicated that traffic metering improved traffic operations by regulating the flow of vehicles at the gates that were distributed at the boundary of the case study network. The solution

technique found these results after 190 and 310 iterations and reached an optimality gap of 1% for two different demand profiles. These solutions were found in 6.8 and 4.7 hours for the two cases. However, traffic operations require control strategies that can be implemented in real-time to be responsive to unforeseen changes in a network. Moreover, traffic signal timings of intersections inside a network play an important role in traffic metering decisions.

Moreover, a cooperative program for traffic signal and perimeter control with a distributed methodology is developed to find solutions in real-time. The cooperative problem was formulated as a mixed-integer nonlinear program that optimized traffic signal indications and metering levels simultaneously. Although the central program was able to find the optimal solutions, it could not be solved efficiently for real-time applications. Accordingly, a distributed optimization and coordination algorithm was proposed that decomposed the network-level problem into several sub-network-level sub-problems, and thus reduced the problem complexity significantly. Effective coordination between sub-problems was created to improve the quality of the solutions. The distributed coordination exchanged information among sub-problems and re-enforced the re-introduced relaxed constraints. Moreover, the solution technique had a model predictive control structure that estimated the network state at each time step, optimized control variables, and implemented them in the network before getting to the next time step. The methodology was applied to a CTM-represented network of 20 intersections with two demand profiles. The results showed that the methodology found solutions with at most a 3.6% optimality gap in the evaluated cases. The approach was also applied to a simulated network in Vissim and compared its solutions with several scenarios that either optimized signal timings or metering levels. The comparison results showed the benefits of the cooperative control by increasing the network throughput by 6.0% to 12.8% and decreasing the average delay by 10.4% to 27.6% compared to the results of

individual signal timing and traffic metering optimization. Moreover, the runtimes of solving the sub-network-level optimization programs indicated the capability of the methodology for being implemented in real-time (i.e., the optimization runtimes were less than 6 seconds which was the duration of each time step). In addition, the sensitivity of network performance measures to various connected vehicle market penetration rates was also analyzed. The results showed that increasing the penetration rate improved the performance in terms of total travel time, average delay, average speed, and network throughput. The rate of performance improvement from penetration rates of 0 to 60% was higher than penetration rates of more than 60%.

While the locations of gates were assumed to be predetermined in the previous methodologies, an integrated formulation is developed that optimized static metering locations and dynamic metering levels within a limited budget. The integrated approach detected congested areas in a network and protected them by traffic metering. The formulation optimized the metering locations and their corresponding metering levels by considering the installation and operational costs of metering signals as a hard constraint on the available budget in the optimization program. The non-linear flow conservation principles of CTM along with binary metering location and level decision variables made the problem a mixed-integer non-linear program (MINLP) and hence very complex to solve. Therefore, we proposed a solution technique that divided the feasible region of the problem into several sub-regions based on metering locations, constructs two easier MINLPs, decomposed them into primal and master problems, and solved them iteratively. We implemented the solution technique on a case study network of 49 intersections under two demand profiles. The network performance analysis showed that the proposed methodology resulted in better or similar network performance but with 23.08% to 53.85% lower costs compared to a perimeter control scenario. The sensitivity analysis of the number of metering locations showed that the proposed

approach could effectively capture the tradeoffs between network performance and operational costs of traffic metering. Moreover, the run-time comparison of the solution technique with CPLEX showed that CPLEX could not find any solutions for the problem within 24 hours of run-time, whereas the proposed solution technique found the optimal solutions between 0.78 to 3.86 hours in the evaluated cases.

Furthermore, a methodology is proposed to optimize traffic metering signal indications using a deep reinforcement learning (Deep RL) methodology. The methodology addressed the limitations of the existing studies by employing a more detailed traffic flow model to capture queue formation and congestion propagation more accurately and reduce the required computational resources for online implementations. The proposed methodology used gradient descent optimization and Monte-Carlo simulation to train a policy and a value neural network with a series of iterative experiments on a simulated network in Vissim. The developed methodology was applied to a case study network of 20 intersections under two different demand profiles. The results showed that the methodology increased the number of completed trips by 41.2 to 21.3% and decreased the total travel time of vehicles by 3.4 to 15.5% compared to a no-metering strategy. Besides, comparing the results with one of the existing CTM-based approaches showed that the Deep RL methodology was more run-time efficient with relatively similar improvements in the network performance measures. A sensitivity analysis of the size of the hidden layer and the learning rate of the gradient descent approach showed that the hidden layer size of 400 and learning rate of 0.01 had better performance compared to the layer sizes of 200, 600, 800, and rates of 0.001 and 0.1. The integration of a link density estimation algorithm into the Deep RL methodology also showed that similar network performance could be achieved with connected vehicle penetration rates of more than 50% compared to the case of a 100% connected vehicle penetration rate. Moreover, the

addition of the maximum queue length parameter to the Deep RL implementation showed limiting the maximum queue length from 50 to 40 veh/lane resulted in network performance measures that were less than 1% different. However, reducing the maximum queue lengths further increased travel times and decreased network throughput.

Table 8-1 summarizes the methodologies and case studies that are used in each chapter of this dissertation.

Table 8-1 Summary of methodologies and used case studies documents in the dissertation

Information	Values				
Chapter number	3	4	5	6	7
Traffic metering optimization at the network boundary	Yes	Yes	Yes	Yes	Yes
Traffic metering optimization inside a network	No	No	No	Yes	No
Traffic signal timing optimization	No	No	Yes	No	No
Number of boundary gates of the case study	-	13	13	19	13
Number of intersections of the case study	-	20	20	49	20
Number of demand profiles	-	2	2	2	2
Study period (min)	-	75	60-180	50	75

8.2. Limitations and future research suggestions

The developed traffic metering methodologies in this dissertation are network-level strategies that might change travel and origin-destination demand patterns over time. Before any application of traffic metering to an urban network, the economic and social effects of these changes should be further evaluated.

Traffic metering delays vehicles at traffic metering locations to improve traffic operations in other parts of a network. This application might lead to long vehicular queues and increased emission levels at the traffic metering locations. The methodologies in the present dissertation do not provide any recommendations or guidelines to mitigate these side effects.

In the methodologies of Chapters 5 and 7, the location data of connected vehicles were used to estimate vehicular densities across the links of a transportation network. In these methodologies, the effects of communication latency, loss of information, and data collection frequency were not considered. Therefore, the sensitivity of the models to these communication-related issues should be further evaluated.

In addition, the developed integrated signal and metering formulation and solution technique utilize a cycle-free traffic signal control in the optimization programs. However, traffic signal timings with fixed cycle lengths are more desirable for practical implementations. The problem formulation can be modified to consider fixed cycle lengths, and the optimization horizon of each sub-network should be selected long enough to include several cycles.

Below are a few suggestions that can be investigated in future studies:

1. One interesting topic for future research is the development of a traffic metering guideline that can be used to determine in which urban areas traffic metering is effective and when traffic metering control can be started (in terms of the time of the day).

2. Connected and automated vehicles can provide the capability of regulating traffic flow throughput urban areas without the need to have traffic metering signals. Therefore, the control of connected and automated vehicles as moving bottlenecks and studying their effects on regulating traffic flow can be investigated in the future. This analysis can be done for fully automated vehicles ((Mirheli et al., 2019, 2018; Mohebifard and Hajbabaie, 2021a) and different automation

(Mohebifard and Hajbabaie, 2021b; Niroumand et al., 2022; Tajalli et al., 2022; Tajalli and Hajbabaie, 2021) and cooperation levels (Mohebifard and Hajbabaie, 2020; Niroumand et al., 2021).

3. In the methodologies documented in the dissertation, the effects of route guidance on traffic metering were not considered. Therefore, simultaneous route guidance (Mehrabipour et al., 2019a, 2019b; Mehrabipour and Hajbabaie, 2022) and traffic metering methodology can be developed to further improve traffic operations.

4. The signal timings used in the cooperative traffic signal and metering control can be improved in the presence of connected and automated vehicles by considering different phasing strategies such as the addition of white phase (Niroumand et al., 2020b, 2020a). The addition of speed harmonization is another consideration that can improve the performance of traffic signals (Tajalli et al., 2020, 2019; Tajalli and Hajbabaie, 2018b).

5. One promising solution to implement traffic metering for large-scale transportation networks is to integrate macroscopic fundamental (MFD) and cell transmission model (CTM) control techniques. An MFD-based controller can determine the aggregated flow of vehicles that needs to be moved between different regions of a transportation network, and a CTM-based traffic metering methodology optimizes traffic metering rates for each gate to achieve the optimized flows.

6. The deep reinforcement learning methodology in this dissertation is a central controller, while distributed reinforcement learning techniques can provide the potential to develop a methodology that can be applied to other transportation networks than those used for training the model. Therefore, the transferability of a trained model can be further enhanced.

7. The developed traffic metering optimization models are computationally complex since they rely on standard optimization solution techniques such as the branch-and-bound technique in their

core solving operations. Therefore, one of the promising directions for reducing the complexity of this technique is to use reinforcement learning and reduce the runtime of solving the optimization models (Parsonson et al., 2022).

CHAPTER 9. REFERENCES

Abdoos, M., Mozayani, N., Bazzan, A.L.C., 2014. Hierarchical control of traffic signals using Q-learning with tile coding. Appl. Intell. 40, 201–213.

Aboudolas, K., Geroliminis, N., 2013. Perimeter and boundary flow control in multi-reservoir heterogeneous networks. Transp. Res. Part B Methodol. 55, 265–281.

Ahuja, R.K., Magnanti, T.L., Orlin, J.B., 1993. Network flows: theory, algorithms, and applications.

America, P.T. V, 2014. PTV VISTRO User Manual. Ptv Ag 2, 7.

Arel, I., Liu, C., Urbanik, T., Kohls, A.G., 2010. Reinforcement learning-based multi-agent system for network traffic signal control. IET Intell. Transp. Syst. 4, 128–135.

Aslani, M., Seipel, S., Mesgari, M.S., Wiering, M., 2018. Traffic signal optimization through discrete and continuous reinforcement learning with robustness analysis in downtown Tehran. Adv. Eng. Informatics 38, 639–655.

Atkins, 2013. Ramp Metering Feasibility Study for Durham and Wake Counties.

Bakker, B., Whiteson, S., Kester, L., Groen, F.C.A., 2010. Traffic light control by multiagent reinforcement learning systems, in: Interactive Collaborative Information Systems. Springer, pp. 475–510.

Beard, C., Ziliaskopoulos, A., 2006. System Optimal Signal Optimization Formulation. Transp. Res. Rec. J. Transp. Res. Board 1978, 102–112. https://doi.org/10.3141/1978-15

Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems. Numer. Math. 4, 238–252.

Bianchessi, N., Irnich, S., 2019. Branch-and-cut for the split delivery vehicle routing problem with time windows. Transp. Sci. 53, 442–462.

Casas, N., 2017. Deep deterministic policy gradient for urban traffic light control. arXiv Prepr. arXiv1703.09035.

Chen, C., Wei, H., Xu, N., Zheng, G., Yang, M., Xiong, Y., Xu, K., Li, Z., 2020. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control, in: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 3414–3421.

CPLEX, I.B.M.I., 2009. V12. 1: User's Manual for CPLEX. Int. Bus. Mach. Corp. 46, 157.

Daganzo, C.F., 2007. Urban gridlock: Macroscopic modeling and mitigation approaches. Transp. Res. Part B Methodol. 41, 49–62.

Daganzo, C.F., 1995. The cell transmission model, part II: Network traffic. Transp. Res. Part B Methodol. 29, 79–93.

Daganzo, C.F., 1994. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. Transp. Res. Part B Methodol. 28, 269–287.

De Gier, J., Garoni, T.M., Rojas, O., 2011. Traffic flow on realistic road networks with adaptive traffic lights. J. Stat. Mech. Theory Exp. 2011, P04008.

Dell'Amico, M., Righini, G., Salani, M., 2006. A branch-and-price approach to the vehicle routing problem with simultaneous distribution and collection. Transp. Sci. 40, 235–247.

der Pol, E., Oliehoek, F.A., 2016. Coordinated deep reinforcement learners for traffic light control. Proc. Learn. Inference Control Multi-Agent Syst. (at NIPS 2016).

Ding, H., Guo, F., Zheng, X., Zhang, W., 2017. Traffic guidance--perimeter control coupled method for the congestion in a macro network. Transp. Res. Part C Emerg. Technol. 81, 300–316.

Fu, H., Liu, N., Hu, G., 2017. Hierarchical perimeter control with guaranteed stability for dynamically coupled heterogeneous urban traffic. Transp. Res. Part C Emerg. Technol. 83,

18–38.

Garg, D., Chli, M., Vogiatzis, G., 2018. Deep reinforcement learning for autonomous traffic light control, in: 2018 3rd Ieee International Conference on Intelligent Transportation Engineering (Icite). pp. 214–218.

Ge, H., Song, Y., Wu, C., Ren, J., Tan, G., 2019. Cooperative deep Q-learning with Q-value transfer for multi-intersection signal control. IEEE Access 7, 40797–40809.

Genders, W., Razavi, S., 2016. Using a deep reinforcement learning agent for traffic signal control. arXiv Prepr. arXiv1611.01142.

Geoffrion, A.M., 1972. Generalized Benders decomposition. J. Optim. Theory Appl. 10, 237–260.

Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. Transp. Res. Part B Methodol. 42, 759–770.

Geroliminis, N., Haddad, J., Ramezani, M., 2013. Optimal Perimeter Control for Two Urban Regions With Macroscopic Fundamental Diagrams: A Model Predictive Approach. IEEE Trans. Intell. Transp. Syst. 14, 348–359. https://doi.org/10.1109/TITS.2012.2216877

Geroliminis, N., Sun, J., 2011. Properties of a well-defined macroscopic fundamental diagram for urban traffic. Transp. Res. Part B Methodol. 45, 605–617.

Godfrey, J.W., 1969. The mechanism of a road network. Traffic Eng. Control 8.

Gomes, G., Horowitz, R., 2006. Optimal freeway ramp metering using the asymmetric cell transmission model. Transp. Res. Part C Emerg. Technol. 14, 244–262. https://doi.org/10.1016/j.trc.2006.08.001

Gong, Y., Abdel-Aty, M., Cai, Q., Rahman, M.S., 2019. Decentralized network level adaptive signal control by multi-agent deep reinforcement learning. Transp. Res. Interdiscip. Perspect. 1, 100020.

Goodall, N., Smith, B., Park, B., 2013. Traffic signal control with connected vehicles. Transp. Res. Rec. J. Transp. Res. Board 65–72.

Haarnoja, T., Zhou, A., Abbeel, P., Levine, S., 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, in: International Conference on Machine Learning. pp. 1861–1870.

Haddad, J., 2017. Optimal perimeter control synthesis for two urban regions with aggregate boundary queue dynamics. Transp. Res. Part B Methodol. 96, 1–25.

Haddad, J., Geroliminis, N., 2012. On the stability of traffic perimeter control in two-region urban cities. Transp. Res. Part B Methodol. 46, 1159–1176.

Haddad, J., Mirkin, B., 2017. Coordinated distributed adaptive perimeter control for large-scale urban road networks. Transp. Res. Part C Emerg. Technol. 77, 495–515.

Haddad, J., Ramezani, M., Geroliminis, N., 2013. Cooperative traffic control of a mixed network with two urban regions and a freeway. Transp. Res. Part B Methodol. 54, 17–36.

Haddad, J., Shraiber, A., 2014. Robust perimeter control design for an urban region. Transp. Res. Part B Methodol. 68, 315–332.

Hajbabaie, A., 2012. Intelligent dynamic signal timing optimization program. University of Illinois at Urbana-Champaign.

Hajbabaie, A., Benekohal, R., 2013. Traffic Signal Timing Optimization: Choosing the Objective Function. Transp. Res. Rec. J. Transp. Res. Board 10–19.

Hajbabaie, A., Benekohal, R.F., 2015. A program for simultaneous network signal timing optimization and traffic assignment. IEEE Trans. Intell. Transp. Syst. 16, 2573–2586.

Hajbabaie, A., Benekohal, R.F., 2011. Does traffic metering improve network performance efficiency?, in: 2011 14th International IEEE Conference on Intelligent Transportation

Systems (ITSC). IEEE, pp. 1114–1119.

Hajbabaie, A., Medina, J.C., Benekohal, R.F., 2010. Effects of ITS-based left turn policies on network performance, in: 13th International IEEE Conference on Intelligent Transportation Systems. IEEE, pp. 80–84.

Hajbabaie, A., Medina, J.C., Benekohal, R.F., NEXTRANS, C., 2011. Traffic signal coordination and queue management in oversaturated intersections. Purdue Univ. Discov. Park.

Hajbabaie, A., Mohebifard, R., Consortium, P.N.T., others, 2020. Dynamic Metering in Connected Urban Street Networks: Improving Mobility.

Hajibabai, L., Nourbakhsh, S., Ouyang, Y., Peng, F., 2014. Network Routing of Snowplow Trucks with Resource Replenishment and Plowing Priorities: Formulation, Algorithm, and Application. Transp. Res. Rec. J. Transp. Res. Board 16–25.

Hajibabai, L., Ouyang, Y., 2013. Integrated planning of supply chain networks and multimodal transportation infrastructure expansion: model development and application to the biofuel industry. Comput. Civ. Infrastruct. Eng. 28, 247–259.

Hajibabai, L., Saha, D., n.d. Patrol Route Planning for Incident Response Vehicles under Dispatching Station Scenarios. Comput. Civ. Infrastruct. Eng.

He, Q., Irnich, S., Song, Y., 2019. Branch-and-Cut-and-Price for the Vehicle Routing Problem with Time Windows and Convex Node Costs. Transp. Sci.

Herman, R., Prigogine, I., 1979. A two-fluid approach to town traffic. Science (80-.). 204, 148–151.

Huo, Y., Tao, Q., Hu, J., 2020. Cooperative Control for Multi-Intersection Traffic Signal Based on Deep Reinforcement Learning and Imitation Learning. Ieee Access 8, 199573–199585.

Islam, S. M.A.Bin Al, Aziz, H.M.A., Hajbabaie, A., 2021. Stochastic gradient-based optimal

signal control with energy consumption bounds. IEEE Trans. Intell. Transp. Syst. 22, 3054–3067. https://doi.org/10.1109/TITS.2020.2979384

Islam, S.M.A.B. Al, Hajbabaie, A., 2021. An Enhanced Cell Transmission Model for Multi-Class Signal Control. IEEE Trans. Intell. Transp. Syst. 1–12. https://doi.org/10.1109/tits.2021.3101838

Islam, S.M.A.B. Al, Hajbabaie, A., 2017. Distributed coordinated signal timing optimization in connected transportation networks. Transp. Res. Part C Emerg. Technol. 80, 272–285. https://doi.org/10.1016/j.trc.2017.04.017

Islam, S.M.A.B. Al, Hajbabaie, A., Aziz, H.M.A., 2020. A real-time network-level traffic signal control methodology with partial connected vehicle information. Transp. Res. Part C Emerg. Technol. 121, 102830. https://doi.org/10.1016/j.trc.2020.102830

Islam, S M A Bin Al, Tajalli, M., Mohebifard, R., Hajbabaie, A., 2021. Effects of connectivity and traffic observability on an adaptive traffic signal control system. Transp. Res. Rec. 2675, 800–814.

Ji, Y., Daamen, W., Hoogendoorn, S., Hoogendoorn-Lanser, S., Qian, X., 2010. Investigating the shape of the macroscopic fundamental diagram using simulation data. Transp. Res. Rec. J. Transp. Res. Board 40–48.

Ji, Y., Geroliminis, N., 2012. On the spatial partitioning of urban transportation networks. Transp. Res. Part B Methodol. 46, 1639–1656.

Kan, X., Lu, X.-Y., Skabardonis, A., 2017. Coordination of freeway ramp metering and arterial traffic signals. Transportation Research Board, National Research Council, National Academy of Sciences USA, Washington D.C.

Keyvan-Ekbatani, M., Carlson, R.C., Knoop, V.L., Hoogendoorn, S.P., Papageorgiou, M., 2016.

Queuing under perimeter control: Analysis and control strategy, in: 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC). pp. 1502–1507.

Keyvan-Ekbatani, M., Kouvelas, A., Papamichail, I., Papageorgiou, M., 2012. Exploiting the fundamental diagram of urban networks for feedback-based gating. Transp. Res. Part B Methodol. 46, 1393–1403.

Keyvan-Ekbatani, M., Papageorgiou, M., Knoop, V.L., 2015a. Controller design for gating traffic control in presence of time-delay in urban road networks. Transp. Res. Part C Emerg. Technol. 59, 308–322.

Keyvan-Ekbatani, M., Yildirimoglu, M., Geroliminis, N., Papageorgiou, M., 2015b. Multiple concentric gating traffic control in large-scale urban networks. IEEE Trans. Intell. Transp. Syst. 16, 2141–2154.

Keyvan Ekbatani, M., Gao, X., Gayah, V. V, Knoop, V.L., 2016. Combination of traffic-responsive and gating control in urban networks: Effective interactions, in: Proceedings of the 95th Annual Meeting of the Transportation Research Board, Washington (USA), 10-14 Jan. 2016; Authors Version.

Kok, J.R., Vlassis, N., 2005. Using the max-plus algorithm for multiagent decision making in coordination graphs, in: Robot Soccer World Cup. pp. 1–12.

Kouvelas, A., Saeedmanesh, M., Geroliminis, N., 2017. Enhancing model-based feedback perimeter control with data-driven online adaptive optimization. Transp. Res. Part B Methodol. 96, 26–45.

Kouvelas, A., Triantafyllos, D., Geroliminis, N., 2018. Hierarchical Control Design for Large-Scale Urban Road Traffic Networks, in: Transportation Research Board 97th Annual Meeting,.

Kwon, E., Ambadipudi, R.-P., Bieniek, J., 2003. Adaptive coordination of ramp meter and intersection signal for optimal management of freeway corridor, in: Proc. 82th Annu. Meeting Transp. Res. Board.

Lange, S., Riedmiller, M., 2010. Deep auto-encoder neural networks in reinforcement learning, in: The 2010 International Joint Conference on Neural Networks (IJCNN). pp. 1–8.

Laval, J., 2010. The effect of signal timing and network irregularities in the macroscopic fundamental diagram, in: Proceedings of the Summer Meeting of the TFTC TRB Committee. pp. 7–9.

Lawler, E.L., Wood, D.E., 1966. Branch-and-bound methods: A survey. Oper. Res. 14, 699–719.

Levy, J.I., Buonocore, J.J., Von Stackelberg, K., 2010. Evaluation of the public health impacts of traffic congestion: a health risk assessment. Environ. Heal. 9, 65.

Li, L., Lv, Y., Wang, F.-Y., 2016. Traffic signal timing via deep reinforcement learning. IEEE/CAA J. Autom. Sin. 3, 247–254.

Liang, X., Du, X., Wang, G., Han, Z., 2019. A deep reinforcement learning network for traffic light cycle control. IEEE Trans. Veh. Technol. 68, 1243–1253.

Lighthill, M.J., Whitham, G.B., 1955. On kinematic waves. II. A theory of traffic flow on long crowded roads. Proc. R. Soc. A Math. Phys. Eng. Sci. 229, 317–345.

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D., 2015. Continuous control with deep reinforcement learning. arXiv Prepr. arXiv1509.02971.

Lo, H.K., 1999. A novel traffic signal control formulation. Transp. Res. Part A Policy Pract. 33, 433–448.

Lu, X.-Y., Su, D., Spring, J., 2013. Coordination of Freeway Ramp Meters and Arterial Traffic Signals Field Operational Test (FOT). CONTRACT 65, 3763.

Mahmassani, H.S., Williams, J.C., Herman, R., 1984. Investigation of network-level traffic flow relationships: some simulation results. Transp. Res. Rec. 971, 121–130.

Mazloumian, A., Geroliminis, N., Helbing, D., 2010. The spatial variability of vehicle densities as determinant of urban network capacity. Philos. Trans. R. Soc. London A Math. Phys. Eng. Sci. 368, 4627–4647.

Medina, J.C., Hajbabaie, A., Benekohal, R. (Ray) F., 2013. Effects of Metered Entry Volume on an Oversaturated Network with Dynamic Signal Timing. Transp. Res. Rec. J. Transp. Res. Board 2366, 53–60. https://doi.org/10.3141/2356-07

Medina, J.C., Hajbabaie, A., Benekohal, R.F., 2010. Arterial traffic control using reinforcement learning agents and information from adjacent intersections in the state and reward structure, in: 13th International IEEE Conference on Intelligent Transportation Systems. pp. 525–530.

Mehrabipour, M., 2018. Real-time Network-level Signal Timing Optimization. Washington State University.

Mehrabipour, M., Hajbabaie, A., 2022. A Distributed Gradient Approach for System Optimal Dynamic Traffic Assignment. IEEE Trans. Intell. Transp. Syst.

Mehrabipour, M., Hajbabaie, A., 2017. A Cell-Based Distributed-Coordinated Approach for Network-Level Signal Timing Optimization. Comput. Civ. Infrastruct. Eng. 32, 599–616. https://doi.org/10.1111/mice.12272

Mehrabipour, M., Hajibabai, L., Hajbabaie, A., 2019a. A decomposition scheme for parallelization of system optimal dynamic traffic assignment on urban networks with multiple origins and destinations. Comput. Civ. Infrastruct. Eng. 34, 915–931. https://doi.org/10.1111/mice.12455

Mehrabipour, M., Hajibabai, L., Hajbabaie, A., 2019b. A Decomposition Algorithm for System Optimal Dynamic Traffic Assignment on Urban Networks, in: Transportation Research

Board 98th Annual Meeting of the Transportation Research Board.

Mirheli, A., Hajibabai, L., Hajbabaie, A., 2018. Development of a signal-head-free intersection control logic in a fully connected and autonomous vehicle environment. Transp. Res. Part C Emerg. Technol. 92, 412–425. https://doi.org/10.1016/j.trc.2018.04.026

Mirheli, A., Tajalli, M., Hajibabai, L., Hajbabaie, A., 2019. A consensus-based distributed trajectory control in a signal-free intersection. Transp. Res. Part C Emerg. Technol. 100, 161–176. https://doi.org/10.1016/j.trc.2019.01.004

Mohebifard, R., Hajbabaie, A., 2021a. Connected automated vehicle control in single lane roundabouts. Transp. Res. Part C Emerg. Technol. 131, 103308.

Mohebifard, R., Hajbabaie, A., 2021b. Trajectory Control in Roundabouts with a Mixed-fleet of Automated and Human-driven Vehicles. Comput. Civ. Infrastruct. Eng.

Mohebifard, R., Hajbabaie, A., 2020. Effects of Automated Vehicles on Traffic Operations at Roundabouts, in: 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC). pp. 1–6.

Mohebifard, R., Hajbabaie, A., 2019a. Distributed Optimization and Coordination Algorithms for Dynamic Traffic Metering in Urban Street Networks. IEEE Trans. Intell. Transp. Syst. 20, 1930–1941. https://doi.org/10.1109/TITS.2018.2848246

Mohebifard, R., Hajbabaie, A., 2019b. Optimal network-level traffic signal control: A benders decomposition-based solution algorithm. Transp. Res. Part B Methodol. 121, 252–274. https://doi.org/10.1016/j.trb.2019.01.012

Mohebifard, R., Hajbabaie, A., 2018a. Real-Time Adaptive Traffic Metering in a Connected Urban Street Network, in: Transportation Research Board 97th Annual Meeting.

Mohebifard, R., Hajbabaie, A., 2018b. Dynamic traffic metering in urban street networks:

Formulation and solution algorithm. Transp. Res. Part C Emerg. Technol. 93, 161–178. https://doi.org/10.1016/j.trc.2018.04.027

Mohebifard, R., Islam, S.M.A.B. Al, Hajbabaie, A., 2019. Cooperative Traffic Signal and Perimeter Control in Semi-Connected Urban- Street Networks. Transp. Res. Part C Emerg. Technol. 104, 408–427. https://doi.org/10.1016/j.trc.2019.05.023

Mousavi, S.S., Schukat, M., Howley, E., 2017. Traffic light control using deep policy-gradient and value-function-based reinforcement learning. IET Intell. Transp. Syst. 11, 417–423.

Niroumand, R., Hajibabai, L., Hajbabaie, A., 2021. The Effects of Connectivity on Intersection Operations with "White Phase," in: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC). pp. 3839–3844.

Niroumand, R., Hajibabai, L., Hajbabaie, A., 2020a. The Effects of the "White Phase" on Intersection Performance with Mixed-Autonomy Traffic Stream, in: The IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC).

Niroumand, R., Hajibabai, L., Hajbabaie, A., Tajalli, M., 2022. Effects of Autonomous Driving Behavior on Intersection Performance and Safety in the Presence of White Phase for Mixed-Autonomy Traffic Stream. Transp. Res. Rec. 03611981221082580.

Niroumand, R., Tajalli, M., Hajibabai, L., Hajbabaie, A., 2020b. Joint Optimization of Vehicle-group Trajectory and Signal Timing: Introducing the White Phase for Mixed-autonomy Traffic Stream. Transp. Res. Part C Emerg. Technol.

Pandey, V., Wang, E., Boyles, S.D., 2020. Deep reinforcement learning algorithm for dynamic pricing of express lanes with multiple access locations. Transp. Res. Part C Emerg. Technol. 119, 102715.

Parsonson, C.W.F., Laterre, A., Barrett, T.D., 2022. Reinforcement Learning for Branch-and-

Bound Optimisation using Retrospective Trajectories. arXiv Prepr. arXiv2205.14345.

Priemer, C., Friedrich, B., 2009. A decentralized adaptive traffic signal control using V2I communication data, in: 2009 12th International IEEE Conference on Intelligent Transportation Systems. IEEE, pp. 1–6. https://doi.org/10.1109/ITSC.2009.5309870

PTV Group, 2013. PTV Vissim 7 User Manual. PTV AG.

Ramezani, M., Haddad, J., Geroliminis, N., 2015. Dynamics of heterogeneity in urban networks: aggregated traffic modeling and hierarchical control. Transp. Res. Part B Methodol. 74, 1–19.

Rathi, A.K., Lieberman, E.B., 1989. Effectiveness of traffic restraint for a congested urban network: A simulation study.

Richards, P.I., 1956. Shock waves on the highway. Oper. Res.

Sirmatel, I.I., Geroliminis, N., 2017. Economic Model Predictive Control of Large-Scale Urban Road Networks via Perimeter Control and Regional Route Guidance. IEEE Trans. Intell. Transp. Syst.

Su, D., Lu, X.-Y., Horowitz, R., Wang, Z., 2014. Coordinated ramp metering and intersection signal control. Int. J. Transp. Sci. Technol. 3, 179–192.

Sutton, R.S., 1995. Generalization in reinforcement learning: Successful examples using sparse coarse coding. Adv. Neural Inf. Process. Syst. 8.

Tajalli, M., Hajbabaie, A., 2021. Traffic Signal Timing and Trajectory Optimization in a Mixed Autonomy Traffic. IEEE Trans. Intell. Transp. Syst. 1–14. https://doi.org/10.1109/TITS.2021.3058193

Tajalli, M., Hajbabaie, A., 2018a. Distributed optimization and coordination algorithms for dynamic speed optimization of connected and autonomous vehicles in urban street networks.

Transp. Res. Part C Emerg. Technol. 95, 497–515. https://doi.org/10.1016/j.trc.2018.07.012

Tajalli, M., Hajbabaie, A., 2018b. Dynamic Speed Harmonization in Connected Urban Street Networks. Comput. Civ. Infrastruct. Eng. 33, 510–523. https://doi.org/10.1111/mice.12360

Tajalli, M., Mehrabipour, M., Hajbabaie, A., 2020. Network-Level Coordinated Speed Optimization and Traffic Light Control for Connected and Automated Vehicles. IEEE Trans. Intell. Transp. Syst. 1–12. https://doi.org/10.1109/TITS.2020.2994468

Tajalli, M., Mehrabipour, M., Hajbabaie, A., 2019. Cooperative Signal Timing and Speed Optimization in Connected Urban-Street Networks, in: Transportation Research Board 98th Annual Meeting of the Transportation Research Board.

Tajalli, M., Niroumand, R., Hajbabaie, A., 2022. Distributed Cooperative Trajectory and Lane changing Optimization of Connected Automated Vehicles: Freeway Segments with Lane Drop. Transp. Res. Part C.

Tan, T., Bao, F., Deng, Y., Jin, A., Dai, Q., Wang, J., 2019. Cooperative deep reinforcement learning for large-scale traffic grid signal control. IEEE Trans. Cybern. 50, 2687–2700.

Timotheou, S., Panayiotou, C.G., Polycarpou, M.M., 2015. Distributed traffic signal control using the cell transmission model via the alternating direction method of multipliers. IEEE Trans. Intell. Transp. Syst. 16, 919–933.

Varaiya, P., 2013. Max pressure control of a network of signalized intersections. Transp. Res. Part C Emerg. Technol. 36, 177–195.

Yan, Z., Xu, Y., 2020. A multi-agent deep reinforcement learning method for cooperative load frequency control of a multi-area power system. IEEE Trans. Power Syst. 35, 4599–4608.

Yang, K., Menendez, M., Zheng, N., 2019. Heterogeneity aware urban traffic control in a connected vehicle environment: A joint framework for congestion pricing and perimeter

control. Transp. Res. Part C Emerg. Technol. 105, 439–455.

Ying, C., Chow, A.H.F., Chin, K.-S., 2020. An actor-critic deep reinforcement learning approach for metro train scheduling with rolling stock circulation under stochastic demand. Transp. Res. Part B Methodol. 140, 210–235.

Zhang, L., Garoni, T.M., de Gier, J., 2013. A comparative study of macroscopic fundamental diagrams of arterial road networks governed by adaptive traffic signal systems. Transp. Res. Part B Methodol. 49, 1–23.

Zhong, C., Lu, Z., Gursoy, M.C., Velipasalar, S., 2018. Actor-critic deep reinforcement learning for dynamic multichannel access, in: 2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP). pp. 599–603.

Zhou, X., Zhong, M., 2007. Single-track train timetabling with guaranteed optimality: Branch-and-bound algorithms with enhanced lower bounds. Transp. Res. Part B Methodol. 41, 320–341.

Zhu, F., Ukkusuri, S. V, 2013. A cell based dynamic system optimum model with non-holding back flows. Transp. Res. Part C Emerg. Technol. 36, 367–380.