ADAPTIVE DYNAMIC TRAFFIC CONTROL OF URBAN NETWORKS: A MACROSCOPIC FUNDAMENTAL DIAGRAM APPROACH

CAN CHEN

PhD

The Hong Kong Polytechnic University

2024

The Hong Kong Polytechnic University

Department of Civil and Environmental Engineering

Adaptive dynamic traffic control of urban networks: A macroscopic fundamental diagram approach

Can CHEN

A thesis
submitted in partial fulfilment of the requirements
for the degree of

Doctor of Philosophy

May 2024

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____

Can CHEN

Abstract

Urbanization has induced dramatic growth in car usage in metropolises around the world, which results in growing traffic congestion, accidents and pollution. Efficient utilization of existing infrastructures via appropriate traffic control schemes is crucial to handling the fast-growing travel demand. Conventional traffic control methods concentrate on link-level strategies. Oversaturated traffic conditions with queues spilling back to upstream links and the huge spatial dimension would introduce significant challenges to the local traffic signal control strategies at the link level. Hence, under heavily saturated traffic conditions, traffic control strategies capturing network-level congestion should be devised to alleviate network congestion.

The network-level congestion can be significantly alleviated by identifying some critical intersections and regulating them effectively. This finding gives rise to the concept of perimeter control by leveraging the recent advances in the macroscopic fundamental diagrams (MFDs). The MFD intuitively describes a low-scatter relationship between the network vehicle accumulation and production, providing an analytically simple and computationally efficient framework for aggregate modeling of urban traffic network dynamics. Therefore, this dissertation proposes an MFD-based optimal control framework for traffic networks.

Perimeter control, which aims to manipulate the transfer flow at the boundaries of the region, is a promising solution to address the spatial dimension challenge in dealing with network-scale traffic congestion. Existing MFD-based data-driven and feedback perimeter control strategies do not consider the heterogeneity of real-time data measurements. Besides, traditional reinforcement learning (RL) methods for traffic control usually converge slowly for lacking data efficiency. Moreover, conventional optimal perimeter control schemes require exact knowledge of the system dynamics and thus they would be fragile to endogenous uncertainties. To handle these challenges, Study 1 proposes an integral reinforcement learning (IRL) based approach to learning the macroscopic traffic dynamics for adaptive optimal perimeter control. A continuous-time control is developed with discrete gain updates to adapt to the discrete-time sensor data. Different from the conventional RL approaches, the reinforcement interval of the proposed IRL method can be varying with respect to the real-time resolution of data measurements. To reduce the sampling complexity and use the available data more efficiently, the experience

replay (ER) technique is introduced to the IRL algorithm. The proposed method relaxes the requirement on model calibration in a model-free manner that enables robustness against modeling uncertainty and enhances the real-time performance via a data-driven RL algorithm. Numerical examples and simulation experiments are presented to verify the effectiveness and efficiency of the proposed method.

Considering the time-varying nature of the travel demand pattern and the equilibrium of the accumulation state, Study 2 extends the set-point perimeter control (SPC) problem investigated in Study 1 to an optimal tracking perimeter control problem. Unlike the SPC schemes that stabilize the traffic dynamics to the desired equilibrium point, the proposed tracking perimeter control (TPC) scheme will regulate the traffic dynamics to a desired trajectory in a differential framework. Study 2 proposes an adaptive dynamic programming (ADP) approach to solving the optimal TPC problem. The convergence of the ADP based algorithms and the stability of the controlled traffic dynamics are proven via the Lyapunov theory. Numerical experiments are performed to demonstrate the effectiveness of the proposed ADP-based TPC. Compared with the SPC scheme, the proposed TPC scheme achieves both improvements in reducing total travel delay and increasing cumulative trip completion in our case studies.

Coupling perimeter control and regional route guidance (PCRG) is a promising strategy to decrease congestion heterogeneity and reduce delays in large-scale MFD-based urban networks. For MFD-based PCRG, one needs to distinguish between the dynamics of the plant that represents reality and is used as the simulation tool, and the model that contains easier-to-measure states than the plant and is used for devising controllers, i.e., the model-plant mismatch should be considered. Traditional model-based methods require an accurate representation of the plant dynamics as the prediction model. On the other hand, existing data-driven methods do not consider the model-plant mismatch and the limited access to plant-generated data. Therefore, Study 3 develops an iterative adaptive dynamic programming (IADP) based method to address the limited data source induced by the model-plant mismatch. An actor-critic neural network structure is developed to circumvent the requirement of complete information on plant dynamics. Performance comparisons with other PCRG schemes under various scenarios are carried out. The numerical results indicate that the IADP controller trained with a limited data source can achieve comparable performance in minimizing the total travel delay with the benchmark model predictive control (MPC) approach using perfect measurements from the plant. In cases of higher input errors, IADP achieves a better performance than MPC.

Most existing studies on optimal traffic control of MFD-based networks do not consider the effect of expressways passing through urban regions. Ring expressways are built in many megacities (e.g., Beijing) with on- and off-ramps to connect the city's periphery areas where ramp metering is usually desired to protect the freeways from over congestion. Few studies have explored the cooperation of perimeter control, route guidance and ramp metering strategies in improving the whole network mobility. Study 4 proposes a cooperative adaptive dynamic programming (CADP) approach to solve the cooperative control problem for a mixed urban-expressway network. The network is composed of a multi-region urban network modeled by the MFD and a ring expressway going through the periphery regions modeled by the asymmetric cell transmission model. Different from the traditional decentralized ADP (D-ADP) method, the proposed CADP approach trains the agents of perimeter control, route guidance, and ramp metering to fully cooperate in improving the whole network performance. Numerical studies demonstrate that the CADP can significantly reduce the total travel delay compared with the model-based decentralized strategies and the D-ADP strategy. In addition, the city center is well protected from over-congestion by applying the CADP approach.

In conclusion, this thesis contributes to the literature on network-level optimal perimeter control and regional route guidance, and to traffic management of mixed urban-expressway networks.

List of Publications

JOURNALS:

[1] Can Chen, Nikolas Geroliminis, Renxin Zhong* (2024) An iterative adaptive dynamic programming approach for macroscopic fundamental diagram-based perimeter control and route guidance. *Transportation Science*, 58 (4): 896-918. (doi:10.1287/trsc.2023.0091)

[2] Can Chen, Yunping Huang, William H.K. Lam, Tianlu Pan, Shu-Chien Hsu, Agachai Sumalee, Renxin Zhong* (2022) Data efficient reinforcement learning and adaptive optimal perimeter control of network traffic dynamics. *Transportation Research Part C: Emerging technologies*, 142: 103759. (doi:10.1016/j.trc.2022.103759)

[3] Simin Jiang, Tianlu Pan, Renxin Zhong*, Can Chen, Xin'an Li, Shimin Wang (2022) Coordination of Mixed Platoons and Eco-Driving Strategy for a Signal-Free Intersection. *IEEE Transactions on Intelligent Transportation Systems*, 24(6): 6597-6613. (doi:10.1109/TITS.2022.3211934)

[4] Yunping Huang, Can Chen, Zicheng Su, Tianshi Chen, Agachai Sumalee, Tianlu Pan, Renxin Zhong* (2021) Bus arrival time prediction and reliability analysis: An experimental comparison of functional data analysis and Bayesian support vector regression. *Applied Soft Computing*, 111: 107663. (doi:10.1016/j.asoc.2021.107663)

[5] Renxin Zhong, Hengxing Cai, Dabo Xu, Can Chen, Agachai Sumalee, Tianlu Pan* (2020) Dynamic feedback control of day-to-day traffic disequilibrium process. *Transportation Research Part C: Emerging technologies*, 114, 297-321. (doi:10.1016/j.trc.2020.02.005)

CONFERENCES:

[1] Can Chen, Yunping Huang, Hongwei Zhang, Shu-Chien Hsu*, Renxin Zhong* (2024) Tracking perimeter control for two-region macroscopic traffic dynamics: An adaptive dynamic programming approach. *The 27th IEEE International Conference on Intelligent Transportation Systems (Edmonton, Canada)*.

[2] Can Chen, Yunping Huang, Renxin Zhong*, Shu-Chien Hsu (2024) Adaptive cooperative traffic control of a multi-region urban network with a ring expressway. *103rd Transportation Research Board (TRB) Annual Meeting (Washington D.C., U.S)*.

[3] Can Chen, Nikolas Geroliminis*, Renxin Zhong (2023) A robust adaptive dynamic programming approach for MFD perimeter control. *102nd Transportation Research Board (TRB) Annual Meeting (Washington D.C., U.S)*.

[4] Can Chen, Yunping Huang, William H.K. Lam, Tianlu Pan, Shu-Chien Hsu, Agachai Sumalee, Renxin Zhong* (2021) Learning the macroscopic traffic dynamics for adaptive optimal perimeter control with integral reinforcement learning. *24th International Symposium on Transportation and Traffic Theory (Beijing, China)*. (available: https://isttt24.buaa.edu.cn)

*Note: * Corresponding author(s).*

Acknowledgement

First and foremost, I would like to express my sincere gratitude to my supervisors: Dr. Shu-Chien Hsu and Prof. Agachai Sumalee for their attentive guidance, pivotal inspiration, and continued encouragement over the past years. Throughout my research journey, I was struck by the way they looked at the problem and dug deep into it, providing me with a unique perspective and thorough understanding of the subject matter. I would not be able to complete my Ph.D. without their continuous support.

I would like to express my deepest gratitude to Prof. Renxin Zhong, who, although not officially on my committee, has provided unwavering support and guidance throughout my doctoral voyage. He introduced me to the research field during my master's study. I have greatly benefited from our discussions, and his constant help and advice have been instrumental in shaping my academic pursuits.

Special thanks should be given to Prof. Nikolas Geroliminis for supporting me as an exchange student at Urban Transport Systems Laboratory (LUTS), École Polytechnique Fédérale de Lausanne (EPFL), Switzerland. His unwavering passion and dedication to research and teaching deeply set a lifelong example for me. The exchange program was a crucial starting point for our continued research collaboration.

I would also like to thank Dr. Tony N.N. Sze, Prof. Bin Yu, and Dr. Mohsen Ramezani for taking the time to review this dissertation and for providing valuable comments and discussions. Their valuable suggestions and criticisms have greatly enhanced the quality of this research and dissertation.

I want to extend my thanks to my colleagues and friends who helped, inspired, and encouraged me during my Ph.D. study. My sincere gratitude first goes to the lab mates and friends from The Hong Kong Polytechnic University, including but not limited to Dr. Yunping Huang, Dr. Junbiao Su, Dr. Zhongnan Ye, Dr. Ziyue Yuan, Dr. Wei Ma, Mr. Zijian Hu, Mr. Penglin Song, Dr. Julio Ho, Dr. Vahid Asghari, Mr. Zixuan Kang, Mr. Mudasir Hosein, Ms. Shujie Xu, Miss Xiaoyi Liu, Mr. Xinyu Yan, Mr. Hongfeng Liang. I want to express my gratitude to Dr. Zicheng Su and Mr. Enming Liang from The City University of Hong Kong, Mr. Canqiang Weng, Mr. Zheng Huang, Ms. Wenfei Ma, Ms. Yingqi Liu, and Mr. Qinzhou Ma from Sun Yat-Sen University,

Dr. Wentao Huang and Mr. Jing Gao from The Hong Kong University of Science and Technology. I am grateful for the enriching discussions and fruitful collaborations. I would also like to thank Ms. Pengbo Zhu, Mr. Georgios Anagnostopoulos, Dr. Caio Vitor Beojone, Ms. Christine Debossens, Dr. Zhenyu Yang, and Prof. Michel Bierlaire from EPFL for their kind help in my study and life.

Last but not least, my deepest gratitude goes to my dearest parents, Xiaoqing and Shuwu, for their unconditional love and endless support. Thank you for everything. This work is dedicated to you.

Contents

List of Figures

List of Tables

Abbreviations

MFD	Macroscopic Fundamental Diagram
ADP	Adaptive Dynamic Programming
RL	Reinforcement Learning
MPC	Model Predictive Control
TTS	Total time spent
CTC	Cumulative trip completion
CTM	Cell transmission model
ACTM	Asymmetric cell transmission model
AC-NN	Actor-critic neural network
NN	Neural network
HJB	Hamilton-Jacobi-Bellman
IRL	Integral reinforcement learning
ER	Experience replay
IADP	Iterative adaptive dynamic programming
CADP	Cooperative adaptive dynamic programming
TD	Temporal difference
PC	Perimeter control
RG	Route guidance
PCRG	Perimeter control integrated with route guidance
OD	Origin-destination
TUC	Traffic-responsive Urban Control
PI	Proportional-integral

Introduction and objectives

1

1.1 Background and motivations

Urbanization has induced dramatic growth in car usage in metropolises around the world, which results in growing traffic congestion, accidents and pollution. Efficient utilization of existing infrastructures via appropriate traffic control schemes is crucial to handling the fast-growing travel demand. Over the past decades, several traffic control strategies have been proposed and successfully implemented in practice (see Papageorgiou et al., 2003, for an overview). Conventional traffic control methods such as SCOOT (Hunt et al., 1982), SCATS (Lowrie, 1982) and Traffic-responsive Urban Control (TUC) such as ALINEA (see Figure 12 in Papageorgiou et al., 2003), concentrate on link-level strategies. In the case of heterogeneous networks with multiple bottlenecks and heavily directional demand flows, local traffic-responsive metering controls such as TUC may not be optimal or might not achieve the stabilization of the system in a reasonable time period (Kouvelas et al., 2017). Oversaturated traffic conditions with queues spilling back to upstream links and the huge spatial dimension would introduce significant challenges to the local adaptive real-time traffic signal control strategies at the link level, i.e., SCOOT and SCATS (Gayah et al., 2014; Zhong et al., 2018a; Zhong et al., 2018b). Hence, under heavily saturated traffic conditions, traffic control strategies capturing network-level congestion should be devised to alleviate network congestion.

The network-level congestion can be significantly alleviated by identifying some critical intersections and regulating them effectively (Kouvelas et al., 2017). This finding gives rise to the concept of perimeter control by leveraging the recent advances in the macroscopic fundamental diagrams (MFDs). Pioneered by Godfrey (1969), with its existence proven by Daganzo (2007) theoretically, the MFDs have been widely investigated (Haddad and Geroliminis, 2012; Haddad et al., 2013; Keyvan-Ekbatani et al., 2013; Leclercq et al., 2014; Yildirimoglu and Geroliminis, 2014; Saeedmanesh and Geroliminis, 2017). The MFD intuitively describes a low-scatter relationship between the network vehicle accumulation and production, providing an analytically simple and computationally efficient framework for aggregate modeling of

urban traffic network dynamics. Under the MFD framework, a heterogeneous urban traffic network is divided into several homogeneous regions with each admits a well-defined MFD (Ji and Geroliminis, 2012). Under certain regularity conditions, such as stationary (or slow-varying) and evenly distributed demand, well-defined MFDs were evidenced by both simulation-based experiments (Gartner and Wagner, 2004) and empirical investigations (Geroliminis and Daganzo, 2008). In particular, Loder et al. (2019) empirically observed the existence of the MFDs and their critical point variations using billions of vehicle observations from more than 40 cities. Further analytical consideration and empirical evidence have been provided by Daganzo and Geroliminis (2008), Helbing (2009), Ji et al. (2010), Gayah and Daganzo (2011), and Daganzo et al. (2011). However, heterogeneous networks, in essence, do not exhibit a well-defined MFD. Such a network can be modeled by a set of differential equations governing the traffic flow conservation in conjunction with MFDs as long as it can be partitioned into homogeneous subregions with each admits a well-defined MFD (Ji and Geroliminis, 2012).

The adoption of MFDs to model and regulate traffic flows of large-scale urban networks has been widely studied in the last decade. The MFD has evolved as a promising solution for large-scale urban management and in applications like traffic state estimation (Yildirimoglu and Geroliminis, 2014; Ambühl and Menendez, 2016; Mariotte et al., 2020; Ma et al., 2024), perimeter control (Ampountolas et al., 2017; Zhong et al., 2018a; Mohajerpoor et al., 2020; Haddad and Mirkin, 2020; Su et al., 2023; Moshahedi and Kattan, 2023; Tsitsokas et al., 2023; Hu and Ma, 2024), congestion pricing (Gu et al., 2018; Zheng and Geroliminis, 2020), route guidance (Knoop et al., 2012; Sirmatel and Geroliminis, 2018; Hou and Lei, 2020; Jiang et al., 2024), ridesharing (Wei et al., 2020; Ramezani and Valadkhani, 2023; Valadkhani and Ramezani, 2023; Huang et al., 2023), demand management (Yildirimoglu and Ramezani, 2020; Kumarage et al., 2021), departure time choice (Huang et al., 2020; Zhong et al., 2021; Zhong et al., 2020; Ameli et al., 2022) and cruising for parking (Cao and Menendez, 2015; Leclercq et al., 2017), etc.

The perimeter control and regional route guidance are the most significant applications of the MFD. The perimeter control aims to manipulate the transfer flow at the boundaries of the region by identifying critical intersections and regulating them, which is a promising solution to alleviating network-scale traffic congestion. Different from the en-route link-level route guidance strategy, the regional route guidance strategy advises drivers a sequence of regions with a lower cost (in terms of travel time, fuel consumption, etc.) to assist them in reaching their destination, which might improve the overall system performance. Most existing literature on MFD-based perimeter control and regional route guidance has focused on model-based

approaches such as the model predictive control (MPC) adopted by Geroliminis et al. (2013). Model-based controllers generally assume that model parameters are accurately calibrated and perfect knowledge of the network is available. Nevertheless, traffic networks are subject to various uncertainties (e.g., demand noise and model error), making these assumptions difficult and even impossible to be met. It is desired that traffic controllers can well adapt to the changes in traffic conditions and achieve a satisfactory performance when the network traffic dynamics are uncertain or even unknown.

Reinforcement learning (RL), a concept under the umbrella of artificial intelligence, has gained recent attention due to its success in video games and Go (Mnih et al., 2015; Silver et al., 2016). In an RL setting, an agent learns to optimize a long-term goal-oriented reward through policy learning by interacting with the environment and evaluating the performance of its actions based on feedback. Then the agent seeks to improve its performance over time (Sutton and Barto, 2018). Adaptive dynamic programming (ADP) is an RL reformulation in the economics and management communities, which provides an approximate solution to the optimal control problem given by the Bellman optimality principle. The ADP has been used to design optimal traffic signal controls for large-scale networks, with simulation results showing that low-complexity parametrization of the Hamilton-Jacobi-Bellman (HJB) equation achieves an adequate compromise between network efficiency and computational complexity (Baldi et al., 2019). The RL/ADP approach can address the model error and external uncertainty in a "model-free" manner. The RL/ADP circumvents the necessity of perfect system information by learning with trials and errors from interactions with the environment. However, traditional RL/ADP methods do not consider the heterogeneity in real-time data resolution and the limited access to plant[1] data that are very difficult to collect. It is desired that the RL/ADP-based perimeter control and regional route guidance strategies can be robust to the heterogeneous data resolution and work well in the unknown plant environment even though they are trained with parsimonious data.

With the expansion of the city radius and the emergence of city agglomeration, urban networks connected by arterials cannot satisfy the travel demand of citizens. Highway and expressway are built to connect the city center and satellite city in the suburbs. It is necessary to consider the travel demand of the highways when designing perimeter control and route guidance strategies. However, few existing works on MFD-based perimeter control and regional route guidance have taken this into account. Haddad et al. (2013) and Hu and Ma (2024) have integrated

[1]The real system is termed 'plant' while the simplified dynamics used for controller design is termed 'model'. These concepts will be further explained in the following sections.

perimeter control and ramp metering for mixed urban-expressway networks where the expressway traffic is modeled by the cell transmission model (CTM) (Daganzo, 1994). Case studies in Haddad et al. (2013) are limited in region size while Hu and Ma (2024) do not incorporate the regional route choice model in the perimeter controller design.

Considering the previous works on MFD-based urban traffic control, three research directions that are not well explored in the literature are identified: (i) development of "model-free" perimeter control that incorporates the heterogeneous data resolution, (ii) integration of adaptive data-driven perimeter control with regional route guidance considering the model and plant dissimilarity and limited plant data available, and (iii) coordination of various network-level traffic management schemes in improving the operation of large-scale mixed traffic networks. This thesis contributes to the literature on network-level adaptive optimal perimeter control and regional route guidance, and on traffic management of mixed urban-expressway networks.

1.2 Research objectives

This dissertation integrates theories and methods from multiple disciplines including control theory, traffic flow modeling, and machine learning. The interconnection of different components of this dissertation are illustrated in Figure 1.1.

The goal of this dissertation is to develop adaptive optimal control strategies for large-scale urban traffic networks to improve travelers' mobility and network performance. In light of the aforementioned research background and motivations, the objectives can be categorized into three distinct groups: (i) development of "model-free" adaptive control methods for perimeter-controlled urban networks, (ii) design of traffic management schemes coupling regional route guidance actuation and perimeter control for heterogeneous networks, and (iii) cooperation of perimeter control, route guidance, and ramp metering in large-scale mixed urban-expressway networks. Objectives of individual chapters according to dissertation structure are listed as follows:

"Model-free" optimal perimeter control

- The main objective of Chapter 3 (i.e., Study 1) is to devise a data-efficient adaptive perimeter controller for the MFD framework without relying on any knowledge on the traffic dynamics, i.e., "model-free". Traffic networks are

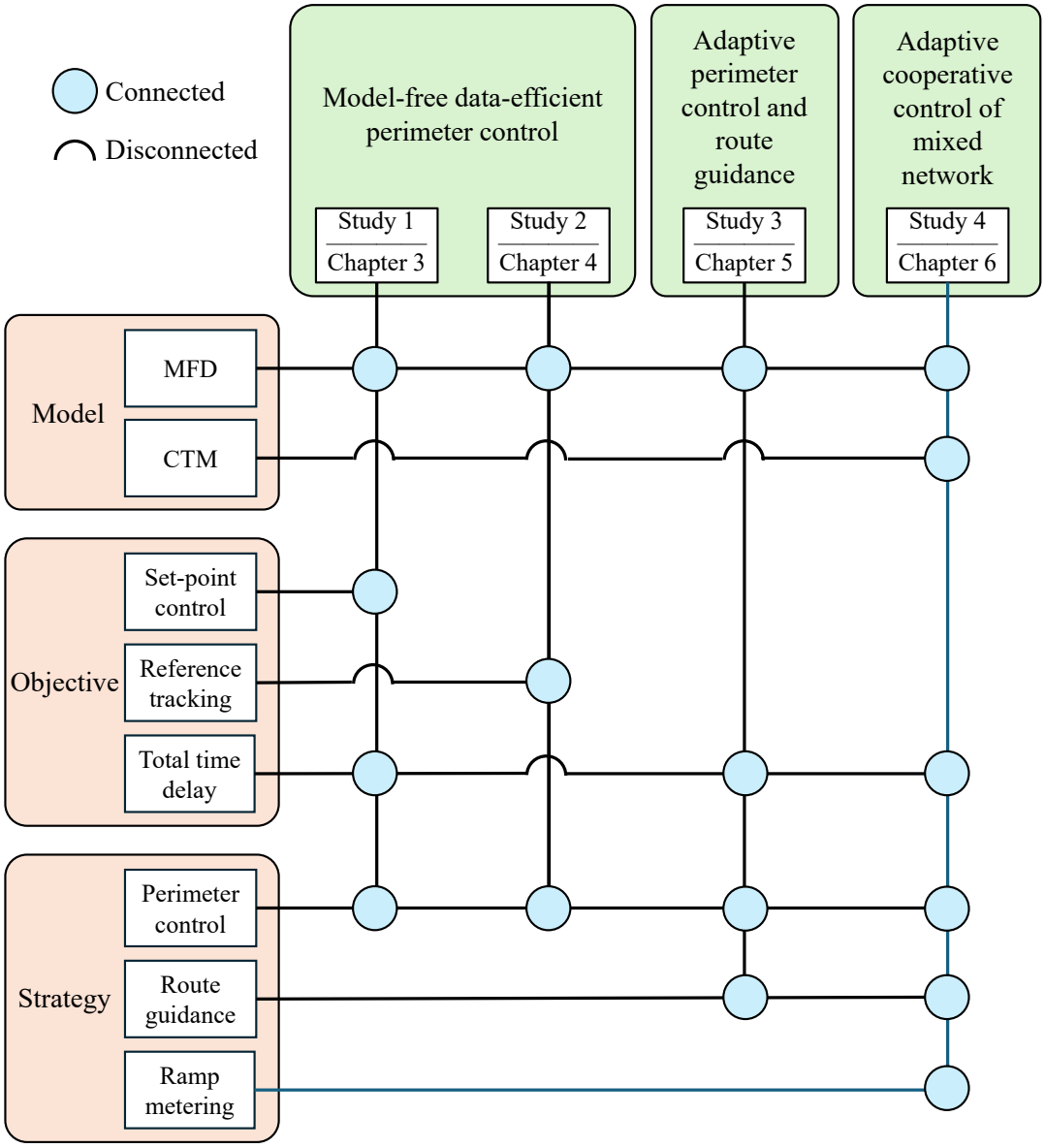


Figure 1.1 The interconnection of different components of the thesis

subject to model errors and demand uncertainties. Thus the MFD parameters are uncertain and even unknown. We aim to circumvent the requirement of perfect system information when designing the optimal perimeter controller. Traditional data-driven methods such as RL lack data efficiency and do not consider the heterogeneity in real-time data resolution. We try to enhance the data efficiency of the RL approach and make it robust to the time-varying data resolution.

- The key objective of Chapter 4 (i.e., Study 2) is to develop a trajectory-stabilizing perimeter controller that tracks a desired reference. Because of the time-varying nature of the travel demand and supply, an inappropriate choice of the single setpoint could degrade the perimeter control performance and the MFD system's stability. Instead, a trajectory reference that better fits the demand and supply nature is desirable. We attempt to extend the stability analysis of a single equilibrium (or its invariant set) studied in Chapter 3 to that of a Lipschitz continuous trajectory. An adaptive tracking perimeter controller will be devised.

Iterative adaptive perimeter control and regional route guidance

- The chief objective of Chapter 5 (i.e., Study 3) is to address the hurdle induced by model-plant mismatch in optimizing perimeter control and regional route guidance strategies. As region size increases, regional route guidance systems are necessary for the traffic control of multi-region MFD-based networks. One needs to distinguish the model used for optimization from the plant that is subject to heterogeneity and contains difficult-to-measure states. Hence, it is desired that one can circumvent the requirement for plant-generated data while using merely the parsimonious model data for learning the MFD traffic dynamics. We aim to develop an optimal perimeter control and route guidance strategy that can be directly implemented in the plant while using only measurements from the model.

Cooperative control of mixed urban-expressway networks

- The main objective of Chapter 6 (i.e., Study 4) is to further extend the strategies developed in Chapter 5 to handle more complicated traffic networks. With the expansion of the city radius, freeways/expressways are built to connect different parts of the city. Other than arterial roads, expressways are playing an important role in the traffic management of a megacity. One should explore the effect of these expressways and their control strategies when devising traffic control schemes for urban networks. We attempt to develop a cooperative control strategy for a mixed urban-expressway network. The complexity of the model structure could render an invalid solution to the cooperative control problem of large-scale networks. We try to address this difficulty using data-driven approaches.

1.3 Organization

This dissertation consists of seven chapters, and there are five main chapters (excluding Chapter 1 and Chapter 7). The presentation of the main five chapters is organized as follows.

Chapter 2 revisits the main applications of MFDs in network-level traffic control and management. We first review the MFD-based perimeter control. In general, there are two main goals of regional traffic regulation exploiting perimeter control. One is to manipulate the regional accumulation to the desired equilibrium (the so-called set-point control), and the other is to achieve the best network mobility in terms of maximum trip completion or minimum total travel delay. Existing methods of feedback-based perimeter control leveraging the MFD can be categorized into two main types, model-based and model-free. The importance of model-free perimeter control is highlighted due to uncertain or unknown traffic dynamics, which might make model-based methods invalid. The ADP/RL is a promising approach to solving optimal control problems with no requirement for perfect system knowledge. We revisit the recent applications of ADP/RL methods in perimeter control. Note that regional route guidance can be integrated with perimeter control to enhance urban network mobility. We then review the existing works on regional route guidance leveraging MFDs and on cooperative traffic control of urban network dynamics. Finally, we present the preliminary knowledge on ADP and integral reinforcement learning (IRL). We outline how the general optimal control couples the nonlinear system dynamics, the Bellman equation, and the neural network framework.

Chapter 3 develops an integral reinforcement learning approach to learning the macroscopic traffic dynamics for adaptive optimal perimeter control. A continuous-time perimeter controller that can adapt to the discrete-time heterogeneous sensor data is devised. The experience replay technique is utilized to boost the data efficiency of the IRL algorithm. Convergence of the algorithm and stability of the controlled traffic dynamics are proven via the Lyapunov theory. Preliminary results of this work are presented in:

- Can Chen, Yunping Huang, William H.K. Lam, Tianlu Pan, Shu-Chien Hsu, Agachai Sumalee, Renxin Zhong (2021) Learning the macroscopic traffic dynamics for adaptive optimal perimeter control with integral reinforcement learning. *24th International Symposium on Transportation and Traffic Theory (Beijing, China)*. (available: https://isttt24.buaa.edu.cn)

Chapter 3 is a stand-alone article published as:

- Can Chen, Yunping Huang, William H.K. Lam, Tianlu Pan, Shu-Chien Hsu, Agachai Sumalee, Renxin Zhong (2022) Data efficient reinforcement learning and adaptive optimal perimeter control of network traffic dynamics. *Transportation Research Part C: Emerging technologies*, 142: 103759. (doi:10.1016/j.trc.2022.103759)

Chapter 4 extends the results of Chapter 3 in terms of the stability of a single equilibrium (or its invariant set) to a desired trajectory. A trajectory stability concept in the MFD framework is proposed in this chapter, which can better fit the dynamic nature of travel demand and supply. Upon the determination of the trajectory reference, an adaptive tracking perimeter control scheme is devised to regulate the traffic dynamics to the desired trajectory. Preliminary results of this work are presented in:

- Can Chen, Yunping Huang, Hongwei Zhang, Shu-Chien Hsu, Renxin Zhong (2024) Tracking perimeter control for two-region macroscopic traffic dynamics: An adaptive dynamic programming approach. *The 27th IEEE International Conference on Intelligent Transportation Systems (Edmonton, Canada)*.

while a journal article is under preparation.

Chapter 5 proposes an integrated strategy that couples perimeter control and regional route guidance in the management of MFD systems. The model used for optimization is distinguished from the plant that represents reality due to their differences in structure and the inherent network uncertainty. An iterative adaptive dynamic programming approach is proposed to address the limited data source induced by the model-plant mismatch. An actor-critic neural network framework is employed to circumvent the necessity of the plant-generated data that are very difficult to measure. Preliminary results of this work are presented in:

- Can Chen, Nikolas Geroliminis, Renxin Zhong (2023) A robust adaptive dynamic programming approach for MFD perimeter control. *102nd Transportation Research Board (TRB) Annual Meeting (Washington D.C., U.S)*.

Chapter 5 is a stand-alone article published as:

- Can Chen, Nikolas Geroliminis, Renxin Zhong (2024) An iterative adaptive dynamic programming approach for macroscopic fundamental diagram-based perimeter control and route guidance. *Transportation Science*, 58 (4): 896-918. (doi:10.1287/trsc.2023.0091)

Ring expressways are built in many megacities (e.g., Beijing) with on- and off-ramps connecting the city's periphery areas where ramp metering is usually implemented. Chapter 6 considers a mixed urban-expressway traffic network. We propose a cooperative adaptive dynamic programming-based control model for a multi-region urban network modeled by the MFD with a ring expressway modeled by the asymmetric cell transmission model (ACTM). Due to the complexity and strong nonlinearity of the system dynamics, solving the optimal control strategy explicitly is extremely difficult. Hence, a multi-agent actor-critic neural network framework is developed to approach the solution of the optimization problem. Preliminary results of this work are presented in:

- Can Chen, Yunping Huang, Renxin Zhong, Shu-Chien Hsu (2024) Adaptive cooperative traffic control of a multi-region urban network with a ring expressway. *103rd Transportation Research Board (TRB) Annual Meeting (Washington D.C., U.S)*.

while a journal article is under preparation.

Finally, Chapter 7 gives a summary of this thesis. Some topics for future research are also highlighted in this chapter.

Literature review

2

The adoption of macroscopic fundamental diagrams (MFDs) to model and regulate traffic flows of large-scale urban networks has been widely studied in the last decades. The MFD intuitively provides a concave relationship between the network traffic density (or accumulation) and throughput (or trip completion rate) as shown in Figure 2.1, which has evolved as a promising solution for large-scale urban management and in applications like traffic state estimation (Yildirimoglu and Geroliminis, 2014; Ambühl and Menendez, 2016; Mariotte et al., 2020), perimeter control (Ampountolas et al., 2017; Zhong et al., 2018a; Mohajerpoor et al., 2020; Haddad and Mirkin, 2020; Su et al., 2023; Moshahedi and Kattan, 2023; Tsitsokas et al., 2023; Hu and Ma, 2024), congestion pricing (Gu et al., 2018; Zheng and Geroliminis, 2020), route guidance (Knoop et al., 2012; Sirmatel and Geroliminis, 2018; Hou and Lei, 2020; Jiang et al., 2024), ridesharing (Wei et al., 2020; Ramezani and Valadkhani, 2023; Valadkhani and Ramezani, 2023), demand management (Yildirimoglu and Ramezani, 2020; Kumarage et al., 2021), departure time choice (Huang et al., 2020; Zhong et al., 2021; Zhong et al., 2020; Ameli et al., 2022) and cruising for parking (Cao and Menendez, 2015; Leclercq et al., 2017), etc. Among the aforementioned applications, perimeter control and regional route guidance are the most significant ones, which are promising ways to alleviate urban congestion and improve network mobility.

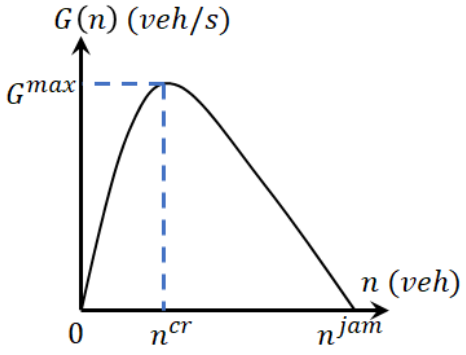


Figure 2.1 The accumulation MFD

This chapter reviews the literature related to the applications of MFD in perimeter control, route guidance, and cooperative mixed-network traffic management. It begins with a review of MFD-based perimeter control in Section 2.1. This is followed by a revisit of the perimeter control coupled with regional route guidance for MFD traffic systems in Section 2.2. Then Section 2.3 reviews the existing works on traffic control of mixed urban-expressway networks. Note that adaptive dynamic programming (ADP) is the key research method in this thesis. Finally, preliminaries of ADP and integral reinforcement learning are presented in Section 2.4.

2.1 MFD-based perimeter control

Considerable research efforts have been dedicated to devising optimal network traffic control strategies based on MFDs. The perimeter control is believed to be a promising solution to address the spatial dimension challenge while considering the network-scale traffic congestion. Gating/perimeter control, usually actuated by traffic signals installed on the boundaries between regions, is used to manipulate the intertransfer flows between regions. Recent studies showed that feedback-based perimeter control is efficient in mitigating congestion in the protected urban networks by exploiting MFDs. One goal is to manipulate the regional accumulation to the desired equilibrium (e.g., to operate the protected regions around the critical accumulation that maximizes flow), i.e., set-point control. Aboudolas and Geroliminis (2013) used linear-quadratic-integral (LQI) and linear-quadratic-regulator (LQR) to operate the MFD system to approach the equilibrium points while Keyvan-Ekbatani et al. (2012) utilized a proportional-integral (PI) controller considering system uncertainty. Keyvan-Ekbatani et al. (2013), Keyvan-Ekbatani et al. (2015a), and Keyvan-Ekbatani et al. (2015b) solved the set-point control problem using the PI controller with consideration of boundary queue in MFDs, and different kinds of uncertainty and disturbance were included in the simulations. Haddad and Mirkin (2016) proposed a transfer function embedded with time delay to deal with the set-point control problem.

Another goal of using perimeter control is to achieve the maximum trip completion flow or to minimize the total travel time of the road network by properly restricting the traffic inflow to the network. Daganzo (2007) applied the MFD framework to devise a control rule that maximizes the network trip completion rate. Geroliminis et al. (2013) and Ramezani et al. (2015) solved the optimal perimeter control problem within a standard two-region MFD system by model predictive control

(MPC) while Haddad et al. (2013) implemented MPC on a mixed network. Other optimal perimeter controls of the MFD system using MPC were in a hierarchical scheme (Zhou et al., 2016; Fu et al., 2017). Aalipour et al. (2018) derived an analytical optimal control policy by solving the Hamilton-Jacobi-Bellman (HJB) equation for maximizing the trip completion rates.

Apart from optimal control using the MFD framework, the robust perimeter control problem of the MFD-based network traffic was also addressed in previous studies using linear matrix inequalities, e.g., Haddad and Shraiber (2014) and Haddad (2015). All the above methods require linearization of the MFD function except for Zhong et al. (2018a) and Sirmatel and Geroliminis (2021). Sirmatel and Geroliminis (2019) developed a nonlinear moving horizon estimation scheme for large-scale urban networks subject to measurement noises in state and inflow demand. Li et al. (2021b) proposed a sliding mode controller for two-region MFD-based networks considering cordon queues and heterogeneous transfer flows.

Other recent efforts were put to devising resilient perimeter control under cyber-attacks (Mercader and Haddad, 2021) and in hyper-congested networks (Gao et al., 2022), real-time state estimation in multi-region MFD urban networks (Saeedmanesh et al., 2021), multi-region extension for the M-model that captures the effects of remaining travel distance dynamics (Sirmatel et al., 2021), perimeter control for congested areas against state degradation risk (Ding et al., 2020b), optimal perimeter control considering coupled/decoupled controllers (Haddad, 2017a), aggregate boundary queue dynamics (Haddad, 2017b), and perimeter control with dynamic boundary (Li et al., 2021a; Ding et al., 2022; Hamedmoghadam et al., 2022).

The aforementioned studies on perimeter control can be regarded as model-based traffic responsive control. Specifically, previous studies on the feedback-based perimeter control were derived under one common assumption that the model parameters can be accurately calibrated. For optimal perimeter control, in particular, it is generally assumed that perfect knowledge of the network is available and the parameters will not change during the planning horizon. Moreover, local linearization around the desired equilibrium is widely performed to simplify the control design. Apart from model-based traffic responsive control, considerable research efforts have focused on adaptive perimeter control which adapts to a controlled system with time-varying and/or uncertain parameters or external disturbances such as travel demand noise. By considering the boundary queue that can have a negative impact on upstream queue modeling, Kouvelas et al. (2017) introduced an online adaptive parameters optimization algorithm for perimeter control. Haddad

and Zheng (2020) designed the distributed adaptive perimeter control laws with control gains varying with time considering state delays and interconnection delays. Since traffic networks are subject to various uncertainties, parameters of MFDs are uncertain and time-varying. Also, the travel demand and traffic control strategies can significantly affect the shape of the MFD (Geroliminis and Boyacı, 2012). The performance of these control strategies increasingly deteriorates with increasing disturbance prediction and model errors (Zhong et al., 2014; Baldi et al., 2019). Nevertheless, as specified in Kouvelas et al. (2017), in many cases the adopted models are calibrated once and would not be re-calibrated regularly. This causes a defect in their field experiments. Despite the vast literature related to modeling and control with MFDs, the design of dynamic control policies to various exogenous disturbances that can affect the dynamics is seldom considered. To adapt the real-time observation and then the control to operate the traffic network optimally, it is necessary to keep adjusting the model parameters (Kouvelas et al., 2017). However, this process can be a heavy computational burden and difficult to be implemented in real-time (Modares et al., 2014). For an ever-changing traffic environment subject to various exogenous disturbances, a predefined model-based traffic responsive policy may become suboptimal or even impractical. Yet in the literature, to the best knowledge of the authors, few existing studies dealt with the problem of devising the adaptive control strategies for MFD systems with (partially) unknown system dynamics. Lei et al. (2019) and Ren et al. (2020) devised a "model-free" perimeter controller for a multi-region MFD-based network via the iterative learning control by assuming recurrent traffic conditions that the traffic dynamics would not admit a significant change in a day-to-day time-scale during the learning period.

The emerging big data technology gives rise to data-driven approaches to solving the aforementioned difficulties. Rooted in computer science, the reinforcement learning (RL) has attracted increasing attention recently for its success in video games (Mnih et al., 2015) and Go (Silver et al., 2016). Under the RL setting, an agent optimizes a goal-oriented long-term reward via policy learning. At each step, the RL agent interacts with the environment and evaluates the performance of its action based on the feedback from the environment. The agent then tries to improve the performance of subsequent actions (Sutton and Barto, 2018). A reformulation of RL is called adaptive dynamic programming (ADP) in economics and management communities. The RL and ADP bridge the gap between optimal control and adaptive control. In an off-line manner, the RL and ADP provide an approximate solution to the optimal control problem obtained from the Pontryagin's minimum principle and the dynamic programming principle (i.e., the HJB equation). Solving the HJB equation takes the center stage in deriving optimal control strategies. However, the

HJB equation is generally intractable to be solved by analytical approaches for strong nonlinearity, possible discontinuities in the solution and the curse of dimensionality. To handle the curse of dimensionality in optimal traffic signal control design for large-scale networks, Baldi et al. (2019) parametrized the solution of the HJB equation using an appropriate Lyapunov function. The simulation results showed that the approximately optimal traffic signal control design via low-complexity parametrization of the HJB equation can provide a satisfactory trade-off between computational complexity and network performance. However, there is a lack of analytical proof of the convergence as well as the explicit consideration of saturated constraints on the system state and input in Baldi et al. (2019). A conventional ADP based RL algorithm was proposed by Su et al. (2020) to provide an analytical optimal perimeter control law for the MFD dynamics. Both convergence and stability of the closed-loop system were achieved. However, conventional approximation techniques for solving the HJB equations require complete or partial knowledge of the system dynamics and are normally off-line. Thus, they cannot handle modeling uncertainties and be deployed for real-time applications.

The RL and ADP in the data-driven control community give rise to a promising solution for optimal perimeter control problems in a "model-free" manner. Data-driven deep reinforcement learning has been incorporated in solving traffic optimization problems (Kheterpal et al., 2018). Zhou and Gayah (2021) proposed a deep RL based scheme for the two-region perimeter control problems, which can achieve comparable performances to the MPC approach. Traditional RL methods do not consider the heterogeneity in data resolution and are usually trained off-line requiring intensive data. To overcome these difficulties, Study 1 of this thesis will develop an integral reinforcement learning (IRL) based approach for adaptive optimal perimeter control in MFD systems. The IRL approach to be proposed enables online tuning of the reinforcement interval to adapt to the real-time data resolution and ensures the data richness for online training. More discussion on the contributions of Study 1 can be found in Chapter 3. Moreover, conventional feedback-based perimeter controls aim to regulate the accumulation state to a single predefined set point. Due to the time-varying demand pattern and supply function, a predefined set point (or its invariant set) may not be an appropriate control objective and could degrade network mobility. Study 2 of this thesis fills this gap by developing an adaptive tracking perimeter control for the MFD framework. More discussion on the contributions of Study 2 can be found in Chapter 4.

2.2 Regional route guidance leveraging MFDs

Regional route guidance, which is a promising approach to alleviating urban traffic congestion, has been incorporated into the MFD framework in the past decade. Different from the conventional link-based route guidance strategy, the regional route guidance system advises drivers a sequence of subregions with a lower cost (in terms of travel time, fuel consumption, etc.) to assist them in reaching their destination, which might improve the overall system performance.

Previous studies have utilized MFDs in devising link-level routing strategies. Knoop et al. (2012) developed dynamic routing strategies at the link level using the aggregated information from multiple grid subnetworks with MFDs. Leclercq and Geroliminis (2013) further studied the influence of route choice in the MFD for a two-bin network with parallel routes under various traveler's behavior realism (e.g., steady-state/dynamic user equilibrium and system optimum).

Route guidance systems were also developed for multi-region MFD-based networks. Yildirimoglu and Geroliminis (2014) developed a regional route guidance strategy based on dynamic user equilibrium (DUE), while Yildirimoglu et al. (2015) extended the analysis to a route guidance system based on dynamic system optimum (DSO). Batista et al. (2019) and Batista and Leclercq (2019) further extended the regional dynamic traffic assignment framework for MFDs in Yildirimoglu and Geroliminis (2014) to consider the variability of trip lengths inside the regions. Yildirimoglu et al. (2018) proposed a hierarchical control strategy composed of an upper-level regional route guidance scheme for minimizing the total delay and a lower-level path assignment mechanism for actuating the output of the upper-level scheme. Huang et al. (2020) and Zhong et al. (2020) investigated the DUE and DSO problems for multi-region MFD systems with time-varying delays to model simultaneous route choice and departure time choice, respectively. Zhong et al. (2021) investigated the DUE problem of departure time choice in an isotropic urban network governed by a trip-based model with identical travelers.

Recent efforts have been devoted to the integration of perimeter control and regional route guidance (PCRG) for improving traffic efficiency in multi-region MFD-based networks. A dynamic simple route choice model was firstly integrated into traffic management of an MFD-based network consisting of a freeway and two homogeneous regions by Haddad et al. (2013). Using a Logit model, Ramezani et al. (2015) developed a hierarchical control framework of model-based perimeter control for region- and subregion-based MFD systems, while iterative learning control was employed by Lei et al. (2019) for model-free perimeter control with route choice. It

is worth noting that these works utilized the Logit model to generate route choice splits. They regard the mean regional path travel time as the utility and assume a constant per origin-destination (OD) regional pair. As discussed in Batista et al. (2019), the true trip patterns in a city network are unknown and time-varying, making it very difficult to properly set and calibrate regional trip distances for the application of MFD-based models. An alternative representation of the travel time is the estimated experienced travel time (Yildirimoglu and Geroliminis, 2014), which depends on the trip distance distribution and the spatial mean speed. Other works are dedicated to optimizing the perimeter control and the controllable route guidance strategies simultaneously. Sirmatel and Geroliminis (2018) proposed an economic MPC scheme integrating perimeter control and regional route guidance to improve mobility. Hou and Lei (2020) employed a constrained adaptive predictive framework combined with MPC for designing the PCRG strategies. Fu et al. (2021) developed a PCRG strategy to prevent the protected region from congestion, wherein Colored Petri Nets were used to enhance the MFD model for capturing macroscopic characteristics (e.g., transfer flows and travel delays) of urban traffic systems.

The differences between the "model" used for control design and the "plant" used for simulating the real traffic system, i.e., the model-plant mismatch, have been considered in model-based PCRG strategies. To our best knowledge, no existing works on model-free PCRG have taken the model-plant mismatch into account. To fill this gap, Study 3 of this thesis will develop an iterative adaptive dynamic programming approach to handling the challenges brought by the model-plant mismatch. More discussion on the contributions of Study 3 can be found in Chapter 5.

2.3 Traffic control of mixed networks

With the expansion of the city radius and the emergence of city agglomeration, urban networks connected by arterials cannot satisfy the travel demand of citizens. Highway and expressway are built to connect the city center and satellite city in the suburbs. For example, the 6th Ring Road connects seven districts in Beijing while 55% of Beijing citizens live beside the 5th Ring Road. The Inner Ring Road diverted approximately 20% of the traffic flow of the central urban area in Guangzhou, which connects to the Guangzhou Ring Expressway through seven radial routes. Highways and urban streets can have varying traffic characteristics and flow-density relationships. Generally, vehicles on highways travel at a higher speed than on urban

streets. The expressway is not connected to the urban network directly but through on-ramps and off-ramps.

In general, the regulation objectives for arterial roads in the urban region and the freeways/highways are different and may conflict with each other. It is necessary to consider the travel demand of the highways when designing perimeter control and route guidance strategies. To the best knowledge of the authors, Haddad et al. (2013) initially considered the cooperative control of two regions and a freeway using perimeter control and ramp metering, wherein the urban network is modeled using a two-region MFD system and the highway surpass is modeled with asymmetric cell transmission model (ACTM). Ding et al. (2020a) proposed an integral control for macroscopic traffic guidance, ramp-coordinated control, and MFD subregion perimeter control in a road network consisting of three neighboring regions and one freeway running through. Han et al. (2020) modeled the freeway traffic with MFDs and proposed a hierarchical ramp metering control strategy to minimize the total time spent in the freeway. Yocum and Gayah (2022) devised a coordinated traffic management that combines perimeter flow control and variable speed limits (VSL), wherein the network topology is consistent with Haddad et al. (2013). For more realistic modeling of the urban network, Hu and Ma (2024) developed a demonstration-guided reinforcement learning approach for perimeter control and ramp metering of the Hong Kong network. In most literature, it is assumed that the freeway runs through the urban network, and this network representation is still far from realistic. Ring expressways are becoming more common and pivotal in megacities nowadays. However, a cooperative control model integrating perimeter control, route guidance, and ramp metering for a multi-region urban network with ring expressways remains to be explored. Due to the increased region size, the model complexity makes it impossible to solve the optimal cooperative control problem explicitly. To address these gaps, Study 4 of this thesis will propose a cooperative adaptive dynamic programming method that trains the agents of perimeter control, route guidance, and ramp metering to cooperate in improving network performance. More discussion on the contributions of Study 4 can be found in Chapter 6.

2.4 Preliminaries: adaptive dynamic programming and integral reinforcement learning

The RL and ADP help the optimal control circumvent the requirement of complete knowledge of the system dynamics so that uncertainties and changes in dynamics can be incorporated into the optimal control framework. Compared with the off-line nature of the conventional optimal control framework, the RL and ADP can find the optimal solution online in real-time using a data-driven mechanism meanwhile robustness and adaptiveness can be well achieved.

In this section, we outline how the HJB equation couples the performance functional, the nonlinear system dynamics, the IRL Bellman equation, and the neural network framework.

Consider the nonlinear system in the affine form as

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))u(t) \\ y(t) &= l(x(t)) \end{aligned} \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$, and $y(t) \in \mathbb{R}^p$ represent the state, the control input, and the output of (2.1), respectively. We call $f(x(t)) \in \mathbb{R}^n$ the drift dynamics, $g(x(t)) \in \mathbb{R}^{n\times m}$ the input dynamics, and $l(x(t)) \in \mathbb{R}^p$ the output dynamics, respectively. It is assumed that $f(0) = 0$ and $f(x(t)) + g(x(t))u(t)$ is locally Lipschitz and the system is stabilizable.

In the optimal regulation problem, the objective is to design an optimal control input such that the controlled state of (2.1) converges to the desired equilibrium by minimizing a cost functional defined as

$$J(x(0), u) = \int_0^\infty \mathcal{L}(x(t), u(t))\mathrm{d}t \equiv \int_0^\infty (N(x(t)) + U(u(t)))\mathrm{d}t$$

where $N(x) \succeq 0$ and $U(u) \succeq 0$, with $\succeq$ denotes positive semi-definite. Generally, we choose $U(u) = u^T Ru,\ R = R^T \succ 0$ and $R \in \mathbb{R}^{m\times m}$ for unconstrained control case with $\succ$ denotes positive definite. However, for many real applications, u is saturated, i.e., $\mid u_\xi \mid \leq \lambda$, $\xi = 1, \ldots, m$, where $\lambda > 0$ is the performance limit of the

actuator. Abu-Khalaf et al. (2008) proposed the following generalized non-quadratic functional to consider the effect of saturation on control input u.

$$U(u) = 2\int_0^u \lambda \tanh^{-T}(s/\lambda) R \mathrm{d}s,\ \lambda > 0$$

Without loss of generality, let $R = \mathrm{diag}(\gamma_1, \ldots, \gamma_m)$ be a positive definite matrix of proper dimension.

Definition 2.4.1 (*Admissible Control*) A control policy μ is admissible to (2.1), if $\mu(x)$ is continuous on Ω, $\mu(0) = 0$ and μ stabilizes the system (2.1) on Ω with the value function (2.2) being finite for $\forall x_0 \in \Omega$. ∎

The value function for an admissible control policy can be defined as

$$V(x,u) = \int_t^\infty \mathcal{L}(x(\tau), u(\tau))\mathrm{d}\tau \equiv \int_t^\infty (N(x(\tau)) + U(u(\tau)))\mathrm{d}\tau \tag{2.2}$$

The Hamiltonian function is

$$H\left(x, u, \frac{\partial V}{\partial x}\right) = \mathcal{L}(x,u) + \left(\frac{\partial V}{\partial x}\right)^T (f(x) + g(x)u)$$

Since the integrand of the performance functional does not depend on time explicitly and the terminal time is fixed (or infinite time) while (2.1) is an autonomous dynamical system, the optimality is given by $H\left(x, u, \frac{\partial V}{\partial x}\right) = 0$, i.e., the Bellman optimality equation

$$\mathcal{L}(x, u^\star) + \left(\frac{\partial V^\star}{\partial x}\right)^T (f(x) + g(x)u^\star) = 0 \tag{2.3}$$

The optimal control can be obtained as

$$u^\star = -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} g^T(x) \frac{\partial V^\star}{\partial x}\right) \tag{2.4}$$

Since sensors collect data and transfer them to the controllers with prescribed time resolutions, we cannot apply the Bellman equation (2.3) directly in practice. Moreover, (2.3) involves the exact system dynamics $f(x)$ and $g(x)$. To relax this requirement and to consider sensor data measurements, an equivalent formulation of the the Bellman equation (2.3) that does not involve the drift dynamics can be established

$$V(x(t)) = \int_t^{t+\Delta t} (N(x(\tau)) + U(u(\tau)))\mathrm{d}\tau + V(x(t+\Delta t))$$

for any time $t \geq 0$ and time interval $\Delta t > 0$. Δt is termed as the reinforcement interval, which can be adjusted in real-time according to the resolution of sensor data and the learning rate of the RL based algorithms. This equation is called IRL Bellman equation. By iterating on the IRL Bellman equation and updating the control policy, we can obtain both the value function and the optimal control.

Given an admissible policy u_0, for $j = 0, 1, \ldots$, given u_j, solve for the value $V_{j+1}(x)$ using the following IRL Bellman equation in iteration.

$$V_{j+1}(x(t)) = \int_t^{t+\Delta t} (N(x(\tau)) + U(u_j(\tau)))\mathrm{d}\tau + V_{j+1}(x(t+\Delta t)) \tag{2.5}$$

on convergence, set $V_{j+1}(x(t)) = V_j(x(t))$. Update the control policy $u_{j+1}(x(t))$ using

$$u_{j+1}(x(t)) = -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} g^T(x(t)) \frac{\partial V_{j+1}}{\partial x}\right) \tag{2.6}$$

(2.5)-(2.6) are known as an on-policy RL algorithm.

To uniformly approximate the value function in (2.5), we can use the following neural-network-type structure.

$$\hat{V}(x) = \hat{W}_c^T \phi_1(x)$$

where $\phi_1(x) : \mathbb{R}^n \to \mathbb{R}^N$ is the basis function vector and N is the number of basis functions. With this value function approximation, its partial derivative $\frac{\partial \hat{V}}{\partial x}$ can be approximated accordingly. Using the above approximated value function, the constrained optimal control in (2.4) can be generated by

$$\hat{u} = -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} g^T(x) \hat{W}_c^T \frac{\partial \phi_1}{\partial x}\right)$$

Incorporating these approximations into the IRL Bellman equation yields

$$e(t) = \Delta\phi_1(x(t))^T \hat{W}_c + \int_{t-\Delta t}^{t} (N(x(\tau)) + U(\hat{u}(\tau)))\mathrm{d}\tau$$

where $\Delta\phi_1(x(t)) = \phi_1(x(t)) - \phi_1(x(t-\Delta t))$ and e is the temporal difference (TD) error after using current approximated critic weight $\hat{W}_c$. To avoid the case that there are insufficient real-time data for updating the weights of the learning network and to use the data in the history stack efficiently, we consider $\Delta\phi_1(x(t_j))$ as evaluated values of $\Delta\phi_1$ at the recorded time t_j. Then, we define the Bellman equation error

(i.e., TD error) at the recorded time t_j using the current critic weight estimation $\hat{W}_c$ as

$$e(t_j) = \Delta\phi_1(t_j)^T \hat{W}_c + \int_{t_j-\Delta t}^{t_j} (N(x(\tau)) + U(\hat{u}(\tau)))\mathrm{d}\tau$$

Recent transition samples (historical data) are stored and repeatedly presented to the gradient-based update rule of the weights of the learning network (2.7) so as to speed up the computation and to obtain an easy-to-check convergent condition for the IRL algorithm. This process is known as the experience replay (ER) technique. The weights of the learning network are updated via minimizing simultaneously the instantaneous TD error (the first part of (2.7) from real-time measurement) and the TD errors for the stored transition samples (the second part of (2.7)), which is given as

$$\begin{aligned}\dot{\hat{W}}_c = &-\alpha_c \frac{\Delta\phi_1(x(t))}{(\Delta\phi_1(x(t))^T \Delta\phi_1(x(t)) + 1)^2} e(t) \\ &-\alpha_c \sum_{j=1}^{l} \frac{\Delta\phi_1(x(t_j))}{(\Delta\phi_1(x(t_j))^T \Delta\phi_1(x(t_j)) + 1)^2} e(t_j)\end{aligned} \tag{2.7}$$

The optimal policy (2.6) implemented by the on-policy IRL algorithms does not require the knowledge of $f(x)$. However, it still relies on the input dynamics $g(x)$. To get rid of $g(x)$, we may adopt an off-policy IRL algorithm that the control implemented (nearly optimal) can be different from the optimal control (2.6). Towards this, we rewrite the affine dynamics as

$$\dot{x}(t) = f(x(t)) + g(x(t))u_j(t) + g(x(t))(u(t) - u_j(t)) \tag{2.8}$$

where $u_j(t)$ is the policy to be updated and $u(t)$ is the behavior policy that is actually implemented to the system dynamics to generate the data for learning. Differentiating the value function $V(x)$ along the system trajectory (2.8) and using (2.6) yields

$$\begin{aligned}\dot{V}_j &= \left(\frac{\partial V_j(x)}{\partial x}\right)^T (f + gu_j) + \left(\frac{\partial V_j(x)}{\partial x}\right)^T g(u_{j+1} - u_j) \\ &= -N(x) - 2\varrho^T(u_{j+1}) R(u_{j+1} - u_j) - 2\int_0^{u_j} \varrho^T(s) R \mathrm{d}s\end{aligned}$$

where $\varrho(s) = \lambda \tanh^{-1}(s/\lambda)$. Integrating the above equation yields the off-policy IRL Bellman equation

$$V_j(x(t+\Delta t)) - V_j(x(t)) = \int_t^{t+\Delta t} \Big(- N(x) - 2\varrho^T(u_{j+1}) R(u_{j+1} - u_j) - 2\int_0^{u_j} \varrho^T(s) R \mathrm{d}s \Big) \mathrm{d}\tau \tag{2.9}$$

For an implemented control policy $u(t)$, the off-policy IRL Bellman equation (2.9) can be solved for both value function V_j and updated policy u_{j+1} simultaneously without requiring any knowledge about the system dynamics.

On-policy and off-policy RL algorithms are devised in the literature (see Liu et al., 2020, for a comprehensive review). Their essential difference lies in how the target policy and the behavior policy are implemented. The target policy is what we are learning about, i.e., the optimal control law or the solution to the HJB equation. The target policy can be regarded as the ideal optimal policy. The behavior policy generates the action and behavior, which can be regarded as the policy implemented. The target policy and the behavior policy are the same for on-policy RL algorithms while they are different for off-policy algorithms. Generally, similar to the decision process of human beings, the off-policy algorithms can learn the optimal policies but implement suboptimal policies.

3 Learning the macroscopic traffic dynamics for adaptive optimal perimeter control with integral reinforcement learning

Existing data-driven and feedback traffic control strategies do not consider the heterogeneity of real-time data measurements. Besides, traditional reinforcement learning (RL) methods for traffic control usually converge slowly for lacking data efficiency. Moreover, conventional optimal perimeter control schemes require exact knowledge of the system dynamics and thus they would be fragile to endogenous uncertainties. To handle these challenges, this work will propose an integral reinforcement learning (IRL) based approach to learning the macroscopic traffic dynamics for adaptive optimal perimeter control. This work aims to make the following primary contributions to the transportation literature: (a) A continuous-time control will be developed with discrete gain updates to adapt to the discrete-time sensor data. Different from the conventional RL approaches, the reinforcement interval of the proposed IRL method can vary with respect to the real-time resolution of data measurements. Approximate optimization methods will be carried out to address the curse of dimensionality of the optimal control problem with consideration on the resolution of data measurement. (b) To reduce the sampling complexity and use the available data more efficiently, the experience replay (ER) technique will be introduced to the IRL algorithm. (c) The proposed method will relax the requirement on model calibration in a "model-free" manner that enables robustness against modeling uncertainty and enhances the real-time performance via a data-driven RL algorithm. (d) The convergence of the IRL-based algorithms and the stability of the controlled traffic dynamics will be proven via the Lyapunov theory. The optimal control law will be parameterized and then approximated by neural networks (NN), which can moderate the computational complexity. Both state and input constraints will be considered while no model linearization will be required. Numerical examples and simulation experiments will be presented to verify the effectiveness and efficiency of the proposed method.

3.1 Introduction

The growth of urbanization has led to a significant increase in car usage across metropolitan cities worldwide, resulting in a surge in traffic congestion, accidents, and pollution. Managing the rapidly growing travel demand requires the efficient use of existing infrastructure through appropriate traffic control schemes. The conventional traffic control methods, such as local traffic signal control strategies, focus on link-level strategies, which may not be optimal or might not achieve the stabilization of the system for heterogeneous networks with multiple bottlenecks and heavily directional demand flows. Network-level traffic control strategies should be developed to alleviate network congestion when confronted with conditions.

The perimeter control is one of the most significant applications of MFDs. The perimeter control, which aims to manipulate the transfer flow at the boundaries of the region, is a promising solution to address the spatial dimension challenge in dealing with network-scale traffic congestion. According to the network topology and partitioning, the urban network can be modeled as a single-region (Haddad and Shraiber, 2014), two-region (Zhong et al., 2018b), or multi-region MFD system (Sirmatel and Geroliminis, 2018). Apart from model-based perimeter control schemes, recent research efforts have been dedicated to data-driven perimeter control strategies, e.g., iterative learning control by Ren et al. (2020) and deep reinforcement learning (RL) by Zhou and Gayah (2021).

Considering that traffic data are collected from sensors in a discrete-time manner, we would like to establish a continuous-time control (MFD dynamics) with discrete gain updates (adapting to the sensor data). Generally, the sample time interval of the data collected by a type of sensor is fixed. Thus, sensors of various types deployed in a traffic network would be heterogeneous with different resolutions of data measurements. It would be much better if the reinforcement intervals can be varying with respect to the real-time resolutions of data measurements, i.e., the reinforcement intervals can be selected online to ensure the data-driven RL algorithms do have rich data. Existing works utilizing RL such as Su et al. (2020) and Zhou and Gayah (2021) do not consider such issues. Different from the traditional online RL approaches, the reinforcement intervals of the integral reinforcement learning (IRL) need not be identical and can be adjusted online, which consequently is more suitable for real-world traffic data measurement and allows adaptive online learning to guarantee real-time performance. Based on the idea of IRL, an equivalent Bellman equation, namely the IRL Bellman equation was developed. An online policy iteration algorithm was developed for the optimal

control problem of continuous-time systems via solving the IRL Bellman equation in Vrabie et al. (2009). This adaptive optimal control does not explicitly employ the knowledge on system dynamics, i.e., “model-free".

The actor-critic (AC) structure contributes significantly to the success of RL algorithms. In the AC structure, the actor deploys a control policy to the system or environment, while the critic evaluates the cost induced by the implemented control policy and provides reward signals to the actor. The actor-critic dual neural networks (AC-NN) can be used to circumvent the “curse of dimensionality”. Despite the adaptive learning capability, traditional RL approaches usually converge slowly for lacking data efficiency, which is a major obstacle to real-time applications. Experience replay (ER) technique, also known as concurrent learning, provides a promising approach to improve the efficiency of RL algorithms[1]. The ER technique uses historical and current data simultaneously in a “smart” manner. It has been found that the AC structure can be integrated with the ER technique to improve the data efficiency and convergent speed of RL algorithms (Modares et al., 2014).

To handle the aforementioned challenges, this study makes the following primary contributions to the transportation literature.

- Robustness to heterogeneous data resolutions. Unlike the conventional RL algorithms, the reinforcement intervals of the proposed IRL approach can be selected online to adapt to heterogeneous real-time data resolutions. The introduction of the ER technique to RL algorithms can speed up their convergence when limited real-time data are available due to unexpected longer sample time intervals.

- Data efficiency. In the ER technique, a number of recent samples are stored in a database and are presented repeatedly to the underlying RL algorithm, which enhances their data efficiency. An easy-check rank condition is introduced to verify the data richness requirement and reduce sampling complexity.

- Model-free against modeling uncertainties. It is desirable for the controller to handle the modeling uncertainties. Unlike the previous studies which rely on exact knowledge of the underlying system dynamics, a key advantage of the proposed method is that the exact knowledge of the traffic model is no longer needed. Also, the proposed approach does not rely on the widely used model linearization.

[1] Another benefit of ER is conquering the difficulty arising in the persistently exciting condition for nonlinear systems.

- Incorporating real-time data-driven components for adaptiveness. It is necessary to enable the controller to adapt to the real-time traffic conditions, e.g., traffic incidents. To this end, an online adaptive data-driven perimeter controller is devised.

- Convergence and stability guaranteed. Unlike many existing studies in the transportation literature that use RL algorithms without proof of convergence nor stability, this work guarantees the closed-loop stability of the overall system by leveraging the RL with Lyapunov theory. The input and state constraints are explicitly considered in the proposed IRL algorithms.

This chapter is organized as follows: Section 3.2 discusses the optimal perimeter control problem formulation. Section 3.3 develops a model-free data-driven IRL method for optimal adaptive perimeter control. Then Section 3.4 performs the implementations of the proposed online iterative learning algorithm with NNs. Numerical results are presented in Section 3.5 and a microscopic simulation experiment is provided in Section 3.6. Finally, Section 3.7 provides concluding remarks. For convenience, we summarize the standalone key notation used in this chapter in Table 3.1.

For a better grasp of the logical structure of this chapter, the flow for reasoning is depicted in Figure 3.1. Consider the MFD based system in the affine form as (3.5). In the optimal regulation problem, the objective is to design an optimal perimeter control such that the accumulation state converges to the desired equilibrium by minimizing a cost functional defined by (3.6). This optimal cost function and perimeter control can be derived via solving an equivalent HJB equation (3.12). Note that the solution of (3.12) may be intractable due to its strong nonlinearity. One of the most common methods to resolve this difficulty is the policy iteration method (3.21)-(3.22). However, because of the heterogeneity of real-time data measurements and the lack of complete knowledge on the system dynamics, we cannot apply the policy iteration method directly in practice. Hence, an equivalent formulation of the policy iteration method, namely, the IRL Bellman equation given by (3.37) is established, which can adapt to the time-varying real-time data resolution and does not involve the system dynamics, i.e., "model-free". Finally, based on (3.37), an online iterative learning scheme with experience replay adaptation law (3.48) are established, which can be implemented with the AC-NN framework (3.45) to boost the computational efficiency and approximate the optimal perimeter controller $\tilde{u}^*$.

Table 3.1 List of key notations

Symbol	Meaning
$\mathbb{R}$	The set of all real numbers
$\mathbb{R}^m$	The Euclidean space of all real m-vectors
$\mathbb{R}^{n\times m}$	The space of all $n \times m$ real matrices
T	The transposition symbol
Ω	A compact set of $\mathbb{R}^n$
$C^m(\Omega)$	The class of functions having continuous m-th derivative on Ω
$\|x(t)\|_{L_2}$	The L_2 norm of continuous-time function $x(t)$ while we use $\|x(t)\|$ for brevity
$\succ$	A matrix $R \succ 0$ means that it is positive definite and $\succeq$ denotes positive semi-definite
$\otimes$	Kronecker product: $\forall\ A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{p\times q}$, $A \otimes B \triangleq \begin{bmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{bmatrix}$
$\odot$	Hadamard product: $\forall\ A, B \in \mathbb{R}^{m\times n}$, $A \odot B \triangleq \begin{bmatrix} a_{11}b_{11} & \cdots & a_{1n}b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1}b_{m1} & \cdots & a_{mn}b_{mn} \end{bmatrix}$
Z_i	The directly reachable regions from region i except itself, i.e., $i \notin Z_i$
$n_{ij}(t)$	Number of vehicles in region i with destination to region j at time t
$n_i(t)$	Accumulation or total number of vehicles in region i at time t, and $n_i(t) = n_{ii}(t) + \sum_{j\in Z_i} n_{ij}(t)$
$q_{ij}(t)$	Travel demand defined as a flow in which its origin is region i and destination is region j
$u_{ij}(t)$	Perimeter controllers controlling the ratio of the transfer flow that transfers from region i to region j at time t
$G_i(n_i(t))$	MFD that maps the network accumulation $n_i(t)$ to trip completion rate for region i at time t
α_n	The dimension of the accumulation state n
α_u	the dimension of the perimeter control u
α_q	the dimension of the travel demand q

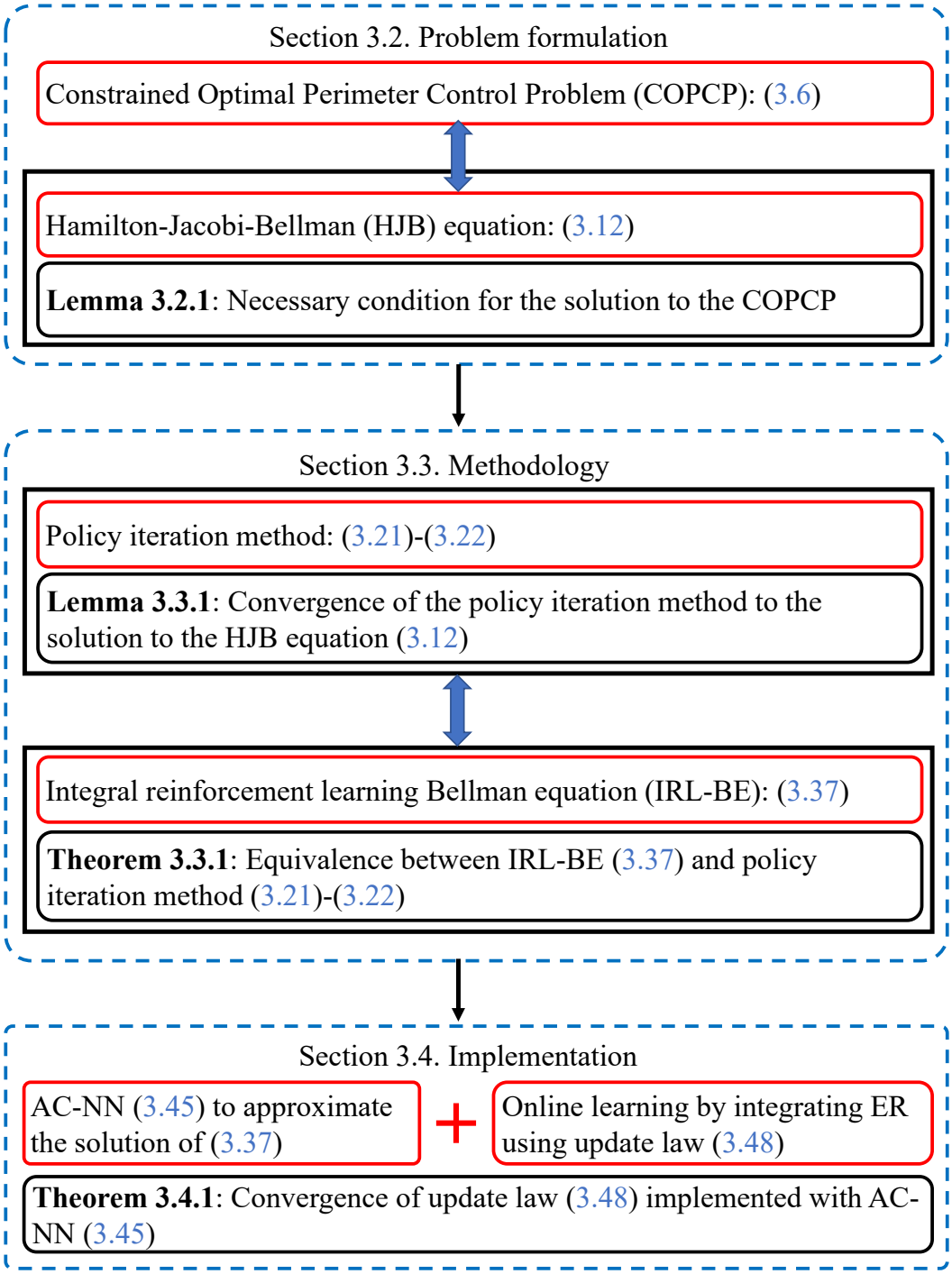


Figure 3.1 Summary of the main results of Study 1

3.2 Problem statement

In this section, we first recapitulate the dynamics for a traffic network modeled by multi-region MFD systems. We then discuss the optimal perimeter control problem formulation.

3.2.1 The multi-region MFD framework

A heterogeneous urban network decomposed into L ($L > 1$) homogeneous subregions wherein each region admits a well-defined MFD and average within-region trip distance is considered in line with Haddad (2015) and Zhong et al. (2018b). Let the state vector be $n(t) \triangleq [n_{11}(t), \ldots, n_{ij}(t), \ldots, n_{LL}(t)]^T \in \mathbb{R}^{\alpha_n}$ and the travel demand vector be $q(t) \triangleq [q_{11}(t), \ldots, q_{ij}(t), \ldots, q_{LL}(t)]^T \in \mathbb{R}^{\alpha_q}$, respectively. The control vector is $u(t) \triangleq [u_{12}(t), \ldots, u_{ij}(t), \ldots, u_{Lj}(t)]^T \in \mathbb{R}^{\alpha_u}$, where $u_{ij}(t)$ controls the ratio of the transfer flow that transfer from region i to j at time t. Note that $n_i(t)$ and $u_{ij}(t)$ are subject to heterogeneous constraints as given by (3.2a)-(3.2b). The dynamic flow conservation equations of the multi-region MFD system are then formulated as follows:

$$\frac{\mathrm{d}n_{ii}(t)}{\mathrm{d}t} = -\frac{n_{ii}(t)}{n_i(t)} G_i(n_i(t)) + \sum_{j \in Z_i} \frac{n_{ji}(t)}{n_j(t)} G_j(n_j(t)) u_{ji}(t) + q_{ii}(t) \tag{3.1a}$$

$$\frac{\mathrm{d}n_{ij}(t)}{\mathrm{d}t} = -\frac{n_{ij}(t)}{n_i(t)} G_i(n_i(t)) u_{ij}(t) + q_{ij}(t) \tag{3.1b}$$

$$n_i(t) = n_{ii}(t) + \sum_{j \in Z_i} n_{ij}(t) \tag{3.1c}$$

subject to

$$0 \leq n_i(t) \leq n_i^{jam} \tag{3.2a}$$

$$0 \leq u_{ij}^{\min} \leq u_{ij}(t) \leq u_{ij}^{\max} \leq 1 \tag{3.2b}$$

where $i = 1, \ldots, L$ and $j \neq i$. The state dynamics (3.1a)-(3.1b) can be written in the following affine form (Su et al., 2020):

$$\dot{n}(t) = F(n(t)) + S(n(t))u(t) \tag{3.3}$$

One significant traffic management purpose is to devise perimeter control $u(t)$ to regulate the cross-boundary flows such that the network accumulations $n(t)$ can converge to the desired equilibrium n^*, i.e., set-point control (Zhong et al., 2018a;

Zhong et al., 2018b). The steady state n^* and the corresponding control input u^* can be solved from the steady-state equations (Haddad and Shraiber, 2014; Zhong et al., 2018b):

$$\frac{\mathrm{d}n_{ii}^*}{\mathrm{d}t} = 0 = -\frac{n_{ii}^*}{\bar{n}_i} G_i(\bar{n}_i) + \sum_{j \in Z_i} \frac{n_{ji}^*}{\bar{n}_j} G_j(\bar{n}_j) u_{ji}^* + q_{ii}^* \tag{3.4a}$$

$$\frac{\mathrm{d}n_{ij}^*}{\mathrm{d}t} = 0 = -\frac{n_{ij}^*}{\bar{n}_i} G_i(\bar{n}_i) u_{ij}^* + q_{ij}^* \tag{3.4b}$$

$$\bar{n}_i = n_{ii}^* + \sum_{j \in Z_i} n_{ij}^* \tag{3.4c}$$

subject to

$$0 \leq \bar{n}_i \leq n_i^{jam}, \quad 0 \leq u_{ij}^{\min} \leq u_{ij}^* \leq u_{ij}^{\max} \leq 1$$

where q_{ii}^* and q_{ij}^* are nominal demand patterns.

It is a common practice to perform a coordinate transformation to reformulate the set-point control problem into a stabilization problem (Zhong et al., 2018a; Zhong et al., 2018b). We define $\tilde{n}(t) = [\tilde{n}_1(t), \ldots, \tilde{n}_{\alpha_n}(t)]^T \in \mathbb{R}^{\alpha_n}$ and $\tilde{u}(t) = [\tilde{u}_1(t), \ldots, \tilde{u}_{\alpha_u}(t)]^T \in \mathbb{R}^{\alpha_u}$ as the new state vector and new control vector, respectively. $\tilde{n} = n - n^*$ denotes the difference between the actual accumulation and the desired steady-state accumulation, while $\tilde{u} = u - u^*$ is the difference between the actual control input and the steady-state control input. After the coordinate transformation, the multi-region MFD system (3.1a)-(3.1c) can be expressed by the following standard affine form:

$$\dot{\tilde{n}}(t) = \mathbf{F}(\tilde{n}(t)) + \mathbf{S}(\tilde{n}(t))\tilde{u}(t) \tag{3.5}$$

Both the state vector and the control vector of system (3.5) are restricted into some compact sets say $\tilde{n}(t) \in \Omega \subset \mathbb{R}^{\alpha_n}$ and $\tilde{u}(t) \in \mathcal{U} \subset \mathbb{R}^{\alpha_u}$, where Ω and $\mathcal{U}$ are the universal sets of $\tilde{n}$ and $\tilde{u}$, respectively. $\mathbf{F}$ and $\mathbf{S}$ are unknown Lipschitz continuous nonlinear functions on $\Omega \subset \mathbb{R}^{\alpha_n}$ containing the origin.

In Appendix A.1, we present the dynamics in the affine form for the two-region and the three-region MFD systems, which are widely investigated in the literature.

3.2.2 Optimal perimeter control of multi-region MFD system

Set-point control and minimizing the total time spent (TTS) are two main objectives considered in the optimal perimeter control problem of MFD systems. In this subsection, we present the formulation of constrained optimal perimeter control problem (COPCP) for multi-region MFD systems considering heterogeneous cross-boundary capacities.

As a special case, Su et al. (2020) showed that the set-point control problem of the two-region MFD system could be modeled as a constrained optimal control problem. We will extend the formulation of set-point constrained optimal perimeter control problem (S-COPCP) for the two-region MFD system to general multi-region MFD systems while considering the heterogeneous cross-boundary capacities. We will also derive the necessary condition for the S-COPCP of multi-region MFD systems. Next, we will present the COPCP for minimizing TTS (T-COPCP) of the multi-region MFD system and derive the optimal perimeter control law for the T-COPCP.

3.2.2.1 Set-point COPCP (S-COPCP) of the Multi-region MFD System

Consider the multi-region MFD system (3.5), find the perimeter controller $\tilde{u}$ to minimize the following objective function:

$$\min_{\tilde{u}} J(\tilde{n}_0) = \int_0^\infty \mathcal{L}(\tilde{n}(t), \tilde{u}(t))\mathrm{d}t \tag{3.6}$$
$$\textbf{subject to (3.5)}$$

where $\tilde{n} \in \Omega \subset \mathbb{R}^{\alpha_n}$ and $\tilde{u} \in \mathcal{U} \subset \mathbb{R}^{\alpha_u}$.

The utility function for the S-COPCP is given by

$$\mathcal{L}(\tilde{n}(t), \tilde{u}(t)) \triangleq N(\tilde{n}(t)) + U(\tilde{u}(t)) \tag{3.7}$$

where $N(\tilde{n})$ represents the cumulative error between the system state and the desired equilibrium, and $U(\tilde{u})$ is the required control effort for unconstrained control case. Generally, $N(\tilde{n}) \triangleq \tilde{n}^T Q \tilde{n} \succeq 0$ with $Q \in \mathbb{R}^{\alpha_n \times \alpha_n}$ and $Q \succ 0$, and $U(\tilde{u}) \triangleq \tilde{u}^T R \tilde{u} \succeq 0$ with $R \in \mathbb{R}^{\alpha_u \times \alpha_u}$ and $R \succ 0$.

Without loss of generality, let $R = \mathrm{diag}(\gamma_1, \ldots, \gamma_{\alpha_u}) \in \mathbb{R}^{\alpha_u \times \alpha_u}$ with $\gamma_{k_u} > 0$, $k_u = 1, \ldots, \alpha_u$. To handle the heterogeneous cross-boundary capacities (3.2b) in the

perimeter controller design, i.e., $\tilde{u}_{k_u}^{\min} \leq \tilde{u}_{k_u} \leq \tilde{u}_{k_u}^{\max}$, inspired by Abu-Khalaf et al. (2006) and Lyshevski (1998), for each $\tilde{u}_{k_u}$ we define the following function:

$$U_{k_u}(\tilde{u}_{k_u}) = 2\underline{v}_{k_u}\gamma_{k_u}\int_{\overline{v}_{k_u}}^{\tilde{u}_{k_u}} \tanh^{-1}\left(\frac{v_{k_u}-\overline{v}_{k_u}}{\underline{v}_{k_u}}\right)\mathrm{d}v_{k_u}$$

where $\overline{v}_{k_u} = \frac{\tilde{u}_{k_u}^{\max}+\tilde{u}_{k_u}^{\min}}{2}$, $\underline{v}_{k_u} = \frac{\tilde{u}_{k_u}^{\max}-\tilde{u}_{k_u}^{\min}}{2}$. Based on the features of inverse hyperbolic tangent function, $U_{k_u}(\tilde{u}_{k_u})$ can be regarded as a penalty function which limits the input $\tilde{u}_{k_u}$ to $(\tilde{u}_{k_u}^{\min}, \tilde{u}_{k_u}^{\max})$. Figure 3.2 shows that the saturation actuator (blue dotted line) developed by the proposed penalty function can well approximate the non-smooth control constraint (green solid line) in a smooth manner. Thus, $U(\tilde{u})$ is defined as

$$\begin{aligned} U(\tilde{u}) &= \sum_{k_u=1}^{\alpha_u} U_{k_u}(\tilde{u}_{k_u}) = \sum_{k_u=1}^{\alpha_u} 2\underline{v}_{k_u}\gamma_{k_u}\int_{\overline{v}_{k_u}}^{\tilde{u}_{k_u}} \tanh^{-1}\left(\frac{v_{k_u}-\overline{v}_{k_u}}{\underline{v}_{k_u}}\right)\mathrm{d}v_{k_u} \\ &= 2\underline{v}^T R\int_{\overline{v}}^{\tilde{u}} \tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v \end{aligned} \tag{3.8}$$

where $\overline{v} \triangleq [\overline{v}_1, \ldots, \overline{v}_{\alpha_u}]^T \in \mathbb{R}^{\alpha_u}$, $\underline{v} \triangleq [\underline{v}_1, \ldots, \underline{v}_{\alpha_u}]^T \in \mathbb{R}^{\alpha_u}$.

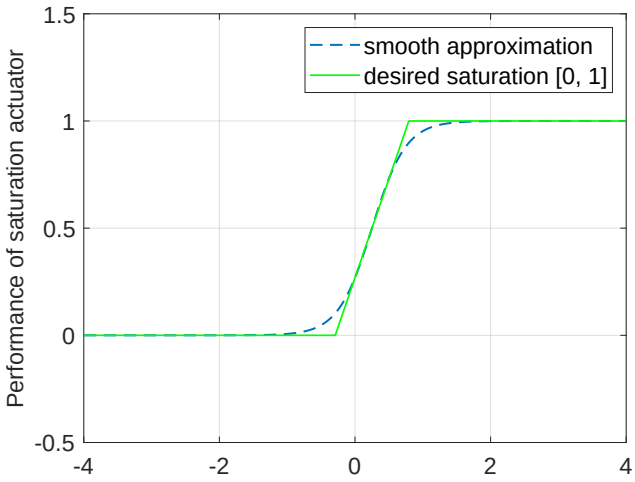


Figure 3.2 Performance of the proposed saturation actuator

The value function $V : \mathbb{R}^{\alpha_n} \to \mathbb{R}$ is defined as

$$\begin{aligned} V(\tilde{n}(t)) &= \int_t^\infty \mathcal{L}(\tilde{n}(\tau), \tilde{u}(\tau))\mathrm{d}\tau \\ &\equiv \int_t^\infty (N(\tilde{n}(\tau)) + U(\tilde{u}(\tau)))\mathrm{d}\tau \\ &= \int_t^\infty \left(\tilde{n}^T(\tau)Q\tilde{n}(\tau) + 2\underline{v}^T R\int_{\overline{v}}^{\tilde{u}(\tau)} \tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v\right)\mathrm{d}\tau \end{aligned} \tag{3.9}$$

Following the development in Section 2.4, we can obtain the following Bellman equation

$$\mathcal{L}(\tilde{n}(t), \tilde{u}(t)) + \left(\frac{\partial V}{\partial \tilde{n}}\right)^T (\mathbf{F}(\tilde{n}) + \mathbf{S}(\tilde{n})\tilde{u}) = 0 \tag{3.10}$$

Now we can present the necessary condition for the solution to the S-COPCP of the multi-region MFD system.

Lemma 3.2.1 Suppose that V^* is the optimal value function for the S-COPCP of the multi-region MFD system. It follows that

1. the constrained optimal perimeter control is given by

$$\tilde{u}^* = -\underline{v} \odot \tanh(D^*) + \overline{v}, \text{ with } D^* = \frac{1}{2\underline{v}} \odot \left(R^{-1}\mathbf{S}^T \frac{\partial V^*}{\partial \tilde{n}}\right) \tag{3.11}$$

   where $D^* = [D_1^*, \ldots, D_{\alpha_u}^*]^T \in \mathbb{R}^{\alpha_u}$ is the unconstrained optimal control input;

2. the necessary condition for the solution to the S-COPCP, i.e., (V^*, D^*) should satisfy the following equation:

$$0 = \tilde{n}^T Q \tilde{n} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{F} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S}\overline{v} + \underline{v}^{2T} R \ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \tag{3.12}$$

   where $\mathbf{1}_{\alpha_u} \in \mathbb{R}^{\alpha_u}$ is a column vector with each element equal to 1.

∎

Proof 3.2.1 1) Assume that V^* is the optimal value function which satisfies (3.10), then it yields the following HJB equation

$$\begin{aligned}
H\left(\tilde{n}, \tilde{u}, \frac{\partial V^*}{\partial \tilde{n}}\right) &= \min_{\tilde{u}} \left[\mathcal{L}(\tilde{n}, \tilde{u}) + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T (\mathbf{F}(\tilde{n}) + \mathbf{S}(\tilde{n})\tilde{u})\right] \\
&= \min_{\tilde{u}} \left[\tilde{n}^T Q \tilde{n} + 2\underline{v}^T R \int_{\overline{v}}^{\tilde{u}} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \right. \\
&\quad \left. + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T (\mathbf{F}(\tilde{n}) + \mathbf{S}(\tilde{n})\tilde{u})\right] = 0 \\
&= \min_{\tilde{u}} \left[\tilde{n}^T Q \tilde{n} + \sum_{k_u=1}^{\alpha_u} 2\underline{v}_{k_u} \gamma_{k_u} \int_{\overline{v}_{k_u}}^{\tilde{u}_{k_u}} \tanh^{-1}\left(\frac{v_{k_u} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) \mathrm{d}v_{k_u} \right. \\
&\quad \left. + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{F}(\tilde{n}) + \sum_{k_u=1}^{\alpha_u} \sum_{k_n=1}^{\alpha_n} \frac{\partial V^*}{\partial \tilde{n}_{k_n}} S_{k_n,k_u} \tilde{u}_{k_u}\right]
\end{aligned} \tag{3.13}$$

where S_{k_n,k_u} denotes the k_n-th element of the k_u-th column of $\mathbf{S}$.

The optimal constrained perimeter control $\tilde{u}_{k_u}^*$ is calculated by applying the stationary (optimal) condition $\partial H/\partial \tilde{u}_{k_u}^* = 0$, i.e.,

$$\frac{\partial H}{\partial \tilde{u}_{k_u}^*} = 2\underline{v}_{k_u}\gamma_{k_u}\tanh^{-1}\left(\frac{\tilde{u}_{k_u}^* - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) + \sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u} = 0$$

Then it follows that

$$\begin{aligned}
&\tanh^{-1}\left(\frac{\tilde{u}_{k_u}^* - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) = -\frac{1}{2\underline{v}_{k_u}\gamma_{k_u}}\sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u} \\
\Rightarrow\quad &\frac{\tilde{u}_{k_u}^* - \overline{v}_{k_u}}{\underline{v}_{k_u}} = \tanh\left(-\frac{1}{2\underline{v}_{k_u}\gamma_{k_u}}\sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u}\right) \\
&\qquad\qquad = -\tanh\left(\frac{1}{2\underline{v}_{k_u}\gamma_{k_u}}\sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u}\right) \\
\Rightarrow\quad &\tilde{u}_{k_u}^* = -\underline{v}_{k_u}\tanh\left(\frac{1}{2\underline{v}_{k_u}\gamma_{k_u}}\sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u}\right) + \overline{v}_{k_u}
\end{aligned}$$

Let $D_{k_u}^* = \frac{1}{2\underline{v}_{k_u}\gamma_{k_u}}\sum_{k_n=1}^{\alpha_n}\frac{\partial V^*}{\partial \tilde{n}_{k_n}}S_{k_n,k_u}$ be the k_u-th unconstrained optimal control input. The optimal control $\tilde{u}_{k_u}^*$ is obtained as

$$\tilde{u}_{k_u}^* = -\underline{v}_{k_u}\tanh(D_{k_u}^*) + \overline{v}_{k_u} \tag{3.14}$$

Therefore, the constrained optimal perimeter control is given by

$$\tilde{u}^* = -\underline{v}\odot\tanh(D^*) + \overline{v},\ \text{with}\ D^* = \frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^T\frac{\partial V^*}{\partial \tilde{n}}\right) \tag{3.15}$$

2) Let $\hat{v} = \frac{1}{\underline{v}}\odot(v - \overline{v})$. Substituting (3.15) into (3.8), we have

$$\begin{aligned}
U(\tilde{u}^*) &= 2\underline{v}^T\odot\underline{v}^T R\int_0^{\frac{1}{\underline{v}}\odot(\tilde{u}^* - \overline{v})}\tanh^{-1}(\hat{v})\mathrm{d}\hat{v} = 2\underline{v}^{2T}R\int_0^{-\tanh(D^*)}\tanh^{-1}(\hat{v})\mathrm{d}\hat{v} \\
&= 2\underline{v}^{2T}R\cdot\left(\hat{v}\odot\tanh^{-1}(\hat{v}) + \frac{1}{2}\ln(\mathbf{1}_{\alpha_u} - \hat{v}^2)\right)\Big|_0^{-\tanh(D^*)} \\
&= 2\underline{v}^{2T}R\left(\tanh(D^*)\odot D^* + \frac{1}{2}\ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*))\right) \\
&= 2\underline{v}^{2T}R(D^*\odot\tanh(D^*)) + \underline{v}^{2T}R\ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \\
&= 2\underline{v}^{2T}R\left(\frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^T\frac{\partial V^*}{\partial \tilde{n}}\right)\odot\tanh(D^*)\right) + \underline{v}^{2T}R\ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \\
&= \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T\mathbf{S}(\tilde{n})(\underline{v}\odot\tanh(D^*)) + \underline{v}^{2T}R\ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*))
\end{aligned} \tag{3.16}$$

Then substituting (3.15)-(3.16) into (3.13), the HJB equation can further be expressed by

$$
\begin{aligned}
0 &= \tilde{n}^T Q \tilde{n} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}^*) + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S}(\underline{v} \odot \tanh(D^*)) \\
&\quad + \underline{v}^{2T} R \ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \\
&= \tilde{n}^T Q \tilde{n} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{F} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^* + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S} \cdot (\underline{v} \odot \tanh(D^*)) \\
&\quad + \underline{v}^{2T} R \ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \\
&= \tilde{n}^T Q \tilde{n} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{F} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S} \cdot (-\underline{v} \odot \tanh(D^*) + \overline{v}) \\
&\quad + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S} \cdot (\underline{v} \odot \tanh(D^*)) + \underline{v}^{2T} R \ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*)) \\
&= \tilde{n}^T Q \tilde{n} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{F} + \left(\frac{\partial V^*}{\partial \tilde{n}}\right)^T \mathbf{S}\overline{v} + \underline{v}^{2T} R \ln(\mathbf{1}_{\alpha_u} - \tanh^2(D^*))
\end{aligned}
\tag{3.17}
$$

That is, if (V^*, D^*) is the solution to the COPCP, (V^*, D^*) should satisfy (3.17). This completes the proof. ∎

Note that (3.12) is the HJB equation for the S-COPCP of the multi-region MFD system. To find the optimal feedback control policy $\tilde{u}^*$ that minimizes (3.6), it is necessary to solve the HJB equation (3.12) for the value function V^* and unconstrained control D^*, and then substitute them into (3.11). However, the HJB equation (3.12) is extremely difficult to solve due to its strong nonlinearity. In the subsequent sections, a data-driven online algorithm will be presented to find an approximate solution to (3.12) without requiring the system dynamics.

3.2.2.2 MinTTS COPCP (T-COPCP) of the Multi-region MFD System

Another commonly adopted perimeter control objective is to minimize the total time spent (TTS) during the simulation period:

$$
\begin{aligned}
&\min_{u} \bar{J}(n_0) = \int_0^{t_f} \left(\sum_{i=1}^{L} n_i(t) + \bar{\lambda} \|u(t)\| \right) \mathrm{d}t \\
&\textbf{subject to } (3.3)
\end{aligned}
\tag{3.18}
$$

where n and u are constrained by (3.2a)-(3.2b). The last term of the value function (3.18) is to damp oscillation of the control input, where $\bar{\lambda}$ is a positive constant to

adjust the weight of the norm. Different from S-COPCP, the formulation of T-COPCP does not require coordinate transformation.

The Hamiltonian function can be formulated as:

$$\bar{H}\left(n, u, \frac{\partial \bar{V}}{\partial n}\right) = \sum_{i=1}^{L} n_i(t) + \bar{\lambda}\|u(t)\| + \left(\frac{\partial \bar{V}}{\partial n}\right)^T \cdot (F(n) + S(n)u) \tag{3.19}$$

where $\bar{V}(n(t)) = \int_t^{t_f} \left(\sum_{i=1}^{L} n_i(\tau) + \bar{\lambda}\|u(\tau)\|\right) \mathrm{d}\tau$.

Similar to the deduction of Lemma 3.2.1, the corresponding constrained optimal control law is

$$u^* = -\underline{v}' \odot \tanh(\bar{D}^*) + \overline{v}', \text{ with } \bar{D}^* = \frac{1}{\underline{v}'} \odot \left(\frac{1}{2\bar{\lambda}} S^T \frac{\partial \bar{V}^*}{\partial n}\right) \tag{3.20}$$

where $\overline{v}' \triangleq [\overline{v}'_1, \ldots, \overline{v}'_{\alpha_u}]^T \in \mathbb{R}^{\alpha_u}$, $\underline{v}' \triangleq [\underline{v}'_1, \ldots, \underline{v}'_{\alpha_u}]^T \in \mathbb{R}^{\alpha_u}$ with $\overline{v}'_{k_u} = \frac{u_{k_u}^{\max} + u_{k_u}^{\min}}{2}$, $\underline{v}'_{k_u} = \frac{u_{k_u}^{\max} - u_{k_u}^{\min}}{2}$.

Note that (3.3) and (3.5) are in the same affine form. The theoretical results developed for system (3.5) (regarding S-COPCP) can be applied to system (3.3) (regarding T-COPCP).

3.3 Data-driven IRL based adaptive optimal perimeter control

Parallel to the development in Section 2.4, to relax the requirement of system knowledge and consider sensor data measurements, we will establish an equivalent formulation of the HJB equation (3.12) that does not involve the system dynamics. Towards this, in this section a recapitulation of the policy iteration method for solving (3.12) will be presented. Based on the policy iteration method, a data-driven model-free adaptive optimal perimeter controller, which considers the heterogeneous discrete-time sensor data, is developed through the lens of the integral reinforcement learning (IRL).

Note that it is difficult to give an analytical solution to (3.12) due to the strong nonlinearity. Policy iteration is one of the most common methods to resolve this difficulty. The policy iteration method considering heterogeneous cross-boundary capacities is as follows:

1. (Policy evaluation) Given an initial admissible control policy $\tilde{u}^0(\tilde{n})$, find $V^k(\tilde{n})$ successively approximated by solving the following equation with $V^k(0)=0$

$$\mathcal{L}(\tilde{n},\tilde{u}^k)+\left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T(\mathbf{F}+\mathbf{S}\tilde{u}^k)=0, k=0,1,\ldots \tag{3.21}$$

2. (Policy improvement) Update the control policy simultaneously by

$$\tilde{u}^{k+1}(\tilde{n})=-\underline{v}\odot\tanh(D^{k+1})+\overline{v}, D^{k+1}=\frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^T\frac{\partial V^{k+1}}{\partial \tilde{n}}\right) \tag{3.22}$$

where k is the iterative index. The policy evaluation is implemented to update the iterative value function that satisfies the Bellman equation (3.10). Then based on value iteration, the policy improvement is implemented to obtain the iterative control law sequence that minimizes the total cost in each period. From the policy improvement, we can always find another control law sequence that is better, or at least no worse. The following lemma demonstrates the convergence of V^k and $\tilde{u}^k$ (i.e., D^k) by iterating (3.21)-(3.22) to the optimal value function V^* and optimal perimeter control $\tilde{u}^*$ (i.e., D^*).

Lemma 3.3.1 Let $V^k(\tilde{n})\in C^1(\Omega)$ on Ω where $V^k(\tilde{n})\geq 0$, $V^k(0)=0$ and $\tilde{u}^k(\tilde{n})$ is admissible to (3.5), $k=0,1,\ldots$. If $(V^{k+1}(\tilde{n}),\tilde{u}^k(\tilde{n}))$ and $(V^{k+2}(\tilde{n}),\tilde{u}^{k+1}(\tilde{n}))$ both satisfy (3.10) with the boundary condition $V^{k+1}(0)=0$, $V^{k+2}(0)=0$, then

1. the obtained control policies $\tilde{u}^{k+1}(\tilde{n})$ in (3.22) are admissible for (3.5) on Ω;
2. $V^*(\tilde{n})\leq V^{k+2}(\tilde{n})\leq V^{k+1}(\tilde{n}),\forall\tilde{n}\in\Omega$;
3. $\lim_{k\to\infty}V^k(\tilde{n})=V^*(\tilde{n})$;
4. $\lim_{k\to\infty}\tilde{u}^k(\tilde{n})=\tilde{u}^*(\tilde{n})$.

∎

To prove this lemma, we need the following Lemma 3.3.2.

Lemma 3.3.2 For a monotonically increasing odd function $\varrho(x)$, we have

1. $\varrho(x)\cdot(y-x)-\int_x^y\varrho(s)\mathrm{d}s\leq 0,\forall x,y$;
2. $\varrho(x)\cdot(y-x)-\int_0^y\varrho(s)\mathrm{d}s\leq 0,\forall x,y$.

∎

Proof 3.3.1 1) Without loss of generality, we assume that $y \geq x$.

Note that $\varrho(x)$ is monotonically increasing and odd. Thus,

$$\varrho(x)\begin{cases} <0, & x<0 \\ =0, & x=0 \\ >0, & x>0 \end{cases}, \ \varrho(-x)=-\varrho(x)$$

Then we have $\int_x^y \varrho(s)\mathrm{d}s \geq 0$. Moreover, $\int_x^y \varrho(s)\mathrm{d}s = 0$ if and only if $y = x$.

Let $\varphi(y) = \varrho(x)\cdot(y-x) - \int_x^y \varrho(s)\mathrm{d}s$, then

$$\frac{\mathrm{d}\varphi}{\mathrm{d}y} = \varrho(x) - \varrho(y)\begin{cases} >0, & y<x \\ =0, & y=x \\ <0, & y>x \end{cases}$$

The extreme value of $\varphi(y)$ is obtained at $y = x$, i.e., $\varphi(x) = 0 - \int_x^x \varrho(s)\mathrm{d}s \leq 0$. $\varphi(y) = 0$ if and only if $y = 0$. Thus, we have

$$\varrho(x)\cdot(y-x) - \int_x^y \varrho(s)\mathrm{d}s \leq 0$$

and $\varrho(x)\cdot(y-x) - \int_x^y \varrho(s)\mathrm{d}s = 0$ if and only if $y = x$.

2) The left side of the second part of Lemma 3.3.2 can be written as

$$\varrho(x)\cdot(y-x) - \int_0^y \varrho(s)\mathrm{d}s = \varrho(x)\cdot(y-0) - \int_0^y \varrho(s)\mathrm{d}s - \varrho(x)x \tag{3.23}$$

Based on the first part, we have $\varrho(x)\cdot(y-0) - \int_0^y \varrho(s)\mathrm{d}s \leq 0$. Because $\varrho(x)$ and x are monotonically increasing and odd, one has $\varrho(x)x \geq 0$, i.e., $-\varrho(x)x \leq 0$. Thus, we have

$$\varrho(x)\cdot(y-x) - \int_0^y \varrho(s)\mathrm{d}s \leq 0$$

Moreover, $\varrho(x)\cdot(y-x) - \int_0^y \varrho(s)\mathrm{d}s = 0$ if and only if $y = x = 0$. This completes the proof. ∎

Now we present the proof of Lemma 3.3.1.

Proof 3.3.2 1) Taking the derivative of V^{k+1} along the system $\mathbf{F} + \mathbf{S}\tilde{u}^{k+1}$ trajectory, we have

$$\dot{V}^{k+1} = \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{F} + \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} \tag{3.24}$$

Based on (3.9) and (3.21), we have

$$
\begin{aligned}
&N(\tilde{n})+2\underline{v}^{T}R\int_{\overline{v}}^{\tilde{u}^{k}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v+\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{F}+\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{S}\tilde{u}^{k}=0\\
\Rightarrow&\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{F}=-\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{S}\tilde{u}^{k}-N(\tilde{n})-2\underline{v}^{T}R\int_{\overline{v}}^{\tilde{u}^{k}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v
\end{aligned}
\tag{3.25}
$$

From (3.22), one has

$$
\begin{aligned}
&\tilde{u}^{k+1}-\overline{v}=-\underline{v}\odot\tanh\left(\frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^{T}\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)\right)\\
\Rightarrow&\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})=-\tanh\left(\frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^{T}\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)\right)\\
\Rightarrow&\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)=-\frac{1}{2\underline{v}}\odot\left(R^{-1}\mathbf{S}^{T}\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)\\
\Rightarrow&-2\underline{v}\odot\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)=R^{-1}\mathbf{S}^{T}\frac{\partial V^{k+1}}{\partial\tilde{n}}\\
\Rightarrow&\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{S}=-2\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R
\end{aligned}
\tag{3.26}
$$

Substitute (3.25)-(3.26) into (3.24), it follows that

$$
\begin{aligned}
\dot{V}^{k+1}=&-N(\tilde{n})-2\underline{v}^{T}R\int_{\overline{v}}^{\tilde{u}^{k}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v-\left(\frac{\partial V^{k+1}}{\partial\tilde{n}}\right)^{T}\mathbf{S}\tilde{u}^{k}\\
&-2\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R\tilde{u}^{k+1}\\
=&-N(\tilde{n})-2\underline{v}^{T}R\int_{\overline{v}}^{\tilde{u}^{k}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v\\
&+2\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R\tilde{u}^{k}\\
&-2\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R\tilde{u}^{k+1}\\
=&-N(\tilde{n})-2\Bigg(\underline{v}^{T}R\int_{\overline{v}}^{\tilde{u}^{k}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v\\
&-\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R\tilde{u}^{k}\\
&+\underline{v}^{T}\odot\tanh^{-T}\left(\frac{1}{\underline{v}}\odot(\tilde{u}^{k+1}-\overline{v})\right)R\tilde{u}^{k+1}\Bigg)
\end{aligned}
$$

Then we have

$$\begin{aligned}\dot{V}^{k+1} &= -N(\tilde{n}) + 2\Bigg(\underline{v}^T \odot \tanh^{-T}\left(\frac{1}{\underline{v}} \odot (\tilde{u}^{k+1} - \overline{v})\right) R(\tilde{u}^k - \tilde{u}^{k+1}) \\ &\quad - \underline{v}^T R \int_{\overline{v}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v\Bigg) \\ &= -N(\tilde{n}) + 2\Bigg(\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u}\gamma_{k_u} \tanh^{-1}\left(\frac{\tilde{u}_{k_u}^{k+1} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right)(\tilde{u}_{k_u}^k - \tilde{u}_{k_u}^{k+1}) \\ &\quad - \sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u}\gamma_{k_u} \int_{\overline{v}_{k_u}}^{\tilde{u}_{k_u}^k} \tanh^{-1}\left(\frac{v_{k_u} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) \mathrm{d}v_{k_u}\Bigg) \\ &= -N(\tilde{n}) + 2\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u}\gamma_{k_u}\Bigg(\tanh^{-1}\left(\frac{\tilde{u}_{k_u}^{k+1} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right)(\tilde{u}_{k_u}^k - \tilde{u}_{k_u}^{k+1}) \\ &\quad - \int_{\overline{v}_{k_u}}^{\tilde{u}_{k_u}^k} \tanh^{-1}\left(\frac{v_{k_u} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) \mathrm{d}v_{k_u}\Bigg)\end{aligned} \tag{3.27}$$

Let $\varrho(x) \triangleq \tanh^{-1}(x/\underline{v}_{k_u})$ for $\forall x \in \mathbb{R}$, $s_{k_u}^k = \tilde{u}_{k_u}^k - \overline{v}_{k_u}$ and $\hat{v}_{k_u} = v_{k_u} - \overline{v}_{k_u}$. Then (3.27) is rewritten as follows

$$\dot{V}^{k+1} = -N(\tilde{n}) + 2\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u}\gamma_{k_u}\left(\varrho(s_{k_u}^{k+1})(s_{k_u}^k - s_{k_u}^{k+1}) - \int_0^{s_{k_u}^k} \varrho(\hat{v}_{k_u})\mathrm{d}\hat{v}_{k_u}\right)$$

Since $\tanh^{-1}(\cdot)$ is a monotonically increasing odd function, $\varrho(x)$ is monotonically increasing and odd. By Lemma 3.3.2, the following inequality holds

$$\varrho(s_{k_u}^{k+1})(s_{k_u}^k - s_{k_u}^{k+1}) - \int_0^{s_{k_u}^k} \varrho(\hat{v}_{k_u})\mathrm{d}\hat{v}_{k_u} \leq 0$$

Recall that $\underline{v} > 0$ and $\gamma > 0$, we have $\dot{V}^{k+1} \leq 0$ and $V^{k+1}(\tilde{n})$ is a Lyapunov function for $\tilde{u}^{k+1}$ on Ω.

Because the nonlinear function $\mathbf{S}$ is continuous and $V^{k+1}(0) = 0$, $\tilde{u}^{k+1}(\tilde{n}) \in C^1(\Omega)$ and $\tilde{u}^{k+1}(0) = 0$, i.e., the obtained control policies $\tilde{u}^{k+1}(\tilde{n})$ in (3.22) are admissible as per Definition 2.4.1 for system (3.5) on Ω.

2) First, we prove that $V^{k+2}(\tilde{n}(t)) \leq V^{k+1}(\tilde{n}(t))$.

Considering $V(\tilde{n})$ defined in (3.9) along the system $\mathbf{F} + \mathbf{S}\tilde{u}^{k+1}$ trajectory, we have

$$V^{k+2}(\tilde{n}) - V^{k+1}(\tilde{n}) = -\int_t^\infty \left(\frac{\partial(V^{k+2} - V^{k+1})^T}{\partial \tilde{n}}(\mathbf{F} + \mathbf{S}\tilde{u}^{k+1})\right) \mathrm{d}\tau \tag{3.28}$$

Since $(V^{k+1}(\tilde{n}), \tilde{u}^k(\tilde{n}))$ and $(V^{k+2}(\tilde{n}), \tilde{u}^{k+1}(\tilde{n}))$ both satisfy (3.10), we can obtain

$$\left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{F} = -N(\tilde{n}) - 2\underline{v}^T R \int_{\overline{v}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^k \tag{3.29a}$$

$$\left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{F} = -N(\tilde{n}) - 2\underline{v}^T R \int_{\overline{v}}^{\tilde{u}^{k+1}} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v - \left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} \tag{3.29b}$$

Substituting (3.29a)-(3.29b) into (3.28), we get

$$\begin{aligned}
V^{k+2}(\tilde{n}) - V^{k+1}(\tilde{n}) = & -\int_t^\infty \Bigg(\left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{F} - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{F} \\
& + \left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} \Bigg) \mathrm{d}\tau \\
= & -\int_t^\infty \Bigg(-N(\tilde{n}) - 2\underline{v}^T R \int_{\overline{v}}^{\tilde{u}^{k+1}} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \\
& - \left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} + N(\tilde{n}) \\
& + 2\underline{v}^T R \int_{\overline{v}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v + \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^k \\
& + \left(\frac{\partial V^{k+2}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}\tilde{u}^{k+1} \Bigg) \mathrm{d}\tau \\
= & -\int_t^\infty \Bigg(2\underline{v}^T R \int_{\tilde{u}^{k+1}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \\
& + \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}(\tilde{u}^k - \tilde{u}^{k+1}) \Bigg) \mathrm{d}\tau
\end{aligned} \tag{3.30}$$

Substituting (3.26) into (3.30), one obtains

$$
\begin{aligned}
V^{k+2}(\tilde{n}) - V^{k+1}(\tilde{n}) = & -\int_t^{\infty} \Bigg(2\underline{v}^T R \int_{\tilde{u}^{k+1}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \\
& - 2\underline{v}^T \odot \tanh^{-T}\left(\frac{1}{\underline{v}} \odot (\tilde{u}^{k+1} - \overline{v})\right) R(\tilde{u}^k - \tilde{u}^{k+1}) \Bigg) \mathrm{d}\tau \\
= & \, 2\int_t^{\infty} \Bigg(\underline{v}^T \odot \tanh^{-T}\left(\frac{1}{\underline{v}} \odot (\tilde{u}^{k+1} - \overline{v})\right) R(\tilde{u}^k - \tilde{u}^{k+1}) \\
& - \underline{v}^T R \int_{\tilde{u}^{k+1}}^{\tilde{u}^k} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \Bigg) \mathrm{d}\tau \\
= & \, 2\int_t^{\infty} \Bigg(\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u} \gamma_{k_u} \tanh^{-1}\left(\frac{\tilde{u}_{k_u}^{k+1} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) (\tilde{u}_{k_u}^k - \tilde{u}_{k_u}^{k+1}) \\
& - \sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u} \gamma_{k_u} \int_{\tilde{u}_{k_u}^{k+1}}^{\tilde{u}_{k_u}^k} \tanh^{-1}\left(\frac{v_{k_u} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) \mathrm{d}v_{k_u} \Bigg) \mathrm{d}\tau \\
= & \, 2\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u} \gamma_{k_u} \int_t^{\infty} \Bigg(\tanh^{-1}\left(\frac{\tilde{u}_{k_u}^{k+1} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) (\tilde{u}_{k_u}^k - \tilde{u}_{k_u}^{k+1}) \\
& - \int_{\tilde{u}_{k_u}^{k+1}}^{\tilde{u}_{k_u}^k} \tanh^{-1}\left(\frac{v_{k_u} - \overline{v}_{k_u}}{\underline{v}_{k_u}}\right) \mathrm{d}v_{k_u} \Bigg) \mathrm{d}\tau \\
= & \, 2\sum_{k_u=1}^{\alpha_u} \underline{v}_{k_u} \gamma_{k_u} \int_t^{\infty} \left(\varrho(s_{k_u}^{k+1})(s_{k_u}^k - s_{k_u}^{k+1}) - \int_{s_{k_u}^{k+1}}^{s_{k_u}^k} \varrho(\hat{v}_{k_u}) \mathrm{d}\hat{v}_{k_u} \right) \mathrm{d}\tau
\end{aligned}
\tag{3.31}
$$

By Lemma 3.3.2, the following inequality holds

$$
\varrho(s_{k_u}^{k+1})(s_{k_u}^k - s_{k_u}^{k+1}) - \int_{s_{k_u}^{k+1}}^{s_{k_u}^k} \varrho(\hat{v}_{k_u}) \mathrm{d}\hat{v}_{k_u} \leq 0
$$

Thus, the right side of (3.31) is negative semi-definite. It follows that $V^{k+2}(\tilde{n}) - V^{k+1}(\tilde{n}) \leq 0$, i.e., $V^{k+2}(\tilde{n}) \leq V^{k+1}(\tilde{n})$.

Next, we prove that $V^*(\tilde{n}) \leq V^{k+2}(\tilde{n})$.

Since $(V^*, \tilde{u}^*)$, which satisfies (3.10), is the optimal solution to the COPCP defined by (3.5)-(3.6), for $\forall \tilde{n} \in \Omega$, we have $\min_{\tilde{u}} V(\tilde{n}) = V^*(\tilde{n})$ and $\tilde{u}^* = \arg\min_{\tilde{u}} \int_t^{\infty} \mathcal{L}(\tilde{n}(\tau), \tilde{u}(\tau)) \mathrm{d}\tau = \arg\min_{\tilde{u}} V(\tilde{n}(t))$.

Suppose $\exists\, k$ such that $V^{k+2}(\tilde{n})$, which also satisfies (3.10), is smaller than $V^*(\tilde{n})$, i.e., $V^{k+2}(\tilde{n}) < V^*(\tilde{n})$. This means that $V^*(\tilde{n})$ is not the optimal solution to the COPCP. By contradiction, we can deduce that $V^*(\tilde{n}) \leq V^{k+2}(\tilde{n})$.

3) It follows from the second part of Lemma 3.3.1 that $\{V^k\}_{k=0}^{\infty}$ is a monotonically decreasing sequence with the lower bounded $V^*(\tilde{n})$, then V^k converges pointwise to V^{∞}. Because of the uniqueness of $V(\tilde{n})$ with $\tilde{n} \in \Omega$ (Lewis et al., 2012; Lyashevskiy, 1996), we can get that $V^{\infty} = V^*$, which means that $\lim_{k\to\infty} V^k(\tilde{n}) = V^*(\tilde{n})$.

4) Since $\lim_{k\to\infty} V^k(\tilde{n}) = V^*(\tilde{n})$, according to (3.22), it can be deduced that $\lim_{k\to\infty} \tilde{u}^k(\tilde{n}) = \tilde{u}^*(\tilde{n})$. The proof is completed. ∎

Lemma 3.3.1 indicates that using the policy iteration method, $(V^k, \tilde{u}^k)$ (i.e., (V^k, D^k)) can approximate the optimal solution $(V^*, \tilde{u}^*)$ (i.e., (V^*, D^*)) to the HJB equation (3.12). However, (3.21)-(3.22) requires identification of the MFD dynamics $\mathbf{F}$ and $\mathbf{S}$. To enable a data-driven method without requiring calibration of the MFD dynamics (3.5), the main idea is to get rid of the system dynamics in the HJB equation (3.12). We can adopt an off-policy IRL algorithm that the control implemented can be different from the optimal control (3.22). Towards this, we rewrite the traffic dynamics (3.5) as

$$\dot{\tilde{n}} = \mathbf{F}(\tilde{n}) + \mathbf{S}(\tilde{n})\tilde{u}^k + \mathbf{S}(\tilde{n})(\tilde{u} - \tilde{u}^k) \tag{3.32}$$

where $\tilde{u}^k$ is the policy to be updated and $\tilde{u}$ is the behavior policy that is actually implemented to the system dynamics to generate the data for learning.

Remark 3.3.1 On-policy and off-policy are two important RL methods. The policy iteration (3.21)-(3.22) can be regarded as a class of on-policy methods. When using the on-policy methods, the learned control policy should be applied to generate data simultaneously even before it converges. Although the on-policy methods can provide nearly unbiased estimates of the policy gradient, they (e.g., Sarsa) are usually data-intensive and their learning process is time-consuming. Furthermore, the data usage of on-policy learning methods is low because the samples generated previously would be discarded along with each policy changes. Thus, the implementation of on-policy learning method is generally difficult. Unlike the on-policy learning, the off-policy learning evaluates the target policy when executing other behavior policies.

There are several practical reasons that the implemented control can differ from the optimal control to be learned. As discussed in Zhong et al. (2018b), the traffic managers may have difficulties in calibrating a detailed functional form and its steady state for the time-varying travel demand and the MFD dynamics. Thus, they may not be able to implement the model-based optimal control (that is a 'miracle' to the manager) in the learning process. On the other hand, the traffic

managers definitely have a preferable network condition (or state) for management purposes. The implemented control $\tilde{u}$ can be arbitrary enables the traffic managers to enforce their preference as a 'priori' for traffic management purposes. This can be regarded as a superiority of the proposed off-policy IRL-based learning algorithms. This implemented control can stimulate the network dynamics so that the learning algorithms can observe the evolution of traffic states and the network performance to adjust the adaptive optimal control iteratively. ∎

Now we derive the IRL Bellman equation that does not involve the system dynamics $\mathbf{F}$ and $\mathbf{S}$, i.e., "model-free". The time derivative of $V^{k+1}(\tilde{n}(t))$ for the $\{k+1\}$-th iteration equals

$$\frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} = \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}) \tag{3.33}$$

Subtracting (3.21) from (3.33), we have

$$\begin{aligned} \frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} &= \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}) - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \\ &= \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}(\tilde{u} - \tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \end{aligned} \tag{3.34}$$

From the second equation of (3.22), we have

$$\begin{aligned} & 2\underline{v} \odot D^{k+1} = R^{-1}\mathbf{S}^T \frac{\partial V^{k+1}}{\partial \tilde{n}} \\ \Rightarrow & \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}R^{-1} = (2\underline{v} \odot D^{k+1})^T \\ \Rightarrow & \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S} = 2(\underline{v} \odot D^{k+1})^T R \end{aligned} \tag{3.35}$$

Substituting (3.35) into (3.34) yields

$$\frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} = 2(\underline{v} \odot D^{k+1})^T R(\tilde{u} - \tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \tag{3.36}$$

Integrating both sides of (3.36) on the interval $[t, t+\Delta t]$, we obtain

$$\begin{aligned} V^{k+1}(\tilde{n}(t+\Delta t)) - V^{k+1}(\tilde{n}(t)) = & \int_t^{t+\Delta t} 2(\underline{v} \odot D^{k+1})^T R(\tilde{u} - \tilde{u}^k)\mathrm{d}\tau \\ & - \int_t^{t+\Delta t} \mathcal{L}(\tilde{n}, \tilde{u}^k)\mathrm{d}\tau \end{aligned}$$

That is

$$V^{k+1}(\tilde{n}(t)) = \int_t^{t+\Delta t} \mathcal{L}(\tilde{n}, \tilde{u}^k)\mathrm{d}\tau - \int_t^{t+\Delta t} 2(\underline{v} \odot D^{k+1})^T R(\tilde{u} - \tilde{u}^k)\mathrm{d}\tau + V^{k+1}(\tilde{n}(t + \Delta t)) \tag{3.37}$$

for any time $t \geq 0$ and time interval $\Delta t > 0$. As introduced in Section 2.4, Δt is termed as the reinforcement interval. There is a trade-off between the learning rate and the reinforcement interval. It is found by Modares et al. (2014) that the larger the reinforcement interval Δt is, the smaller the learning rate should be chosen.

(3.37) is called IRL Bellman equation, which no longer involves the model information of the traffic dynamics. Thus, solving (3.37) instead of the HJB equation (3.12), we can obtain a data-driven IRL based adaptive optimal perimeter controller, which is "model-free".

Note that the convergence of the iteration sequence $\{(V^{k+1}, D^{k+1})\}$ by using (3.21)-(3.22) to the optimality has been checked by Lemma 3.3.1. Hence, we only need to justify the equivalence between the policy iterative equations (3.21)-(3.22) and the IRL Bellman equation (3.37), whereby the convergence and optimality of the IRL approach can also be derived.

Theorem 3.3.1 The IRL Bellman equation (3.37) gives the same solution to the value function as the Bellman equation (3.21) and the same updated control policy as (3.22). ∎

Proof 3.3.3 The proof of Theorem 3.3.1 is divided into two fold.

1) First, we prove that (3.21)-(3.22) $\Rightarrow$ (3.37). Provided that (V^{k+1}, D^{k+1}) is the solution of the policy iterative equations (3.21)-(3.22), from the derivation of (3.37), one can easily deduce that (V^{k+1}, D^{k+1}) is the solution of (3.37).

2) Next, we prove that (3.37) $\Rightarrow$ (3.21)-(3.22). Provided that (V^{k+1}, D^{k+1}) is the solution of the IRL Bellman equation (3.37) and that $D^{k+1} = \frac{1}{2\underline{v}} \odot \left(R^{-1}\mathbf{S}^T \frac{\partial V^{k+1}}{\partial \tilde{n}}\right)$.

Dividing both sides of (3.37) by Δt and taking limit results in

$$\begin{aligned} &\lim_{\Delta t \to 0} \frac{V^{k+1}(\tilde{n}(t + \Delta t)) - V^{k+1}(\tilde{n}(t))}{\Delta t} \\ =&\lim_{\Delta t \to 0} \frac{\int_t^{t+\Delta t} 2(\underline{v} \odot D^{k+1})^T R(\tilde{u} - \tilde{u}^k)\mathrm{d}\tau - \int_t^{t+\Delta t} \mathcal{L}(\tilde{n}, \tilde{u}^k)\mathrm{d}\tau}{\Delta t} \\ \Rightarrow &\frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} = 2(\underline{v} \odot D^{k+1})^T R(\tilde{u} - \tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \end{aligned} \tag{3.38}$$

Substituting $D^{k+1} = \frac{1}{2\underline{v}} \odot \left(R^{-1}\mathbf{S}^T \frac{\partial V^{k+1}}{\partial \tilde{n}}\right)$ into (3.38), we have

$$\frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} = \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}(\tilde{u} - \tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \tag{3.39}$$

Combining (3.32) and (3.39), it follows that

$$\begin{aligned}
\frac{\mathrm{d}V^{k+1}}{\mathrm{d}t} &= \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{S}(\tilde{u} - \tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) + \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{F} - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T \mathbf{F} \\
&= \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}) - \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}^k) - \mathcal{L}(\tilde{n}, \tilde{u}^k) \\
\Rightarrow \mathcal{L}(\tilde{n}, \tilde{u}^k) &+ \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}^k) = \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}) - \frac{\mathrm{d}V^{k+1}}{\mathrm{d}t}
\end{aligned} \tag{3.40}$$

From (3.33) we have the right side of (3.40) equals to 0. Hence,

$$\mathcal{L}(\tilde{n}, \tilde{u}^k) + \left(\frac{\partial V^{k+1}}{\partial \tilde{n}}\right)^T (\mathbf{F} + \mathbf{S}\tilde{u}^k) = 0 \tag{3.41}$$

(3.41) is the same as (3.21). This completes the proof. ∎

Note that we have proven the equivalence between the Bellman equation and the IRL Bellman equation, which does not involve the traffic dynamics (model-free). By iterating V^k on the IRL Bellman equation and updating the control policy D^k (i.e., $\tilde{u}^k$), we can approach both the optimal value function V^* and the optimal perimeter control $\tilde{u}^*$. In the subsequent section, we will develop an online learning approach via the IRL Bellman equation (3.37) to approximate the optimal value function and perimeter controller.

3.4 Online learning by integrating experience replay

For the implementation of the conventional off-line learning based RL approach, sufficient historical data must be collected beforehand and the collected data set would be used repeatedly during the learning process. This implies that only recurrent traffic conditions can be well handled by the conventional off-line learning based RL approach. To adapt to new or unseen data samples and possible changes of

traffic conditions, an online iterative learning approach based on the IRL is proposed in this section. Employing off-policy methods, the proposed online learning approach can be integrated with the ER technique to reduce the requirement on real-time data samples and simultaneously reduce the computational burden.

The online (incorporated with ER) learning method is constructed via the actor-critic (AC) neural network (NN) framework. The neural networks can learn the unknown macroscopic traffic dynamics and achieve the adaptive optimal perimeter control with the IRL. The critic (i.e., policy evaluation) NN and the actor (i.e., policy improvement) NN are tuned sequentially. The flow chart of the proposed IRL algorithm is shown in Figure 3.3. The algorithm starts by evaluating the cost of a given initial admissible control policy and then uses this information to obtain a new and improved control policy that generates a lower associated cost than the previous one does. These two steps of policy evaluation and policy improvement are repeated until the actual policy remains unchanged after the policy improvement step, whereby the convergence to the optimal controller is achieved. The convergence and stability analysis are evidenced via the Lyapunov theory.

Accordingly, at any time $t > \Delta t$ with reinforcement interval $\Delta t > 0$, given that $\tilde{u}'$ is an admissible control, the IRL Bellman equation (3.37) can thus be rewritten as follows

$$\begin{aligned} V(\tilde{n}(t-\Delta t)) = &\int_{t-\Delta t}^{t} \mathcal{L}(\tilde{n}(\tau), \tilde{u}'(\tau))\mathrm{d}\tau \\ &- \int_{t-\Delta t}^{t} 2(\underline{v} \odot D(\tau))^T R(\tilde{u} - \tilde{u}'(\tau))\mathrm{d}\tau + V(\tilde{n}(t)) \end{aligned} \tag{3.42}$$

We utilize an AC-NN framework to approximate the value function and the control policy (i.e., the solution of (3.42)) simultaneously:

$$V(\tilde{n}) = w_V^T \phi_V(\tilde{n}) + \varepsilon_V(\tilde{n}), \quad D(\tilde{n}) = w_D^T \phi_D(\tilde{n}) + \varepsilon_D(\tilde{n}) \tag{3.43}$$

where $\phi_V : \mathbb{R}^{\alpha_n} \to \mathbb{R}^{K_V}$, $\phi_D : \mathbb{R}^{\alpha_n} \to \mathbb{R}^{K_D}$ are vectors of linearly independent activation functions, $w_V^T \in \mathbb{R}^{K_V}$, $w_D^T \in \mathbb{R}^{K_D \times \alpha_u}$ are the NN weights of appropriate dimensions, $\varepsilon_V(\tilde{n})$ and $\varepsilon_D(\tilde{n})$ are the approximation errors of the critic NN and the actor NN, respectively.

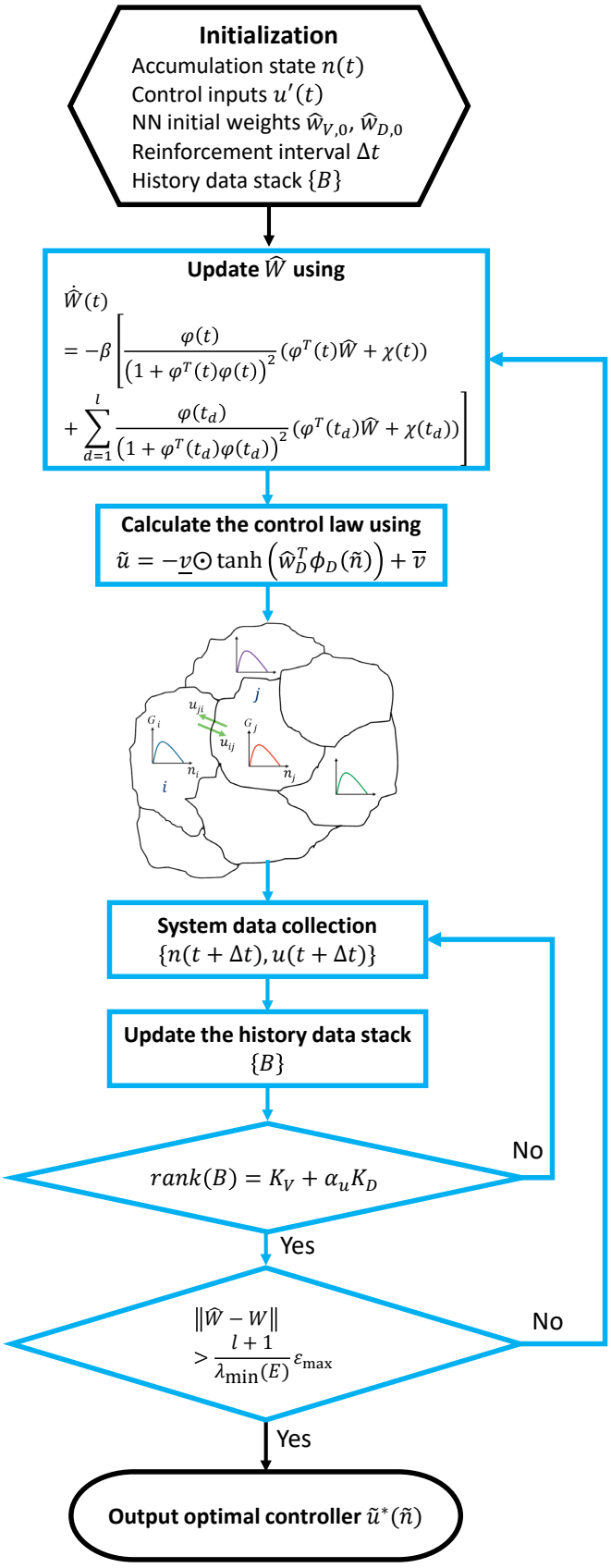


Figure 3.3 The online iterative learning algorithm

Using the approximations (3.43) in (3.42) and considering $\varepsilon_V = 0$, $\varepsilon_D = 0$ for the ideal weights w_V and w_D, one has

$$\begin{aligned}\varepsilon_B(t) \triangleq & \int_{t-\Delta t}^{t} \mathcal{L}(\tilde{n}(\tau), \tilde{u}'(\tau))\mathrm{d}\tau - \int_{t-\Delta t}^{t} 2(\underline{v} \odot w_D^T \phi_D(\tilde{n}(\tau)))^T R(\tilde{u} - \tilde{u}'(\tau))\mathrm{d}\tau \\ & + w_V^T(\phi_V(\tilde{n}(t)) - \phi_V(\tilde{n}(t-\Delta t)))\end{aligned} \tag{3.44}$$

where $\varepsilon_B(t)$ is the Bellman equation error at time t. ε_B is assumed to be bounded on the compact set Ω given the ideal weights w_V and w_D (Modares et al., 2014). That is, there exists a bound $\varepsilon_{\max}$ such that $\|\varepsilon_B\| \leq \varepsilon_{\max}$.

Note that the ideal weights w_V and w_D that provide the best approximate solution for (3.44) are unknown. Hence, the estimations of value function and control policy are given by

$$\hat{V}(\tilde{n}) = \hat{w}_V^T \phi_V(\tilde{n}), \quad \hat{D}(\tilde{n}) = \hat{w}_D^T \phi_D(\tilde{n}) \tag{3.45}$$

where $\hat{w}_V \in \mathbb{R}^{K_V}$, $\hat{w}_D \in \mathbb{R}^{K_D \times \alpha_u}$ are estimations of w_V and w_D, respectively. These estimations are usually learned from training data.

Using (3.45) in (3.42), the approximation error of the IRL Bellman equation, i.e., the TD error, at time t is given by

$$\begin{aligned} e(t) = & \hat{V}(\tilde{n}(t)) - \hat{V}(\tilde{n}(t-\Delta t)) - \int_{t-\Delta t}^{t} 2(\underline{v} \odot \hat{D}(\tau))^T R(\tilde{u} - \tilde{u}'(\tau))\mathrm{d}\tau \\ & + \int_{t-\Delta t}^{t} \mathcal{L}(\tilde{n}(\tau), \tilde{u}'(\tau))\mathrm{d}\tau \\ = & \phi_V^T(\tilde{n}(t))\hat{w}_V - \phi_V^T(\tilde{n}(t-\Delta t))\hat{w}_V - \int_{t-\Delta t}^{t} 2\big(\underline{v} \odot (\hat{w}_D^T \phi_D(\tilde{n}(\tau)))\big)^T \\ & \cdot R(\tilde{u} + \underline{v} \odot \tanh(\hat{w}_D^T \phi_D(\tilde{n}(\tau))) - \overline{v})\mathrm{d}\tau \\ & + \int_{t-\Delta t}^{t} \Bigg(\tilde{n}^T(\tau) Q \tilde{n}(\tau) \\ & + 2\underline{v}^T R \int_{\overline{v}}^{-\underline{v} \odot \tanh(\hat{w}_D^T \phi_D(\tilde{n}(\tau))) + \overline{v}} \tanh^{-1}\left(\frac{1}{\underline{v}} \odot (v - \overline{v})\right) \mathrm{d}v \Bigg) \mathrm{d}\tau \end{aligned} \tag{3.46}$$

Let $\hat{W} = [\hat{w}_V^T, vec^T(\hat{w}_D)]^T \in \mathbb{R}^{K_V + \alpha_u K_D}$ be the estimated weight of the AC-NNs, where $vec(\hat{w}_D) \in \mathbb{R}^{\alpha_u K_D}$ is the vectorization of matrix $\hat{w}_D \in \mathbb{R}^{K_D \times \alpha_u}$. Thus, (3.46) can be rewritten as

$$e(t) = \varphi^T(\tilde{n}(t), \tilde{u}(t))\hat{W} + \chi(\tilde{n}(t)) \tag{3.47}$$

where

$$\varphi(\tilde{n}(t),\tilde{u}(t)) = \begin{bmatrix} \phi_V(\tilde{n}(t)) - \phi_V(\tilde{n}(t-\Delta t)) \\ \int_{t-\Delta t}^{t}(2\underline{v}\odot R(\tilde{u}+\underline{v}\odot\tanh(\hat{w}_D^T\phi_D(\tilde{n}(\tau))) - \overline{v}))\otimes\phi_D(\tilde{n}(\tau))\mathrm{d}\tau \end{bmatrix}$$

$$\begin{aligned}\chi(\tilde{n}(t)) &= \int_{t-\Delta t}^{t}\mathcal{L}(\tilde{n}(\tau),\tilde{u}'(\tau))\mathrm{d}\tau \\ &= \int_{t-\Delta t}^{t}\Bigg(\tilde{n}^T(\tau)Q\tilde{n}(\tau) \\ &\quad + 2\underline{v}^T R\int_{\overline{v}}^{-\underline{v}\odot\tanh(\hat{w}_D^T\phi_D(\tilde{n}(\tau)))+\overline{v}}\tanh^{-1}\left(\frac{1}{\underline{v}}\odot(v-\overline{v})\right)\mathrm{d}v\Bigg)\mathrm{d}\tau\end{aligned}$$

To enable online learning, we use the gradient-descent method to update the estimated AC-NN weights. Both real-time data and historical data are used to estimate the weights of the NNs to guarantee the data richness and efficiency.

As discussed in Section 2.4, the ER technique can be integrated with the IRL algorithm to speed up the computation. Based on the generalized least-squares (GLS) principle, we aim to update the estimated weight vector $\hat{W}$ to minimize $\|e(t)\| + \sum_{d=1}^{l}\|e(t_d)\|$, where the first part denotes the instantaneous TD error and the second part denotes the TD errors for the stored transition samples. In order to ensure the existence of the solution, we need the following assumption.

Assumption 3.4.1 Define $B = [\varphi(t_1),\ldots,\varphi(t_l)]$ as a matrix of the stored data, where l is the number of samples stored in the history stack. There are as many linearly independent elements as the number of corresponding NN's hidden neurons for the stored data matrix B such that $\mathrm{rank}(B) = K_V + \alpha_u K_D$.

This rank condition is to verify the richness of the stored data, i.e., whether it is sufficient to solve the GLS problem and to guarantee the convergence to a near-optimal control (Modares et al., 2014). Based on (3.47), for the online iterative learning, the gradient-based adaptation law with ER is given by

$$\begin{aligned}\dot{\hat{W}}(t) &= -\beta\left(\frac{\varphi(t)}{(1+\varphi^T(t)\varphi(t))^2}e(t) + \sum_{d=1}^{l}\frac{\varphi(t_d)}{(1+\varphi^T(t_d)\varphi(t_d))^2}e(t_d)\right) \\ &= -\beta\Bigg(\frac{\varphi(t)}{(1+\varphi^T(t)\varphi(t))^2}(\varphi^T(t)\hat{W}+\chi(t)) \\ &\quad + \sum_{d=1}^{l}\frac{\varphi(t_d)}{(1+\varphi^T(t_d)\varphi(t_d))^2}(\varphi^T(t_d)\hat{W}+\chi(t_d))\Bigg)\end{aligned} \tag{3.48}$$

where $\beta > 0$ is the learning rate, t is the current time and the index d refers to the d-th sample data ($d = 1, \ldots, l$) stored in the history stack B. In (3.48), the first term is a gradient-descent update law for minimizing $\|e(t)\|$, while the second term attempts to minimize $\sum_{d=1}^{l} \|e(t_d)\|$.

Denote the optimal value of weight by $W = [w_V^T, vec^T(w_D)]^T$ and recall that the estimated weight is defined by $\hat{W} = [\hat{w}_V^T, vec^T(\hat{w}_D)]^T$. The following theorem demonstrates the convergence of the weight estimation error of AC-NNs, $\tilde{W}(t) = W - \hat{W}(t)$, using Lyapunov method.

Theorem 3.4.1 If the stored data B for AC-NNs (3.45) with the ER adaptation law (3.48) satisfy Assumption Assumption 3.4.1,

1. for bounded ε_B, the weight estimation error $\tilde{W}(t) = W - \hat{W}(t)$ converges exponentially to the residual set $R_s = \{\tilde{W} \mid \|\tilde{W}(t)\| \leq c \cdot \varepsilon_{\max}\}$, where $c > 0$ is a constant;

2. the system state $\tilde{n}$ is asymptotically stable.

∎

Proof 3.4.1 1) Based on (3.44), (3.46), (3.47) and $\tilde{W}(t) = W - \hat{W}(t)$, the TD errors for he current time t and the recorded time t_d can be rewritten respectively as

$$\begin{aligned} e(t) &= \varphi^T(t)\hat{W} + \chi(t) = \varphi^T(t)W - \varphi^T(t)\tilde{W} + \chi(t) \\ &= -\varphi^T(t)\tilde{W} + (\chi(t) + \varphi^T(t)W) = -\varphi^T(t)\tilde{W} + \varepsilon_B(t) \qquad (3.49a) \\ e(t_d) &= \varphi^T(t_d)\hat{W} + \chi(t_d) = \varphi^T(t_d)W - \varphi^T(t_d)\tilde{W} + \chi(t_d) \\ &= -\varphi^T(t_d)\tilde{W} + (\chi(t_d) + \varphi^T(t_d)W) = -\varphi^T(t_d)\tilde{W} + \varepsilon_B(t_d) \qquad (3.49b) \end{aligned}$$

From $\tilde{W}(t) = W - \hat{W}(t)$, one has $\dot{\tilde{W}}(t) = -\dot{\hat{W}}(t)$. Substituting (3.49a)-(3.49b) into (3.48) and denoting $\bar{\varphi} = \varphi/(1 + \varphi^T\varphi)$ and $m = 1 + \varphi^T\varphi$, we can obtain

$$\begin{aligned} \dot{\tilde{W}}(t) &= \beta \left(\frac{\varphi(t)}{(1 + \varphi^T(t)\varphi(t))^2} e(t) + \sum_{d=1}^{l} \frac{\varphi(t_d)}{(1 + \varphi^T(t_d)\varphi(t_d))^2} e(t_d) \right) \\ &= \beta \left(\frac{\bar{\varphi}(t)}{m(t)} (-\varphi^T(t)\tilde{W} + \varepsilon_B(t)) + \sum_{d=1}^{l} \frac{\bar{\varphi}(t_d)}{m(t_d)} (-\varphi^T(t_d)\tilde{W} + \varepsilon_B(t_d)) \right) \\ &= -\beta \left(\frac{\bar{\varphi}(t)}{m(t)} \varphi^T(t) + \sum_{d=1}^{l} \frac{\bar{\varphi}(t_d)}{m(t_d)} \varphi^T(t_d) \right) \tilde{W} + \beta \left(\frac{\bar{\varphi}(t)}{m(t)} \varepsilon_B(t) + \sum_{d=1}^{l} \frac{\bar{\varphi}(t_d)}{m(t_d)} \varepsilon_B(t_d) \right) \end{aligned}$$

$$= -\beta \left(\bar{\varphi}(t)\bar{\varphi}^T(t) + \sum_{d=1}^{l} \bar{\varphi}(t_d)\bar{\varphi}^T(t_d) \right) \tilde{W} + \beta \bar{\varepsilon}_B \tag{3.50}$$

where $\bar{\varepsilon}_B = \frac{\bar{\varphi}(t)}{m(t)}\varepsilon_B(t) + \sum_{d=1}^{l} \frac{\bar{\varphi}(t_d)}{m(t_d)}\varepsilon_B(t_d)$.

Now we choose the Lyapunov function as

$$L = \frac{1}{2\beta}\tilde{W}^T(t)\tilde{W}(t) \tag{3.51}$$

Differentiating (3.51) along the trajectories of (3.50), one has

$$\begin{aligned} \dot{L} &= \frac{1}{\beta}\tilde{W}^T\dot{\tilde{W}} \\ &= \frac{1}{\beta}\tilde{W}^T \cdot \left(-\beta \left(\bar{\varphi}(t)\bar{\varphi}^T(t) + \sum_{d=1}^{l} \bar{\varphi}(t_d)\bar{\varphi}^T(t_d) \right) \tilde{W} + \beta \bar{\varepsilon}_B \right) \\ &= -\tilde{W}^T \left(\bar{\varphi}(t)\bar{\varphi}^T(t) + \sum_{d=1}^{l} \bar{\varphi}(t_d)\bar{\varphi}^T(t_d) \right) \tilde{W} + \tilde{W}^T \bar{\varepsilon}_B \end{aligned} \tag{3.52}$$

If Assumption Assumption 3.4.1 is satisfied, then $\bar{\varphi}(t)\bar{\varphi}^T(t) + \sum_{d=1}^{l} \bar{\varphi}(t_d)\bar{\varphi}^T(t_d) > 0$. Suppose that ε_B is bounded by $\varepsilon_{\max}$, i.e., $\|\varepsilon_B\| \leq \varepsilon_{\max}$, $\dot{L}$ is negative definite provided that

$$\|\tilde{W}(t)\| > \frac{l+1}{\lambda_{\min}(E)}\varepsilon_{\max} = c \cdot \varepsilon_{\max} \tag{3.53}$$

where $c = \frac{l+1}{\lambda_{\min}(E)} > 0$ and $\lambda_{\min}(E)$ is the minimum eigenvalue of E with $E = \bar{\varphi}(t)\bar{\varphi}^T(t) + \sum_{d=1}^{l} \bar{\varphi}(t_d)\bar{\varphi}^T(t_d)$. Hence, the weight estimation error $\tilde{W}$ converges exponentially to the residual set $R_s = \{\tilde{W} \mid \|\tilde{W}(t)\| \leq c \cdot \varepsilon_{\max}\}$.

2) For system (3.5), define Lyapunov function candidate as (3.9). Take the time derivative of V and we can obtain

$$\dot{V} = -\mathcal{L}(\tilde{n}, \tilde{u}) = -N(\tilde{n}) - U(\tilde{u})$$

Recall that $N(\tilde{n})$ and $U(\tilde{u})$ are positive definite functions. Then we have $V(\tilde{n}(t)) \geq 0$, $\dot{V} \leq 0$ and $V(\tilde{n}(t)) = 0$ if and only if $\tilde{n} = 0$, i.e., $n = n^*$. That is, $V(\tilde{n})$ is a Lyapunov function. The closed-loop system is thus asymptotically stable. This completes the proof. ∎

Theorem 3.4.1 indicates that using the gradient-based adaptation law with ER (3.48), the AC-NN framework (3.45) can approximate the optimal value function $V^*(\tilde{n})$ and perimeter control policy $\tilde{u}^*(\tilde{n})$. The value function (3.9) is proven to be

a Lyapunov function for the MFD dynamics. Hence, the initial accumulation state n_0 can be asymptotically stabilized by the obtained perimeter controller at the desired steady state n^*.

3.5 Numerical experiments

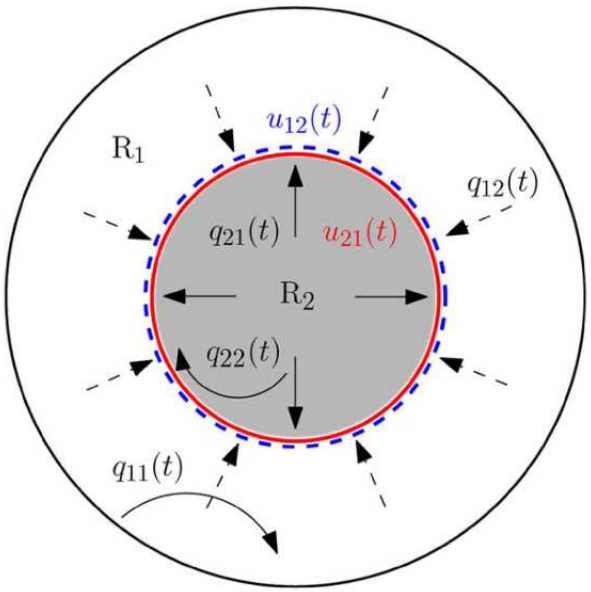
3.5.1 Settings of the test environment

To test the performance of the proposed method, two scenarios with different purposes and settings are simulated (see Table 3.2). The network topologies used in these numerical examples are shown in Figure 3.4. For set-point control objective, the two-region MFD system (Haddad, 2015) with constant demand pattern is considered in Scenario 1 for demonstration of the convergent speed and computation efficiency of the proposed method. For min TTS control objective, a three-region MFD system as in Zhong et al. (2018b) with time-varying travel demand is considered in Scenario 2. The robustness and adaptiveness of the proposed method are validated by conducting experiments under various demand patterns. The subregion MFD functions of all the examples are assumed to be the same. The true, but unknown MFD functions and the parameters are given in Table 3.2 to generate the I/O data for learning the traffic dynamics only. Note that they are not involved in the controller design. For the examples in Scenario 1, the cost functions $N(\tilde{n}) = \tilde{n}^T Q \tilde{n}$ and $U(\tilde{u})$ is defined by (3.8), where $Q = 10^{-2} \cdot \mathcal{I}_{\alpha_n}$, $R = \mathcal{I}_{\alpha_u}$ with $\mathcal{I}_x$ denoting the identity matrix of dimension x. For the experiment in Scenario 2, the objective function is defined by (3.18), where $\bar{\lambda} = 1$.

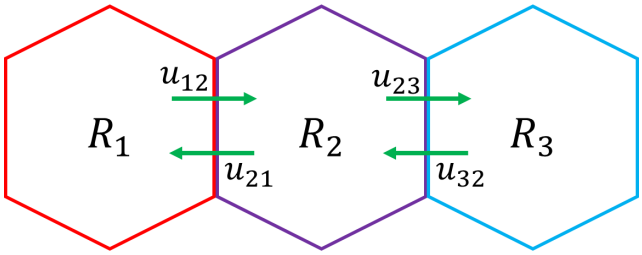
The stabilizing control law by Haddad (2015) embedded with a sequence of randomly generated deviations is adopted to initialize the online learning algorithm. The AC-NN framework is employed for approximating the optimal value function and control policy in all examples. Let $\phi_V \in \mathbb{R}^{K_V}$ and $\phi_D \in \mathbb{R}^{K_D}$ denote the activation functions of the online learning approach.

For Scenario 1, inspired by Abu-Khalaf and Lewis (2005), we adopt an AC-NN framework with 84 critic NN hidden neurons and 4 actor NN hidden neurons, i.e., $K_V = 84$ and $K_D = 4$. Suppose $x = [x_1, x_2, x_3, x_4]^T$, the activation function of critic NN is

$$\phi_V^{p_V}(x) = x_1^i x_2^j x_3^m x_4^n \tag{3.54}$$



(a) The two-region MFD network topology, slightly adapted from Geroliminis et al. (2013)



(b) The three-region MFD network topology

Figure 3.4 Network topologies

where $i+j+m+n=6$ and $p_V = 1,\ldots,84$, and the activation function of actor NN is

$$\phi_D^{p_D}(x) = x_{p_D} \tag{3.55}$$

where $p_D = 1,\ldots,4$.

For Scenario 2, we set $K_V = 210$ and $K_D = 7$. Suppose $x = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]^T$, the activation function of critic NN is

$$\phi_V^{p_V}(x) = x_{k_1}^i x_{k_2}^j x_{k_3}^m x_{k_4}^n \tag{3.56}$$

where $i + j + m + n = 6$, $p_V = 1, \ldots, 210$ and $k_1, k_2, k_3, k_4 \in \{1, \ldots, 7\}$, and the activation function of actor NN is

$$\phi_D^{p_D} = x_{p_D} \tag{3.57}$$

where $p_D = 1, \ldots, 7$.

The sample size and replay buffer (history data stack) size for AC-NN updates in all the examples are 250 and 1000, respectively. The computer processor is Intel Core i7-9850 CPU 2.60 GHz, and the simulation platform is MATLAB R2022a.

3.5.2 Set-point control

In this subsection, we apply the proposed IRL based online iterative learning approach to the two-region network with constant travel demand. A two-region network as shown in Figure 3.4(a) is considered in this scenario. As explained, the objective of set-point perimeter control is to regulate the network traffic state to the desired stable equilibrium. Comparison in terms of control performance and computational efficiency is made between the proposed IRL approach and other existing controllers, e.g., the state-of-the-art MPC by Geroliminis et al. (2013) and the neuro-dynamic programming (N-DP) method by Su et al. (2020). Note that MPC is a model-based controller, and that N-DP requires partial information of the MFD system, while the IRL does not rely on any knowledge of the traffic dynamics.

Table 3.2 Scenario description

	Network	Demand	Controllers	MFD parameters
1-A	2-region	constant	IRL, N-DP	$G_i(n_i) = \frac{1.4877 \cdot 10^{-7} n_i^3 - 2.9815 \cdot 10^{-3} n_i^2 + 15.0912 n_i}{3600}$ (veh/s)
1-B			IRL (various reinforcement intervals)	
1-C			IRL, MPC	
2	3-region	time-varying	IRL, MPC	$n_i^{jam} = 10000$ (veh), $n_i^{cr} = 3392$ (veh)

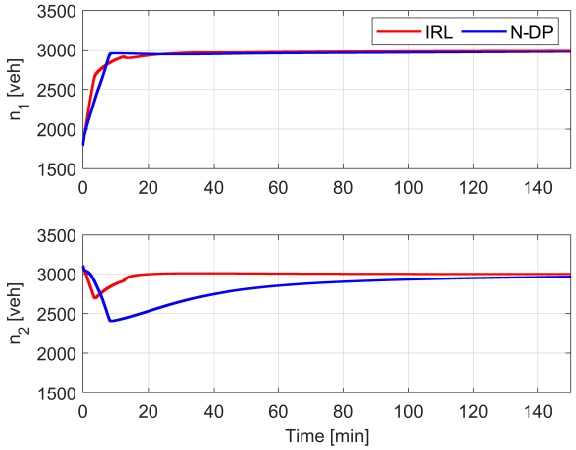
3.5.2.1 Scenario 1-A: Comparison between the IRL and the N-DP approaches

In Scenario 1-A, we present a comparison between the proposed off-policy learning based IRL approach and the on-policy learning based N-DP approach by Su et al. (2020). In line with Haddad (2015), $\bar{n} = [3000, 3000]^T$ (veh), which is close to the critical accumulation, is chosen as the desired equilibrium. In addition, the demand pattern is set to be constant as $q = [1.6, 1.6, 1.6, 1.6]^T$ (veh/s). Thus, the steady-state accumulation for each direction and the corresponding control inputs as solved from the steady-state equations (3.4a)-(3.4c) are $n^* = [1538.9, 1461.1, 1461.1, 1538.9]^T$ (veh) and $u^* = [0.5267, 0.5267]^T$. The initial regional accumulations are set to be $[1800, 3100]^T$ (veh) with OD-specific initial accumulations being $n_{11}(0) = 540$ (veh), $n_{12}(0) = 1260$ (veh), $n_{21} = 2170$ (veh), $n_{22}(0) = 930$ (veh).

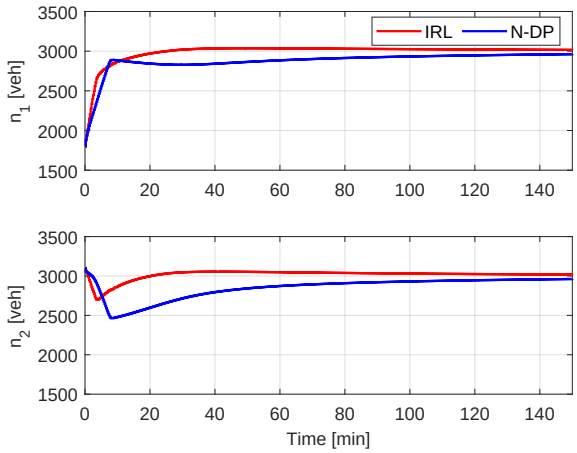
Note that N-DP requires input data with high resolution to solve the HJB equation for the optimal controller. For a fair comparison, the first case is that the sample time interval (and thus reinforcement interval) and the control update step are set as 1 second for both methods. To consider more practical situations, in the second case and third case, we set the sample time interval and the control update step to 15 seconds (Haddad, 2015) and 30 seconds, respectively. Sensitivity analysis of the reinforcement interval for the IRL is presented in Scenario 1-B.

As shown in Figure 3.5(a), when $\Delta t = 1$, both the IRL and the N-DP can regulate the accumulation states to the desired equilibrium $[3000, 3000]^T$ (veh) in an asymptotic manner. Specifically, the N-DP controller achieves a shorter settling time[2] than the IRL approach for accumulation state $n_1(t)$, while the IRL is much better than the N-DP in the settling time for $n_2(t)$. Note that the N-DP control algorithm has been well-trained in an off-line manner before it is applied. However, only by interacting with the environment and learning the macroscopic traffic dynamics online, the proposed IRL approach can achieve settling times of around 20 minutes for both n_1 and n_2. Besides, $n_2(t)$ has experienced an overshoot to around 2500 (veh) applying the N-DP based controller, while the overshoot induced by the IRL approach is much smaller. Moreover, applying the N-DP, the increase of the reinforcement interval slows down the convergent speed of accumulation states (see Figure 3.5(b)) or even cannot regulate them to the desired equilibria (see Figure 3.5(c)). However, the IRL approach can stabilize the accumulations at the desired steady states in all the

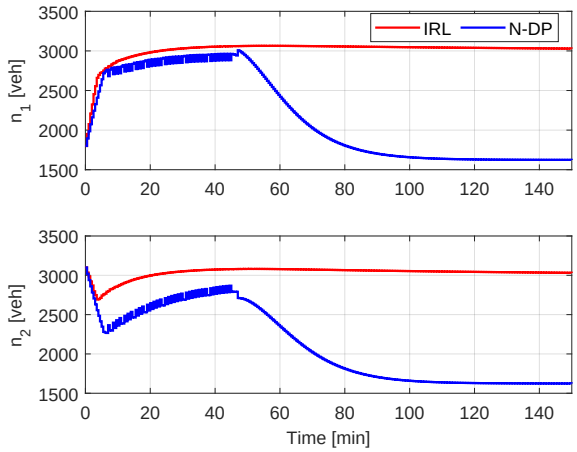
[2]The settling time is the time required for the dynamics to reach and stay within a small range of certain percentage (usually 5% or 2%) of the desired steady state (see Fig. 3.23 and Chapter 3.4.3 in Franklin et al., 2015). In our case, the settling time is the time required for $\tilde{n}_i(t)$ to reach and stay within 2% of the steady state $\bar{n}_i$.



(a) State evolutions with $\Delta t = 1s$



(b) State evolutions with $\Delta t = 15s$



(c) State evolutions with $\Delta t = 30s$

Figure 3.5 Simulation results of Scenario 1-A

cases within around 20 minutes. These results indicate that the IRL approach can achieve decent convergence and stability of the accumulation states under different Δt, while the control performance of N-DP deteriorates as Δt increases.

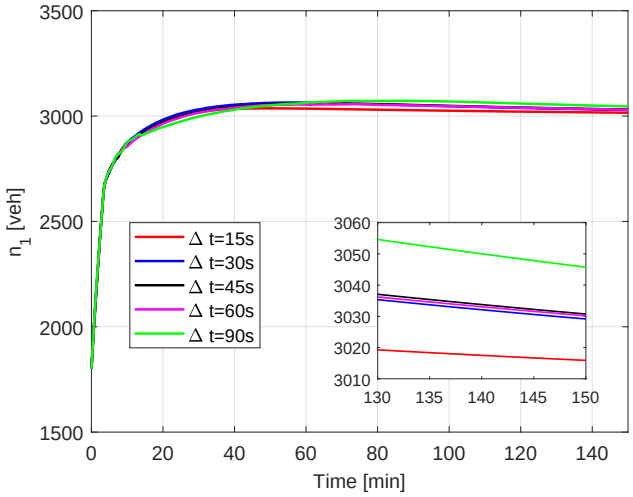
There are also differences in the computational complexity and data usage efficiency. Note that 20000 samples are used for off-line training in each iteration (40 iterations in total) for the N-DP approach, whereas only 250 samples are used in each iteration for the IRL approach. This is the advantage of integrating the ER technique with IRL, i.e., fast convergence of the iterative learning process can be guaranteed. Take the first case as an example, because of the reduction of samples used for each iteration, the total computation time of the IRL approach is less than 8 seconds while that of the N-DP approach is more than 2 minutes. Different from the conventional RL methods (e.g., N-DP) which are usually trained off-line and data intensive, the improvement in data usage efficiency by integrating IRL with ER indicates the real-time applicability of the proposed IRL based perimeter control schemes.

3.5.2.2 Scenario 1-B: Sensitivity analysis of the reinforcement interval

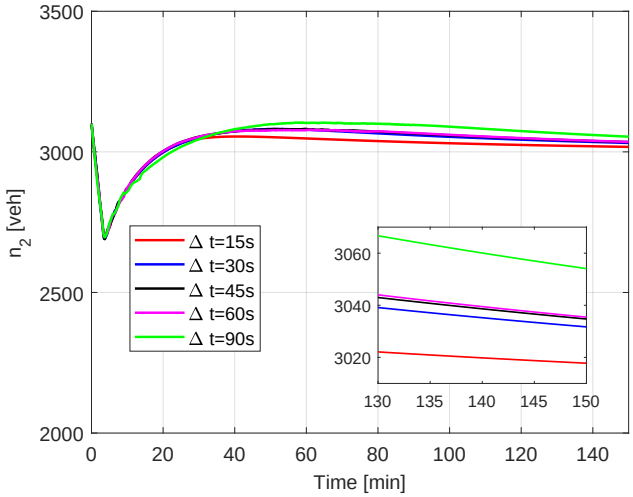
In Scenario 1-B, sensitivity analysis of the reinforcement interval Δt for the proposed IRL method is performed using the network and settings of Scenario 1-A. We assume that the data resolution of traffic sensors is identical to the reinforcement interval so that one data sample is collected in a reinforcement interval. The larger Δt is, the lower frequency the sensors upload traffic data to the management center. For instance, $\Delta t = 60$s means the sensors upload data every 1 minute. Note that the larger the reinforcement interval is, the smaller learning rate that could be chosen to achieve the AC-NN weights convergence (Modares et al., 2014). In this example, each learning rate $\beta \in \{0.01, 0.007, 0.005, 0.003, 0.0001, 0.00007\}$ is chosen respectively for each reinforcement interval $\Delta t \in \{15s, 20s, 30s, 45s, 60s, 90s\}$.

Regarding the nature of perimeter control actuation approaches, e.g., traffic signal controls which can be changed only with a new traffic signal cycle, the control update step cannot be smaller than the sample time interval (i.e., the reinforcement interval). Therefore, the sensitivity analysis of reinforcement interval for the IRL algorithm is divided into the following two folds.

The first case is that the control update intervals are equal to the tested reinforcement intervals. Figure 3.6(a) and Figure 3.6(b) present the accumulation trajectories $n_1(t)$ and $n_2(t)$ over time under different Δt, respectively, while the control input

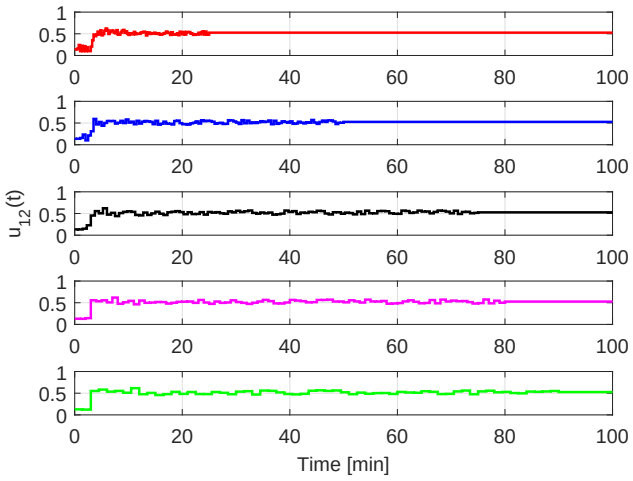


(a) State n_1 evolution

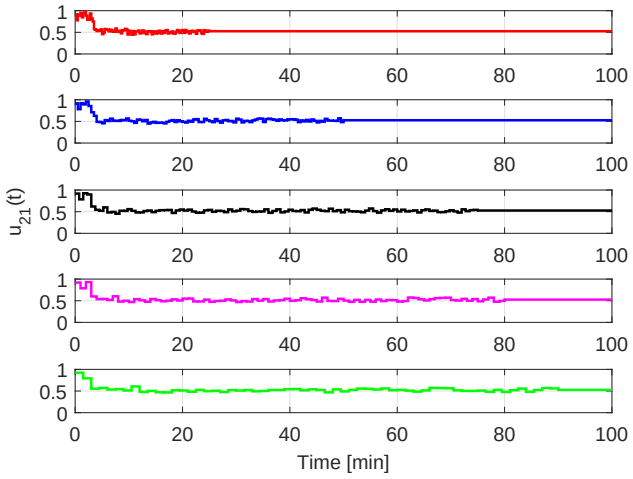


(b) State n_2 evolution

Figure 3.6 State evolution results of Scenario 1-B with reinforcement intervals equal to control update steps



(a) Control u_{12} evolution



(b) Control u_{21} evolution

Figure 3.7 Control input results of Scenario 1-B with reinforcement intervals equal to control update steps

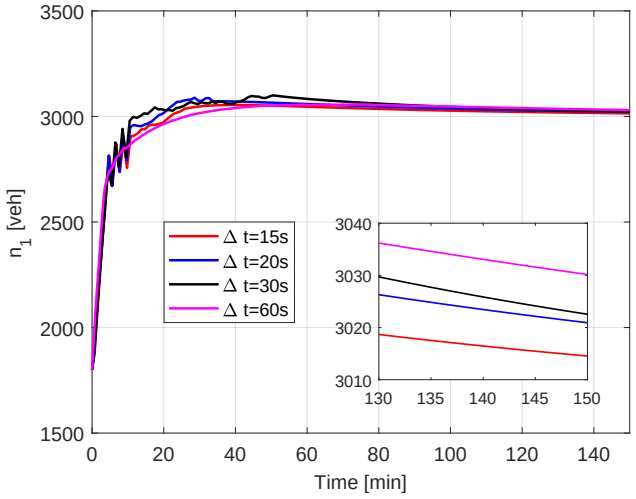
evolutions are illustrated by Figure 3.7(a) and Figure 3.7(b). When $\Delta t = 15$s, the results show that both the initial states and controls converge very fast to the desired equilibrium. As Δt increases, the convergent speed of the perimeter control gain decreases. This slows down the convergent speed of the accumulation states. This is because the algorithm has to wait longer to collect new data to update the weights of the AC-NNs, which also results in less frequent updates of the control inputs. However, Figure 3.6(a) and Figure 3.6(b) indicate that the variation of Δt does not significantly influence the convergence of the accumulation states. That is to say, the proposed online learning approach is robust to the variation of real-time data resolution.

Next, we fix the control update steps to 60 seconds while vary the reinforcement intervals in $\Delta t \in \{15s, 20s, 30s, 60s\}$. Figure 3.8(a)-Figure 3.8(b) shows that the accumulation states can converge to the desired steady states in around 30 minutes. We can also observe that as Δt increases, both the convergent speeds of the perimeter control gain and the accumulation states decrease. The IRL based perimeter controller with $\Delta t = 15s$ still achieves the shortest settling time. In the early stage of the training and implementation of the IRL controllers, the larger difference between Δt and the control update step, the stronger oscillation of the accumulation state occurs. However, with the control update steps fixed, the variation of Δt still does not significantly affect the convergence of the accumulation states.

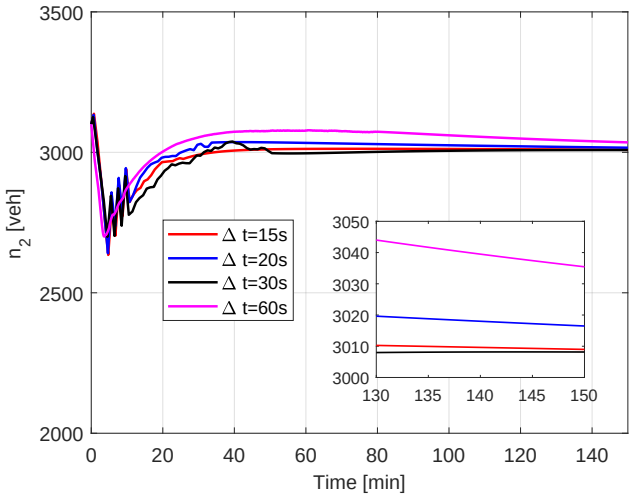
These results imply the feasibility of online tuning of the reinforcement interval to adapt to heterogeneous real-time sensor data resolution without affecting the system stability. This is a key advantage of the proposed IRL based online learning algorithm over the traditional RL based methods.

3.5.2.3 Scenario 1-C: Comparison between the IRL and the MPC approaches

Scenario 1-C adopts the same settings as Scenario 1-A except the initial condition, the set-point value and the reinforcement interval. Unlike Scenario 1-A corresponding to a mild traffic condition where all regions are regulated in an uncongested regime (i.e., below the critical accumulation), the initial accumulation state values are set to far exceed the critical accumulations, i.e., $[4300, 3700]^T$ (veh) with $n_{11}(0) = 430$ (veh), $n_{12}(0) = 3870$ (veh), $n_{21} = 370$ (veh), $n_{22}(0) = 3330$ (veh). Besides, in Scenario 1-C, both regions are regulated around set points in the congested regimes, e.g., $\bar{n} = [4000, 4000]^T$ (veh). Performance comparison is conducted between the IRL and the state-of-the-art MPC method, where for both the controllers the sample time



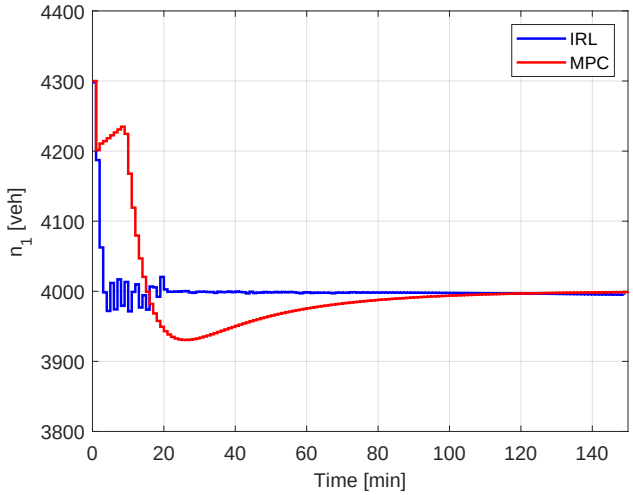
(a) State n_1 evolution



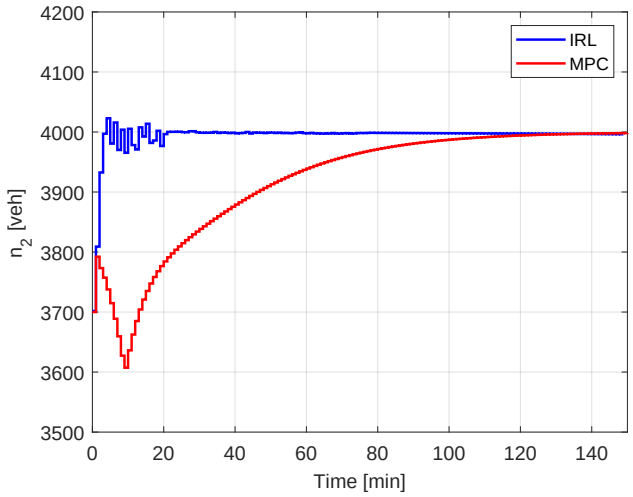
(b) State n_2 evolution

Figure 3.8 Simulation results of Scenario 1-B with fixed control update steps

interval and the control update step are set as 60 seconds. For the MPC controller, the prediction horizon is set to be 30 (i.e., 30 minutes simulation time).



(a) State n_1 evolution



(b) State n_2 evolution

Figure 3.9 Simulation results of Scenario 1-C

Figure 3.9 shows that both the IRL and the MPC controllers can stabilize the accumulation states at the desired equilibrium. Regulated by the IRL controller, both n_1 and n_2 converge very fast to the steady states, while regulated by the MPC controller, one can observe small overshoots of the accumulation states. The settling time and the average CPU time per control update step[3] of different control schemes are reported

[3]The CPU time is defined as the average computation time per control update step. They were measured by the tic and toc functions of MATLAB R2022a. We present the average value of 10 tests for each controller.

in Table 3.3. Despite no model knowledge available, the IRL approach can achieve a 20-minute settling time, which indicates that the IRL can have a decent control performance in a congested traffic situation. The CPU times per control update step of both methods are extremely small, which are far less than the 60-second control update step. These results imply the real-time applicability of the proposed model-free IRL approach.

Table 3.3 Summary of settling time and computation time for Scenario 1-C

	State	IRL	MPC
Settling time (min)	n_1	≈ 22	≈ 29
	n_2	≈ 21	≈ 53
CPU time per step (sec)	–	6.57×10^{-6}	1.79×10^{-2}

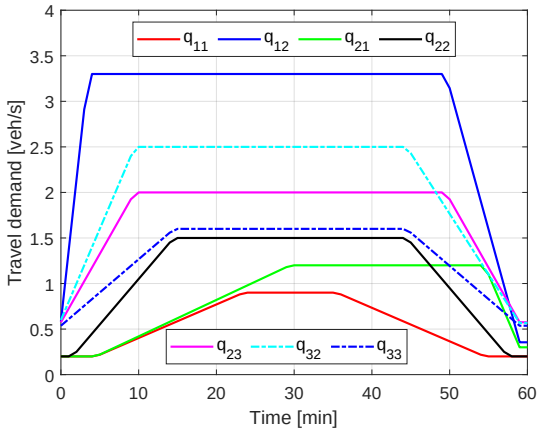
3.5.3 TTS minimization

In this subsection, the objective function is related to minimizing the total time spent (TTS) for the urban network subject to uncertainties in travel demands. We apply the proposed IRL based perimeter controller to the three-region network shown by Figure 3.4(b) with a time-varying demand pattern. The perimeter controller is subject to heterogeneous cross-boundary capacities, i.e.,

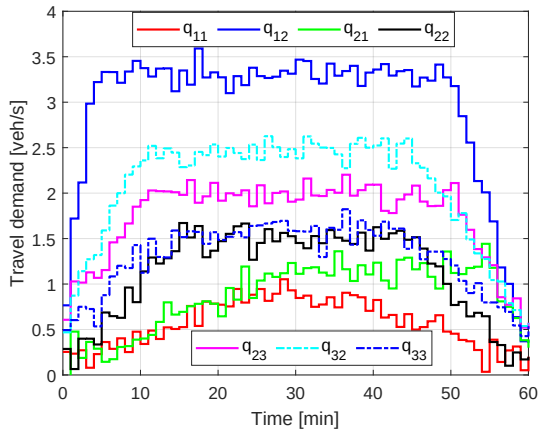
$$0.1 \leq u_{12} \leq 0.7,\ 0.3 \leq u_{21} \leq 1,\ 0.4 \leq u_{23} \leq 1,\ 0.2 \leq u_{32} \leq 0.9$$

3.5.3.1 Scenario 2: Three-region MFD system with uncertain time-varying travel demand

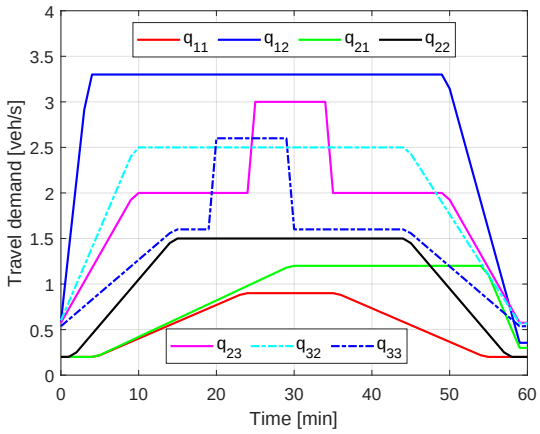
In this scenario, a time-varying demand pattern is used to mimic a scenario of peak-hour traffic with congestion onset, stationary congestion and congestion dissolving processes. Comparisons between the IRL approach and MPC are made under three different travel demand patterns, i.e., 1) the nominal deterministic travel demand pattern (Figure 3.10(a)), 2) the nominal demand pattern subject to external disturbances (Figure 3.10(b)), and 3) the travel demand pattern subject to an abrupt change during the stationary congestion period (Figure 3.10(c)). The initial accumulation state is set as $n(0) = [5400, 5500, 2000]^T$ (veh). Namely, n_1 and n_2 are initiated in a very congested state while n_3 in an uncongested initial state. For both the IRL and the MPC, the control update interval is set as 60 seconds.



(a) Nominal demand



(b) Noisy demand



(c) Abrupt changes of demand

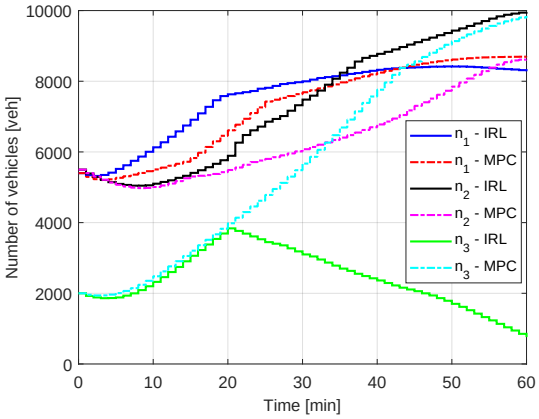
Figure 3.10 Demand patterns of Scenario 2

The accumulation evolution results are given in Figure 3.11, where the evolution of regional accumulations for the nominal, noisy, and abrupt-change demand cases

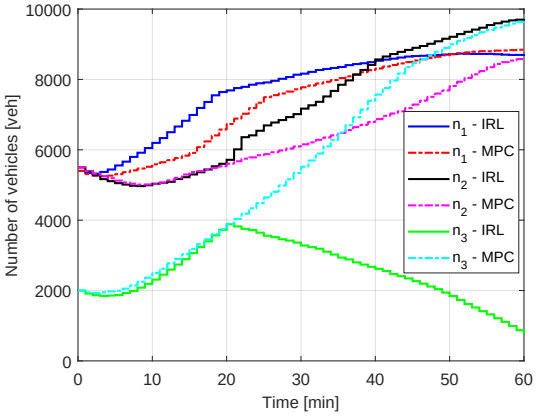
are presented by Figure 3.11(a), Figure 3.11(b), and Figure 3.11(c), respectively. Figure 3.12 and Figure 3.13 showcase the control input evolution and TTS evolution, respectively. The achieved TTS and the average CPU time per control update step of different control schemes are summarized in Table 3.4.

Figure 3.11(a), Figure 3.11(b) and Figure 3.11(c) illustrate that in all the demand cases, Region 1 and 2 are congested at the beginning while Region 3 is uncongested, and all the regional accumulation states experience increase during the early stage due to the increase in inflow demands. It is noteworthy that the congestion in Region 3 regulated by the IRL starts to dissipate after 20 minutes, while the accumulation state of Region 3 regulated by MPC continues being increasingly congested. As observed from Figure 3.13(a), Figure 3.13(b) and Figure 3.13(c), the IRL is superior to the MPC in minimizing TTS for the whole network. Based on Table 3.4, in the abrupt-change demand case, the IRL achieves a 12% decrease in TTS over MPC, while the same performance metrics are 11% and 10% respectively for the nominal and noisy demand cases. These results demonstrate that the proposed IRL based control strategy can well learn and adapt to the dynamic nature of the travel demand and hence guarantee the robustness of the traffic dynamics.

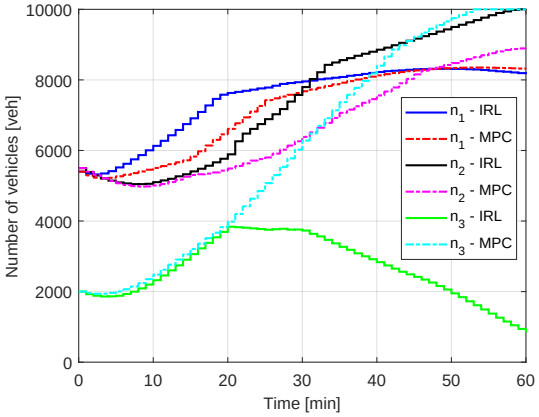
To close the discussion, the numerical results indicate that the proposed IRL based adaptive perimeter controller can not only stabilize the network accumulation states at the desired equilibrium, but also achieve improvement in min TTS compared to the state-of-the-art MPC scheme. These results demonstrate the effectiveness and efficiency of IRL under a variety of data resolutions. In addition, the proposed approach has been examined under various traffic conditions and demand patterns, which implies a promising application of IRL for macroscopic traffic control. The perimeter control in essence is a type of gating control actualized on the boundaries to regulate the cross-boundary traffic flows between different regions. Such kinds of perimeter control are deployed in many metropolises such as Guangzhou and Hong Kong utilizing the existing infrastructure. For example, such perimeter control has been implemented on the cross-Zhujiang-river bridges connecting two busy business districts to manage the peak-hour traffic. Similar perimeter control strategies are also implemented on existing infrastructures in Hong Kong, such as the Hung Hom Cross Harbor Tunnel connecting Hong Kong Island and Kowloon area. For a detailed discussion on the potential applications of the perimeter control, readers may refer to Zhong et al. (2018a) and Zhong et al. (2018b).



(a) States under nominal demand

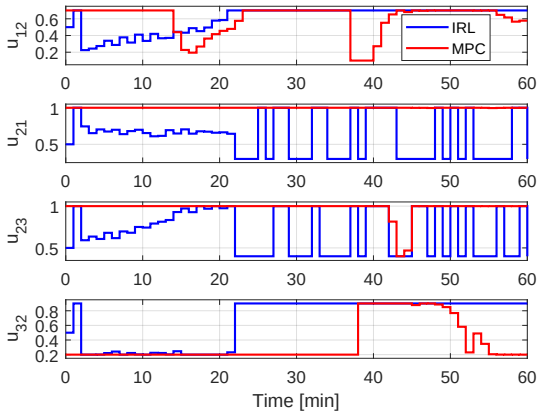


(b) States under noisy demand

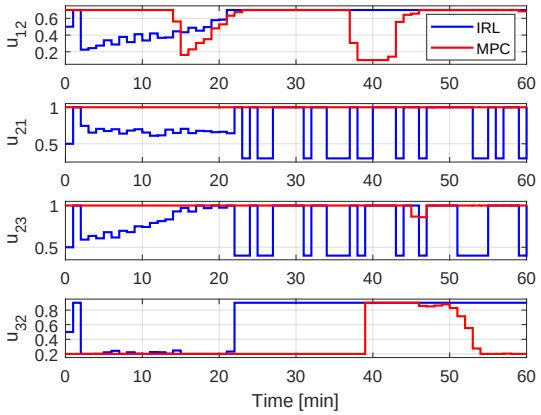


(c) States under abrupt-change demand

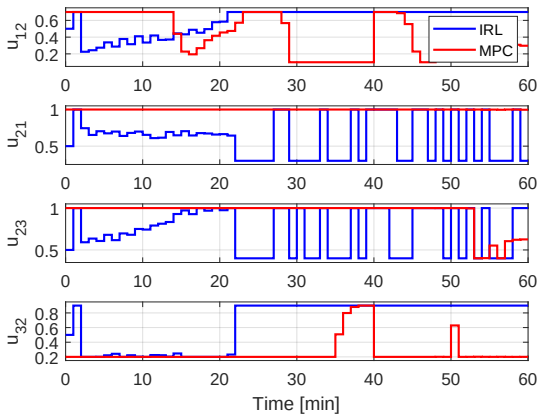
Figure 3.11 Accumulation state evolution of Scenario 2



(a) Controls under nominal demand

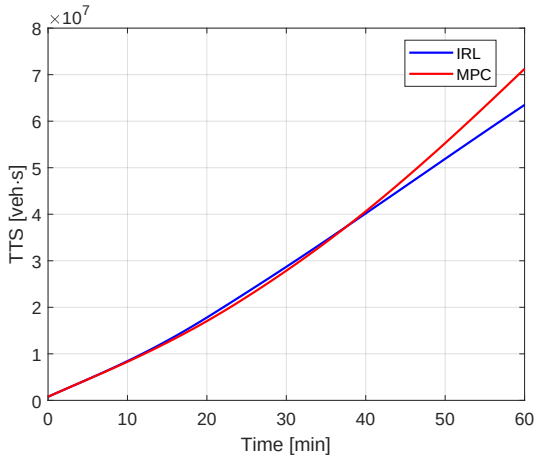


(b) Controls under noisy demand

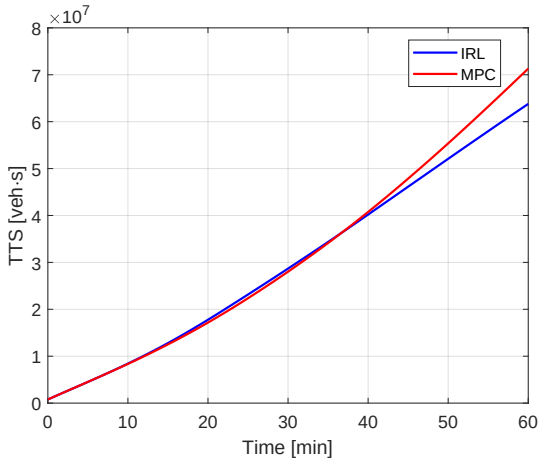


(c) Controls under abrupt-change demand

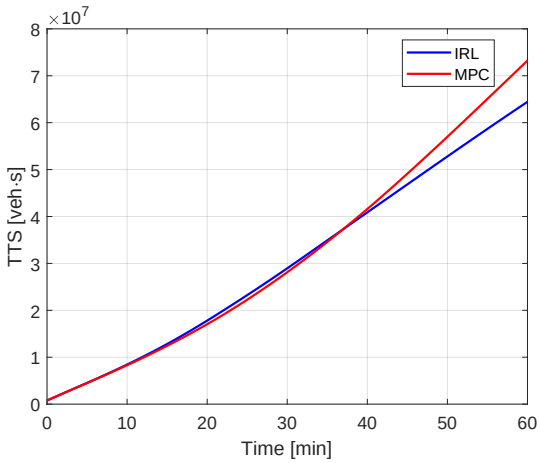
Figure 3.12 Perimeter control input evolution of Scenario 2



(a) TTS under nominal demand



(b) TTS under noisy demand



(c) TTS under abrupt-change demand

Figure 3.13 TTS evolution over time of Scenario 2

Table 3.4 Summary of TTS and computation time for Scenario 2

	nominal demand		noisy demand		abrupt-change demand	
	IRL	MPC	IRL	MPC	IRL	MPC
TTS ($\times 10^7$ veh·sec)	6.35	7.12	6.38	7.13	6.44	7.32
CPU time per step (sec)	1.07×10^{-5}	5.85×10^{-1}	1.14×10^{-5}	5.93×10^{-1}	1.24×10^{-5}	5.99×10^{-1}

3.6 Microscopic simulation

To further demonstrate the applicability of the proposed IRL approach to perimeter control of MFD based networks, a microscopic simulation example is presented in this section. Both training of the proposed IRL algorithm and its performance evaluation are conducted using SUMO as the environment (Lopez et al., 2018). The simulation and calculation are implemented in Python 3.6 and MATLAB R2022a.

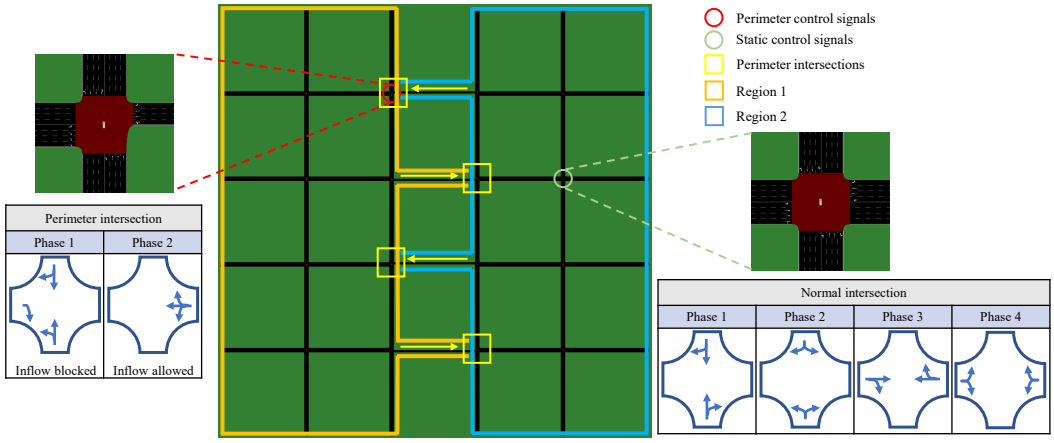


Figure 3.14 The simulated grid network

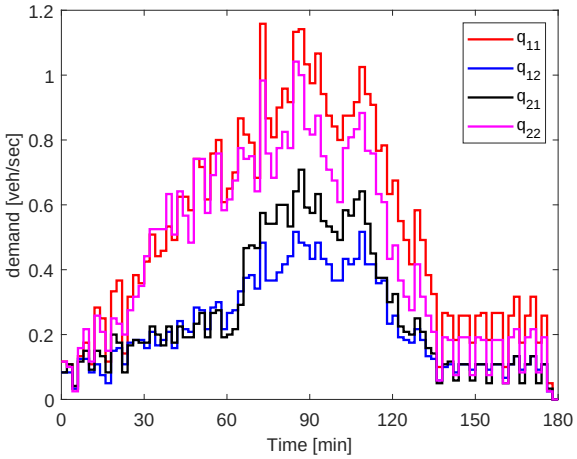
The microscopic simulation is carried out using a grid road network as depicted in Figure 3.14, which comprises 2 regions (regions 1 and 2 surrounded by orange lines and blue lines, respectively), 16 signalized intersections (12 normal intersections applying an identical static signal plan and 4 perimeter intersections applying the IRL based signal plans for perimeter control actuation) and 76 links. All links are 500 meters long and comprise 4 lanes. There are 4 special links connecting the two network regions as marked alongside yellow arrows. Note that these 4 links are unidirectional and that their end nodes are signalized intersections working as the perimeter controllers (as marked with yellow rectangles). For the perimeter intersections, a two-phased signal with a 120-second cycle time is adopted (see Figure 3.14). Both the sample time interval and the control update step are equal to the perimeter control signal cycle time. Perimeter control inputs are implemented by changing the green duration ratios of the corresponding perimeter intersections. Let $GR_{ij}(k)$ denotes the green duration ratio of phase 2 at the k-th ($k = 1, \ldots, 90$) control update step for actualizing the computed result of $u_{ij}(k)$ and $GR_{ij}(k) = u_{ij}(k)$. The calculation of the green light duration $\mathcal{G}_{ij}(k)$ of phase 2 is $\mathcal{G}_{ij}(k) = GR_{ij}(k) \cdot CT_{ij} = u_{ij}(k) \cdot CT_{ij}$ where $CT_{ij} = 120$ (sec) is the cycle time of the perimeter intersections for u_{ij}. For instance, if the computed result of a control

input is 0.6, the green light duration is set as 72 seconds for the 120-second cycle. All the normal intersections have four phases and the same cycle time 100 seconds. An identical static signal plan is adopted for the normal intersections. The total simulation time is 3 hours.

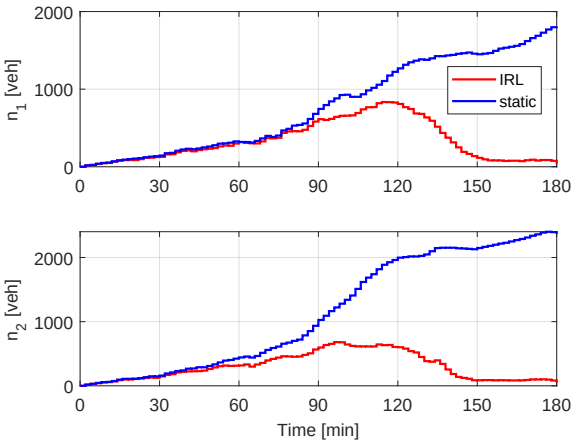
The perimeter control objective in this microscopic simulation is to minimize the TTS. Both regions are initially empty at the beginning. A time-varying travel demand pattern associated with a 10% coefficient of variation to represent the stochasticity is adopted, which mimics the morning-peak traffic with congestion onset and dissolving processes (see Figure 3.15(a)). The accumulation evolutions are depicted in Figure 3.15(b). As observed from the results, by merely manipulating the green duration ratios of the four perimeter intersections, the IRL scheme can regulate the accumulation states below the critical accumulation and achieve a significant improvement over the static scheme in avoiding congestion. Figure 3.15(c) shows that the IRL scheme can guarantee a low TTS at around 6.81×10^6 (veh·sec), while the static scheme results in a very high TTS at around 2.14×10^7 (veh·sec). The flow-accumulation plots by the proposed IRL approach and the static scheme are shown in Figure 3.16(a) and Figure 3.16(b), respectively. One can see that the IRL scheme results in a higher maximal throughput than the static scheme, whereas observed from the simulation process, the static scheme induces severe congestion and even gridlocks in both regions. These microscopic simulation results validate the effectiveness of the proposed IRL method in optimal perimeter control for MFD based traffic networks.

3.7 Conclusions

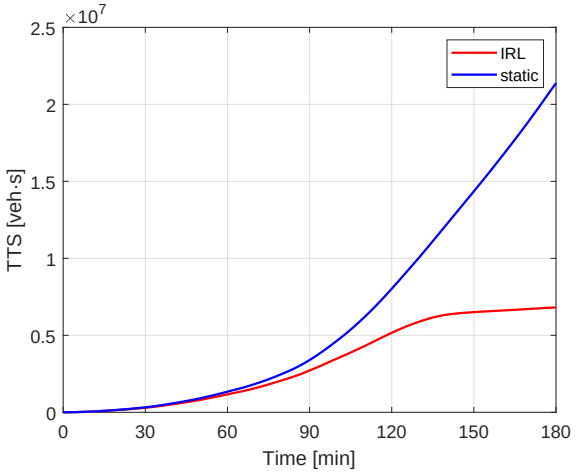
This study developed a data-driven IRL based framework for learning macroscopic urban traffic dynamics for adaptive constrained optimal perimeter control. An online adaptive optimal perimeter control scheme with continuous-time control and discrete gain updates was established to adapt to the discrete-time nature of traffic data. To further consider the heterogeneous traffic sensors with different resolutions of data measurements, the reinforcement interval of the proposed IRL based perimeter control could be selected online to ensure data richness for the data-driven RL algorithms and allow adaptive online learning to guarantee real-time performance. An actor-critic dual neural network structure was developed to approximate the optimal control and the objective function, respectively. The actor-critic dual neural networks could be used to circumvent the “curse of dimensionality”



(a) Demand pattern

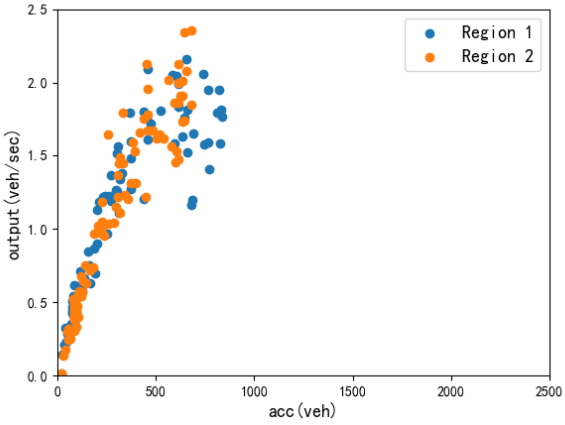


(b) State evolution

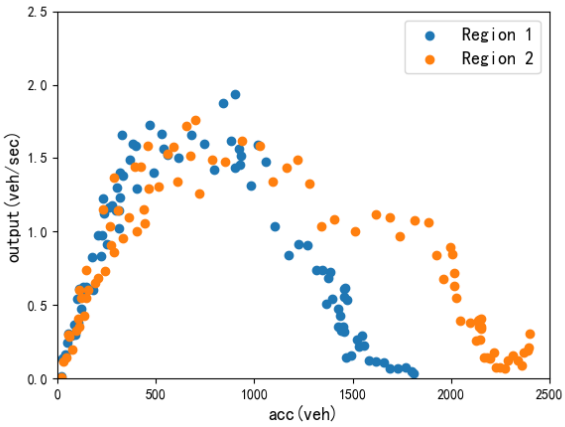


(c) TTS

Figure 3.15 The microscopic simulation results



(a) Using the IRL method



(b) Using the static scheme

Figure 3.16 Flow-accumulation plots in the microscopic simulation

in solving the HJB equations involved. Integrating the experience replay technique, the proposed online learning approach could adapt to the real-time traffic conditions by using the historical and real-time data simultaneously in a "smart" manner. This proposed optimal perimeter control did not explicitly employ the knowledge of traffic network dynamics, i.e., "model-free". The convergence of learning algorithms and the stability of the traffic dynamics under control were proven via the Lyapunov theory. The proposed online iterative learning approach was tested under various traffic conditions (e.g., constant demand, time-varying demand with and without uncertainties, unknown MFD model), where the convergence, adaptiveness and robustness of the network traffic state were achieved. Both numerical examples and microscopic simulation experiments were presented to validate the applicability

of the proposed method to optimal perimeter control for MFD based networks. In addition, the comparison results indicated that the proposed IRL approach could achieve both good control performance and computational efficiency.

Considering the dynamic nature of travel demand and supply, future efforts can be dedicated to investigating a novel trajectory stability concept, instead of the stability of the desired equilibrium point, to fit such dynamic travel demand and supply. In Chapter 4, we explore the trajectory stability of the MFD system and focus on the design of adaptive optimal tracking perimeter control.

4 Tracking perimeter control for two-region macroscopic traffic dynamics: An adaptive dynamic programming approach

The perimeter control by leveraging the concept of the macroscopic fundamental diagram (MFD) can alleviate network-level congestion by identifying critical intersections and regulating them effectively. Considering the time-varying nature of the travel demand pattern and the equilibrium of the accumulation state, this study attempts to reformulate the conventional set-point perimeter control (SPC) problem for the two-region MFD system into an optimal tracking perimeter control problem (OTPCP). Unlike the SPC schemes that stabilize the traffic dynamics to the desired equilibrium point, the proposed tracking perimeter control (TPC) scheme will regulate the traffic dynamics to a desired trajectory in a differential framework. Deriving the augmented system and the tracking Hamilton-Jacobi-Bellman (HJB) equation takes center in solving the OTPCP. Due to the inherent network uncertainties, such as uncertain dynamics of heterogeneity and demand disturbance, the system dynamics could be uncertain or even unknown. To address these issues, this study will propose an adaptive dynamic programming (ADP) approach to solving the tracking HJB equation, which requires no knowledge of the system dynamics. Finally, numerical experiments will be performed to demonstrate the effectiveness of the proposed ADP-based TPC. Compared with the SPC scheme, results will show that the proposed TPC scheme achieves both an improvement in reducing total travel time and an enhancement in cumulative trip completion.

4.1 Introduction

The adoption of the macroscopic fundamental diagrams (MFDs) to model and regulate the traffic flow of large-scale urban networks has been extensively studied in the last decade (Haddad and Geroliminis, 2012; Haddad et al., 2013; Keyvan-Ekbatani et al., 2013; Leclercq et al., 2014; Yildirimoglu and Geroliminis, 2014). The MFD intuitively provides an aggregate, low-scatter relationship between the network vehicle density (veh/km) or accumulation (veh) and network outflow or

trip completion flow rate (veh/h). Leveraging the concept of MFDs, the perimeter control aims to manipulate the transfer flow at the boundaries of the region, which is a promising solution to alleviating network-scale traffic congestion. Considerable research efforts have been dedicated to devising optimal network traffic control strategies based on MFDs. Apart from previous literature on maximizing the network traffic throughput by leveraging perimeter control (Daganzo, 2007; Geroliminis et al., 2013; Ramezani et al., 2015; Haddad et al., 2013; Zhou et al., 2016; Fu et al., 2017; Aalipour et al., 2018), considerable research efforts have been devoted to devising perimeter control strategies that regulate the network accumulation to the desired equilibrium, i.e., set-point perimeter control (Aboudolas and Geroliminis, 2013; Keyvan-Ekbatani et al., 2012; Keyvan-Ekbatani et al., 2013; Keyvan-Ekbatani et al., 2015a; Keyvan-Ekbatani et al., 2015b; Haddad and Mirkin, 2016). The robust perimeter control problem of the MFD-based system was also addressed in previous studies, e.g., Haddad and Shraiber (2014), Haddad (2015), and Zhong et al. (2018a).

A critical assumption adopted in traffic control (including the perimeter control and signal control) is that the steady state of the system can be achieved and the equilibria of the system can be determined. Under this assumption, the stability of fixed equilibrium points in the sense of Lyapunov is widely applied in traffic control. Considering the dynamic nature of traffic demand and supply, especially for fast time-varying cases, identification of the steady state is an extremely difficult and unclear task in practice (Zhong et al., 2018a; Zhong et al., 2018b). Some recent studies have attempted to optimize set-points for traffic control by updating them based on real-time traffic state estimations/measurements (Wang et al., 2021; Mohajerpoor et al., 2020), and others by using data-driven approaches (Kouvelas et al., 2017) or Nash equilibrium seeking schemes (Kutadinata et al., 2016). Yu and Hou (2020) optimized the set-point using model predictive control (MPC) with MFD and used this set-point with an iterative learning method to design traffic signal timing plans. They conducted the planning and control synchronously, which may cause the curse of dimensionality in large-scale urban networks with many intersections that are managed distributedly. However, there is a complex and unclear relationship between network traffic performance and desired set point, with no unified or clear definition of the best set point in an ever-changing environment.

The aforementioned studies on perimeter control can be regarded as model-based traffic responsive control, which assumes that model parameters are accurately calibrated and perfect knowledge of the network is available. Note that traffic networks are subject to various uncertainties (e.g., demand noise and model error), making these assumptions difficult and even impossible to be met. Recently,

model-free methods, such as iterative learning control (Lei et al., 2019; Ren et al., 2020), model-free adaptive predictive control (Li and De Schutter, 2022), and deep reinforcement learning (RL) (Zhou and Gayah, 2021), have been proposed to address the problem of devising adaptive perimeter control strategies for MFD systems with unknown system dynamics. A reformulation of RL is called adaptive dynamic programming (ADP) in economics and management communities. The RL and ADP bridge the gap between optimal control and adaptive control. In an off-line manner, the ADP method provides an approximate solution to the optimal control problem obtained from the Pontryagin's minimum principle and the dynamic programming principle (i.e., the Hamilton-Jacobi-Bellman (HJB) equation). Su et al. (2020) proposed a conventional ADP-based perimeter controller for the two-region MFD system, which requires partial knowledge of the system dynamics and thus cannot handle modeling uncertainties. Recently, we developed a completely model-free integral reinforcement learning (IRL) approach integrated with experience replay for adaptive perimeter control of multi-region MFD systems, which enables online tuning of the reinforcement interval to adapt to the real-time heterogeneous data resolution. This study further extends the results of Study 1 in terms of the stability of a single equilibrium (or its invariant set) to a desired trajectory.

In this study, we introduce a trajectory stability concept in the MFD framework to better fit the dynamic nature of traffic demand and supply. We leverage recent advances in control design based on the optimal tracking control theory that extends the stability analysis to constructive feedback perimeter control design through the concept of an augmented affine system. Unlike the conventional perimeter control schemes that stabilize the traffic dynamics to the desired equilibrium point, the proposed perimeter control scheme will regulate the traffic dynamics to a desired trajectory (e.g., time-varying with respect to the demand and supply) in a differential framework, instead of the stability of the desired equilibrium point or its invariant set. Few existing studies have addressed the tracking perimeter control (TPC) problem for MFD-based traffic networks except Haddad and Mirkin (2017). Based on this work, Haddad and Zheng (2020) investigated the effect of constant time delays. Local linearization around the desired equilibrium was performed in both works to simplify the controller design. Different from the existing works, we explicitly consider both state and control constraints in the solution of the TPC problem and no model linearization is required.

In this study, first, the optimal tracking perimeter control problem (OTPCP) is transformed into the minimization of a nonquadratic performance function subject to an augmented system composed of the original system and the command generator system. Then, an ADP algorithm is proposed to generate the optimal solution

to the associated HJB equation without complete knowledge of the augmented dynamics.

The remainder of this chapter is organized as follows: Section 4.2 proposes a novel problem formulation of the OTPCP for the two-region MFD system. Section 4.3 develops a model-free ADP approach to solving the OTPCP. Then numerical examples are presented in Section 4.4. Finally, Section 4.5 concludes this study.

4.2 Optimal tracking perimeter control of a two-region MFD system

In this section, we first present a recapitulation of the feedback-based set-point perimeter control (SPC) problem for a two-region MFD system. Considering the time-varying nature of the travel demand pattern and the equilibrium of accumulation state, we reformulate the conventional feedback-based SPC problem into an OTPCP. The standard solution to OTPCP for a two-region MFD system is given. Note that the standard solution to the OTPCP is to solve the steady-state part controller and the feedback part controller separately, which requires perfect knowledge of the system dynamics. To simplify the solution and to circumvent the requirement of perfect system information, we propose a reformulation of the OTPCP for the two-region MFD system, which converts the conventional way that solves the steady-state and feedback parts separately to merely solving a single optimal feedback control by deriving an augmented system and the associated tracking HJB equation.

4.2.1 The OTPCP for a two-region MFD system

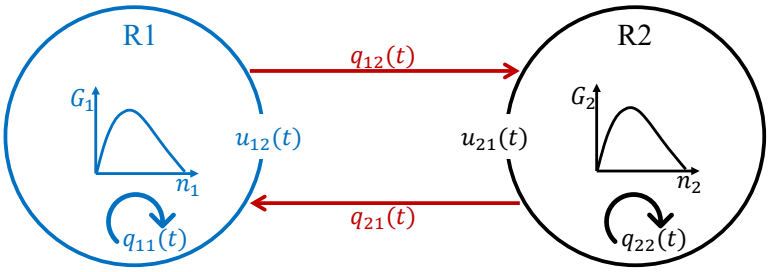


Figure 4.1 The two-region MFD system

To begin with, we recapitulate the two-region MFD system model. An urban network with two regions as shown in Figure 4.1 is of great significance in investigating gating control on the periphery. Assume that the urban network is composed of two homogeneous regions that both admit well-defined MFDs. A two-region MFD system can be used to model the macroscopic traffic dynamics. The MFD is a function that depicts a nonlinear relationship between the regional accumulation $n_i(t)$ (veh) and the trip completion rate $G_i(n_i(t))$ (veh/s) at time t, $i = 1,\ 2$. The regional accumulation $n_i(t)$ represents the number of vehicles in region i with $0 \leq n_i \leq n_i^{jam}$ where n_i^{jam} is the jam accumulation, i.e., the maximum vehicle number in region i. Let $q_{ij}(t)$ (veh/s) denote the travel demand generated in region i with destination to region j. By distinguishing whether the origin and destination of the travel demand are in the same region or not, the travel demand can be divided into endogenous and exogenous travel demand. Corresponding to the travel demand, four state variables, denoted by $n_{ij}(t)$ (veh) are identified. These state variables represent the accumulations contributed by the travel demand from region i to region j. By definition, we have $n_i(t) = \sum_j n_{ij}(t)$. Meanwhile, the perimeter control variables are introduced to the system, denoted as $u_{12}(t)$ and $u_{21}(t)$ with $0 \leq u_{ij}(t) \leq 1,\ i \neq j$, which are utilized to control the transfer flow between R1 and R2 on the border. The sending function of the transfer flow from region i with destination region j can be calculated by $\frac{n_{ij}(t)}{n_i(t)} G_i(n_i(t))$, and the completed transfer flow is determined by the perimeter control, i.e., $u_{ij}(t)\frac{n_{ij}(t)}{n_i(t)} G_i(n_i(t))$. On the other hand, the completed internal flow is defined as $\frac{n_{ii}(t)}{n_i(t)} G_i(n_i(t))$.

Now we introduce the formulation of the SPC problem. Based on flow conservation, the dynamics of the two-region MFD system can be regarded as a class of non-affine system

$$\dot{n}(t) = K(n(t), u(t), q(t)) \tag{4.1}$$

where $n(t) = [n_{11}(t), n_{12}(t), n_{21}(t), n_{22}(t)]^T \in \mathbb{R}_+^4$, $u(t) = [u_{12}(t), u_{21}(t)]^T \in \mathbb{R}_+^2$, and $q(t) = [q_{11}(t), q_{12}(t), q_{21}(t), q_{22}(t)]^T \in \mathbb{R}_+^4$ are the accumulation state of the system, the perimeter control, and the travel demand, respectively. Here $K(n, u, q)$ has the following well-known form (Geroliminis et al., 2013):

$$K(n,u,q) \triangleq \begin{bmatrix} -\frac{n_{11}}{n_1} G_1(n_1) + \frac{n_{21}}{n_2} G_2(n_2) u_{21} + q_{11} \\ -\frac{n_{12}}{n_1} G_1(n_1) u_{12} + q_{12} \\ -\frac{n_{21}}{n_2} G_2(n_2) u_{21} + q_{21} \\ -\frac{n_{22}}{n_2} G_2(n_2) + \frac{n_{12}}{n_1} G_1(n_1) u_{12} + q_{22} \end{bmatrix} \tag{4.2}$$

subject to

$$0 \leq n_i(t) \leq n_i^{jam},\ 0 \leq u_{ij}^{\min} \leq u_{ij}(t) \leq u_{ij}^{\max} \leq 1$$

For the SPC problem, the perimeter control is designed to manipulate the cross-boundary flows such that the accumulation state n can track a desired steady state. It is a common practice to perform a coordinate transformation to reformulate the SPC problem into a stabilization problem (Zhong et al., 2018a; Haddad and Zheng, 2020). Suppose that there exist an equilibrium $n^* = [n_{11}^*, n_{12}^*, n_{21}^*, n_{22}^*]^T$, $u^* = [u_{12}^*, u_{21}^*]^T$ and $q^* = [q_{11}^*, q_{12}^*, q_{21}^*, q_{22}^*]^T$ such that $\dot{n}^* = K(n^*, u^*, q^*) = 0$, i.e.,

$$\begin{bmatrix} -\frac{n_{11}^*}{n_1^*}G_1(n_1^*) + \frac{n_{21}^*}{n_2^*}G_2(n_2^*)u_{21}^* + q_{11}^* \\ -\frac{n_{12}^*}{n_1^*}G_1(n_1^*)u_{12}^* + q_{12}^* \\ -\frac{n_{21}^*}{n_2^*}G_2(n_2^*)u_{21}^* + q_{21}^* \\ -\frac{n_{22}^*}{n_2^*}G_2(n_2^*) + \frac{n_{12}^*}{n_1^*}G_1(n_1^*)u_{12}^* + q_{22}^* \end{bmatrix} = 0 \tag{4.3}$$

Let $\tilde{n}(t) = n(t) - n^*$ and $\tilde{u}(t) = u(t) - u^*$ denote the new state vector and new control input, respectively. Combining (4.2) and (4.3), we can rewrite the original traffic dynamics (4.1) as:

$$\dot{\tilde{n}}_{11} = -\frac{\tilde{n}_{11} + n_{11}^*}{\tilde{n}_1 + n_1^*}G_1(\tilde{n}_1 + n_1^*) + \frac{\tilde{n}_{21} + n_{21}^*}{\tilde{n}_2 + n_2^*}G_2(\tilde{n}_2 + n_2^*) \cdot (\tilde{u}_{21} + u_{21}^*) + q_{11} \tag{4.4a}$$

$$\dot{\tilde{n}}_{12} = -\frac{\tilde{n}_{12} + n_{12}^*}{\tilde{n}_1 + n_1^*}G_1(\tilde{n}_1 + n_1^*) \cdot (\tilde{u}_{12} + u_{12}^*) + q_{12} \tag{4.4b}$$

$$\dot{\tilde{n}}_{21} = -\frac{\tilde{n}_{21} + n_{21}^*}{\tilde{n}_2 + n_2^*}G_2(\tilde{n}_2 + n_2^*) \cdot (\tilde{u}_{21} + u_{21}^*) + q_{21} \tag{4.4c}$$

$$\dot{\tilde{n}}_{22} = -\frac{\tilde{n}_{22} + n_{22}^*}{\tilde{n}_2 + n_2^*}G_2(\tilde{n}_2 + n_2^*) + \frac{\tilde{n}_{12} + n_{12}^*}{\tilde{n}_1 + n_1^*}G_1(\tilde{n}_1 + n_1^*) \cdot (\tilde{u}_{12} + u_{12}^*) + q_{22} \tag{4.4d}$$

subject to

$$-n_i^* \le \tilde{n}_i(t) \le n_i^{jam} - n_i^*$$
$$-u_{ij}^* \le u_{ij}^{\min} - u_{ij}^* \le \tilde{u}_{ij}(t) \le u_{ij}^{\max} - u_{ij}^* \le 1 - u_{ij}^*$$

Now we present the formulation of OTPCP. We denote the reference trajectory by $\tilde{n}_d(t) = [\tilde{n}_{d,11}(t), \tilde{n}_{d,12}(t), \tilde{n}_{d,21}(t), \tilde{n}_{d,22}(t)]^T$. The trajectory $\tilde{n}_d(t)$ is assumed to be bounded and could be generated by the following Lipschitz continuous command generator dynamics (Modares and Lewis, 2014; Zhang et al., 2017):

$$\dot{\tilde{n}}_d = \theta(\tilde{n}_d(t)) \tag{4.5}$$

and $\theta(0) = 0$.

The target of the OTPCP is to find an optimal controller $\tilde{u}^*(t)$ to make the state $\tilde{n}$ can track the desired state $\tilde{n}_d$. We define $e_d(t) = \tilde{n}(t) - \tilde{n}_d(t)$ as the tracking error, and the tracking error dynamics are given by

$$\begin{aligned} \dot{e}_d &= \dot{\tilde{n}} - \dot{\tilde{n}}_d = K(n, u, q) - \theta(\tilde{n}_d) \\ &= K(e_d + \tilde{n}_d + n^*, \tilde{u} + u^*, q) - \theta(\tilde{n}_d) \end{aligned} \tag{4.6}$$

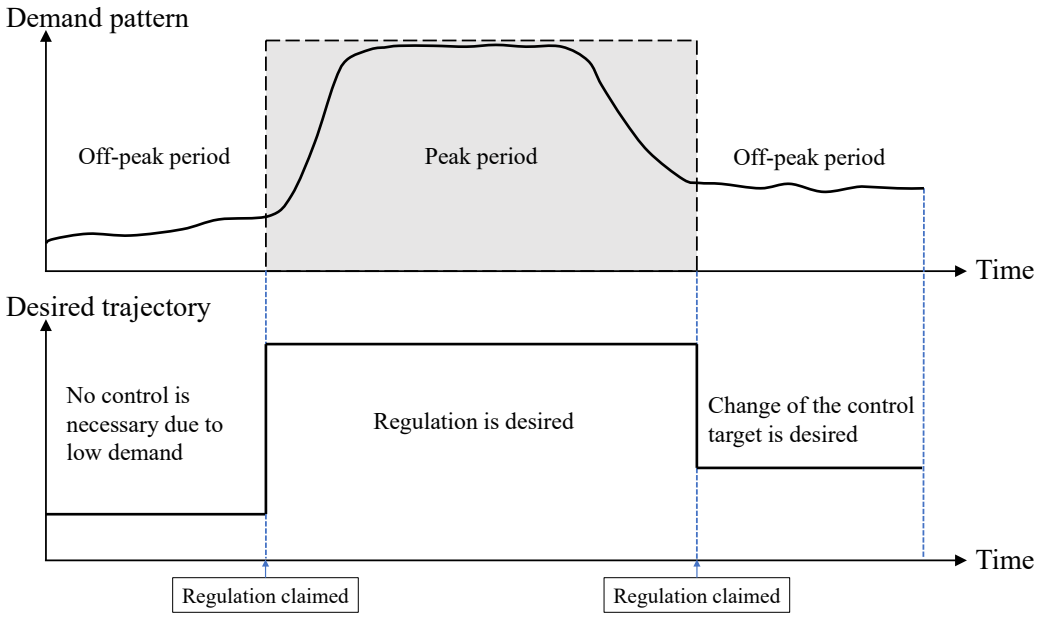


Figure 4.2 Demand pattern and desired state trajectory

Remark 4.2.1 For different periods of within-day traffic (i.e., off-peak period and peak period), the desired control targets should be different and fit the dynamics of the demand pattern. As shown in Figure 4.2, during the first off-peak period (e.g., 0:00–7:00), no control is necessary as the travel demand is low. Regulation is then claimed at the onset of the congestion. During the peak period (e.g., 7:00–11:00) with high travel demand, a steady accumulation state (target equilibrium) less than but close to the critical accumulation of the MFD is desired because operating the protected region around the critical accumulation maximizes its throughput (Zheng et al., 2016; Zhong et al., 2018b). After the peak-period congestion dissolves, the second off-peak period (e.g., 11:00–16:00) commences, during which the demand level is medium. It is not necessary to set the target equilibrium to be close to the critical accumulation that is larger than the steady state yielded by the demand pattern. Hence, the change of the control target is desired for the second off-peak period.

It is desired to design a reference signal associated with the performance of the network traffic flows. Moreover, a reference signal, if bounded by the desired invariant set of the steady states, would be more desirable because that means the control target is more achievable, controllable, and practical.

4.2.2 The Standard solution to the OTPCP

Note that the non-affine macroscopic traffic dynamics (4.4) can be expressed by (4.7).

$$\begin{aligned}\begin{bmatrix}\dot{\tilde{n}}_{11}\\ \dot{\tilde{n}}_{12}\\ \dot{\tilde{n}}_{21}\\ \dot{\tilde{n}}_{22}\end{bmatrix} = &\begin{bmatrix}-\frac{\tilde{n}_{11}+n^*_{11}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)+\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\cdot u^*_{21}+q_{11}\\ -\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)\cdot u^*_{12}+q_{12}\\ -\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\cdot u^*_{21}+q_{21}\\ -\frac{\tilde{n}_{22}+n^*_{22}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)+\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)\cdot u^*_{12}+q_{22}\end{bmatrix}\\ &+\begin{bmatrix}0 & \frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\\ -\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1) & 0\\ 0 & -\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\\ \frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1) & 0\end{bmatrix}\cdot\begin{bmatrix}\tilde{u}_{12}\\ \tilde{u}_{21}\end{bmatrix}\end{aligned} \tag{4.7}$$

Let $f(\tilde{n})$ and $s(\tilde{n})$ be defined by (4.8) and (4.9), respectively.

$$f(\tilde{n}) = \begin{bmatrix}-\frac{\tilde{n}_{11}+n^*_{11}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)+\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\cdot u^*_{21}+q_{11}\\ -\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)\cdot u^*_{12}+q_{12}\\ -\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\cdot u^*_{21}+q_{21}\\ -\frac{\tilde{n}_{22}+n^*_{22}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)+\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1)\cdot u^*_{12}+q_{22}\end{bmatrix} \tag{4.8}$$

$$s(\tilde{n}) = \begin{bmatrix}0 & \frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\\ -\frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1) & 0\\ 0 & -\frac{\tilde{n}_{21}+n^*_{21}}{\tilde{n}_2+n^*_2}G_2(\tilde{n}_2+n^*_2)\\ \frac{\tilde{n}_{12}+n^*_{12}}{\tilde{n}_1+n^*_1}G_1(\tilde{n}_1+n^*_1) & 0\end{bmatrix} \tag{4.9}$$

Then (4.4) can be rewritten as a standard affine form system as follows (Zhong et al., 2018a):

$$\dot{\tilde{n}} = f(\tilde{n}) + s(\tilde{n})\cdot\tilde{u} \tag{4.10}$$

As reported by Modares and Lewis (2014), the standard solution to the optimal tracking control problem is composed of two parts: 1) the steady-state part of the control input $\tilde{u}_s(t)$ that guarantees perfect tracking of the reference trajectory, and 2) the feedback part of the control input $\mu(t)$ that stabilizes the tracking error dynamics in an optimal manner, i.e.,

$$\tilde{u}(t) = \tilde{u}_s(t) + \mu(t)$$

First, suppose perfect information on the dynamics is available and the inverse of the input dynamics $s^{-1}(\tilde{n}_d)$ exists, the desired reference trajectory $\tilde{n}_d(t)$ can be presented by

$$\dot{\tilde{n}}_d = f(\tilde{n}_d) + s(\tilde{n}_d) \cdot \tilde{u}_s \tag{4.11}$$

Hence, based on (4.5) and (4.11), we have

$$\theta(\tilde{n}_d) = f(\tilde{n}_d) + s(\tilde{n}_d) \cdot \tilde{u}_s$$

Then the steady-state part control input $\tilde{u}_s(t)$ can be obtained by

$$\tilde{u}_s = s^{-1}(\tilde{n}_d) \cdot (\theta(\tilde{n}_d) - f(\tilde{n}_d)) \tag{4.12}$$

As reported by Zhang et al. (2018), if $s^{-1}(\tilde{n}_d)$ does not exist, $\tilde{u}_s(t)$ can be developed by

$$\tilde{u}_s = [s(\tilde{n}_d)^T s(\tilde{n}_d)]^{-1} s(\tilde{n}_d)^T [\theta(\tilde{n}_d) - f(\tilde{n}_d)] \tag{4.13}$$

Second, the feedback part of the control $\mu(t)$ can be obtained by minimizing the following performance function:

$$V(t) = \int_t^{\infty} r(e_d(\tau), \mu(\tau)) \mathrm{d}\tau \tag{4.14}$$

where $r(e_d, \mu) = Q_d(e_d) + U(\mu)$, $Q_d(e_d) = e_d^T Q e_d$, Q is a symmetric positive definite matrix of proper dimension and $U(\mu)$ is a positive definite function.

4.2.3 A reformulation for OTPCP

In this subsection, an augmented system for the OTPCP is presented. First, the OTPCP is transformed into the minimization of a nonquadratic performance function subject to an augmented system composed of the original system and the command generator system. Then a tracking Hamilton-Jacobi-Bellman (HJB) equation for the augmented system is derived.

According to the affine system (4.10) and the reference trajectory (4.5), we can define the augmented system state $N(t) = [e(t)^T, \tilde{n}_d(t)^T]^T$ and the augmented system as

$$\dot{N}(t) = F(N(t)) + S(N(t)) \cdot \mu(t) \tag{4.15}$$

where

$$F(N) = \begin{bmatrix} f(\tilde{n}) + s(\tilde{n})\tilde{u}_s - \theta(\tilde{n}_d) \\ \theta(\tilde{n}_d) \end{bmatrix}$$
$$S(N) = \begin{bmatrix} s(\tilde{n}) \\ 0 \end{bmatrix}$$

Considering $\tilde{n}(t) = e(t) + \tilde{n}_d(t)$, we have

$$F(N) = \begin{bmatrix} f(e + \tilde{n}_d) + s(e + \tilde{n}_d)\tilde{u}_s - \theta(\tilde{n}_d) \\ \theta(\tilde{n}_d) \end{bmatrix}$$
$$S(N) = \begin{bmatrix} s(e + \tilde{n}_d) \\ 0 \end{bmatrix}$$

For this augmented system, we introduce the following performance function.

$$V(N(t)) = \int_t^\infty \bar{Q}(N(\tau)) + U(\mu(\tau))\mathrm{d}\tau \tag{4.16}$$

where

$$\begin{aligned} \bar{Q}(N(t)) &= N(t)^T \hat{Q} N(t) \\ &= \begin{bmatrix} e_d \\ \tilde{n}_d \end{bmatrix}^T \begin{bmatrix} Q & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} e_d \\ \tilde{n}_d \end{bmatrix} \\ &= Q_d(e_d(t)) \end{aligned}$$

and $U(\mu)$ is a positive definite integrand function defined as

$$U(\mu) = 2\int_0^\mu \left(\lambda \tanh^{-1}\left(\frac{\upsilon}{\lambda}\right)\right) R d\upsilon \tag{4.17}$$

where $\upsilon \in \mathbb{R}^2$, λ is the saturating bound for the actuators and without loss of generality, $R = diag(\gamma_1, \gamma_2)$ is a positive semidefinite symmetric matrix. This nonquadratic performance function is used in the optimal regulation problem of constrained-input systems to deal with the input constraints. In fact, using this nonquadratic performance function, the following constraints are always satisfied, i.e., $|\mu_i(t)| \leq \lambda, \quad i = 1, 2$.

By constructing the augmented system (4.15), the conventional standard solution to the OTPCP is transformed into solving the optimal feedback part $\mu(N)$ for the augmented system, whereas the solution to the steady-state control $\tilde{u}_s$ has been substituted in the dynamics $F(N)$. It is worth noting that the performance functions $V(N)$ and $\mu(N)$ in this process are not linked to the reference trajectory.

The solution of the OTPCP of the two-region macroscopic traffic dynamics can be converted to solving the nonlinear tracking HJB equation. To begin with, we introduce the concept of admissible control.

Definition 4.2.1 A feedback control policy $\mu(N)$ is admissible with respect to (4.15), if the control $\mu \in \Lambda(\Omega)$, $\mu(0) = 0$, is continuous on Ω and stabilizes the error dynamics (4.6) with finite performance function $V(N)$, $\forall N \in \Omega$. ∎

Differentiating the performance function (4.16) along the augmented system (4.15), we can obtain the following tracking Bellman equation

$$\dot{V}(N(t)) = -N^T \hat{Q} N - U(\mu(N)) \tag{4.18}$$

Using (4.17), (4.18) can be further expressed as

$$0 = N^T \hat{Q} N + 2\int_0^{\mu} \left(\lambda \tanh^{-1}\left(\frac{\upsilon}{\lambda}\right)\right) R d\upsilon + \nabla V^T(N) \cdot (F(N) + S(N)\mu) \tag{4.19}$$

where $\nabla V(N)$ denotes the partial derivative of V with respect to the state N. Suppose V^* is the optimal value function. Then it satisfies the following tracking HJB equation

$$H(N, \mu, V^*) = N^T \hat{Q} N + 2\int_0^{\mu} \left(\lambda \tanh^{-1}\left(\frac{\upsilon}{\lambda}\right)\right) R d\upsilon + (\nabla V^*(N))^T \cdot (F(N) + S(N)\mu)$$

Applying the stationary condition $\partial H / \partial \mu^* = 0$, the optimal control policy is given by

$$\begin{aligned} \mu^*(N) &= \arg \min_{\mu \in \Lambda(\Omega)} H(N, \mu, V^*) \\ &= -\lambda \tanh\left(\frac{1}{2\lambda} R^{-1} S(N)^T \nabla V^*(N)\right) \end{aligned} \tag{4.20}$$

Substituting (4.20) into (4.17) results in

$$U(\mu^*) = \lambda \nabla V^{*T}(N) S(N) \tanh(D^*) + \lambda^2 \bar{R} \ln(\underline{\mathbf{1}} - \tanh^2(D^*)) \tag{4.21}$$

where $D^* = (1/2\lambda)R^{-1}S(N)^T\nabla V^*(N)$, $\underline{\mathbf{1}}$ is a column vector with all elements being ones and $\bar{R} = [\gamma_1, \gamma_2] \in \mathbb{R}^{1\times 2}$. Substituting (4.20) and (4.21) into (4.19) yields

$$H(N, \mu^*, \nabla V^*) = N^T\hat{Q}N + \nabla V^{*T}(N)F(N) + \lambda^2\bar{R}\ln(\underline{\mathbf{1}} - \tanh^2(D^*)) = 0 \quad (4.22)$$

To solve the OTPCP, one solves the tracking HJB equation (4.22) for the optimal value V^*. Then the optimal control is given as a feedback $\mu(V^*)$ in terms of (4.22) using (4.20).

4.3 Adaptive optimal tracking perimeter controller design

In this section, we propose a model-free ADP approach to solving the OTPCP for the two-region MFD system, i.e., solving the tracking HJB equation (4.22).

Due to the strong nonlinearity of the tracking HJB equation (4.22), it is extremely difficult to obtain the analytical solution to (4.22). The offline policy iteration method is one of the most common approaches to resolving this difficulty (Lewis and Vrabie, 2009). First, we revisit the offline policy iteration method, based on which the model-free ADP algorithm is derived, to solve the HJB equation (4.22). The principle of the offline policy iteration method consists of the following two iterative steps to calculate the Bellman equation and the optimal controller:

1. (Policy evaluation) Given an initial admissible control policy $\mu^{(0)}(N)$ and initial cost $V^{(0)} = 0$, find $V^{(k)}(N)$ successively approximated by solving the following equation

$$\begin{aligned} &N^T\hat{Q}N + U(\mu^{(k)}(N)) + \left(\nabla V^{(k)}(N)\right)^T \\ &\cdot \left(F(N) + S(N)\mu^{(k)}(N)\right) = 0,\ k = 0, 1, \ldots \end{aligned} \quad (4.23)$$

2. (Policy improvement) Update the control policy simultaneously by

$$\begin{aligned} \mu^{(k+1)}(N) &= -\lambda\tanh\left(D^{(k+1)}\right) \\ D^{(k+1)} &= \frac{1}{2\lambda}R^{-1}S(N)^T\nabla V^{(k)}(N) \end{aligned} \quad (4.24)$$

where k is the iterative index. Proof of convergence of this offline policy iteration is similar to that of Lemma 3.3.1.

Due to the inherent network uncertainties, such as uncertain dynamics of heterogeneity and demand disturbance, the MFD parameters could be time-varying and uncertain, i.e., $F(N)$ and $S(N)$ could be uncertain and even unknown. To implement the model-free method, an improved data-driven algorithm is developed by eliminating the dynamics in the iteration procedure. Denote $\mu^{(k)}$ as the policy to be updated and μ as the behavior policy that is actually implemented to generate the data for learning. Then we can rewrite the augmented system as:

$$\dot{N} = F(N) + S(N) \cdot \mu^{(k)} + S(N) \cdot \left(\mu - \mu^{(k)}\right) \tag{4.25}$$

Taking the derivative of $V^{(k+1)}(N)$ along the system trajectory (4.25) yields

$$\begin{aligned}\frac{\mathrm{d}V^{(k+1)}(N)}{\mathrm{d}t} &= \left(\nabla V^{(k+1)}(N)\right)^T \left(F(N) + S(N)\mu^{(k)} + S(N)\left(\mu - \mu^{(k)}\right)\right) \\ &= \left(\nabla V^{(k+1)}(N)\right)^T \left(F(N) + S(N)\mu^{(k)}\right) \\ &\quad + \left(\nabla V^{(k+1)}(N)\right)^T S(N)\left(\mu - \mu^{(k)}\right)\end{aligned}$$

According to (4.23)-(4.24), we have

$$\begin{aligned}\frac{\mathrm{d}V^{(k+1)}(N)}{\mathrm{d}t} =& - N^T \hat{Q} N - 2\lambda \int_0^{\mu^{(k)}} \left(\tanh^{-1}(\frac{\upsilon}{\lambda})\right)^T R \mathrm{d}\upsilon \\ &+ 2\lambda \left(\tanh^{-1}\left(\frac{\mu^{(k+1)}}{\lambda}\right)\right)^T \cdot R\left(\mu^{(k)} - \mu\right)\end{aligned} \tag{4.26}$$

Integrating both sides of (4.26) over the time interval $[t, t+\Delta t]$, the ADP algorithm is obtained, which is detailed by (4.27) in Algorithm 1.

From (4.27), we can see that the proposed ADP algorithm does not require any information on the system dynamics. $V^{(k+1)}$ and $\mu^{(k+1)}$ are solved simultaneously using only the collected system data. By now, we have extended the results in Chen et al. (2022) from SPC to TPC.

Algorithm 1 ADP algorithm

Input: initial admissible control policy $\mu^{(0)}(N)$ and initial cost $V^{(0)} = 0$

Output: $V^{(k)}(N)$ and $\mu^{(k)}(N)$

According to the control policy $\mu^{(k)}$, $\mu^{(k+1)}$ and $V^{(k+1)}$ can be solved simultaneously as follows:

$$\begin{aligned} & V^{(k+1)}(N(t+\Delta t)) - V^{(k+1)}(N(t)) = \\ & -\int_t^{t+\Delta t} \left(N(\tau)^T \hat{Q} N(\tau) + 2\lambda \int_0^{\mu^{(k)}(\tau)} (\tanh^{-1}(\upsilon/\lambda))^T R \mathrm{d}\upsilon \right) \mathrm{d}\tau \\ & + 2\lambda \int_t^{t+\Delta t} \left(\tanh^{-1}\left(\mu^{(k+1)}(N(\tau))/\lambda\right)\right)^T \cdot \\ & R\left(\mu^{(k)}(N(\tau)) - \mu(\tau)\right) \mathrm{d}\tau \end{aligned} \tag{4.27}$$

On convergence, set $V^{(k+1)}(N) = V^{(k)}(N)$ and the optimal control is $\mu^* = \mu^{(k+1)}(N)$.

4.4 Numerical experiments

To show the validity of the proposed ADP algorithm for optimal tracking perimeter control of the two-region MFD system, we provide two illustrative examples. In both examples, the MFD functions for the two regions are assumed to be the same, which are in line with those in Haddad (2015), i.e.,

$$G_i(n_i) = \frac{1.4877 \cdot 10^{-7} n_i^3 - 2.9815 \cdot 10^{-3} n_i^2 + 15.0912 n_i}{3600} \tag{4.28}$$

For both regions, based on (4.28), the jam accumulation is $n_i^{jam} = 10000$ (veh), the maximum trip completion rate (i.e., the maximum throughput) is $G_i^{\max} = 6.3$ (veh/s), and the according critical accumulation state is $n_i^{cr} = 3392$ (veh). The sample time interval is 60 s.

4.4.1 Example 1: Time-varying travel demand

In Example 1, we mimic a realistic scenario of peak-hour traffic. For different periods of peak-hour traffic (e.g., congestion onset, stationary congestion, and congestion dissolving), the desired control targets should be different and fit the dynamic nature

of the travel demand. For the first hour, the travel demand is at a medium level, i.e., $q(t) = [1.2, 1.6, 1.0, 1.4]^T$ (veh/s) and the set points are $[n_1^*, n_2^*] = [2000, 2000]$ (veh), i.e., around 60% of the critical accumulation states. For the next 2.5 hours with stationary congestion, the travel demand is $q(t) = [1.6, 1.6, 1.6, 1.6]^T$ (veh/s) and the set points are set to be $[n_1^*, n_2^*] = [3000, 3000]$ (veh), i.e., around 90% of the critical accumulation states. After that, as the congestion dissolves, the travel demand decreases to $q(t) = [0.9, 0.9, 0.9, 0.9]^T$ (veh/s), and the set points are modified to a much lower level, i.e., $[n_1^*, n_2^*] = [1500, 1500]$ (veh). Based on (4.3), the corresponding equilibrium points of the accumulation and the perimeter control input are given in Table 4.1. The initial OD-specific initial accumulations are $n_{11}(0) = 450$ (veh), $n_{12}(0) = 1050$ (veh), $n_{21} = 1750$ (veh), $n_{22}(0) = 750$ (veh).

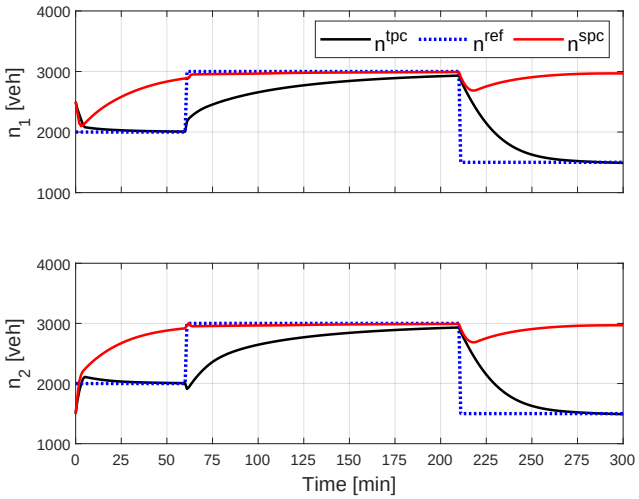
Table 4.1 The equilibrium points of Example 1

Time	$[n_{11}^*, n_{12}^*, n_{21}^*, n_{22}^*]$	$[u_{12}^*, u_{21}^*]$
0:00-1:00	$[814.5, 1185.5, 889.3, 1110.7]$	$[0.50, 0.42]$
1:01-3:30	$[1538.9, 1461.1, 1461.1, 1538.9]$	$[0.53, 0.53]$
3:31-5:00	$[591.6, 908.4, 908.4, 591.6]$	$[0.33, 0.33]$

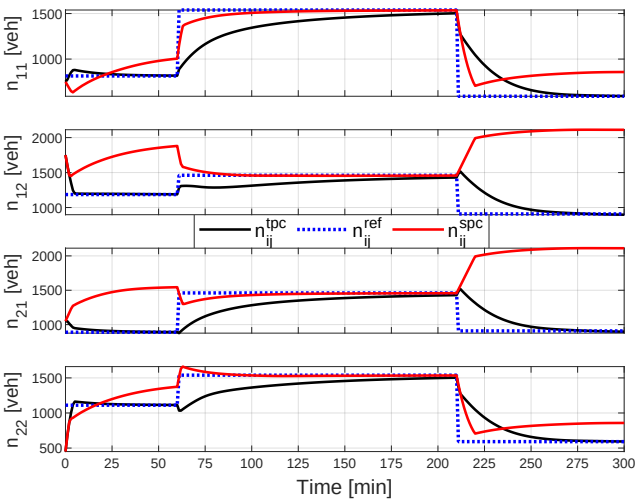
In Example 1, we compare the performance of the proposed ADP-based TPC against that of the ADP-based SPC. The SPC aims to track a fixed set point $[n_1^*, n_2^*] = [3000, 3000]$ (veh) regardless of the changes in the demand pattern. The results of the accumulation state evolution, OD-specific state evolution, and the control input of Example 1 are shown in Figure 4.3(a), Figure 4.3(b) and Figure 4.4, respectively.

Figure 4.3(a) and Figure 4.3(b) indicate that the proposed ADP-based TPC scheme can adapt to the changes of the travel demand pattern and regulate the accumulation states (black solid lines) to the corresponding set points (blue dotted lines). Instead of tracking the time-dependent reference trajectory, the SPC scheme succeeds in stabilizing the accumulation states to a fixed set-point in this time-varying demand case (red solid lines). As shown in Figure 4.4(a), for the first hour, the TPC inputs converge very fast to the first equilibrium point. Then the TPC shows a fast chattering behavior respectively at the beginning of the second hour and the beginning of the last 1.5 hours. This is because the TPC scheme captures the changes in the demand pattern and attempts to track the desired trajectory that better fits the dynamic nature of the demand. For the SPC scheme (see Figure 4.4(b)), the chattering behavior at the beginning of the first hour is greater but much milder hereafter.

A summary of two important performance indices: 1) minimizing the total time spent (TTS) and 2) maximizing the cumulative trip completion (CTC), achieved by

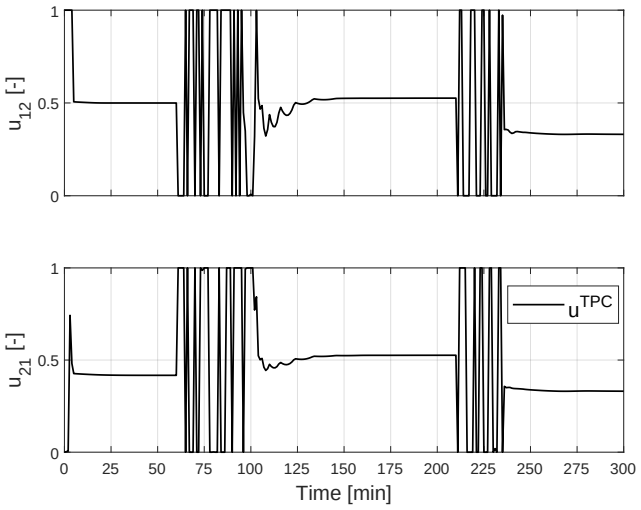


(a) Regional state $n_i(t)$

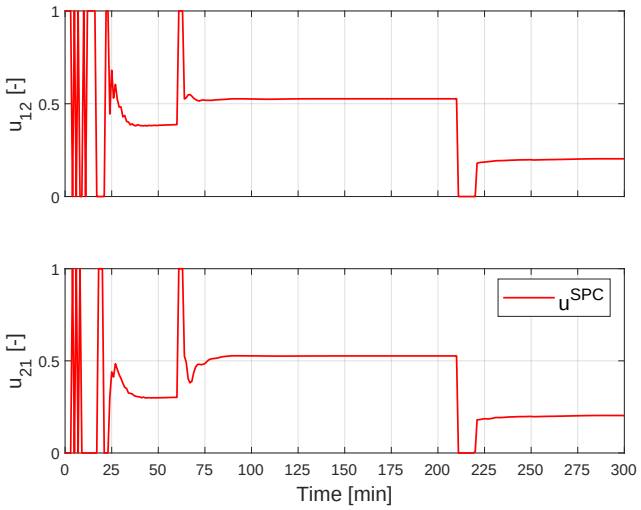


(b) OD-specific state $n_{ij}(t)$

Figure 4.3 Accumulation state evolutions of Example 1



(a) Tracking perimeter control



(b) Set-point perimeter control

Figure 4.4 Perimeter control inputs of Example 1

Table 4.2 Performance in TTS (veh·s) and CTC (veh) of Example 1

Controller	TTS ($\times 1e7$ veh·s)	CTC ($\times 1e4$ veh)
TPC	8.311 (-20.01%)	9.698 (+3.15%)
SPC	10.391 (-)	9.402 (-)

the two controllers is given by Table 4.2. The performance comparison is depicted in Figure 4.5. Note that the proposed TPC scheme achieves a 20.01% reduction in TTS compared with the SPC scheme. When performing congestion offset, the SPC scheme restricts the transfer flows between the two regions to keep regulating the accumulation state to the critical point, which makes the network denser and leads

to unnecessary travel delays. On the contrary, by defining a reference signal that is bounded by more practical control targets, the proposed TPC scheme not only achieves a significant improvement in minimizing the TTS compared with the SPC strategy, but also can outperform the latter in facilitating the CTC.

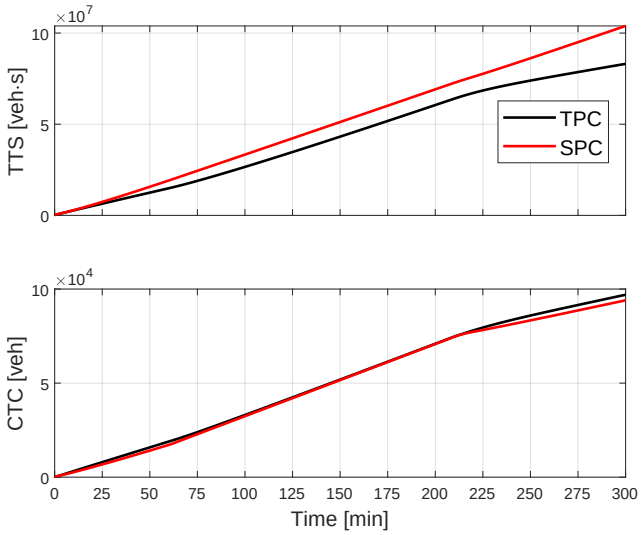


Figure 4.5 Performances in minTTS and maxCTC of Example 1

4.4.2 Example 2: Time-varying accumulation reference trajectories

Note that the accumulation reference trajectories in Example 1 are a priori given constant accumulation points. Now in this case study, we test the proposed ADP tracking perimeter controller for time-varying accumulation reference trajectories. A sinusoidal wave with amplitude $10\sqrt{5}$ (veh) and period $80\sqrt{5}\pi$ (s) is adopted for the reference trajectory of $\tilde{n}_{ij}(t)$. A fixed travel demand pattern is adopted, i.e., $q(t) = [1.56, 1.56, 1.56, 1.56]^T$ (veh/s). The set-point is selected as $[n_1^*, n_2^*] = [3000, 3000]$ (veh). Based on (4.3), the corresponding equilibrium points of the OD-specific accumulation state and the perimeter control input are $n^* = [1500.5, 1499.5, 1499.5, 1500.5]^T$ (veh) and $u^* = [0.5003, 0.5003]^T$, respectively. The initial OD-specific initial accumulations are $n_{11}(0) = 540$ (veh), $n_{12}(0) = 1260$ (veh), $n_{21} = 2170$ (veh), $n_{22}(0) = 930$ (veh).

In this case study, we compare the performance of the proposed model-free ADP-based TPC and the tracking controller solved by the standard solution assuming perfect system knowledge available (i.e., (4.12) for the feedforward part and (4.20)

for the feedback part). The results of the accumulation state evolution and the control input of Example 2 are shown in Figure 4.6 and Figure 4.7, respectively. As observed from the accumulation state evolution, the proposed ADP-based TPC scheme achieves a comparative performance with the standard TPC scheme. These results demonstrate the effectiveness of the model-free proposed ADP-based TPC in tracking the time-varying accumulation reference trajectories.

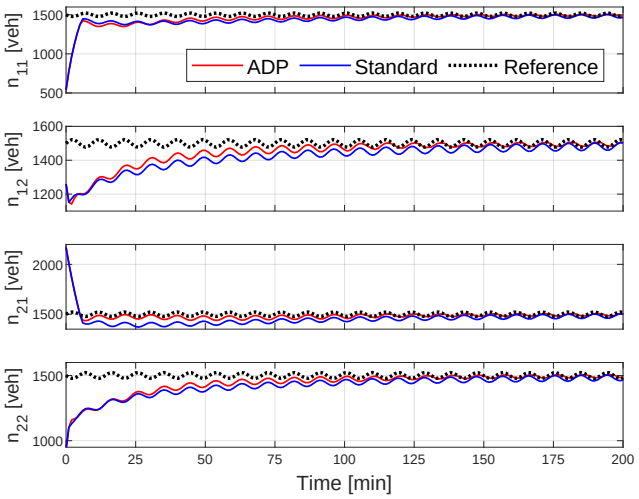


Figure 4.6 Accumulation state evolutions of Example 2

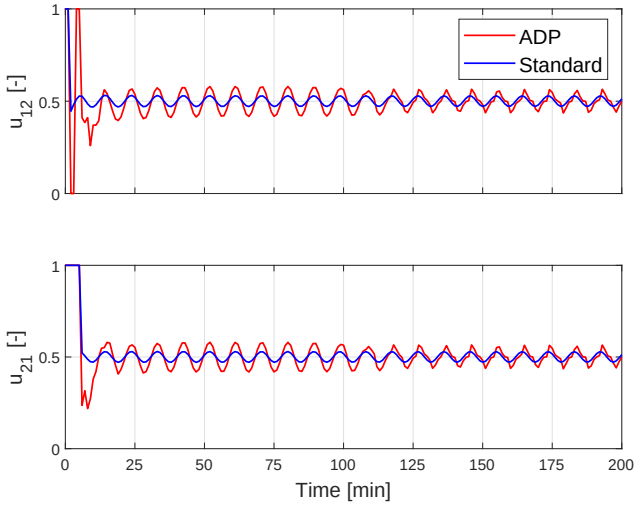


Figure 4.7 Perimeter control inputs of Example 2

4.5 Conclusion

This chapter leverages the trajectory stability concept that can better fit the dynamic nature of traffic demand to devise an adaptive tracking perimeter control (TPC) strategy for two-region macroscopic traffic dynamics. An offline learning adaptive dynamic programming (ADP) approach integrated with the AC-NN framework is proposed to approximate the optimal solution to the OTPCP, which requires no knowledge of the macroscopic traffic dynamics, i.e., model-free. Compared with the traditional set-point perimeter control (SPC) scheme that tracks a pre-defined set point, the proposed ADP-based TPC scheme can well adapt to the changes in the traffic condition (e.g., time-varying travel demand) and regulate the accumulation state to a desired reference trajectory that better fits the dynamics of the demand. Numerical experiments demonstrate not only the effectiveness of the ADP-based TPC in optimal tracking control of the two-region MFD system but also the improvement in network traffic efficiency (e.g., minTTS and maxCTC) compared to the SPC scheme.

The adaptive optimal perimeter control schemes investigated in Chapter 3 and Chapter 4 are limited in region size. With the increase in region size, regional route guidance systems can be integrated into network-level traffic management. Coupling perimeter control and route guidance is believed to enhance network traffic mobility while bringing challenges to data-driven traffic controller design. A major hurdle in optimizing the network traffic performance is the dissimilarity between the plant dynamics that represent the reality and the model used for optimization. Chapter 5 attempts to tackle the difficulties in devising adaptive perimeter control and regional route guidance strategies in case of model-plant mismatch.

5 An iterative adaptive dynamic programming approach for macroscopic fundamental diagram-based perimeter control and route guidance

Macroscopic fundamental diagrams (MFDs) have been widely adopted to model the traffic flow of large-scale urban networks. Coupling perimeter control and regional route guidance (PCRG) is a promising strategy to decrease congestion heterogeneity and reduce delays in large-scale MFD-based urban networks. For MFD-based PCRG, one needs to distinguish between the dynamics of (a) the plant that represents reality and is used as the simulation tool, and (b) the model that contains easier-to-measure states than the plant and is used for devising controllers, i.e., the model-plant mismatch should be considered. Traditional model-based methods (e.g., model predictive control (MPC)) require an accurate representation of the plant dynamics as the prediction model. However, due to the inherent network uncertainties, such as uncertain dynamics of heterogeneity and demand disturbance, MFD parameters could be time-varying and uncertain. On the other hand, existing data-driven methods (e.g., reinforcement learning) do not consider the model-plant mismatch and the limited access to plant-generated data, e.g., subregional OD-specific accumulations. Therefore, we aim to develop an iterative adaptive dynamic programming (IADP) based method to address the limited data source induced by the model-plant mismatch. An actor-critic neural network structure will be developed to circumvent the requirement of complete information on plant dynamics. Performance comparisons with other PCRG schemes under various scenarios will be carried out. The numerical results will indicate that the IADP controller trained with a limited data source can achieve comparable performance with the "benchmark" MPC approach using perfect measurements from the plant. The results will also validate the IADP's robustness against various uncertainties (e.g., demand noise, MFD error, and trip distance heterogeneity) when minimizing the total time spent in the urban network. These results can demonstrate the great potential of the proposed scheme in improving the efficiency of multi-region MFD systems.

5.1 Introduction

Depending on the network topology and partitioning, the urban network can be modeled as a single-region (Haddad and Shraiber, 2014), two-region (Zhong et al., 2018b), or multi-region MFD system (Sirmatel and Geroliminis, 2018). As the number of regions increases, regional route guidance is introduced in the MFD system to assist drivers in reaching their destinations. Leclercq and Geroliminis (2013) investigated the route choice in a two-bin MFD network with parallel routes which advises drivers a sequence of subregions that has a lower cost (in terms of travel time, fuel consumption, etc.) to improve the overall system performance. Dynamic user equilibrium (DUE) and dynamic system optimum (DSO) conditions for multi-region MFD system with regional route choice departure time choice were investigated in Huang et al. (2020) and Zhong et al. (2020), respectively.

The integration of perimeter control and regional route guidance (PCRG) is a promising approach to improving traffic efficiency in multi-region MFD-based networks. In the previous literature on PCRG strategies, Ramezani et al. (2015) is the first to distinguish the *model* and the *plant* when devising PCRG schemes for MFD-based urban networks. Following their work, we regard the region-based model (see Figure 5.1(a)) as the *model*, which considers an urban network partitioned into a small number of regions with scattered MFDs due to the link density heterogeneity. Moreover, we consider the more detailed subregion-based model (see Figure 5.1(b)) as the *plant*, which further divides the above regions into smaller subregions with low-scatter MFDs. The region-based *model* is essential for developing control strategies as it contains aggregated states that are easier to monitor than the more detailed *plant* states. The subregion-based *plant* with more detailed dynamics can replicate the reality wherein the developed strategies are actually implemented. Note that the *model* and the *plant* have different structures, which is one of the reasons for the model-plant mismatch. Therefore, it usually requires tedious translation of the regional control signals into the subregion ones (Ramezani et al., 2015; Yildirimoglu et al., 2018). Moreover, the subregion-based *plant* can incorporate a route choice model into the MFD framework (Yildirimoglu et al., 2015), whereas there is no route choice modeling in the region-based *model*. This is done on purpose not only to simplify the *model*, but also to create stronger model-plant disimilarity, which is more challenging when solving the optimal PCRG (OPCRG) problem.

Ramezani et al. (2015) reported that there exists a mismatch between the *model* and the *plant* induced by the link density heterogeneity. They found that ignoring the effect of such heterogeneity when designing optimal control schemes may lead

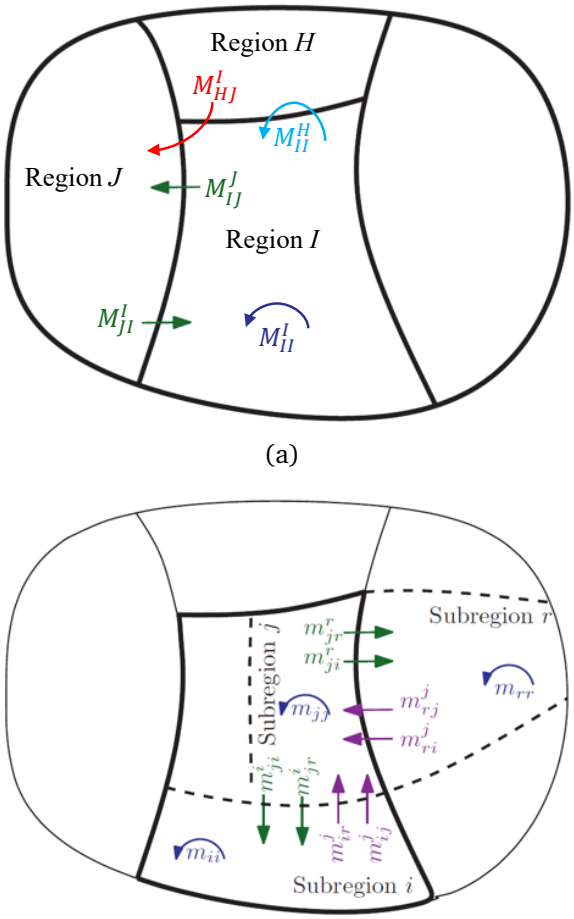


Figure 5.1 Network topology, reprinted (with modification) from Ramezani et al. (2015). (a) Region- and (b) Subregion-based models.

to non-optimal results. They proposed a hierarchical perimeter control framework to study the dynamics of heterogeneity. The heterogeneity effect is modeled using a negative binomial distribution and incorporated in the region-based model. However, such explicit heterogeneity modeling requires a large amount of high-resolution link-level data. Moreover, the regional MFD models need to be re-calibrated regularly to adapt to real-time traffic conditions, which can be a heavy computational burden. Therefore, an ADP based model-free optimal PCRG scheme is desired, which is also computationally efficient in implementation.

The dynamic regional trip distance is another cause of the model-plant mismatch. Many studies assumed that the average regional trip distance stays constant during the planning horizon (e.g., Geroliminis et al., 2013; Sirmatel and Geroliminis, 2018). In practice, the average regional trip distance changes over time. Batista et al.

(2021) proved that the constant average regional trip distance assumption could degrade the MPC control performance. They proposed an unscented Kalman filter (UKF) framework that dynamically adjusts the accumulations and trip distances to improve the prediction model of the MPC. To our best knowledge, few RL/ADP based PCRG schemes have studied the effect of the time-varying average regional trip distance on their control performance.

Previous studies on RL/ADP based perimeter control did not consider the limited access to the plant data (Su et al., 2020; Zhou and Gayah, 2021; Chen et al., 2022). Plant states such as the subregional OD-specific accumulation and travel demand are extremely difficult to measure. The regional states are more aggregated but easier to monitor. In this study, we investigate a scenario that has not been considered by previous RL/ADP PCRG schemes, wherein we develop a data-driven OPCRG scheme that can be directly applied in the subregion-based *plant* using merely the aggregated region-based data.

To address the aforementioned challenges, we develop an iterative adaptive dynamic programming (IADP) based OPCRG strategy for heterogeneous urban networks. An actor-critic neural network (AC-NN) framework is employed to simultaneously approximate the optimal value function associated with the HJB equation and to parametrize the adaptive OPCRG strategy. The principle contributions are summarized as follows:

- Model-free against the model-plant mismatch. Rather than eliminating the dissimilarity between the *model* and the *plant*, the proposed IADP approach tackles the model-plant mismatch by circumventing the necessity of perfect information on the system dynamics.
- OPCRG strategy trained with limited data. To our best knowledge, it is the first time for RL/ADP-based PCRG controllers to address the difficulties induced by the model-plant mismatch. It will be good to ensure that the data used for training are quite different than the ones used for testing. Despite being trained without the detailed subregional state data, the IADP approach can approximate the OPCRG scheme.
- Robustness against time-varying unknown regional trip distances. The IADP approach can well adapt to the dynamics of the heterogeneous regional trip distances and achieve a comparable performance with the MPC scheme with exact measurements of the average regional trip distances.

- Generalizability in different environments. We examine the performance of the IADP agent trained in an environment without uncertainties in various unseen scenarios (e.g., different levels of trip distance heterogeneity, MFD error, and demand noise). Without extra training, the IADP approach retains satisfactory control performances.
- Computational efficiency in MFD systems with a large region size. Sirmatel and Geroliminis (2018) and Yildirimoglu et al. (2018) reported that the MPC route guidance schemes cannot retain real-time feasibility when the network (sub)region size is much more than seven. As will be shown in the numerical experiments, the IADP approach is computationally efficient even when implemented in a sixteen-subregion network.

The remainder of the chapter is organized as follows: Section 5.2 introduces the MFD-based modeling of large-scale urban networks. In Section 5.3, we formulate the OPCRG problem and derive its standard solution based on the Bellman optimality principle. Then we propose a two-phase IADP-based approach for MFD-based OPCRG without exact plant dynamics. Section 5.4 presents numerical examples. Finally, Section 5.5 concludes the study.

5.2 MFD-based modeling of large-scale urban networks

In this section, we introduce the region-based *model* and the subregion-based *plant*, see Figure 5.1. Note that the IADP OPCRG scheme to be developed in the next section only uses partial information that mainly comes from the region-based *model*. On the other hand, the subregion-based *plant* is regarded as a black box and merely used for implementing the devised PCRG schemes. Following Yildirimoglu et al. (2015), the MFD model investigated in this study is the accumulation-based model.

5.2.1 Region-based model

First, we consider a network as a set of r regions denoted by $\mathcal{R} = \{1, 2, \ldots, r\}$. $Q_{IJ}(t)$ (veh/s) denotes the travel demand originating from Region I with a destination in Region J at time t. $N_{IJ}(t)$ (veh) represents the accumulation of vehicles in Region I

that are headed towards Region J. $N_I(t)$ (veh) denotes the total accumulation of vehicles in Region I, which is the summation of $N_{IJ}(t)$ over all regions in $\mathcal{R}$, i.e., $N_I(t) = \sum_{J\in\mathcal{R}} N_{IJ}(t)$; $I,\ J \in \mathcal{R}$.

Dynamics of an r-region MFDs network are given as follows (Yildirimoglu et al., 2015):

$$\begin{aligned}\dot{N}_{II}(t) =& Q_{II}(t) - M^I_{II}(t) - \sum_{H\in\mathcal{V}_I\setminus\{I\}} U_{IH}(t)\cdot M^H_{II}(t)\\ &+ \sum_{H\in\mathcal{V}_I\setminus\{I\}} U_{HI}(t)\cdot M^I_{HI}(t) \qquad (5.1a)\\ \dot{N}_{IJ}(t) =& Q_{IJ}(t) - \sum_{H\in\mathcal{V}_I\setminus\{I\};I\neq J} U_{IH}(t)\cdot M^H_{IJ}(t)\\ &+ \sum_{H\in\mathcal{V}_I\setminus\{I\};I\neq J} U_{HI}(t)\cdot M^I_{HJ}(t) \qquad (5.1b)\end{aligned}$$

where $\mathcal{V}_I$ is the set of regions that are directly reachable from Region I. U_{IH}, $0 \leq U_{IH} \leq 1$, denotes the perimeter controller, which exists between every two adjacent Regions I and H and constrains the transfer flows from I to H; $H \in \mathcal{V}_I\setminus\{I\}$. $M^H_{IJ}(t)$ represents the transfer flow for accumulation in Region I with a final destination in J through the next immediate Region H, while the internal trip completion rate for accumulation in Region I with a destination within I (without going through another region) is represented by $M^I_{II}(t)$ (veh/s). Note that paths including more than one crossing over the boundaries between the regions are permitted (e.g., the path of transfer flow M^H_{II} is $I \to H \to I$, see Figure 5.1(a)).

Internal trip completion rates and transfer flows are calculated corresponding to the ratio between accumulations as:

$$M^I_{II}(t) = \frac{N_{II}(t)}{N_I(t)} \cdot \frac{P_I\left(N_I(t), \sigma_I\left(N_I(t)\right)\right)}{L_{II}(t)} \qquad (5.2a)$$

$$M^H_{IJ}(t) = \frac{N_{IJ}(t)}{N_I(t)} \cdot \frac{P_I\left(N_I(t), \sigma_I\left(N_I(t)\right)\right)}{L_{IH}(t)} \qquad (5.2b)$$

where $P_I\left(N_I(t), \sigma_I\left(N_I(t)\right)\right)$ (veh·m/s) denotes the MFD production for Region I. $\sigma_I(N_I(t))$ is the heterogeneity variance at $N_I(t)$. $L_{II}(t)$ (m) and $L_{IH}(t)$ (m) denote the average trip lengths for trips in Region I and for trips from Region I to H, respectively. We will further discuss $\sigma_I(N_I(t))$ that captures the spatial heterogeneity of Region I in Section 5.2.4.

5.2.2 Subregion-based plant

Let us consider Region $I \in \mathcal{R}$ with spatial heterogeneity in subregional density, which consists of subregions as seen in Figure 5.1(b). In this chapter, capital letters and lowercase letters are used for variables related to regions and subregions, respectively. The subregion-based plant is more detailed but shares the same form of dynamics as the region-based model, following Ramezani et al. (2015).

Let $\mathcal{SR}$ be the set of all subregions in $\mathcal{R}$. $q_{ij}(t)$ (veh/s) denotes the demand from Subregion i to j. $n_{ij}(t)$ (veh) represents the accumulation in Subregion i with destination in Subregion j. $n_i(t)$ (veh) is the total accumulation in Subregion i and $n_i(t) = \sum_{j \in \mathcal{SR}} n_{ij}(t)$. $p_i(t)$ (veh·m/s) defines the MFD production for Subregion i, which is the total distance traveled by all vehicles in Subregion i and equal to the sum of the transfer and internal flows multiplied by the average trip length l_i (m) in Subregion i. $u_{ij} \in [0, 1]$ is the perimeter controller that controls the transfer flows on the border between Subregions i and j.

For $\forall i \in \mathcal{SR}$, the mass conservation equations for the subregions are given as follows:

$$\dot{n}_{ii}(t) = q_{ii}(t) - m_{ii}(t) + \sum_{h \in \mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hi}(t) \tag{5.3a}$$

$$\dot{n}_{ij}(t) = q_{ij}(t) - \sum_{h \in \mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ij}(t) + \sum_{h \in \mathcal{H}_i; h \neq j} u_{hi}(t) \cdot \hat{m}^i_{hj}(t), \quad \forall\, j \in \mathcal{H}_i \backslash \{i\} \tag{5.3b}$$

$$\dot{n}_{ir}(t) = q_{ir}(t) - \sum_{h \in \mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ir}(t) + \sum_{h \in \mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hr}(t), \quad \forall\, r \in \mathcal{SR} \backslash \mathcal{H}_i \tag{5.3c}$$

where $m_{ii}(t)$ (veh/s) denotes the transfer flow from Subregion i with final destination Subregion i, while $m^h_{ij}(t)$ (veh/s) is the transfer flow for accumulation in i with final destination j through the next immediate Subregion h, $h \in \mathcal{H}_i$ with $\mathcal{H}_i$ the set of subregions that are directly reachable from Subregion i. $m_{ii}(t)$ and $m^h_{ij}(t)$ are defined respectively as follows:

$$m_{ii}(t) = \frac{n_{ii}(t)}{n_i(t)} \cdot \frac{p_i(n_i(t))}{l_i} \tag{5.4a}$$

$$m^h_{ij}(t) = \theta^h_{ij}(t) \cdot \frac{n_{ij}(t)}{n_i(t)} \cdot \frac{p_i(n_i(t))}{l_i} \tag{5.4b}$$

where $\theta^h_{ij}(t) \in [0, 1]$ is the transfer flow ratio from Subregion i with destination in Subregion j that goes immediately through Subregion h, and hence $\sum_{h \in \mathcal{H}_i} \theta^h_{ij}(t) = 1$.

Note that high accumulation in a subregion can limit the inflow from the boundary. Therefore, the definition of capacity-restricted transfer flow from Subregion i to j passing through h immediately, $\hat{m}_{ij}^h(t)$, is introduced (Ramezani et al., 2015; Yildirimoglu et al., 2015; Sirmatel and Geroliminis, 2018):

$$\hat{m}_{ij}^h(t) = \min\left[m_{ij}^h(t),\ \frac{m_{ij}^h(t)}{\sum_{k\in\mathcal{SR};k\neq i} m_{ik}^h(t)} \cdot r_{ih}(n_h(t))\right] \tag{5.5}$$

where $r_{ih}(\cdot)$ (veh/s) is the receiving flow capacity of Subregion $h \in \mathcal{H}_i$, from Subregion i. We consider that the receiving capacity is a piecewise function of $n_h(t)$ as follows:

$$r_{ih}(n_h(t)) = \begin{cases} r_{ih}^{\max}, & 0 \le n_h(t) \le \alpha \cdot n_h^{jam} \\ -\frac{r_{ih}^{\max}}{(1-\alpha)\cdot n_h^{jam}} \cdot n_h(t) + \frac{r_{ih}^{\max}}{1-\alpha}, & \alpha \cdot n_h^{jam} < n_h(t) \le n_h^{jam} \end{cases} \tag{5.6}$$

5.2.3 Transferring subregion-based control variables to region-based ones

This study aims at providing perimeter control and route guidance strategies in the subregion-based plant by utilizing the aggregated states $N_{IJ}(t)$ in the region-based model where the regional control decisions $U_{IJ}(t)$ are replaced with $u_{ij}(t)$ and $\theta_{ij}^h(t)$. This procedure requires the transfer of variables from the subregion-based plant to the region-based model.

Note from (5.3a)-(5.3c) that perimeter controllers exist between every two neighboring subregions. However, we do not intend to regulate inter-transfers between any two subregions, but only at the boundary of the region-based model. That is, subregions that are not attached to the boundary between the regions, will not be controlled and the according inputs are set to be 1. Unlike Ramezani et al. (2015) that developed a hierarchical perimeter control framework, we focus on the design of low-level perimeter controls, u_{ij}, to control the subregional accumulation and minimize the total time delay of the whole network. For a given boundary between regions, thus, the high-level perimeter controls, U_{IJ}, are automatically calculated by

$$U_{IJ}(t) = \frac{\sum_{i\in\mathcal{SR}_I}\sum_{h\in\mathcal{SR}_J\cap\mathcal{H}_i}\sum_{j\in\mathcal{R}\setminus\{i\}} u_{ih}(t)\cdot\hat{m}_{ij}^h(t)}{\sum_{i\in\mathcal{SR}_I}\sum_{h\in\mathcal{SR}_J\cap\mathcal{H}_i}\sum_{j\in\mathcal{R}\setminus\{i\}}\hat{m}_{ij}^h(t)} \tag{5.7}$$

where $\mathcal{SR}_I$ is the set of subregions that belongs to Region I.

The assumption behind estimating the average trip length for internal and external trips, $L_{II}(t)$ and $L_{IH}(t)$, is that the region-based model and subregion-based plant should be consistent and exhibit identical internal and external region outflows if the information is perfect (Ramezani et al., 2015). $M_{II}^{I}(t)$ in the region-based model is equivalent to the sum of all $m_{ii}(t)$, $i \in \mathcal{SR}_I$, and $M_{IJ}^{H}(t)$ in the region-based model is equivalent to the sum of all $m_{ij}^{h}(t)$, $i \in \mathcal{SR}_I$, $h \in \mathcal{SR}_H \cap \mathcal{H}_i$ and $j \in \mathcal{SR}_J$. Based on (5.2a)-(5.2b), we have $L_{II}(t) = \frac{N_{II}(t)}{N_I(t)} \cdot \frac{P_I(N_I(t),\sigma_I(N_I(t)))}{M_{II}^{I}(t)}$ and $L_{IH}(t) = \frac{N_{IJ}(t)}{N_I(t)} \cdot \frac{P_I(N_I(t),\sigma_I(N_I(t)))}{M_{IJ}^{H}(t)}$. Thus, $L_{II}(t)$ and $L_{IH}(t)$ can be written in detail as follows (Yildirimoglu et al., 2015):

$$L_{II}(t) = \frac{\sum_{i\in\mathcal{SR}_I}\sum_{j\in\mathcal{SR}_I} n_{ij}(t)}{\sum_{i\in\mathcal{SR}_I} n_i(t)} \cdot \frac{\sum_{i\in\mathcal{SR}_I} p_i(n_i(t))}{\sum_{i\in\mathcal{SR}_I} m_{ii}(t)} \tag{5.8a}$$

$$L_{IH}(t) = \frac{\sum_{i\in\mathcal{SR}_I}\sum_{j\in\mathcal{SR}_J} n_{ij}(t)}{\sum_{i\in\mathcal{SR}_I} n_i(t)} \cdot \frac{\sum_{i\in\mathcal{SR}_I} p_i(n_i(t))}{\sum_{i\in\mathcal{SR}_I}\sum_{h\in\mathcal{SR}_H\cap\mathcal{H}_i}\sum_{j\in\mathcal{SR}_J} \hat{m}_{ij}^{h}(t)} \tag{5.8b}$$

Note that paths including more than one crossing over the boundaries between the subregions are prohibited (Sirmatel and Geroliminis, 2018). The route choice of subregion-based plant meets this assumption.

5.2.4 Region-based model considering spatial heterogeneity

In the region-based model, the heterogeneity dynamics are integrated into the regional MFDs by considering the heterogeneous distribution of spatial density (heterogeneous $n_i(t)$). Inspired by Ramezani et al. (2015) and Geroliminis and Sun (2011), the real production MFD function is defined as

$$P_I\left(N_I(t), \sigma_I\left(N_I(t)\right)\right) = |\mathcal{SR}_I| \cdot \left(d_3^I \left(\frac{N_I(t)}{|\mathcal{SR}_I|}\right)^3 + d_2^I \left(\frac{N_I(t)}{|\mathcal{SR}_I|}\right)^2 + d_1^I \frac{N_I(t)}{|\mathcal{SR}_I|}\right) \cdot \left(a^I \cdot e^{b^I \cdot \left(\sigma_I(N_I(t)) - \sigma_I^h\right)} + (1 - a^I)\right) \tag{5.9}$$

where $|\mathcal{SR}_I|$ denotes the number of subregions in Region I, $\sigma_I\left(N_I(t)\right)$ denotes the variance that captures the spatial heterogeneity, σ_I^h is the standard deviation of summation of $|\mathcal{SR}_I|$ negative binomial distributions with mean occupancy $N_I(t)/|\mathcal{SR}_I|$, d_3^I, d_2^I, and d_1^I are the estimated nominal MFD parameters, and a^I, b^I are the estimated parameters that regulate the extent of subregional density heterogeneity effect on the region production.

Let $\phi_I(N_I(t)) = a^I \cdot \left(e^{b^I \cdot \left(\sigma_I(N_I(t)) - \sigma_I^h\right)} - 1\right)$. Then (5.9) can be expressed by

$$\begin{aligned} P_I\left(N_I(t), \sigma_I\left(N_I(t)\right)\right) &= \bar{P}_I\left(N_I(t)\right) \cdot \left(\phi_I\left(N_I(t)\right) + 1\right) \\ &= \phi_I\left(N_I(t)\right) \cdot \bar{P}_I\left(N_I(t)\right) + \bar{P}_I\left(N_I(t)\right) \end{aligned} \tag{5.10}$$

where $\bar{P}_I(N_I(t)) = |\mathcal{SR}_I| \cdot \left(d_3^I \cdot \left(\frac{N_I(t)}{|\mathcal{SR}_I|}\right)^3 + d_2^I \cdot \left(\frac{N_I(t)}{|\mathcal{SR}_I|}\right)^2 + d_1^I \cdot \frac{N_I(t)}{|\mathcal{SR}_I|}\right)$ is the nominal production MFD function. (5.10) means that: The production MFD in Region I is composed of 1) the exponential term considering the heterogeneity, and 2) the production term assuming homogeneous condition corresponding to the upper bound (low-scatter) MFD.

Combining (5.10) and (5.2a)-(5.2b), we obtain

$$\hat{M}_{II}^I(t) = \frac{N_{II}(t)}{N_I(t)} \cdot \frac{\phi_I\left(\sigma_I\left(N_I(t)\right)\right) \cdot \bar{P}_I\left(N_I(t)\right) + \bar{P}_I\left(N_I(t)\right)}{L_{II}(t)} \tag{5.11a}$$

$$\hat{M}_{IJ}^H(t) = \frac{N_{IJ}(t)}{N_I(t)} \cdot \frac{\phi_I\left(\sigma_I\left(N_I(t)\right)\right) \cdot \bar{P}_I\left(N_I(t)\right) + \bar{P}_I\left(N_I(t)\right)}{L_{IH}(t)} \tag{5.11b}$$

Let $\bar{M}_{II}^I(t) = \frac{N_{II}(t)}{N_I(t)} \cdot \frac{\bar{P}_I(N_I(t))}{L_{II}(t)}$ and $\bar{M}_{IJ}^H(t) = \frac{N_{IJ}(t)}{N_I(t)} \cdot \frac{\bar{P}_I(N_I(t))}{L_{IJ}(t)}$. Substituting $\bar{M}_{II}^I$ and $\bar{M}_{IJ}^H$ into (5.11a)-(5.11b) and then substituting $\hat{M}_{II}^I$ and $\hat{M}_{IJ}^H$ into (5.1a)-(5.1b) yield that

$$\begin{aligned} \dot{N}_{II}(t) =& Q_{II}(t) - \bar{M}_{II}^I(t) \cdot \left(1 + \phi_I\left(N_I(t)\right)\right) \\ &- \sum_{H \in \mathcal{V}_I \setminus \{I\}} U_{IH}(t) \cdot \bar{M}_{II}^H(t) \cdot \left(1 + \phi_I\left(N_I(t)\right)\right) \\ &+ \sum_{H \in \mathcal{V}_I \setminus \{I\}} U_{HI}(t) \cdot \bar{M}_{HI}^I(t) \cdot \left(1 + \phi_H\left(N_H(t)\right)\right) \end{aligned} \tag{5.12a}$$

$$\begin{aligned} \dot{N}_{IJ}(t) =& Q_{IJ}(t) - \sum_{H \in \mathcal{V}_I \setminus \{I\}; I \neq J} U_{IH}(t) \cdot \bar{M}_{IJ}^H(t) \cdot \left(1 + \phi_I\left(N_I(t)\right)\right) \\ &+ \sum_{H \in \mathcal{V}_I \setminus \{I\}; I \neq J} U_{HI}(t) \cdot \bar{M}_{HJ}^I(t) \cdot \left(1 + \phi_H\left(N_H(t)\right)\right) \end{aligned} \tag{5.12b}$$

5.2.5 Introducing uncertainty in MFD dynamics

Uncertainty in travel demand profiles $Q_{IJ}(t)$, $I, J \in \mathcal{R}$ (i.e., demand disturbance) is considered. The real demand profile is assumed to be composed of a known nominal term and an unknown external disturbance term:

$$Q_{IJ}(t) = \bar{Q}_{IJ}(t) + \epsilon_{IJ}(t) \tag{5.13}$$

Now, integrating the subregion accumulation heterogeneity $\phi_I\left(N_I(t)\right)$ and the real demand profile $Q_{IJ}(t)$ subject to external disturbance, we can derive the dynamics of the region-based model that considers demand uncertainties.

Substituting (5.13) into (5.12a)-(5.12b), we have

$$\begin{aligned}\dot{N}_{II}(t) =&\bar{Q}_{II}(t)+\epsilon_{II}(t)-\bar{M}_{II}^{I}(t)\cdot\left(1+\phi_I\left(N_I(t)\right)\right)\\&-\sum_{H\in\mathcal{V}_I\setminus\{I\}}U_{IH}(t)\cdot\bar{M}_{II}^{H}(t)\cdot\left(1+\phi_I\left(N_I(t)\right)\right)\\&+\sum_{H\in\mathcal{V}_I\setminus\{I\}}U_{HI}(t)\cdot\bar{M}_{HI}^{I}(t)\cdot\left(1+\phi_H\left(N_H(t)\right)\right)\end{aligned}\tag{5.14a}$$

$$\begin{aligned}\dot{N}_{IJ}(t) =&\bar{Q}_{IJ}(t)+\epsilon_{IJ}(t)-\sum_{H\in\mathcal{V}_I\setminus\{I\};I\neq J}U_{IH}(t)\cdot\bar{M}_{IJ}^{H}(t)\cdot\left(1+\phi_I\left(N_I(t)\right)\right)\\&+\sum_{H\in\mathcal{V}_I\setminus\{I\};I\neq J}U_{HI}(t)\cdot\bar{M}_{HJ}^{I}(t)\cdot\left(1+\phi_H\left(N_H(t)\right)\right)\end{aligned}\tag{5.14b}$$

5.3 Adaptive optimal perimeter control and route guidance for MFD networks

In this section, the optimal perimeter control and route guidance (OPCRG) problem with the objective to minimize the network delay or to minimize the total time spent (TTS) is formulated. Due to the model-plant mismatch and the uncertainty in the system dynamics, it is intractable to obtain an analytical solution to the OPCRG problem, i.e., solving the HJB equation explicitly. A data-driven IADP approach to solving the HJB is then developed to resolve this difficulty. An actor-critic neural network (AC-NN) framework is employed to approximate the optimal solution to the HJB equation, which overcomes the challenge posed by the curse of dimensionality. An off-line iterative learning scheme based on the general least-square (GLS) technique is devised to implement the AC-NN framework.

5.3.1 Data-driven IADP for the OPCRG of MFD systems

First, we present the OPCRG problem formulation of the MFD-based traffic dynamics and its standard solution.

The objective of the OPCRG for MFD systems is to minimize the TTS, defined as the integral of the network accumulation with respect to time, as given by (5.15a), by manipulating the perimeter controllers and route guidance system.

$$\min_{u_{ij}(t),\theta_{ij}^h(t)} \int_0^{t_f} \sum_I \sum_J N_{IJ}(t)\mathrm{d}t \tag{5.15a}$$

subject to:

$$0 \le N_{IJ}(t) \le N_I^{jam} \tag{5.15b}$$

$$0 \le \sum_J N_{IJ}(t) \le N_I^{jam} \tag{5.15c}$$

$$u_{\min} \le u_{ij}(t) \le u_{\max} \tag{5.15d}$$

$$0 \le \theta_{ij}^h(t) \le 1 \tag{5.15e}$$

$$\sum_{h\in\mathcal{H}_i} \theta_{ij}^h(t) = 1 \tag{5.15f}$$

$$(5.7) - (5.8),\ (5.14).$$

where N_I^{jam} is the capacity of Region I and $u_{\min}$, $u_{\max}$ denote the lower and upper bounds of the perimeter controller, respectively. Here $t_f > 0$ denotes the planning horizon.

Define the state vector as $\mathbf{N}(t) \in \mathbb{R}^{r^2}$ containing all $N_{IJ}(t)$ terms and the control vector as $\mathbf{\Lambda}(t)$ of proper dimensions containing all $u_{ij}(t)$ (corresponding to $U_{IJ}(t)$ terms) and $\theta_{ij}^h(t)$ terms. We can write the complete state-space model in a compact form as

$$\dot{\mathbf{N}}(t) = \mathbf{f}(\mathbf{N}(t), \mathbf{\Lambda}(t)) \tag{5.16}$$

where $\mathbf{f}$ is the compact form of (5.14a)-(5.14b) combined with (5.7)-(5.8). $\mathbf{f}$ is an unknown nonlinear vector-valued function that is Lipschitz continuous with $\mathbf{f}(0,0) = 0$. Now, to obtain the optimal controller, one needs to solve $\mathbf{\Lambda}(t)$ from (5.15a) subject to the constraint given by (5.15b)-(5.16).

Note that (5.16) serves as the training environment of the proposed IADP algorithm. The data used for training can be simply categorized into the state data and the action (control input) data. We assume for the state data that only the measurements of regional accumulation $N_{IJ}(t)$ are available. This is a more challenging case than the one where more detailed subregion data, e.g., the measurements of subregional accumulation $n_{ij}(t)$, are available. For the action data, although the plant dynamics (5.3) is regarded as a black box to the IADP algorithm, the control strategies $u_{ij}(t)$ and $\theta_{ij}^h(t)$ are devised by transport managers and thus by nature they are available. That is to say, the training data is a group of the regional states and the subregional control input values, i.e., $\{N_{IJ}(t), u_{ij}(t), \theta_{ij}^h(t)\}$. We will prove in the following

subsection that the well-trained IADP approach using the data $\{N_{IJ}(t), u_{ij}(t), \theta_{ij}^h(t)\}$ obtained from (5.16) can approximate the OPCRG strategy for the subregion-based plant (5.3). Before that, we need to present the standard solution to the OPCRG problem.

Chen et al. (2022) developed an IRL-based adaptive perimeter controller by constructing the HJB equation associated with the performance function and the underlying plant dynamics, which can be reformulated as a dynamical system of the input-affine form. Nevertheless, due to the unknown plant dynamics (5.3a)-(5.3c) and strong nonlinear relationship between U_{IJ}, L_{IH} and u_{ij}, θ_{ij}^h as described by (5.7) and (5.8a)-(5.8b), (5.16) cannot be rewritten as an input-affine nonlinear dynamical system directly.

To overcome this difficulty, the core idea is to introduce the pre-compensator (Murray et al., 2002; Lee and Sutton, 2021) for $\mathbf{\Lambda}$ that is governed by the following dynamic equation

$$\dot{\mathbf{\Lambda}} = \mathbf{A}_1 \cdot \mathbf{\Lambda} + \mathbf{A}_2 \cdot \mathbf{U} \tag{5.17}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are constant matrices and $\mathbf{A}_1$ is a Hurwitz matrix. $\mathbf{U} \in \mathbb{R}^{\bar{d}}$ is the new control input vector where $\bar{d}$ is a positive integer equal to the summation of numbers of all u_{ij} and θ_{ij}^h variables.

Remark 5.3.1 The matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ should be designed to guarantee the global asymptotic stability of $\mathbf{\Lambda}$. Generally, we can let $\mathbf{A}_1$ be a negative-definite matrix while $\mathbf{A}_2$ be a positive-definite matrix. Without loss of generality, in this study, we define $\mathbf{A}_1 \triangleq -\mathcal{I}_{\mathbf{\Lambda}}$ and $\mathbf{A}_2 \triangleq \mathcal{I}_{\mathbf{\Lambda}}$ where $\mathcal{I}_{\mathbf{\Lambda}} \in \mathbb{R}^{\bar{d}}$ is an identity matrix.

There are several reasons why (5.17) that determines the implemented control policy can be devised. On the one hand, given that the exact knowledge on system dynamics is unavailable or the traffic managers have difficulty calibrating the model in the learning process, the implemented control policy may not be an optimal one, which could lead to unexpected behaviors such as poor stability or even instability. By adding this pre-compensator, we are tailoring the phase and gain of the open loop response and therefore changing stability margins which have the effect of determining both rise time and overshoot of the closed-loop system (Ogata et al., 2010). On the other hand, as discussed in Zhong et al. (2018b), there may be a difference between the control that is actually implemented and the optimal control that needs to be learned. The network dynamics can be stimulated by the implemented control, and hence the evolution of traffic states and the network performance can be captured by the learning algorithms. Then the learning

algorithm adjusts the adaptive controller iteratively to achieve the optimal network performance (Chen et al., 2022). ∎

Now we define the augmented state vector as $\mathbf{X} = \left[\mathbf{N}^T, \mathbf{\Lambda}^T\right]^T \in \mathbf{\Omega} \subset \mathbb{R}^{\bar{r}}$ where $\bar{r}$ is a positive integer equal to the summation of numbers of all N_{IJ}, u_{ij} and θ_{ij}^h variables. Then the dynamics of the augmented system in terms of $\mathbf{X}$ is

$$\dot{\mathbf{X}} = \mathbf{F}(\mathbf{X}) + \mathbf{G} \cdot \mathbf{U} \tag{5.18}$$

where

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}(\mathbf{N}, \mathbf{\Lambda}) \\ \mathbf{A}_1 \cdot \mathbf{\Lambda} \end{bmatrix}, \ \mathbf{G} = \begin{bmatrix} 0 \\ \mathbf{A}_2 \end{bmatrix}$$

A standard solution to the OPCRG problem is by constructing the Hamiltonian function. First, the value function is defined as

$$V(\mathbf{X}(t)) = \int_t^{t_f} \mathcal{L}(\mathbf{X}, \mathbf{U}) \, \mathrm{d}\tau \tag{5.19}$$

where $\mathcal{L}$ denotes the cost function, which can be generally chosen as

$$\mathcal{L}(\mathbf{X}, \mathbf{U}) = \| \mathbf{X} \| + \lambda \| \mathbf{U} \|$$

where $\lambda > 0$ is a small constant and $\|\mathbf{a}\|$ denotes the Euclidean norm of a vector $\mathbf{a}$. Minimizing the cost $\mathcal{L}$ means the simultaneous minimization of both the regional accumulation and the PCRG control effort.

Taking the time derivative of both sides of (5.19) and moving the right terms to the left, we can derive

$$(\nabla V)^T \cdot \Big(\mathbf{F}(\mathbf{X}) + \mathbf{G} \cdot \mathbf{U}\Big) + \| \mathbf{X} \| + \lambda \| \mathbf{U} \| = 0$$

where ∇V denotes the partial derivative of $V(\mathbf{X})$ with respect to $\mathbf{X}$.

The Hamiltonian function for (5.18) can be defined as

$$\mathbb{H}(\mathbf{X}, \nabla V, \mathbf{U}) \triangleq (\nabla V)^T \cdot \Big(\mathbf{F}(\mathbf{X}) + \mathbf{G} \cdot \mathbf{U}\Big) + \| \mathbf{X} \| + \lambda \| \mathbf{U} \| \tag{5.20}$$

Note that the integrand of the value function is not explicitly time-dependent and the terminal time is fixed, and that (5.18) is an autonomous dynamical system.

Therefore, the optimality is obtained by letting $\mathbb{H}(\mathbf{X}, \nabla V, \mathbf{U}) = 0$, i.e., the Bellman optimality equation is derived as

$$(\nabla V^*)^T \cdot \left(\mathbf{F}(\mathbf{X}) + \mathbf{G} \cdot \mathbf{U}^*\right) + \parallel \mathbf{X} \parallel + \lambda \parallel \mathbf{U}^* \parallel = 0 \tag{5.21}$$

By applying the stationary condition $\partial \mathbb{H}(\mathbf{X}, \nabla V, \mathbf{U})/\partial \mathbf{U} = 0$, the optimal control is obtained as

$$\mathbf{U}^* = -\frac{1}{2\lambda}\mathbf{G}^T \cdot \nabla V^*(\mathbf{X}) \tag{5.22}$$

Substituting (5.22) into (5.21), we have

$$(\nabla V^*)^T \cdot \mathbf{F} - \frac{1}{4\lambda}(\nabla V^*)^T \cdot \mathbf{G} \cdot \mathbf{G}^T \cdot \nabla V^* + \parallel \mathbf{X} \parallel = 0 \tag{5.23}$$

To find the optimal feedback control policy for the OPCRG problem, it is necessary to solve V^* from the HJB equation (5.23). However, the strong nonlinearity of (5.23) makes it challenging to obtain an analytical solution. In such cases, policy iteration is often employed as one of the most commonly used approaches to overcome this difficulty. The policy iteration algorithm for the OPCRG problem starts with an initial admissible control $\mathbf{U}^0$. For $k = 0, 1, \ldots$, the policy iteration algorithm contains the policy evaluation phase and the policy improvement phase, as described in Algorithm 2. The convergence of the iteration sequence $\{(V^k, \mathbf{U}^k)\}$ by using (5.24)-(5.25) to the optimality $(V^*, \mathbf{U}^*)$ to the HJB equation can be found in Chen et al. (2022).

Algorithm 2 Policy iteration

Input: initial admissible control policy $\mathbf{U}^0(\mathbf{X})$
Output: $V^k(\mathbf{X})$

1: Policy evaluation:
$k = k + 1$.
Update the value function by calculating

$$\left(\nabla V^{k+1}\right)^T \cdot \left(\mathbf{F}(\mathbf{X}) + \mathbf{G} \cdot \mathbf{U}^k\right) + \parallel \mathbf{X} \parallel + \lambda \parallel \mathbf{U}^k \parallel = 0 \tag{5.24}$$

On convergence, set $V^{k+1}(\mathbf{X}) = V^k(\mathbf{X})$.
2: Policy improvement:
Update the control policy by calculating

$$\mathbf{U}^{k+1} = -\frac{1}{2\lambda}\mathbf{G}^T \cdot \nabla V^{k+1}(\mathbf{X}) \tag{5.25}$$

Note that (5.24)-(5.25) require the knowledge of the drift dynamics $\mathbf{F}$. A model-free off-policy IRL Bellman equation is employed to help get rid of $\mathbf{F}$ while it uses system data generated by an implemented behavior policy to solve the HJB equation. Towards this, we rewrite the affine dynamics as

$$\dot{\mathbf{X}}(t) = \mathbf{F}(\mathbf{X}(t)) + \mathbf{G} \cdot \mathbf{U}^k(t) + \mathbf{G} \cdot (\mathbf{U}(t) - \mathbf{U}^k(t)) \tag{5.26}$$

where $\mathbf{U}^k(t)$ is the target policy to be learned and $\mathbf{U}(t)$ is the implemented behavior policy for generating the data for training. Differentiating the value function $V(\mathbf{X})$ along the system trajectory (5.26) and using (5.25), we have

$$V^k(\mathbf{X}(t)) - V^k(\mathbf{X}(t+\Delta t)) = \int_t^{t+\Delta t} \left(\mathcal{L}(\mathbf{X}, \mathbf{U}^k) - 2\lambda \left(\mathbf{U}^{k+1} \right)^T \cdot \boldsymbol{\mathcal{I}}_{\mathbf{U}} \cdot (\mathbf{U} - \mathbf{U}^k) \right) \mathrm{d}\tau \tag{5.27}$$

where $\boldsymbol{\mathcal{I}}_{\mathbf{U}}$ is an identity matrix of appropriate dimensions.

Without any prior knowledge of the system dynamics, the value function V^k and the updated policy $\mathbf{U}^{k+1}$ can be simultaneously obtained from solving the off-policy IRL Bellman equation (5.27) with an implemented control policy $\mathbf{U}(t)$.

5.3.2 A two-phase iterative learning scheme

We adopt an AC-NN framework to approach the solution of the OPCRG problem. The following critic NN and the actor NN are constructed to approximate the value function V^k and the control strategy $\mathbf{U}^k$, respectively.

$$\begin{aligned} V^{k+1}(\mathbf{X}) &= \mathbf{w}_{c,\,k+1}^T \cdot \boldsymbol{\Psi}_c(\mathbf{X}) + \varepsilon_{c,\,k+1} \\ \mathbf{U}^{k+1}(\mathbf{X}) &= \mathbf{w}_{d,\,k+1}^T \cdot \boldsymbol{\Psi}_d(\mathbf{X}) + \boldsymbol{\varepsilon}_{d,\,k+1} \end{aligned} \tag{5.28}$$

where $\boldsymbol{\Psi}_c \in \mathbb{R}^{K_c}$, $\boldsymbol{\Psi}_d \in \mathbb{R}^{K_d}$ are vectors of linearly independent activation functions, $\mathbf{w}_{c,\,k+1} \in \mathbb{R}^{K_c}$, $\mathbf{w}_{d,\,k+1} \in \mathbb{R}^{K_d \times D}$ are the weighting matrices of the proper dimension of the NNs, $\varepsilon_{c,\,k+1}$ and $\boldsymbol{\varepsilon}_{d,\,k+1}$ are the approximation errors of appropriate dimensions.

Based on (5.28), the residual error is defined as

$$\begin{aligned}\zeta^{k+1}(\mathbf{X}(t)) =& \Big(\mathbf{\Psi}_c(\mathbf{X}(t+\Delta t)) - \mathbf{\Psi}_c(\mathbf{X}(t))\Big)^T \cdot \mathbf{w}_{c,\,k+1} \\ &+ \int_t^{t+\Delta t} \Big(\parallel \mathbf{X} \parallel + \lambda \mathbf{\Psi}_d^T(\mathbf{X}) \cdot \mathbf{w}_{d,\,k} \cdot \mathbf{w}_{d,\,k}^T \cdot \mathbf{\Psi}_d(\mathbf{X})\Big)\, \mathrm{d}\tau \\ &- 2\lambda \int_t^{t+\Delta t} \mathbf{\Psi}_d^T(\mathbf{X}) \cdot \mathbf{w}_{d,\,k+1} \cdot \mathcal{I}_{\mathbf{U}} \cdot \Big(\mathbf{U} - \mathbf{w}_{d,\,k}^T \cdot \mathbf{\Psi}_d(\mathbf{X})\Big)\, \mathrm{d}\tau\end{aligned} \tag{5.29}$$

It is assumed that the outputs of the NNs can be expressed by the estimations as follows

$$\begin{aligned}\hat{V}^{k+1}(\mathbf{X}) &= \hat{\mathbf{w}}_{c,\,k+1}^T \cdot \mathbf{\Psi}_c(\mathbf{X}) \\ \hat{\mathbf{U}}^{k+1}(\mathbf{X}) &= \hat{\mathbf{w}}_{d,\,k+1}^T \cdot \mathbf{\Psi}_d(\mathbf{X})\end{aligned} \tag{5.30}$$

where $\hat{\mathbf{w}}_{c,\,k+1}$ and $\hat{\mathbf{w}}_{d,\,k+1}$ are estimations of $\mathbf{w}_{c,\,k+1}$ and $\mathbf{w}_{d,\,k+1}$, respectively, which are usually learned from training data.

Define a strictly increasing time sequence $\{t_m\}_{m=0}^b$, and let $b > 0$ denote the number of collected data samples for estimating $\Big(V^{k+1}(\mathbf{X}), \mathbf{U}^{k+1}(\mathbf{X})\Big)$ by $\Big(\hat{V}^{k+1}(\mathbf{X}), \hat{\mathbf{U}}^{k+1}(\mathbf{X})\Big)$, the residual error $\mathbf{e}^{k+1} = \Big[e_1^{k+1}, \ldots, e_m^{k+1}, \ldots, e_b^{k+1}\Big]^T$ due to the truncation is given by

$$\begin{aligned}e_m^{k+1} =& \hat{V}^{k+1}(\mathbf{X}(t_m)) - \hat{V}^{k+1}(\mathbf{X}(t_{m+1})) - \int_{t_m}^{t_{m+1}} \mathcal{L}(\mathbf{X}, \mathbf{U}^k)\, \mathrm{d}\tau \\ &+ \int_{t_m}^{t_{m+1}} 2\lambda \Big(\hat{\mathbf{U}}^{k+1}\Big)^T \cdot \mathcal{I}_{\mathbf{U}} \cdot \Big(\mathbf{U} - \mathbf{U}^k\Big)\, \mathrm{d}\tau \\ =& \hat{\mathbf{w}}_{c,\,k+1}^T \cdot \Big(\mathbf{\Psi}_c(\mathbf{X}(t_m)) - \mathbf{\Psi}_c(\mathbf{X}(t_{m+1}))\Big) \\ &- \int_{t_m}^{t_{m+1}} \Big(\parallel \mathbf{X} \parallel + \lambda \mathbf{\Psi}_d^T(\mathbf{X}) \cdot \hat{\mathbf{w}}_{d,\,k} \cdot \hat{\mathbf{w}}_{d,\,k}^T \cdot \mathbf{\Psi}_d(\mathbf{X})\Big)\, \mathrm{d}\tau \\ &+ 2\lambda \int_{t_m}^{t_{m+1}} \mathbf{\Psi}_d^T(\mathbf{X}) \cdot \hat{\mathbf{w}}_{d,\,k+1} \cdot \mathcal{I}_{\mathbf{U}} \cdot \Big(\mathbf{U} - \hat{\mathbf{w}}_{d,\,k}^T \cdot \mathbf{\Psi}_d(\mathbf{X})\Big)\, \mathrm{d}\tau\end{aligned} \tag{5.31}$$

Define $\mathbf{W}_{k+1} = \Big[\mathbf{w}_{c,\,k+1}^T, vec(\mathbf{w}_{d,\,k+1})^T\Big]^T \in \mathbb{R}^{\bar{K}}$, $\hat{\mathbf{W}}_{k+1} = \Big[\hat{\mathbf{w}}_{c,\,k+1}^T, vec(\hat{\mathbf{w}}_{d,\,k+1})^T\Big]^T \in \mathbb{R}^{\bar{K}}$ the vectors of the ideal and the estimated AC-NN weights, respectively, where $\bar{K} = K_c + \bar{d} \cdot K_d$ is the corresponding dimension. Here the iterative index is $k \in \{0, 1, \ldots\}$, and the time sequence index is $m \in \{0, \ldots, b\}$. $vec(\cdot)$ denotes the vectorization of a matrix formed by stacking all

the columns of the matrix into a single-column vector. By Kronecker product $\otimes$, let $\boldsymbol{\rho}_m(\hat{\mathbf{W}}_k)$, $\pi_m(\hat{\mathbf{W}}_k)$ be defined as

$$\boldsymbol{\rho}_m(\hat{\mathbf{W}}_k) = \begin{bmatrix} \boldsymbol{\Psi}_c(\mathbf{X}(t_m)) - \boldsymbol{\Psi}_c(\mathbf{X}(t_{m+1})) \\ 2\lambda \int_{t_m}^{t_{m+1}} \mathcal{I}_{\mathbf{U}} \cdot \left(\mathbf{U} - \hat{\mathbf{w}}_{d,\,k+1}^T \cdot \boldsymbol{\Psi}_d(\mathbf{X})\right) \otimes \boldsymbol{\Psi}_d(\mathbf{X})\, \mathrm{d}\tau \end{bmatrix}$$

$$\pi_m(\hat{\mathbf{W}}_k) = \int_{t_m}^{t_{m+1}} \left(\| \mathbf{X} \| + \lambda \boldsymbol{\Psi}_d^T(\mathbf{X}) \cdot \hat{\mathbf{w}}_{d,\,k} \cdot \hat{\mathbf{w}}_{d,\,k}^T \cdot \boldsymbol{\Psi}_d(\mathbf{X}) \right) \mathrm{d}\tau$$

This gives a compact form of the residual error (5.31) of the approximation

$$e_m^{k+1} = \boldsymbol{\rho}_m^T(\hat{\mathbf{W}}_k) \cdot \hat{\mathbf{W}}_{k+1} - \pi_m(\hat{\mathbf{W}}_k) \tag{5.32}$$

Based on the GLS principle, it is desired to determine the estimated AC-NN weight vector $\hat{\mathbf{W}}_{k+1}$ by solving $\min_{\hat{\mathbf{W}}_{k+1}} \|\mathbf{e}^{k+1}\|$. According to (5.32), the solution to this GLS problem is

$$\hat{\mathbf{W}}_{k+1} = \left[\mathbf{P}^T(\hat{\mathbf{W}}_k) \cdot \mathbf{P}(\hat{\mathbf{W}}_k)\right]^{-1} \cdot \mathbf{P}^T(\hat{\mathbf{W}}_k) \cdot \boldsymbol{\Pi}(\hat{\mathbf{W}}_k) \tag{5.33}$$

where

$$\mathbf{P}(\hat{\mathbf{W}}_k) = \left[\boldsymbol{\rho}_0(\hat{\mathbf{W}}_k), \ldots, \boldsymbol{\rho}_b(\hat{\mathbf{W}}_k)\right]^T$$

$$\boldsymbol{\Pi}(\hat{\mathbf{W}}_k) = \left[\pi_0(\hat{\mathbf{W}}_k), \ldots, \pi_b(\hat{\mathbf{W}}_k)\right]^T$$

In order to guarantee the convergence of the IADP control policy to a near-optimal control, a rank condition in the following assumption is adopted to verify the richness of the recorded data, i.e., whether it is sufficient to solve the GLS problem (Modares et al., 2014).

Assumption 5.3.1 The number of sampling data points b should be sufficiently large, i.e.,

$$b \geq \text{rank}(\mathbf{P}(\hat{\mathbf{W}}_k)) = \bar{K} \tag{5.34}$$

$\boldsymbol{\rho}(\hat{\mathbf{W}}_k)$ and $\pi(\hat{\mathbf{W}}_k)$ can be computed with a suitable initial policies weight $\mathbf{w}_{d,\,0}$ using a set of training data. The algorithm is then iterated following (5.33). Accordingly, the unknown value function $\hat{V}^k(\mathbf{X})$ and policy function $\hat{\mathbf{U}}^{k+1}(\mathbf{X})$ can be approximated by (5.30) with the convergent $\hat{\mathbf{W}}_{k+1}$. The analytical result is summarized by the following theorem.

Theorem 5.3.1 Suppose that the convergence of $\hat{\mathbf{W}}_{k+1}$ holds, for $\forall \xi > 0$, there exist integer $k^* > 0$, $K_c^* > 0$ and $K_d^* > 0$, such that if $k > k^*$, $K_c > K_c^*$ and $K_d > K_d^*$, then

1) $|\hat{V}^k(\mathbf{X}) - V^k(\mathbf{X})| \leq \xi, \|\hat{\mathbf{U}}^{k+1} - \mathbf{U}^{k+1}\| \leq \xi$

2) $|\hat{V}^k(\mathbf{X}) - V^*(\mathbf{X})| \leq \xi, \|\hat{\mathbf{U}}^{k+1} - \mathbf{U}^*\| \leq \xi$

hold for all $\mathbf{X} \in \mathbf{\Omega}$. ∎

To ensure the convergence of $\hat{\mathbf{W}}_{k+1}$, the following persistency of excitation (PE) assumption is given.

Assumption 5.3.2 Let $\boldsymbol{\rho}_l(\hat{\mathbf{W}}_k)$ be persistently existed, that is there exist $b_0 > 0$ and $\delta > 0$ such that for all $b \leq b_0$, we have

$$\frac{1}{b}\sum_{l=0}^{b-1} \boldsymbol{\rho}_l(\hat{\mathbf{W}}_k)\boldsymbol{\rho}_l^T(\hat{\mathbf{W}}_k) \geq \delta \boldsymbol{\mathcal{I}}_{\bar{K}}$$

where $\boldsymbol{\mathcal{I}}_{\bar{K}}$ is an identity matrix of appropriate dimensions.

Now we provide the proof of Theorem 5.3.1.

Proof 5.3.1 1) Define the weight estimation error vector $\tilde{\mathbf{W}}_{k+1}$ as

$$\tilde{\mathbf{W}}_{k+1} \triangleq \hat{\mathbf{W}}_{k+1} - \mathbf{W}_{k+1} \tag{5.35}$$

Then it follows from (5.33) and (5.35) that

$$\mathbf{P}^T\mathbf{P}\hat{\mathbf{W}}_{k+1} = \mathbf{P}^T\mathbf{\Pi}$$

i.e.,

$$\mathbf{P}^T\mathbf{P}\tilde{\mathbf{W}}_{k+1} = \mathbf{P}^T\mathbf{\Pi} - \mathbf{P}^T\mathbf{P}\mathbf{W}_{k+1} \tag{5.36}$$

Multiplying $\tilde{\mathbf{W}}_{k+1}^T$ on both sides of (5.36) yields

$$\tilde{\mathbf{W}}_{k+1}^T\mathbf{P}^T\mathbf{P}\tilde{\mathbf{W}}_{k+1} = \left[\mathbf{P}\tilde{\mathbf{W}}_{k+1}\right]^T(\mathbf{\Pi} - \mathbf{P}\mathbf{W}_{k+1}) \tag{5.37}$$

Based on Assumption 5.3.2, the left side of (5.37) satisfies

$$\tilde{\mathbf{W}}_{k+1}^T\mathbf{P}^T\mathbf{P}\tilde{\mathbf{W}}_{k+1} \geq b\delta\boldsymbol{\mathcal{I}}_{\bar{K}}\|\tilde{\mathbf{W}}_{k+1}\| \tag{5.38}$$

Note that we have

$$\begin{aligned}\mathbf{P}^T(\mathbf{\Pi}-\mathbf{P}\mathbf{W}_{k+1}) =& \sum_{l=0}^{b-1}\Bigg[\boldsymbol{\rho}_l^T\Bigg((\mathbf{\Psi}_c(\mathbf{X}_{l+1})-\mathbf{\Psi}_c(\mathbf{X}_l))^T\mathbf{w}_{c,k+1}\\ &+\int_{t_l}^{t_{l+1}}\Big(\parallel\mathbf{X}\parallel+\lambda\mathbf{\Psi}_d^T(\mathbf{X})\cdot\mathbf{w}_{d,\,k}\cdot\mathbf{w}_{d,\,k}^T\cdot\mathbf{\Psi}_d(\mathbf{X})\Big)\,\mathrm{d}\tau\\ &-2\lambda\int_{t_l}^{t_{l+1}}\mathbf{\Psi}_d^T(\mathbf{X})\cdot\mathbf{w}_{d,\,k+1}\cdot\mathcal{I}_{\mathbf{U}}\cdot\Big(\mathbf{U}-\mathbf{w}_{d,\,k}^T\cdot\mathbf{\Psi}_d(\mathbf{X})\Big)\,\mathrm{d}\tau\Bigg)\Bigg]\\ =&\sum_{l=0}^{b-1}\boldsymbol{\rho}_l^T\zeta^{k+1}(\mathbf{X}(t_l))\end{aligned}$$

where $\zeta^{k+1}(\mathbf{X}(t_l))$ denotes the residual error for time interval $[t_l, t_{l+1}]$ instead of $[t, t+\Delta t]$ for (5.29).

Based on (5.37), we have

$$\delta b\|\tilde{\mathbf{W}}_{k+1}\| \le \|\tilde{\mathbf{W}}_{k+1}\|\sum_{l=0}^{b-1}\|\boldsymbol{\rho}_l^T\|\cdot|\zeta_{k+1}(\mathbf{X}(t_l))| \le \|\tilde{\mathbf{W}}_{k+1}\|\sum_{l=0}^{b-1}\|\boldsymbol{\rho}_l^T\|\zeta_{\max} \tag{5.39}$$

where $\zeta_{\max}$ denotes the bound of ζ^{k+1}. Note that $\lim_{\bar{K}\to\infty}\zeta^{k+1}(\mathbf{X}(t_l))=0$. Based on (5.39), we have $\lim_{\bar{K}\to\infty}\tilde{\mathbf{W}}_{k+1}=0$.

Define $\tilde{\mathbf{w}}_{c,k}\triangleq\hat{\mathbf{w}}_{c,k}-\mathbf{w}_{c,k}$ and $\tilde{\mathbf{w}}_{d,k+1}^T\triangleq\hat{\mathbf{w}}_{d,k+1}^T-\mathbf{w}_{d,k+1}^T$. Since

$$\begin{aligned}\hat{V}^k(\mathbf{X})-V^k(\mathbf{X}) &= \tilde{\mathbf{w}}_{c,k}^T\mathbf{\Psi}_c(\mathbf{X})-\varepsilon_{c,\,k}\\ \hat{\mathbf{U}}^{k+1}(\mathbf{X})-\mathbf{U}^{k+1}(\mathbf{X}) &= \tilde{\mathbf{w}}_{d,k+1}^T\mathbf{\Psi}_d(\mathbf{X})-\boldsymbol{\varepsilon}_{d,\,k+1}\end{aligned}$$

and $\lim_{K_c\to\infty}\varepsilon_{c,\,k}=0$, $\lim_{K_d\to\infty}\boldsymbol{\varepsilon}_{d,\,k+1}=0$, we can obtain

$$\begin{aligned}\lim_{K_{c,k}\to\infty}\hat{V}^k &= V^k\\ \lim_{K_{d,k+1}\to\infty}\hat{\mathbf{U}}^{k+1} &= \mathbf{U}^{k+1}\end{aligned}$$

That is to say, there exist integers k^*, $K_c^*>0$ and $K_d^*>0$ for $\forall\,\mathbf{X}\in\Omega$, $\xi>0$ such that if $k>k^*$, $K_c>K_c^*$ and $K_d>K_d^*$, then

$$|\hat{V}^k(\mathbf{X})-V^k(\mathbf{X})|\le\xi,\ \|\hat{\mathbf{U}}^{k+1}-\mathbf{U}^{k+1}\|\le\xi$$

2) Based on the convergence property of Algorithm 2, for $\forall\xi>0$, there exists integer k^* such that for $\forall k\ge k^*$,

$$|V^k(\mathbf{X})-V^*(\mathbf{X})|\le\frac{\xi}{2} \tag{5.40}$$

Based on part 1), there exists integer $K_c^* > 0$ such that

$$|\hat{V}^k(\mathbf{X}) - V^k(\mathbf{X})| \leq \frac{\xi}{2} \tag{5.41}$$

From (5.40) and (5.41), we have

$$|\hat{V}^k(\mathbf{X}) - V^*(\mathbf{X})| \leq |\hat{V}^k(\mathbf{X}) - V^k(\mathbf{X})| + |V^k(\mathbf{X}) - V^*(\mathbf{X})| \leq \frac{\xi}{2} + \frac{\xi}{2} = \xi$$

Similarly, $\|\hat{\mathbf{U}}^{k+1} - \mathbf{U}^*\| \leq \xi$. The proof is completed. ∎

Theorem 5.3.1 indicates that the optimal value function $V^*(\mathbf{X})$ and control policy $\mathbf{U}^*(\mathbf{X})$ can be simultaneously approximated by the AC-NN framework (5.30) applying the GLS-based update law (5.33). The OPCRG strategy is then obtained by the approximated policy function such that the network performance is optimized.

Different from Chen et al. (2022), the proposed IADP algorithm is composed of an *online measurement* phase and a *off-line training* phase as shown in Figure 5.2. First, without knowledge of the accurate traffic dynamics, an *online measurement* phase is required to collect a sufficient amount of data under a given control input $\mathbf{U}$. Then NNs are constructed in the *off-line training* phase to approach the optimal solution of the model-free iterative equation. This AC-NN is then trained using the measured data sequence $\{N_{IJ}(t), u_{ij}(t), \theta_{ij}^h(t)\}$ from the environment, i.e., the compact state-space model (5.16). The AC-NN weights are iterated using the adaptation law given by (5.33). For the *off-line training* phase, $\|\hat{\mathbf{W}}_{k+1} - \hat{\mathbf{W}}_k\| < \varepsilon$ with $\varepsilon > 0$ a small constant is adopted as the phase termination condition. After an *off-line training* phase terminates, the current trained controller will be tested in the plant dynamics (5.3). During the implementation, only the state data $\{N_{IJ}(t)\}$ are required as input to the IADP control agent. After the entire simulation, the control performance index TTS will be evaluated and compared with the TTS achieved by the previously trained controller. The training process will be terminated if the relative performance difference between the current and previous epochs is less than 10^{-4}. Otherwise, we update the initial AC-NN weights, carry out the online measurement to collect new data, and then the next off-line training phase starts.

5.4 Numerical experiments

In this section, we present two case studies to evaluate the performance of the proposed OPCRG controller. Case 1 considers a two-region network mimicking

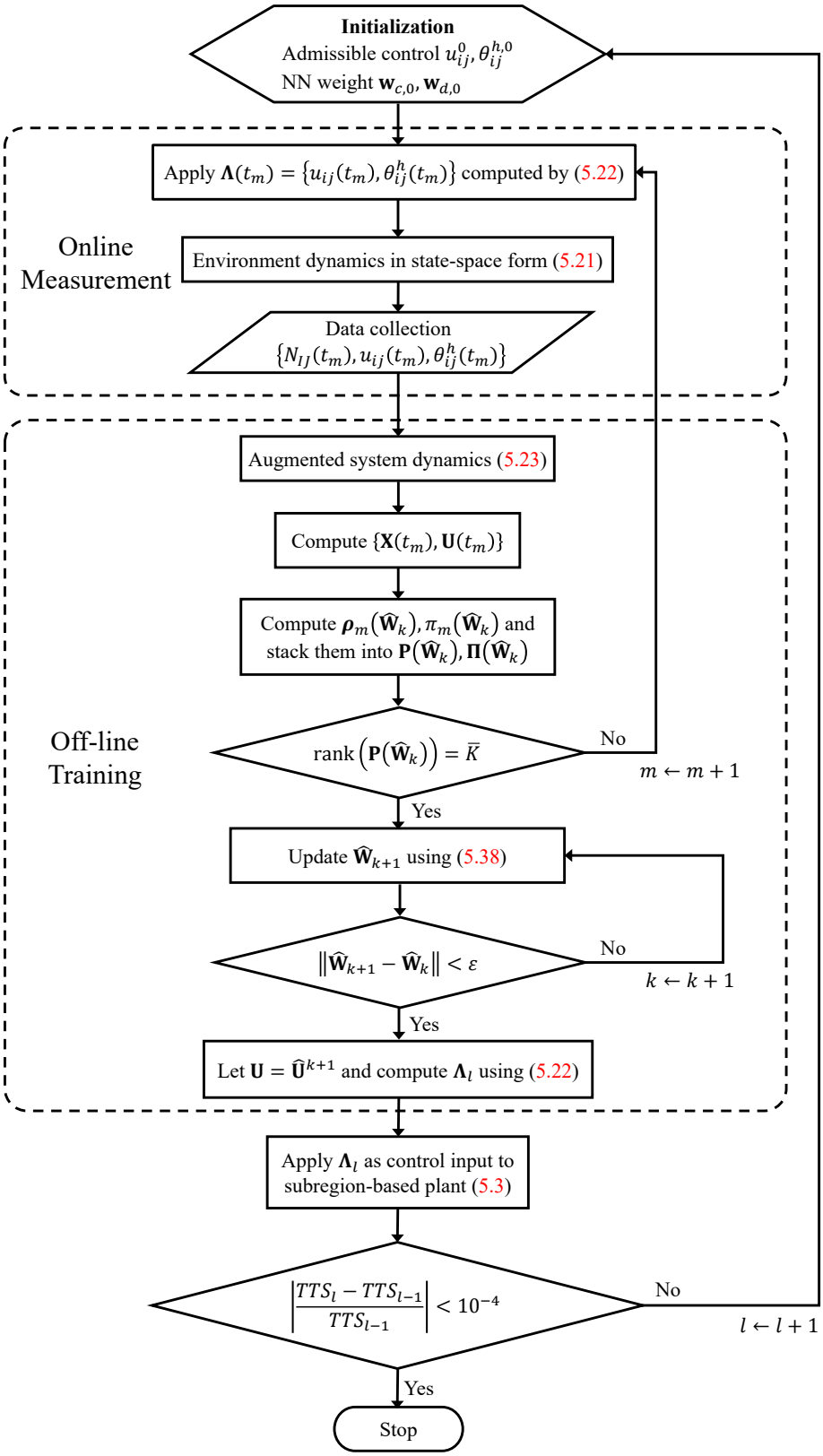


Figure 5.2 Flowchart of the IADP-based control method

the city center and periphery, wherein the periphery comprises 4 subregions, see Figure 5.3(a). In this case, the *model* and the *plant* have different structures and the *plant* model is assumed unknown to the IADP controller. First, we examine the performance of the IADP scheme in a scenario where a limited data source is available. Then we validate the effectiveness of the IADP scheme in cases of heterogeneous time-varying average regional trip distances and MFD errors, respectively. In addition, we examine the performance of the IADP approach in cases with various levels of driver compliance rates. In Case 2, to validate the scalability of the proposed IADP method, we consider a larger network consisting of a city center with 4 subregions and its periphery with 12 subregions, see Figure 5.3(b). Case 2 also examines the IADP's robustness against demand uncertainties. The adopted MFD functions are depicted by Figure 5.4. Every subregion accumulation is initially identical and uncongested in all case studies. The perimeter control input constraint is $0.1 \leq u_{ij} \leq 1$. The sampling interval and the control update interval are 1 minute. Moreover, a case that considers a four-region network, wherein the model and the plant share the same model structure, is presented in Appendix A.2.

Performance comparison is made among various strategies including:

- The proposed IADP: In this scenario, the IADP approach is trained with the parsimonious regional state data and subregional control input data, i.e., $\{N_{IJ}(t), u_{ij}(t), \theta^h_{ij}(t)\}$. After the training is completed, the IADP strategy is implemented in the subregion-based plant (5.3). Note that the IADP is model-free.

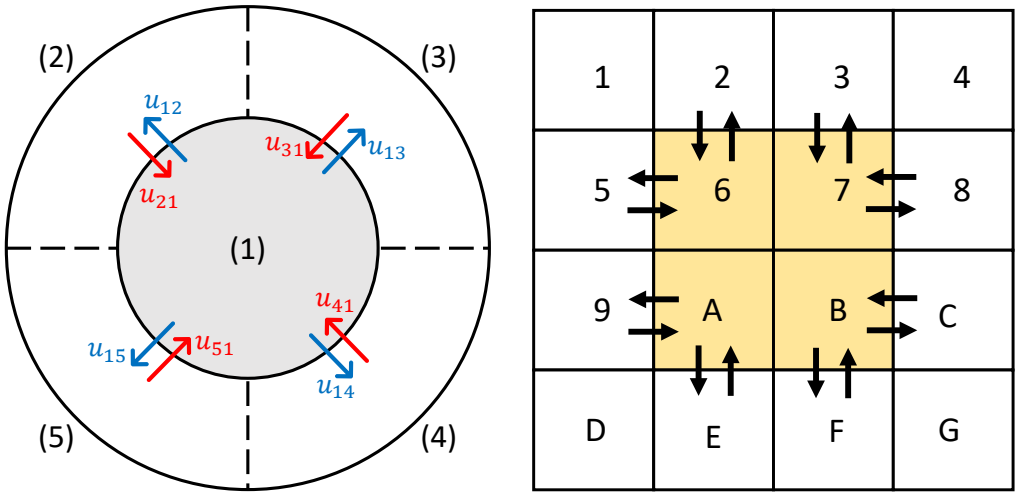
- The IADP trained in the plant environment, IADP-PT: The difference between this scenario and the IADP scenario is that more detailed state data $\{n_{ij}(t)\}$ is available for training the IADP-PT agent.

- The MPC with perfect knowledge of the plant, MPC-PM: Inspired by Kouvelas et al. (2023), the MPC approach with access to perfect plant information (accurate measurement of states $n_{ij}(t)$, average trip distance $l_i(t)$, and OD specific demand $q_{ij}(t)$) is regarded as a benchmark.

- The MPC with imperfect measurements of the plant characteristics, MPC-IPM: In this scenario, the MPC controller has no access to exact information on average trip distances, MFD errors, and demand uncertainties.

- The MPC with the UKF to update regional accumulation N_{IJ} and trip distance L_{IJ}, MPC-UKF: Different from MPC-PM, the prediction model used in this scenario is the parsimonious region-based model (5.1) with state estimation using a UKF method (Batista et al., 2021).

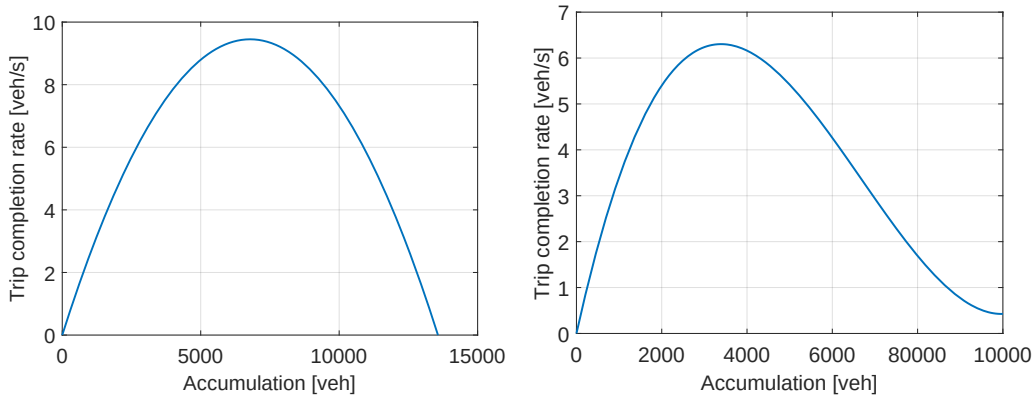
- The proportional-integral perimeter control (Keyvan-Ekbatani et al., 2012) with a Logit-based route guidance strategy, PIL: In this scenario, drivers are free to choose their routes. In simulations, this is captured by calculating Logit-based route split ratios θ_{ij}^h based on travel times of a predefined set of paths connecting subregion i and the destination j. Such predefined set of paths is determined using Dijkstra's algorithm for K-shortest paths (distance-based, $K = 3$ for this study). Note that the θ_{ij}^hs are updated using the Logit model at each control time step. We do not intend to perform SUE or DUE assignments using the Logit model. This PIL strategy is regarded as the baseline.



(a) (b)

Figure 5.3 The tested networks. (a) Two-region network consisting of five subregions (Region 1 in gray, Region 2 in white) for Case 1, and (b) Two-region network consisting of sixteen subregions (Region 1 in white, Region 2 in terrestrial yellow) for Case 2.

For the MPC implementation, the state-of-the-art CasADi toolbox with the IPOPT solver is employed. All computations are performed within MATLAB R2022a on a personal computer equipped with Intel Core i7-9850 CPU 2.65 GHz. In all the examples, the prediction horizon of the MPC controller is 15 min.



(a) (b)

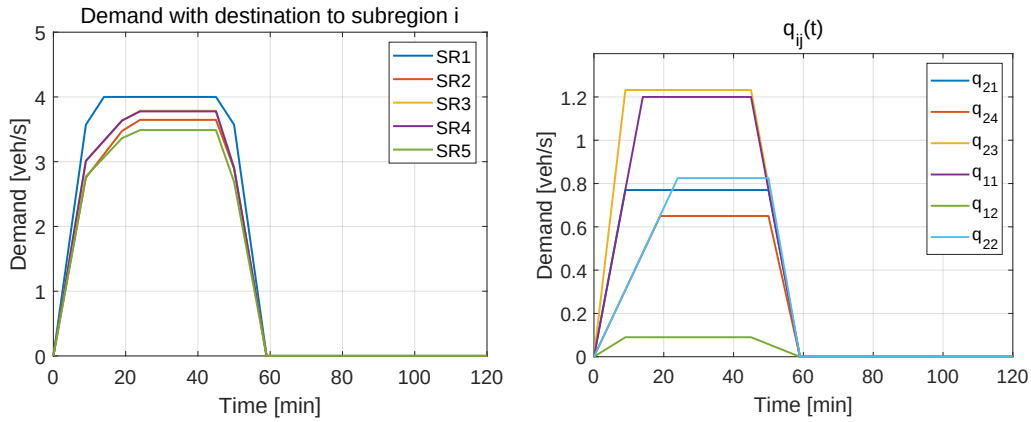
Figure 5.4 The MFD functions. MFD of subregions within (a) the city center, and (b) the periphery.

5.4.1 Case 1: Two-region network consisting of five subregions

5.4.1.1 Example 1: No uncertainties and heterogeneity

In Case 1-Example 1, the city center Region 1 is also denoted as Subregion 1, and the periphery Region 2 consists of four subregions, see Figure 5.3(a). The periphery subregions share an identical MFD as presented in Figure 5.4(b), while the city center is governed by an MFD with a higher capacity and throughput than the periphery as shown in Figure 5.4(a). The subregional MFD function is given by $G_i(n_i(t)) = p_i(n_i(t))/l_i$ (veh/s), $i = 1, \ldots, 5$. To be specific, $n_1^{cr} = 6784$ (veh), $n_1^{jam} = 13568$ (veh), and $\bar{G}_1^{\max} = 9.45$ (veh/s), while $n_i^{cr} = 3392$ (veh), $n_i^{jam} = 10000$ (veh), and $\bar{G}_i^{\max} = 6.3$ (veh/s) for $i = 2, \ldots, 5$. The average subregional trip length for vehicles in the periphery is $l_i = 3600$ (m), $i = 2, 3, 4, 5$; and we set $l_1 = 1.5\ l_2$. The travel demand pattern (see Figure 5.5(a)) mimics a peak period with one peak hour followed by one off-peak hour for congestion dissolving. As the CBD, Subregion 1 attracts more trips than the periphery subregions. Figure 5.5(b) depicts the OD-specific demand regarding Subregion 2 over time. The OD-specific demand profiles regarding Subregions 3, 4, and 5 follow a similar trend as Subregion 2 but are associated with a 10% coefficient of variation to represent the underlying stochasticity.

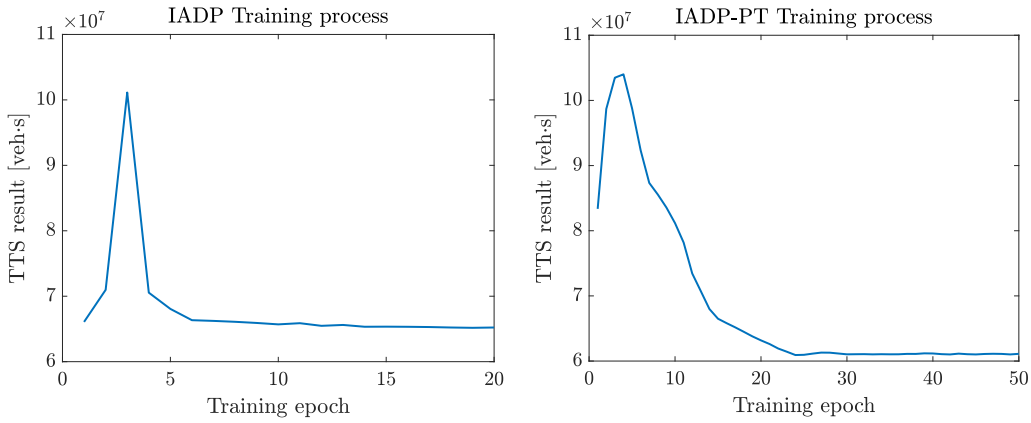
Five PCRG strategies are performed: (1) IADP, (2) IADP-PT, (3) MPC-PM as the benchmark, (4) MPC-UKF, and (5) PIL as the baseline. The key information on the



(a) (b)

Figure 5.5 Demand pattern of Case 1. (a) Demand with destination to subregions, and (b) OD-specific demand.

training and implementation of the IADP and the IADP-PT schemes in this case is given by Table 5.1. As expected, with more detailed data and a larger total epoch number, the IADP-PT approach has both a longer average CPU time per iteration and a longer total training time than the IADP approach. Figure 5.6(a) and Figure 5.6(b) show the training processes of the IADP and the IADP-PT schemes, respectively. For the IADP and the IADP-PT, the accumulative reward converges after around 10 and around 30 training epochs, respectively.

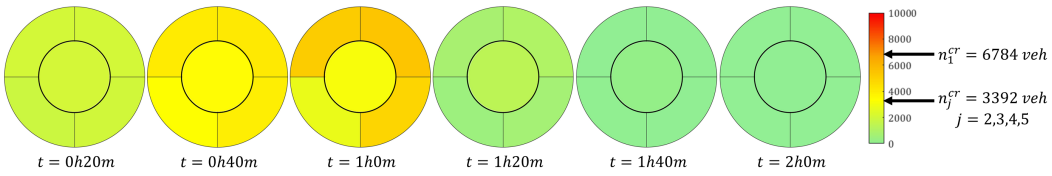


(a) IADP (b) IADP-PT

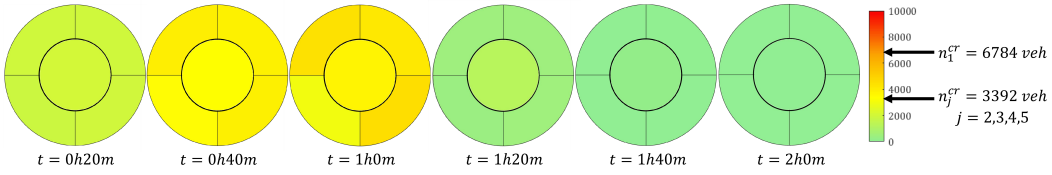
Figure 5.6 The IADP training process of Case 1-Example 1.

Table 5.1 Key information on the IADP training and implementation in Case 1

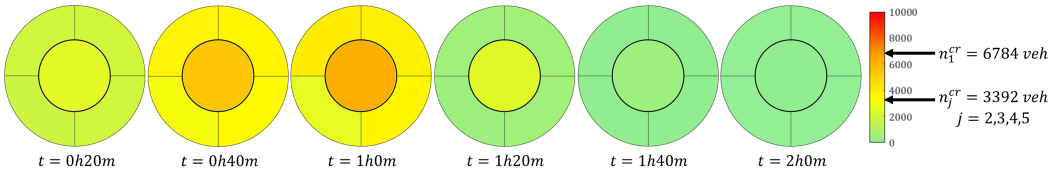
Hyperparameters	IADP	IADP-PT
Max iteration number	20	25
Total epoch number	20	50
Replay memory buffer size	1210	2420
Batch size	484	968
Critic NN basis function	$x_1^m x_2^n x_3^o x_4^p$, $m+n+o+p=4$, $m,n,o,p=0,1,2$	
Actor NN basis function	$\tanh(x_1^{m'} x_2^{n'} x_3^{o'} x_4^{p'})$, $m'+n'+o'+p'=3$, $m',n',o',p'=0,1,2$	
Iteration termination condition	$\|\hat{\mathbf{W}}_{k+1}-\hat{\mathbf{W}}_k\|<\varepsilon$ with $\varepsilon=10^{-3}$	
Training termination condition	$\lvert\frac{TTS_l-TTS_{l-1}}{TTS_{l-1}}\rvert<10^{-4}$	
Computational complexity		
Avg. CPU time per iteration	11.83 s	20.19 s
Avg. CPU time per epoch	236.59 s	504.75 s
Total CPU time for training	≈80 min	≈420 min
Data usage		
Training	$\{N_{IJ}(t), u_{ij}(t), \theta_{ij}^h(t)\}$	$\{n_{ij}(t), u_{ij}(t), \theta_{ij}^h(t)\}$
Implementation	$\{N_{IJ}(t)\}$	$\{n_{ij}(t)\}$



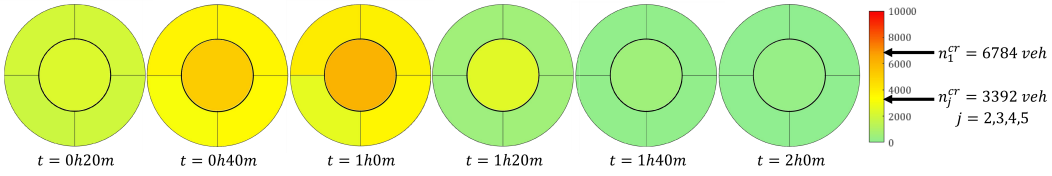
(a) IADP



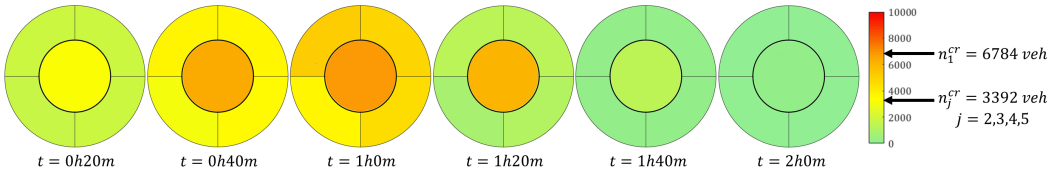
(b) IADP-PT



(c) MPC-PM



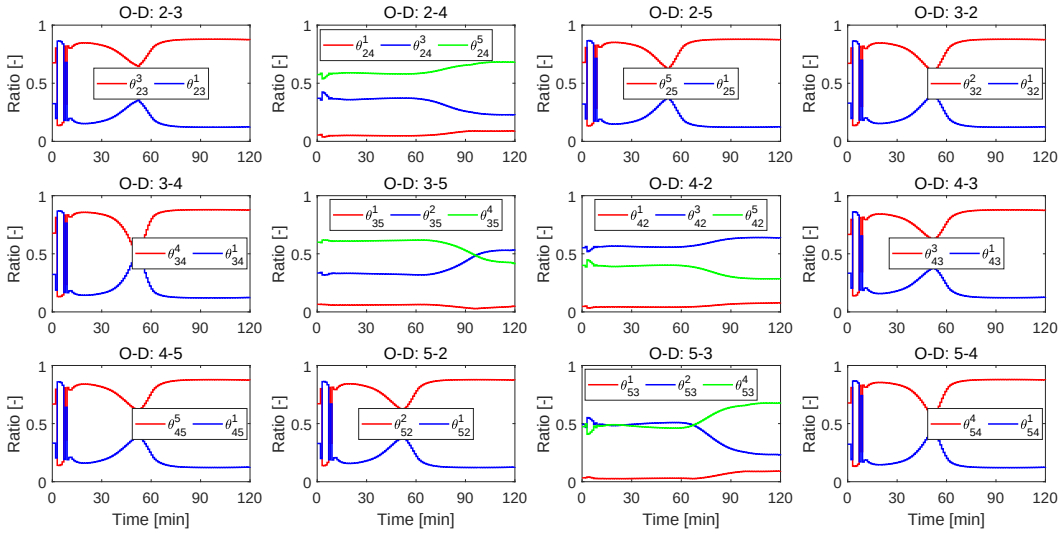
(d) MPC-UKF



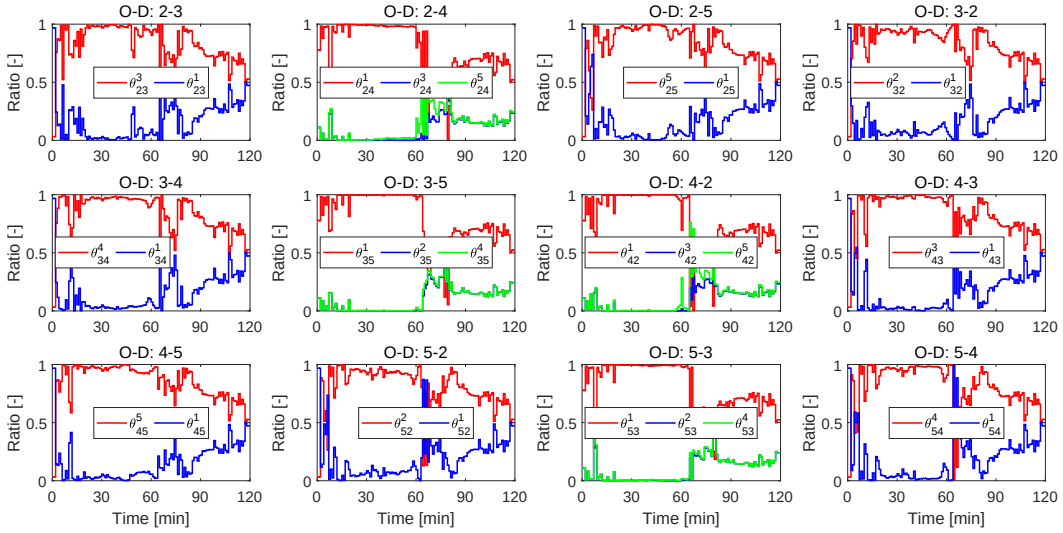
(e) PIL

Figure 5.7 Subregional accumulation evolution of Case 1-Example 1. (a) IADP, (b) IADP-PT, (c) MPC-PM, (d) MPC-UKF, and (e) PIL.

It is desired by the traffic managers that not only the PCRG strategies can minimize the total network delay, but also they can regulate the cross-boundary flows to avoid going through the CBD if possible. Figure 5.7 presents a series of snapshots over time that depict the subregional accumulation state evolution of Case 1-Example 1. As observed in Figure 5.7(e), central congestion cannot be avoided by applying the PIL scheme. Figure 5.7(c) and Figure 5.7(d) show that in the MPC scenarios,



(a) IADP



(b) MPC-PM

Figure 5.8 Route guidance signals of Case 1-Example 1. (a) IADP, and (b) MPC-PM.

the accumulation in the central region escalates to values very close to the critical point during the congestion period. From Figure 5.7(a) and Figure 5.7(b), we observe that both the IADP and IADP-PT schemes significantly decrease the average congestion level in the city center compared to the MPC schemes. Figure 5.8(a) and Figure 5.8(b) present the evolutions of route guidance schemes devised by IADP and MPC-PM, respectively. Applying both strategies, more inter-transfer flows between neighbor subregions in the periphery are assigned to traverse directly to the destination than those suggested to detour through the city center (see the top-left figures entitled 'O-D: 2-3' for example). For transfer flows between diagonal subregions (see e.g. in subfigures entitled 'O-D: 2-4'), more travelers are suggested by the IADP scheme to avoid the city center, while the MPC-PM scheme assigns more travelers to traverse through the city center. This indicates that the IADP scheme is able to assign as less traffic loads as possible to the CBD and hence improves the central region network efficiency, while the MPC scheme does not take this into account. More results of the accumulation, perimeter control, and route split ratio evolution are presented in Appendix A.3.

Table 5.2 summarizes the performance of the five PCRG strategies. The percentage numbers embraced by the parentheses in the second to the fifth columns represent the decrease of TTS compared with the baseline PIL strategy. The MPC-PM scheme achieves a significant improvement in minimizing TTS over the baseline PIL strategy. The MPC-PM scheme performs slightly better than the MPC-UKF scheme because the former uses the exact measurements from the plant. Despite being trained with limited data, the IADP achieves a comparable performance to the MPC-PM. This demonstrates the effectiveness of the proposed IADP method in learning the unknown traffic dynamics without using the detailed plant data and minimizing the total network delay. Moreover, trained with detailed plant-generated data, the IADP-PT approach outperforms the IADP and the MPC-based strategies. This validates the efficiency of the IADP approach in data usage. In addition, the IADP scheme can regulate the cross-boundary flows to avoid passing through the city center without affecting the overall system performance.

Table 5.2 Performance comparison among various PCRG schemes of Case 1-Example 1

	IADP	IADP-PT	MPC-PM	MPC-UKF	PIL
TTS (× 1e7 veh·s)	6.5240 (-20.3%)	6.1155 (-25.3%)	6.5424 (-20.1%)	6.5568 (-19.9%)	8.1889 (-)
Avg. CPU time/step (s)	2.0631e-4	1.8217e-3	1.0271	1.5464	1.9997e-2

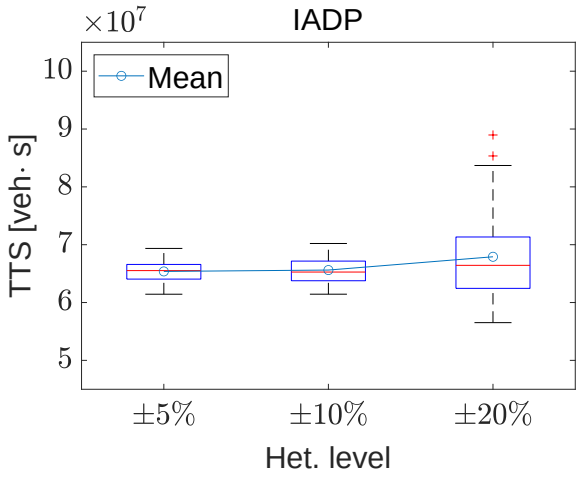
5.4.1.2 Example 2: MFD system subject to regional trip distance heterogeneity

In this example, we examine the performance of the pre-trained IADP scheme in Section 5.4.1.1 in cases of regional trip distance heterogeneity. Different from Example 1, the average regional trip distances for this example are time-varying. Let $\bar{l}_i$ (m) and $l_i(t) = (1+\varsigma(t)) \cdot \bar{l}_i$ (m), $i = 1, \ldots, 5$, denote the static and time-varying average sub-regional trip distances, respectively. The time-varying coefficient $\varsigma(t)$ represents the heterogeneity level. We set three heterogeneity levels: $\varsigma(t) \in \{\pm 5\%, \pm 10\%, \pm 20\%\}$. Note that the scenario of this example is an unseen environment to the IADP. In such a case, we compare the IADP against the benchmark MPC controller with the exact values of the time-varying average trip distance (MPC-PM). The MPC-IPM scheme is adopted as the baseline. For each investigated controller, we carry out fifty Monte-Carlo simulations per heterogeneity level.

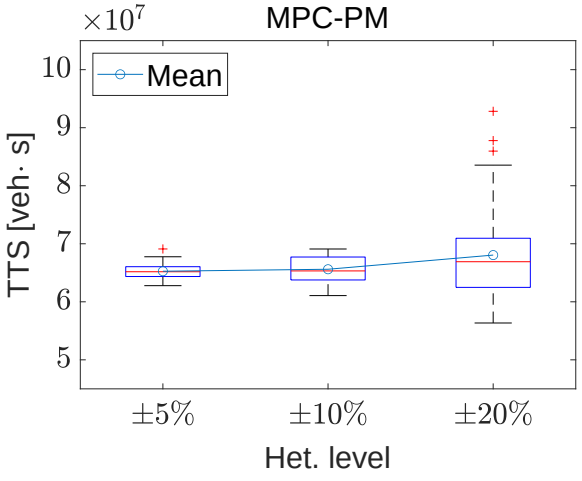
Figure 5.9 presents the TTS results and the associated variances delivered by the three schemes over different heterogeneity levels. Figure 5.9(c) implies that the assumption of static average regional trip distances significantly degrades the performance of the MPC-IPM scheme as the heterogeneity level increases, which is in line with the finding of Batista et al. (2021). In contrast, given the exact values of the dynamic trip lengths, the MPC-PM exhibits a stable performance and outperforms the MPC-IPM. The performance difference between MPC-PM and MPC-IPM becomes more evident as the heterogeneity level increases. The IADP-based controller achieves a comparable performance to the MPC-PM controller, see Figure 5.9(a). This indicates that the IADP approach can well adapt to the changes in the average regional trip distances. Recall that the IADP agent trained in Example 1 is directly employed without extra training. Thus, these results also demonstrate the generalizability of the proposed IADP approach, which is one of the advantages of our method over the perfectly model-based methods.

5.4.1.3 Example 3: MFD system subject to MFD errors

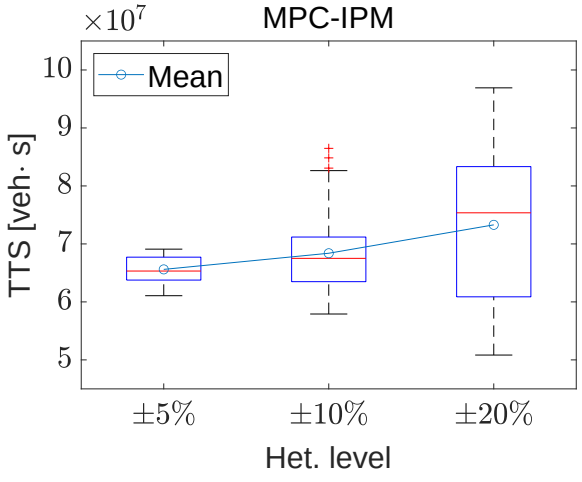
In this example, we study the effect of MFD calibration errors on the performance of the IADP scheme. A uniformly distributed term $\varrho_I \sim U(-\alpha \frac{N_I}{3600}, \alpha \frac{N_I}{3600})$ (veh/s) is added to the basic MFD to represent the actual one that considers the errors caused by the heterogeneous congestion distribution. We set three levels of MFD error: $\alpha \in \{0.05, 0.1, 0.2\}$. The other settings in this example are identical to those



(a)



(b)



(c)

Figure 5.9 Box plot of TTS results of Example 2. Performances of (a) IADP, (b) MPC-PM, and (c) MPC-IPM under various levels of regional trip distance heterogeneity.

in Example 1. The proposed IADP scheme is compared against the MPC-PM and MPC-IPM schemes.

Figure 5.10 shows the control performances over Monte-Carlo runs applying the considered schemes over different MFD error levels. We can observe from Figure 5.10(c) that the performance of the MPC-IPM degrades as the MFD error level increases. With perfect state measurements, the MPC-PM scheme significantly reduces the impact of MFD errors on the control performance compared to the MPC-IPM scheme, see Figure 5.10(b). Although the IADP endures a slight performance variation as the MFD error level increases, it still achieves a comparable performance to the MPC-PM scheme and outperforms MPC-IPM, see Figure 5.10(a). This indicates that the IADP scheme is robust to the MFD calibration errors.

5.4.1.4 Example 4: Driver compliance analysis for IADP

We consider the driver compliance rate in route guidance actuation in this case study. The driver compliance rate (CR) γ indicates the percentage of drivers following the route guidance recommendations of the traffic control scheme. Following Sirmatel and Geroliminis (2018), the realized route guidance command θ_{ij}^h at time t in the simulation is obtained as:

$$\theta_{ij}^h(t) = \gamma\hat{\theta}_{ij}^h(t) + (1-\gamma)\bar{\theta}_{ij}^h(t)$$

where $\hat{\theta}_{ij}^h(t)$ and $\bar{\theta}_{ij}^h(t)$ are the outputs of the IADP and the Logit model, respectively.

Simulations with seven different values of γ are conducted, which are summarized in Figure 5.11. Figure 5.11(c) shows that the whole network efficiency is not significantly influenced by the CR. A maximal 4% decrease in whole network efficiency is observed when γ drops from 1.0 to 0.4, which validates the robustness of the proposed approach against various levels of CRs. We can observe that an increase in the driver compliance rate results in a decrease in the congestion level of the city center, see Figure 5.11(a). With 100% compliance rate, the IADP significantly improves the mobility of the city center. When the CR is high (between 0.6 and 0.9), there is no significant difference in the performance of the IADP approach in alleviating central congestion. However, a low CR (equal to 0.5 or lower) still degrades the performance of the IADP in alleviating central congestion. This implies that CR=0.5 could be the inflection point when applying the IADP in central congestion alleviation.

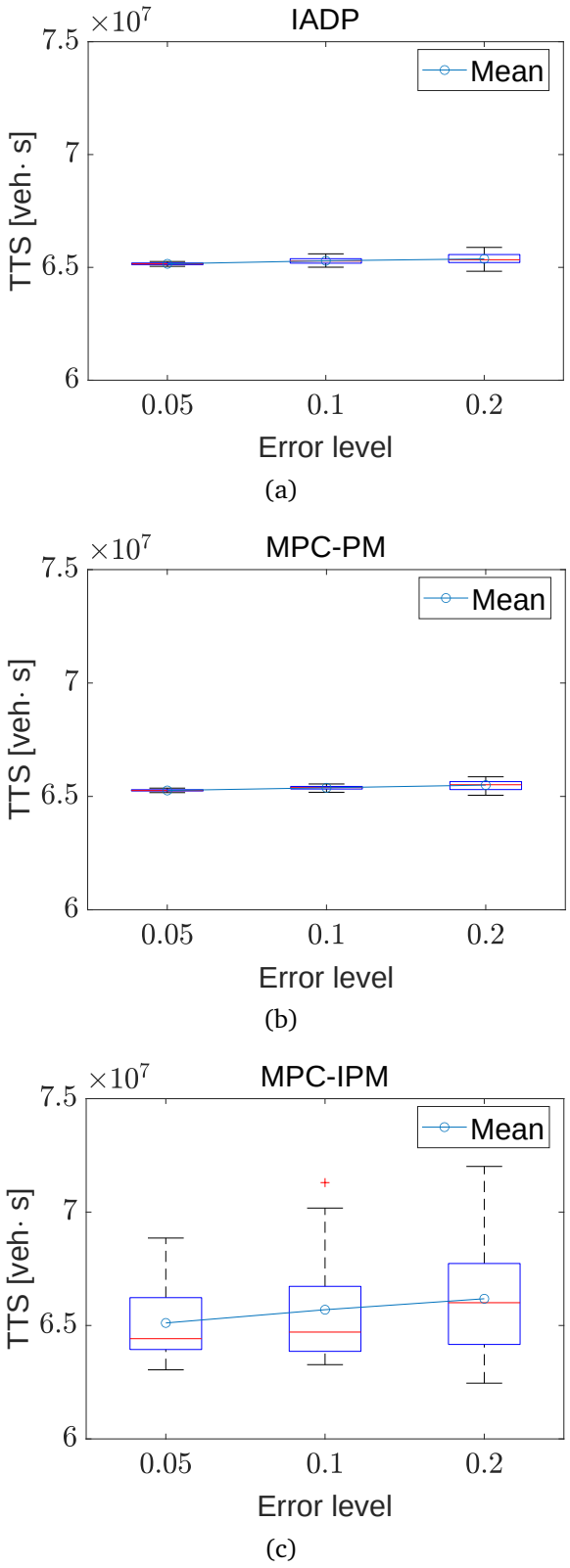
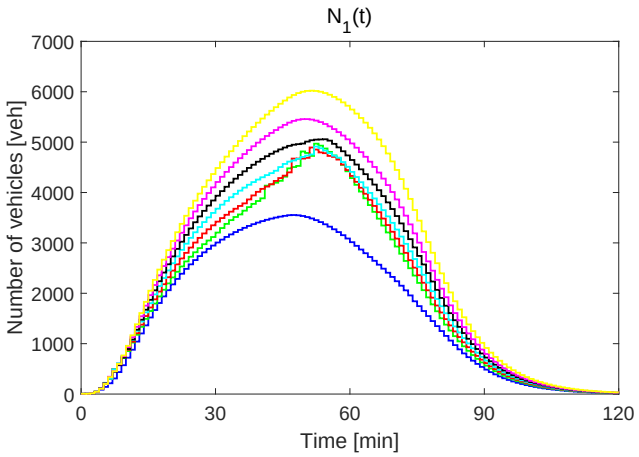
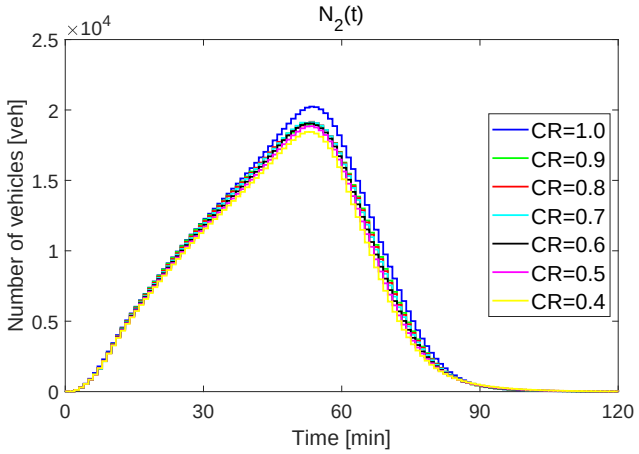


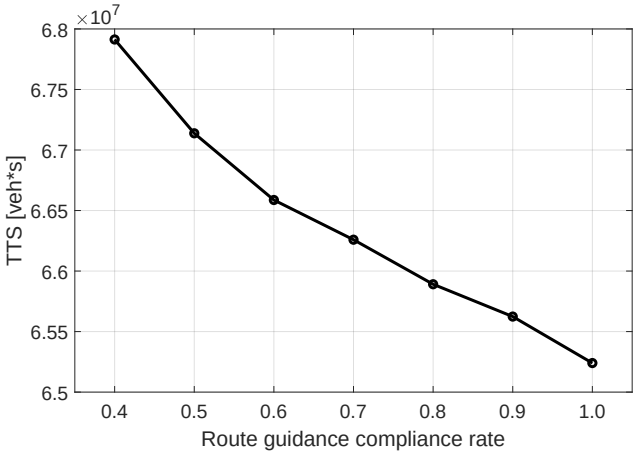
Figure 5.10 Box plot of TTS results of Example 3. Performances of (a) IADP, (b) MPC-PM, and (c) MPC-IPM under various levels of MFD error.



(a)



(b)



(c)

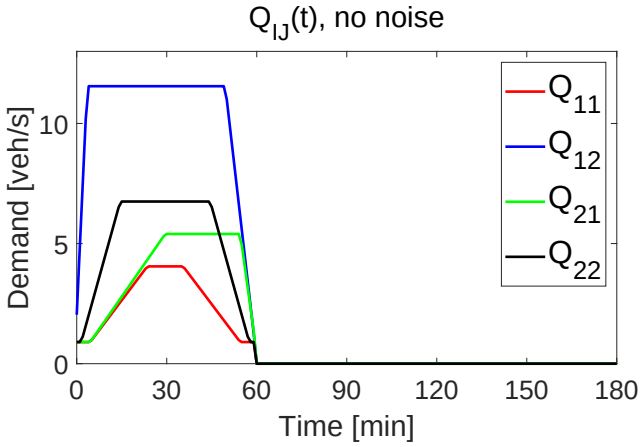
Figure 5.11 Performance comparison under different route guidance compliance rates. (a) State N_1, (b) State N_2, and (c) TTS.

5.4.2 Case 2: Two-region network consisting of sixteen subregions

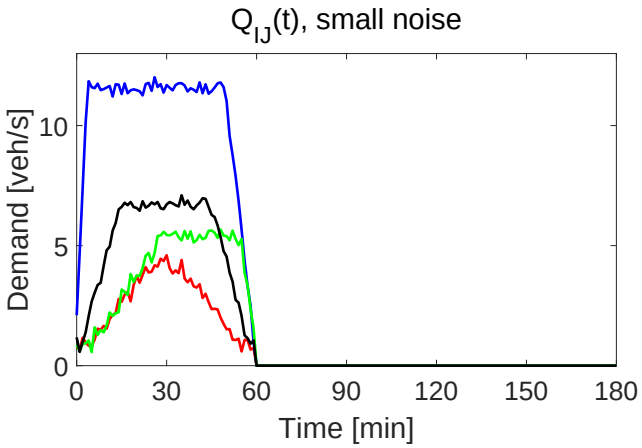
In Case 2, we examine the scalability and robustness against demand disturbance of the proposed IADP scheme. The urban network is composed of the city center (Region 2) and its periphery (Region 1), each further partitioned into 4 and 12 subregions, respectively (see Figure 5.3(b)). In this case, the model-plant dissimilarity is much stronger than in Case 1. In addition, different subregions are governed by different subregional MFDs. The actual subregional MFDs are given by $G_i(n_i(t)) = \eta_i \cdot \bar{G}_i(n_i(t))$ (veh/s), where $\bar{G}_i(n_i(t)) \triangleq p_i(n_i(t))/l_i$ (veh/s) is the basic subregional MFD function, and $\eta_i \sim U(0.9, 1.2)$ denotes the stochastic scale factor. The unit subregional MFDs for Region 1 and Region 2 are shown in Figure 5.4(b) and Figure 5.4(a), respectively. As the city center, subregions in Region 2 have a higher capacity and throughput than subregions in Region 1. The average subregional trip distance l_i is associated with a 10% coefficient of variation to represent the underlying heterogeneity. A base time-varying demand pattern (see Figure 5.12(a)) is adopted, mimicking the morning peak hour and the following two hours of low demand to fully clear the network. Region 1 generates most of the demand towards Region 2 that as the central business district attracts trips. We carry out the experiments in three scenarios with different levels of demand uncertainty. The nominal demand pattern is subject to external disturbance (e.g., measurement noise), as shown by Figure 5.12(b) (small) and Figure 5.12(c) (medium).

The IADP scheme is compared against the MPC+IADP scheme, which employs MPC-IPM as the perimeter controller and IADP as the route guidance strategy. Note that the MPC+IADP scheme has imperfect knowledge of the system (e.g., unknown demand noise and stochastic scale factor that determines the subregional MFDs). The reason why we do not implement MPC for both perimeter control and route guidance is that doing so could lead to real-time intractability. Sirmatel and Geroliminis (2018) and Yildirimoglu et al. (2018) reported that the MPC route guidance schemes cannot retain real-time feasibility when the region size of the network is much more than 7. The PIL strategy is employed as the baseline.

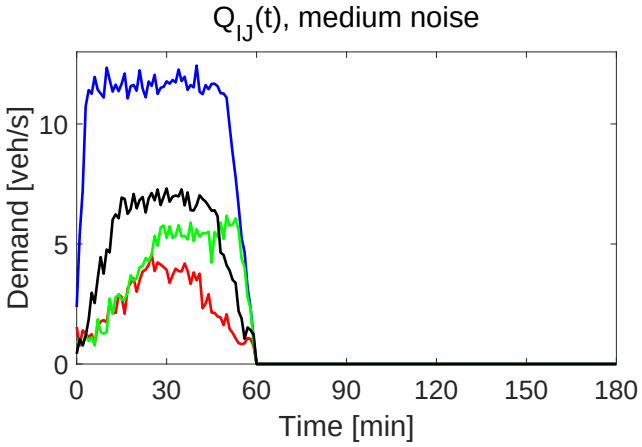
The accumulation evolution results are depicted in Figure 5.13. As can be observed from the results, applying the IADP strategy or the PIL scheme, the regional state evolution behaves similarly in all three cases. Applying the MPC+IADP scheme, the medium disturbance in travel demand induces higher maximal regional accumulation states of both regions than the small disturbance in travel demand. In addition, as depicted in Figure 5.13(b), Figure 5.13(d) and Figure 5.13(f), the IADP scheme



(a) No disturbance



(b) Small disturbance



(c) Medium disturbance

Figure 5.12 Demand profiles of Case 2. (a) nominal $Q_{IJ}(t)$, subject to (b) small and (c) medium disturbances.

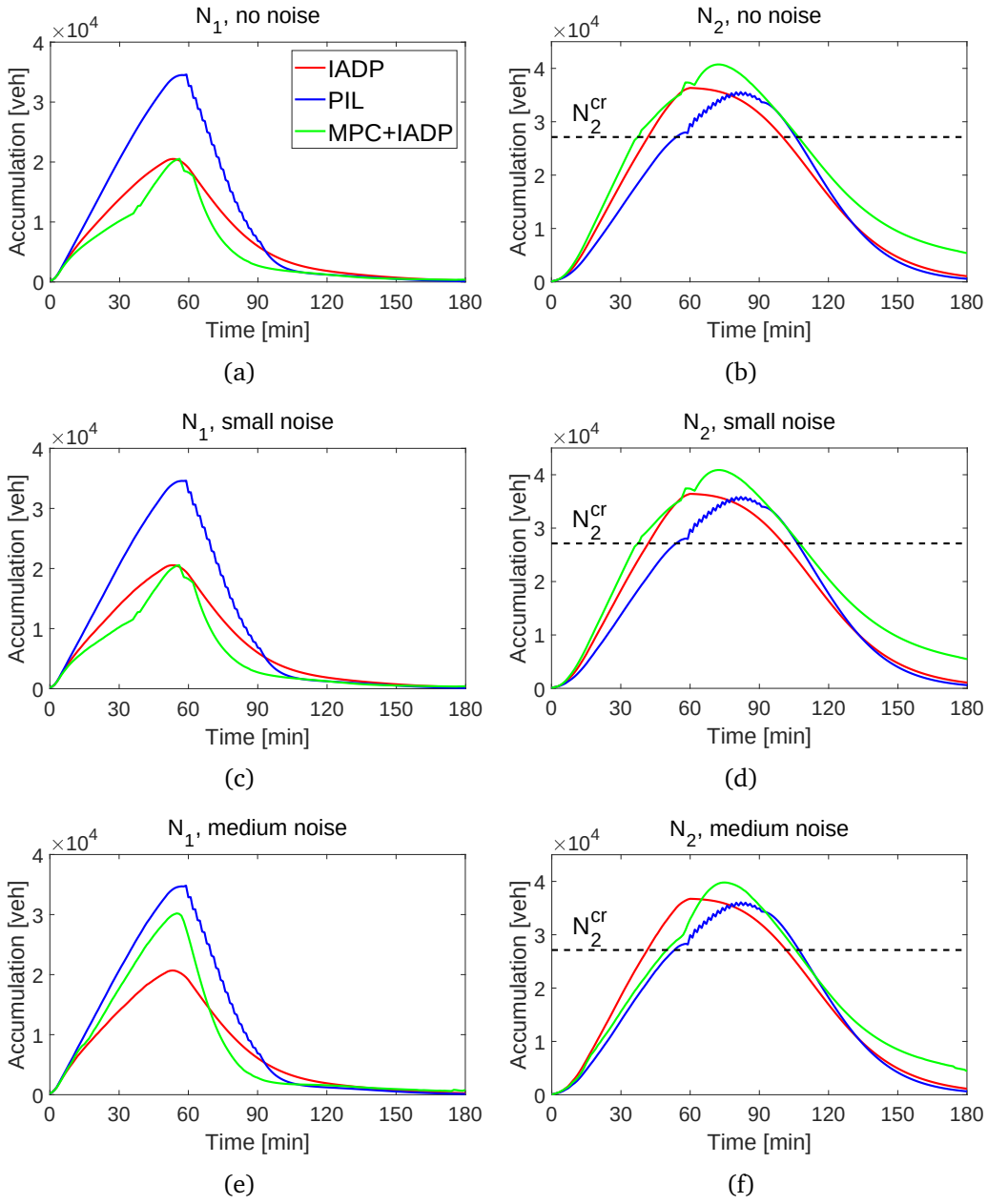


Figure 5.13 Performance of control strategies in Case 2. Evolution of (a) $N_1(t)$ and (b) $N_2(t)$ in the case of no disturbance; Evolution of (c) $N_1(t)$ and (d) $N_2(t)$ in the case of small disturbance; Evolution of (e) $N_1(t)$ and (f) $N_2(t)$ in the case of medium disturbance.

achieves the minimal total delay in Region 2 in both cases, which implies that the IADP scheme can regulate the cross-boundary flows to avoid going through the city center. The performance comparison is summarized in Table 5.3. Note that the MPC+IADP scheme has imperfect knowledge of the dynamics. Hence, when the demand noise level increases, the performance of the MPC+IADP scheme deteriorates. The IADP achieves over 7% improvement in minimizing the TTS in all three cases compared to the other schemes. The effect of increasing the demand noise level on the IADP performance is negligible. These results indicate the robustness of the proposed IADP approach against the demand uncertainty. It is worth noting that the average CPU time per calculation of the IADP is the minimum among the three schemes and is negligible compared to the control input update interval. This implies the real-time feasibility of the proposed OPCRG scheme in urban networks with a large region size.

Table 5.3 Comparison among different strategies in minimizing TTS ($\times$ 1e8 veh·s) of Case 2

	IADP	MPC+IADP	PIL
No demand noise	2.693284	2.884275	2.903544
	(-7.24%)	(-0.66%)	(-)
Small demand noise	2.703573	2.899021	2.922776
	(-7.50%)	(-0.81%)	(-)
Medium demand noise	2.745938	2.955109	2.957374
	(-7.15%)	(-0.07%)	(-)
Avg. CPU time/step (s)	2.3527e-2	3.8858	0.1244

5.5 Conclusions

This study proposed an iterative adaptive dynamic programming (IADP) approach to solving the optimal perimeter control and route guidance (OPCRG) problem for large-scale MFD-based urban networks. Different from the existing model-free methods, to the best of our knowledge, it is the first time that the model-plant mismatch is considered in devising the IADP framework for the PCRG of MFD-based urban networks. Compared with the model-based MPC approach that requires complete knowledge on the system dynamics, the IADP approach does not rely on any information on the plant dynamics but only the regional accumulation data and the subregional control input data. This is one of the prominent advantages of the IADP-based control strategy over the existing model-based controllers. Numerical

examples conducted in various scenarios demonstrated the effectiveness and superiority of the proposed IADP approach in the performance improvement of large-scale MFD-based urban networks by coupling perimeter control schemes and regional route guidance strategies.

There are three main types of MFD frameworks in the literature, as studied and compared in Huang et al. (2024). For future works, utilizing an MFD model (e.g., the trip-based model) for training while using another MFD model (e.g., accumulation-based model) for implementation would be an interesting research direction (Yildirimoglu and Ramezani, 2020).

The urban network considered in Chapter 5 is still far from realistic. Nowadays in megacities, ring expressways have been built to connect different parts of the city. It is desired that various traffic control strategies can cooperate in optimizing the whole network's traffic efficiency. Chapter 6 explores the cooperative control for a multi-region urban network with a ring expressway. With the MFD modeling the urban network and the asymmetric cell transmission model (ACTM) modeling the ring expressway, the next chapter couples three strategies: perimeter control, regional route guidance, and ramp metering.

6 Adaptive cooperative traffic control of a multi-region urban network with a ring expressway

Macroscopic fundamental diagrams (MFDs) have been widely adopted to model the traffic flow of large-scale urban networks. While coupling perimeter control with regional route guidance is an effective strategy to reduce congestion and network delays in large-scale urban networks, most studies overlook the role of expressways passing through urban areas. Ring expressways with on- and off-ramps are built in many megacities (e.g., Beijing) to connect the city's periphery areas. However, few studies have explored the cooperation of perimeter control, route guidance, and ramp metering strategies to improve network mobility. This paper aims to develop a cooperative adaptive dynamic programming (CADP) approach to solve the cooperative control problem for a mixed urban-expressway network. The network is composed of a multi-region urban network modeled by the MFD and a ring expressway going through the periphery regions modeled by the asymmetric cell transmission model. The proposed CADP approach trains the agents of perimeter control, route guidance, and ramp metering to cooperate fully to improve overall network performance. Numerical studies will demonstrate that the CADP outperforms the model-based uncoordinated strategy (i.e., proportional-integral perimeter control and ALINEA ramp metering scheme coupled with a Logit-based route choice modeling) in minimizing the total travel delay. In addition, the CADP-based strategy will effectively utilize the capacity of ring expressways, helping to achieve a better balance of traffic loads for the mixed urban-expressway system.

6.1 Introduction

In most literature (Haddad et al., 2013; Ding et al., 2020a; Yocum and Gayah, 2022), it is assumed that the freeway runs through the urban network, and this network representation is still far from realistic. Ring expressways are becoming more common and pivotal in megacities nowadays. However, a cooperative control

model integrating perimeter control, route guidance, and ramp metering for a multi-region urban network with ring expressways remains to be explored. In this study, we propose a cooperative control model for a multi-region urban network with a ring expressway. The urban network is modeled as a multi-region MFD system, with a ring expressway connecting these regions as shown in Figure 6.1. The expressway is modeled by the ACTM following Haddad et al. (2013) and Gomes and Horowitz (2006). The expressway and the urban network are connected by on-ramps and off-ramps, which are controlled by ramp metering. The perimeter control is used to control the transfer flow between regions. The route guidance is used to guide the vehicles through a sequence of regions/cells at a lower cost. The perimeter control, route guidance, and ramp metering are coordinated to improve the network performance.

Reinforcement learning (RL), a concept under the umbrella of artificial intelligence, has gained recent attention due to its success in video games and Go (Mnih et al., 2015; Silver et al., 2016). Adaptive dynamic programming (ADP) is an RL reformulation in the economics and management communities, which provides an approximate solution to the optimal control problem based on the Bellman optimality principle. Model-free methods such as RL and ADP enable optimal control to bypass the necessity of full knowledge of the model, thus allowing for the integration of uncertainties and dynamics changes into the optimal control. It was shown that the aforementioned RL and ADP-based control approaches can handle different levels of error in MFDs and noise in travel demand (Zhou and Gayah, 2021; Chen et al., 2022). However, most existing studies on RL/ADP based MFD traffic control are limited in network scale and control variable dimensionality. In practice, the traffic management of a mixed urban-expressway network requires the cooperation of various traffic control policies. To address this challenge, we propose a cooperative ADP (CADP) approach to solve the optimal cooperative control of a mixed urban-expressway system.

The remainder of the chapter is organized as follows: Section 6.2 introduces mixed network traffic modeling, including the MFD modeling of the urban road traffic and the ACTM modeling of the ring expressway. Section 6.3 presents the cooperative control problem (CCP) formulation and derives its standard solution based on Bellman's optimality principle, then a CADP-based approach is proposed to approximate the optimal solution to the CCP. Numerical results are presented in Section 6.4. Finally, Section 6.5 concludes the chapter.

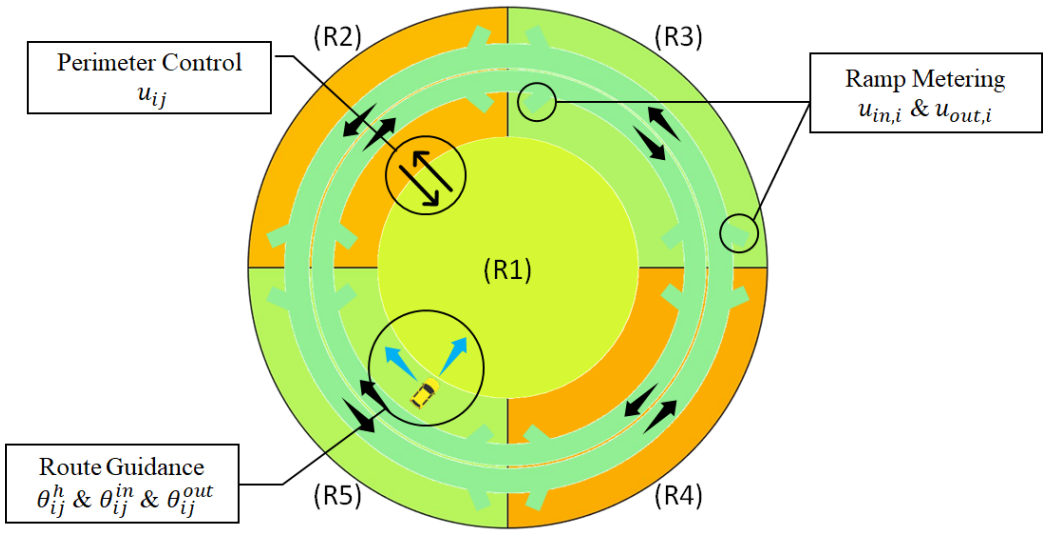


Figure 6.1 The network topology. (a) Mixed urban-expressway network, (b) Inside lanes and (c) Outside lanes of the ring expressway

6.2 Modeling traffic dynamics of the mixed network

This section introduces the modeling of the MFD-based urban traffic and the ACTM-based ring expressway traffic, respectively. To our best knowledge, Haddad et al. (2013) is the first to develop the traffic dynamics that integrate the MFD model and the ACTM model. Following their work, we extend their macroscopic traffic dynamics of a two-region-one-freeway network to a multi-region urban network with a bidirectional ring expressway (see Figure 6.1(a)). The urban network is partitioned into five homogeneous regions[1] with well-defined MFDs, denoted by $\{R1,\ldots,R5\}$, where $R1$ is regarded as the city center while the rest are the periphery regions. In addition, there is a bidirectional ring expressway that passes through all the periphery regions. The bidirectional ring expressway is composed of the inside lanes in a clockwise driving direction (see Figure 6.1(b)) and the outside lanes in a counterclockwise driving direction (see Figure 6.1(c)). The inside ring has only one on-ramp and one off-ramp within a periphery region. This setting also applies to the outside ring. Different from Haddad et al. (2013), the expressway does not carry travel demand and it is neither an origin nor a destination. Hence, a 5×5 origin-destination (O-D) matrix with the corresponding route choices presented by Table 6.1 is associated with the network demand. Let $q_{ij}(t)$ (veh/s) denote the demand from Region i to j at time t with $i, j = 1, \ldots, 5$. $\theta_{ij}^{h}(t) \in [0, 1]$ is the route split ratio for the transfer flow in Region i with final destination j through the next immediate Region h, while $\theta_{ij}^{in}(t), \theta_{ij}^{out}(t) \in [0, 1]$ denote the transfer flow from i to j through the inside and outside ring expressway, respectively. Note that $\sum_{h\in\mathcal{H}_i} \theta_{ij}^{h}(t) + \theta_{ij}^{in}(t) + \theta_{ij}^{out}(t) = 1$, where $\mathcal{H}_i$ is the set of regions that are directly reachable from Region i.

[1] For simplicity, we consider such a five-region partition. However, the network partition can be general and does not affect the application of the proposed method.

Table 6.1 O-D matrix and route choices in the mixed network

Origin	Destination				
	$R1$	$R2$	$R3$	$R4$	$R5$
$R1$	q_{11}: θ_{11}^{1}	q_{12}: θ_{12}^{2}	q_{13}: θ_{13}^{3}	q_{14}: θ_{14}^{4}	q_{15}: θ_{15}^{5}
$R2$	q_{21}: θ_{21}^{1}	q_{22}: θ_{22}^{2}	q_{23}: θ_{23}^{3}, θ_{23}^{in}	q_{24}: θ_{24}^{1}, θ_{24}^{3}, θ_{24}^{5}, θ_{24}^{in}, θ_{24}^{out}	q_{25}: θ_{25}^{5}, θ_{25}^{out}
$R3$	q_{31}: θ_{31}^{1}	q_{32}: θ_{32}^{2}, θ_{32}^{out}	q_{33}: θ_{33}^{3}	q_{34}: θ_{34}^{4}, θ_{34}^{in}	q_{35}: θ_{35}^{1}, θ_{35}^{2}, θ_{35}^{4}, θ_{35}^{in}, θ_{35}^{out}
$R4$	q_{41}: θ_{41}^{1}	q_{42}: θ_{42}^{1}, θ_{42}^{3}, θ_{42}^{5}, θ_{42}^{out}, θ_{42}^{in}	q_{43}: θ_{43}^{3}, θ_{43}^{out}	q_{44}: θ_{44}^{4}	q_{45}: θ_{45}^{5}, θ_{45}^{in}
$R5$	q_{51}: θ_{51}^{1}	q_{52}: θ_{52}^{2}, θ_{52}^{in}	q_{53}: θ_{53}^{1}, θ_{53}^{2}, θ_{53}^{4}, θ_{53}^{out}, θ_{53}^{in}	q_{54}: θ_{54}^{4}, θ_{54}^{out}	q_{55}: θ_{55}^{5}

6.2.1 MFD-based urban traffic modeling

Let $\mathcal{R}$ be the set of all regions in the urban network and $|\mathcal{R}| = 5$. $n_{ij}(t)$ (veh) represents the accumulation in Region i with final region destination j. $n_i(t)$ (veh) is the total accumulation in Region i and $n_i(t) = \sum_{j\in\mathcal{R}} n_{ij}(t)$. $p_i(t)$ (veh·m/s) defines the MFD production for Region i, which is the total distance traveled by all vehicles in Region i and equal to the sum of the transfer and internal flows multiplied by the average trip length $l_i(t)$ (m) in Region i. $u_{ij} \in [0, 1]$ is the perimeter controller that controls the transfer flows on the border between Regions i and j.

For $\forall i \in \mathcal{R}$, denote Δt as the sample time interval, then the mass conservation equations in a discrete-time form for the urban regions are given as follows:

$$n_{ii}(t+1) = n_{ii}(t) + \Delta n_{ii}(t) \cdot \Delta t,\ i \in \mathcal{R} \tag{6.1a}$$

$$n_{ij}(t+1) = n_{ij}(t) + \Delta n_{ij}(t) \cdot \Delta t,\ i \in \mathcal{R},\ j \in \mathcal{H}_i \backslash \{i\} \tag{6.1b}$$

$$n_{ir}(t+1) = n_{ir}(t) + \Delta n_{ir}(t) \cdot \Delta t,\ i \in \mathcal{R},\ r \in \mathcal{R} \backslash \mathcal{H}_i \tag{6.1c}$$

where $\Delta n_{ii}(t),\ \Delta n_{ij}(t),\ \Delta n_{ir}(t)$ are defined as

$$\Delta n_{ii}(t) = \begin{cases} q_{ii}(t) - m_{ii}(t) + \sum_{h\in\mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hi}(t), & i = 1 \\ q_{ii}(t) - m_{ii}(t) + \sum_{h\in\mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hi}(t) + m^{off}_i(t), & i \neq 1 \end{cases} \tag{6.2a}$$

$$\Delta n_{ij}(t) = \begin{cases} q_{ij}(t) - \sum_{h\in\mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ij}(t) + \sum_{h\in\mathcal{H}_i; h\neq j} u_{hi}(t) \cdot \hat{m}^i_{hj}(t), \\ \qquad\qquad \forall\, j \in \mathcal{H}_i \backslash \{i\}, i = 1 \\ q_{ij}(t) - \sum_{h\in\mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ij}(t) + \sum_{h\in\mathcal{H}_i; h\neq j} u_{hi}(t) \cdot \hat{m}^i_{hj}(t) - m^{on}_{ij}(t), \\ \qquad\qquad \forall\, j \in \mathcal{H}_i \backslash \{i\}, i \neq 1 \end{cases} \tag{6.2b}$$

$$\Delta n_{ir}(t) = \begin{cases} q_{ir}(t) - \sum_{h\in\mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ir}(t) + \sum_{h\in\mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hr}(t), \\ \qquad\qquad \forall\, r \in \mathcal{R} \backslash \mathcal{H}_i, i = 1 \\ q_{ir}(t) - \sum_{h\in\mathcal{H}_i} u_{ih}(t) \cdot \hat{m}^h_{ir}(t) + \sum_{h\in\mathcal{H}_i} u_{hi}(t) \cdot \hat{m}^i_{hr}(t) - m^{on}_{ir}(t), \\ \qquad\qquad \forall\, r \in \mathcal{R} \backslash \mathcal{H}_i, i \neq 1 \end{cases} \tag{6.2c}$$

where $m_{ii}(t)$ (veh/s) denotes the transfer flow from Region i with final destination Region i, while $m^h_{ij}(t)$ (veh/s) is the transfer flow for accumulation in i with final destination j through the next immediate Region h, $h \in \mathcal{H}_i$. $m_{ii}(t)$ and $m^h_{ij}(t)$ are defined respectively as follows:

$$m_{ii}(t) = \frac{n_{ii}(t)}{n_i(t)} \cdot \frac{p_i(n_i(t))}{l_i(t)}$$

$$m_{ij}^{h}(t) = \theta_{ij}^{h}(t) \cdot \frac{n_{ij}(t)}{n_i(t)} \cdot \frac{p_i(n_i(t))}{l_i(t)}$$

Note that high accumulation in an urban region can limit the inflow reception from the boundary. Therefore, the definition of capacity-restricted transfer flow from Region i to j passing through h immediately, $\hat{m}_{ij}^{h}(t)$, is introduced (Ramezani et al., 2015; Yildirimoglu et al., 2015; Sirmatel and Geroliminis, 2018):

$$\hat{m}_{ij}^{h}(t) = \min\left[m_{ij}^{h}(t),\ \frac{m_{ij}^{h}(t)}{\sum_{k\in\mathcal{R};k\neq i} m_{ik}^{h}(t)} \cdot r_{ih}(n_h(t))\right]$$

where $r_{ih}(\cdot)$ (veh/s) is the receiving flow capacity of Region $h \in \mathcal{H}_i$, from Region i. We consider that the receiving capacity is a piecewise function of $n_h(t)$ as follows:

$$r_{ih}(n_h(t)) = \begin{cases} r_{ih}^{\max}, & 0 \le n_h(t) \le \alpha \cdot n_h^{jam} \\ -\frac{r_{ih}^{\max}}{(1-\alpha)\cdot n_h^{jam}} \cdot n_h(t) + \frac{r_{ih}^{\max}}{1-\alpha}, & \alpha \cdot n_h^{jam} < n_h(t) \le n_h^{jam} \end{cases}$$

$m_i^{off}(t)$ (veh/s) denotes the inflow from the off-ramp into Region i. $m_{ij}^{on}(t) = \min\left[\theta_{ij}^{y} \cdot \frac{n_{ij}}{n_i} \cdot \frac{p_i(n_i)}{l_i},\ \frac{n_{on,i}^{\max} - n_{on,i}}{\Delta t}\right]$ (veh/s) is the transfer flow from the urban network to enter the on-ramp in Region i, where $n_{on,i}$ (veh) is the queue length of the on-ramp in Region i at time step t, $n_{on,i}^{\max}$ (veh) is the maximum queue length of the on-ramp in Region i, and $y \in \{in,\ out\}$ is the indicator that indicates which ring expressway (inside ring or outside ring) the cross-boundary flow is using.

6.2.2 ACTM-based ring expressway traffic modeling

Following Haddad et al. (2013), we adopt the ACTM in Gomes and Horowitz (2006) to model the ring expressway traffic dynamics.

In the ACTM, both the inside and outside rings of the expressway are divided into L cells, where each cell l of the expressway contains at most one on- or one off-ramp. The number of vehicles in cell l at time t is denoted by $x_l(t)$ (veh), while $f_l(t)$ (veh) is the number of vehicles moving from cell l to $l+1$ during time t. Each cell l has a triangular fundamental diagram with the following parameters: $w_l \in [0,1]$ is the normalized congestion wave speed, $v_l \in [0,1]$ is the normalized free-flow speed, $x_l^{\max}$ (veh/lane) is the jam accumulation, and $\bar{f}_l$ (veh/h/lane) is the mainline capacity.

The on-ramp is fed from $m_{ij}^{on}(t)$ (veh/s), i.e., the maximum output that can flow from the periphery Region i at time t computed by the MFD. The unmetered on-ramp flow $f_{on,l}(t)$ (veh) is the number of vehicles that can enter cell l from its on-ramp during time step t, which is calculated as follows:

$$f_{on,l}(t) = \min\left[n_{on,i}(t) + \sum_{j\in\mathcal{R}\setminus\{i\};h\in\mathcal{H}_i} m_{ij}^{on}(t)\cdot\Delta t,\ \xi_l\cdot(x_l^{\max}-x_l(t)),\ s_{on,i}\cdot\Delta t\right] \quad (6.3)$$

where i is the region that the on-ramp belongs to, $s_{on,i}$ (veh/s) is the maximum number of vehicles that can enter the expressway from the on-ramp belonging to Region i, and $\xi \in [0,1]$ is the on-ramp flow allocation parameter. The on-ramp metering control inputs, denoted by $u_{on,i}(t)$ (-) are introduced at the entrance of the expressway to control the flow entering from Region i to the expressway. The queue dynamic for the on-ramp belonging to Region i with $u_{on,i}(t)$, considering the on-ramp maximum queue length, is as follows:

$$n_{on,i}(t+1) = \min\left[n_{on,i}(t) + \sum_{j\in\mathcal{R}\setminus\{i\};h\in\mathcal{H}_i} m_{ij}^{on}(t)\cdot\Delta t - u_{on,i}(t)\cdot f_{on,l}(t),\ n_{on,i}^{\max}\right] \quad (6.4)$$

The mainline flow in the expressway is calculated as follows:

$$\begin{aligned} f_l(t) = \min\Big[&(1-\beta_l(t))v_l\cdot(x_l(t)+\gamma u_{on,i}(t)f_{on,l}(t)),\ F_l(t),\\ &w_{l+1}\cdot(x_{l+1}^{\max}-x_{l+1}(t)-\gamma u_{on,i}(t)f_{on,l+1}(t))\Big] \end{aligned} \quad (6.5)$$

where $\beta_l(t)$ (-) is the split ratio for the off-ramp (if exists) in cell l, γ (-) $\in [0,1]$ is the on-ramp (if exists) flow blending coefficient, and $F_l(t) = \min[\bar{f}_l,\ (1-\beta_l(t))/\beta_l(t)\cdot \bar{f}_{off,l}]$, where $\bar{f}_{off,l}$ (veh) is the off-ramp capacity. The exit flow of the off-ramp in cell l, $f_{off,l}(t)$ (veh), is determined as follows:

$$f_{off,l}(t) = \frac{\beta_l(t)}{1-\beta_l(t)} f_l(t) \quad (6.6)$$

In our case study, we do not intend to regulate the off-ramp meterings. Then $m_i^{off}(t)$ is the summation of all $f_{off,l}(t)$s if these off-ramps belong to Region i.

Finally, the mainline mass conservation is

$$x_l(t+1) = x_l(t) + f_{l-1}(t) + u_{on,i}(t)f_{on,l}(t) - f_l(t) - f_{off,l}(t) \quad (6.7)$$

where $f_{on,l}(t) = 0$ and/or $f_{off,l}(t) = 0$ if cell l does not contain an on-ramp and/or an off-ramp, respectively.

6.3 Adaptive cooperative traffic controller design

In this section, we propose the cooperative adaptive dynamic programming (CADP) approach to solving the cooperative control problem (CCP) of the mixed urban-expressway system. First, we present the problem formulation of the CCP. The standard solution to the CCP, namely the associated Hamilton-Jacobi-Bellman (HJB) equation, is then presented. Due to the strong nonlinearity of the HJB, it is intractable to obtain the optimal solution to the CCP. Hence, we finally propose the CADP method to approximate the optimal solution to the CCP.

6.3.1 Formulation of cooperative control problem

In the mixed network control problem, there are three types of controllers to minimize the network total delay: the perimeter controllers for the urban regions, the on-ramp meterings for the ring expressway, and the route guidance system for both networks. The aim of cooperative control for the mixed urban-expressway system is to minimize the total time spent (TTS) of the whole network. Let $u(t)$, $u_{on}(t)$ and $\theta(t)$ be the vectors of control variables $u_{ij}(t)$, $u_{on,i}(t)$ and $(\theta_{ij}^{h}(t), \theta_{ij}^{in}(t), \theta_{ij}^{out}(t))$, respectively. The CCP formulation is given as follows:

$$J = \min_{u(t),u_{on}(t),\theta(t)} \sum_{t=0}^{T} \Delta t \cdot \left(\sum_{i} \sum_{j} n_{ij}(t) + \sum_{l} x_l(t) + \sum_{i} n_{on,i}(t) \right) \tag{6.8}$$

subject to

$$0 \le \sum_{j} n_{ij}(t) \le n_i^{jam}$$

$$0 \le x_l(t) \le x_l^{\max}$$

$$0 \le n_{on,i}(t) \le n_{on,i}^{\max}$$

$$u_{\min} \le u_{ij}(t) \le u_{\max}$$

$$u_{\min}^{on} \le u_{on,i}(t) \le u_{\max}^{on}$$

$$0 \le \theta_{ij}^{h}(t), \theta_{ij}^{out}(t), \theta_{ij}^{in}(t) \le 1$$

$$\sum_{h \in \mathcal{H}_i} \theta_{ij}^{h}(t) + \theta_{ij}^{in}(t) + \theta_{ij}^{out}(t) = 1$$

(6.1) – (6.7).

where $u_{\min}$, $u_{\max}$ denote the lower and upper bounds of the perimeter controller; $u_{\min}^{on}$, $u_{\max}^{on}$ are the lower and upper bounds of the ramp metering control; n_i^{jam}, $x_l^{\max}$ and $n_{on,i}^{\max}$ are the capacities of the urban network accumulation, expressway cell accumulation and on-ramp queue length, respectively.

6.3.2 A policy iteration method to solve the CCP

The mixed urban-expressway traffic dynamics of the CCP can be considered as the following three-player game system

$$x(t+1) = f(x(t), u(t), u_{on}(t), \theta(t)) \tag{6.9}$$

where $x(t) \in \mathbf{\Omega} \subset \mathbb{R}^{\bar{r}}$ is the state vector containing all $(n_{ij}(t), x_l(t), n_{on,i}(t))$ terms with $\bar{r}$ a positive integer equal to the summation of numbers of all state variables. $u(t) \in \mathbb{R}^{D_1}$, $u_{on}(t) \in \mathbb{R}^{D_2}$ and $\theta(t) \in \mathbb{R}^{D_3}$ are now regarded as fully-cooperative players of the game system (6.9), taking actions together as a team. $f(\cdot,\cdot,\cdot,\cdot)$ is Lipschitz continuous on the compact set $\mathbf{\Omega}$ containing the origin and $f(0,0,0,0) = 0$. (6.9) is the compact vector form of the state-space model given by (6.1)-(6.7).

Remark 6.3.1 The MFD traffic dynamics can be written as a continuous-time dynamical system as in Haddad (2015). The MFD traffic dynamics can be further rewritten in a control-affine form (Zhong et al., 2018a; Zhong et al., 2018b; Su et al., 2020; Chen et al., 2022). Incorporating the regional route choice model, Ramezani et al. (2015) firstly distinguished the parsimonious regional model for optimization from the more detailed subregional plant that replicates the reality and is used only for simulation. Previous works have been dedicated to addressing model-plant mismatch when designing perimeter control and route guidance strategies (Yildirimoglu et al., 2018; Batista et al., 2019; Batista et al., 2021). Such model-plant mismatch makes it extremely difficult to rewrite the original MFD traffic dynamics into an input-affine form. Hence, the existing ADP approaches designed for affine systems are no longer valid. Recently, Chen et al. (2024) introduced a pre-compensator for the input variable and defined an augmented state variable that contains both the original state and input variables. By doing this, the original MFD traffic dynamics are then expressed by an affine system.

Different from the aforementioned studies, this study investigates the cooperative traffic control of a mixed urban-expressway network. Unlike the continuous-time MFD model utilized in the previous chapters, the ACTM model used for modeling the

ring expressway is generally expressed in the discrete-time form. Thus, we rewrite the mixed urban-expressway traffic dynamics into the discrete-time nonaffine system (6.9). ∎

For optimal control of the fully cooperative game, the cost function associated with the primary objective function (6.8) is given by

$$C(x_0, u, u_{on}, \theta) = \sum_{t=0}^{T} U(x(t), u(t), u_{on}(t), \theta(t))$$

where x_0 is the initial state value, and U is the utility function (also known as the cost-to-go function, see Section 1.1 in Bertsekas and Tsitsiklis, 1996) for the game system generally defined as

$$U(x, u, u_{on}, \theta) = x^T Q x + u^T R_1 u + u_{on}^T R_2 u_{on} + \theta^T R_3 \theta \tag{6.10}$$

Here without loss of generality, Q, R_1, R_2 and R_3 are positive-definite diagonal matrices of proper dimension.

Based on (6.10), the associated value function is given as

$$V(x(t)) = \sum_{\tau=t}^{T} U(x(\tau), u(\tau), u_{on}(\tau), \theta(\tau)) = U(x(t), u(t), u_{on}(t), \theta(t)) + V(x(t+1))$$

Based on Bellman's optimality principle, the optimal value function satisfies the following discrete-time HJB equation

$$V^*(x(t)) = \min_{u(t), u_{on}(t), \theta(t)} \{U(x(t), u(t), u_{on}(t), \theta(t)) + V(x(t+1))\} \tag{6.11}$$

The optimal perimeter control $u^*(\cdot)$, ramp metering $u_{on}^*(\cdot)$, and route guidance $\theta^*(\cdot)$ should satisfy

$$\begin{aligned}\{u^*(t), u_{on}^*(t), \theta^*(t)\} =& \{u^*(x(t)), u_{on}^*(x(t)), \theta^*(x(t))\} \\ =& \arg \min_{u(t), u_{on}(t), \theta(t)} \{U(x(t), u(t), u_{on}(t), \theta(t)) + V^*(x(t+1))\}\end{aligned}$$

Then the optimal value function can be written as

$$V^*(x(t)) = U(x(t), u^*(t), u_{on}^*(t), \theta^*(t)) + V^*((x(t), u^*(t), u_{on}^*(t), \theta^*(t)))$$

Due to the strong nonlinearity and nonanalyticity, solving this discrete-time HJB (6.11) explicitly is extremely difficult. The policy iteration (PI) method can be used

to overcome this difficulty. The PI method contains the policy evaluation and policy improvement steps that are updated through iterations between

$$V^k(x(t)) = U(x(t), u^k(t), u^k_{on}(t), \theta^k(t)) + V^k((x(t), u^k(t), u^k_{on}(t), \theta^k(t))) \tag{6.12}$$

and

$$\begin{aligned}&\{u^{k+1}(x(t)), u^{k+1}_{on}(x(t)), \theta^{k+1}(x(t))\}\\ &\quad = \arg \min_{u(t), u_{on}(t), \theta(t)} \{U(x(t), u(t), u_{on}(t), \theta(t)) + V^k(x(t+1))\}\end{aligned} \tag{6.13}$$

where k is the iteration step. $V^k(x(t))$ and $\{u^{k+1}(x(t)), u^{k+1}_{on}(x(t)), \theta^{k+1}(x(t))\}$ approximate the optimal value and policy functions, respectively.

6.3.3 An off-line iterative learning scheme

To implement the CADP approach based on the PI algorithm, a multi-agent actor-critic neural network (MACNN) framework is constructed to simultaneously approximate the optimal value function and policy functions.

The critic neural network (NN) is adopted to approximate the value function $V^k(x)$ with the ideal NN representation defined as follows

$$V^{k+1}(x) = w^T_{c,\,k+1} \cdot \psi(x) + \varepsilon^c_{k+1} \tag{6.14}$$

where $w_{c,\,k+1} \in \mathbb{R}^{K_c}$ is the ideal weight vector to be learned from data, $\psi \in \mathbb{R}^{K_c}$ is the activation function of the critic NN, K_c is the number of the hidden neurons, and $\varepsilon^c_{k+1} \in \mathbb{R}^1$ is the approximation error.

The estimation of the ideal value function is defined as follows

$$\hat{V}^{k+1}(x) = \hat{w}^T_{c,\,k+1} \cdot \psi(x) \tag{6.15}$$

where $\hat{w}_{c,\,k+1} \in \mathbb{R}^{K_c}$ is the estimated weight vector to be learned from data.

The following actor NNs are used to approximate the ideal policy functions:

$$u^{k+1}(x) = w^T_{d,\,k+1} \cdot \phi(x) + \varepsilon^d_{k+1} \tag{6.16a}$$

$$u^{k+1}_{on}(x) = w^T_{r,\,k+1} \cdot \chi(x) + \varepsilon^r_{k+1} \tag{6.16b}$$

$$\theta^{k+1}(x) = w^T_{p,\,k+1} \cdot \varphi(x) + \varepsilon^p_{k+1} \tag{6.16c}$$

where $\phi(x) \in \mathbb{R}^{K_d}$, $\chi(x) \in \mathbb{R}^{K_r}$, and $\varphi(x) \in \mathbb{R}^{K_p}$ are vectors of activation functions of the actor NNs. K_d, K_r, and K_p are the numbers of the hidden neurons. $w_{d,\,k+1} \in \mathbb{R}^{K_d \times D_1}$, $w_{r,\,k+1} \in \mathbb{R}^{K_r \times D_2}$, and $w_{p,\,k+1} \in \mathbb{R}^{K_p \times D_3}$ are the ideal actor NN weight matrices, ε^d_{k+1}, ε^r_{k+1}, and ε^p_{k+1} are approximation errors of proper dimensions.

The outputs of the actor NNs are expressed by the estimations as follows

$$\hat{u}^{k+1}(x) = \hat{w}^T_{d,\,k+1} \cdot \phi(x) \tag{6.17a}$$

$$\hat{u}^{k+1}_{on}(x) = \hat{w}^T_{r,\,k+1} \cdot \chi(x) \tag{6.17b}$$

$$\hat{\theta}^{k+1}(x) = \hat{w}^T_{p,\,k+1} \cdot \varphi(x) \tag{6.17c}$$

where $\hat{w}_{d,\,k+1} \in \mathbb{R}^{K_d \times D_1}$, $\hat{w}_{r,\,k+1} \in \mathbb{R}^{K_r \times D_2}$, and $\hat{w}_{p,\,k+1} \in \mathbb{R}^{K_p \times D_3}$ are the weight matrices of the proper dimension of the actor NNs to be learned.

Following Chen et al. (2022), the monotone polynomial function is employed as the activation function for value function approximation. The general least-square (GLS) method is used to update the critic NN weight vector. Define the set of sampled data for training the MACNN as $\{x_m, u_m, u_{on,\,m}, \theta_m\}$. $m = 1, 2, \ldots, b$ is the sample index, where $b > 0$ denotes the number of data samples used for one iteration. The approximated value function should satisfy

$$\hat{V}^k(x_m) = U(x_m, \hat{u}^k_m, \hat{u}^k_{on,m}, \hat{\theta}^k_m) + \hat{V}^k(f(x_m, \hat{u}^k_m, \hat{u}^k_{on,m}, \hat{\theta}^k_m)) \tag{6.18}$$

Then $\hat{w}_{c,\,k+1}$ is updated by

$$\hat{w}_{c,\,k+1} = \left[\Psi^T_k \cdot \Psi_k\right]^{-1} \cdot \Psi^T_k \cdot \Pi_k \tag{6.19}$$

where

$$\Psi_k = \begin{bmatrix} \psi(x_1) - \psi(f(x_1, u^k_1, u^k_{on,1}, \theta^k_1)) \\ \psi(x_2) - \psi(f(x_2, u^k_2, u^k_{on,2}, \theta^k_2)) \\ \cdots \\ \psi(x_b) - \psi(f(x_b, u^k_b, u^k_{on,b}, \theta^k_b)) \end{bmatrix}$$

$$\Pi_k = \left[U(x_1, \hat{u}^k_1, \hat{u}^k_{on,1}, \hat{\theta}^k_1), U(x_2, \hat{u}^k_2, \hat{u}^k_{on,2}, \hat{\theta}^k_2), \ldots, U(x_b, \hat{u}^k_b, \hat{u}^k_{on,b}, \hat{\theta}^k_b)\right]^T$$

The target control policies are obtained by

$$\{u^k(x_m), u^k_{on}(x_m), \theta^k(x_m)\} = \arg \min_{u_m, u_{on,m}, \theta_m} \{U(x_m, u_m, u_{on,m}, \theta_m) + \hat{V}^k(x_{m+1})\}$$

Hence, based on the GLS method, the weight vectors of the actor NNs are updated as

$$\hat{w}_{d,\,k+1} = \left[\Phi^T \cdot \Phi\right]^{-1} \cdot \Phi^T \cdot \Upsilon_{k+1} \tag{6.20a}$$

$$\hat{w}_{r,\,k+1} = \left[\Xi^T \cdot \Xi\right]^{-1} \cdot \Xi^T \cdot \Gamma_{k+1} \tag{6.20b}$$

$$\hat{w}_{p,\,k+1} = \left[\Theta^T \cdot \Theta\right]^{-1} \cdot \Theta^T \cdot \Lambda_{k+1} \tag{6.20c}$$

where $\Phi = [\phi(x_1), \phi(x_2), \ldots, \phi(x_b)]^T$, $\Xi = [\chi(x_1), \chi(x_2), \ldots, \chi(x_b)]^T$, $\Theta = [\varphi(x_1), \varphi(x_2), \ldots, \varphi(x_b)]^T$, $\Upsilon_{k+1} = [\hat{u}^{k+1}(x_1), \hat{u}^{k+1}(x_2), \ldots, \hat{u}^{k+1}(x_b)]^T$, $\Gamma_{k+1} = [\hat{u}_{on}^{k+1}(x_1), \hat{u}_{on}^{k+1}(x_2), \ldots, \hat{u}_{on}^{k+1}(x_b)]^T$, and $\Lambda_{k+1} = [\hat{\theta}^{k+1}(x_1), \hat{\theta}^{k+1}(x_2), \ldots, \hat{\theta}^{k+1}(x_b)]^T$.

We now investigate the convergence of the MACNN framework. The convergence of the value function $\hat{V}^k(x)$ given by (6.15) and policy functions $\hat{u}^{k+1}(x)$, $\hat{u}_{on}^{k+1}(x)$, $\hat{\theta}^{k+1}(x)$ given by (6.17) is summarized as follows.

Proposition 6.3.1 Suppose that the convergence of $\hat{w}_{c,\,k+1}, \hat{w}_{d,\,k+1}, \hat{w}_{r,\,k+1}, \hat{w}_{p,\,k+1}$ holds, for $\forall \xi > 0$, there exist integer $k^* > 0$, $K_c^* > 0$, $K_d^* > 0$, $K_r^* > 0$, and $K_p^* > 0$ such that if $k > k^*$, $K_c > K_c^*$, $K_d > K_d^*$, $K_r > K_r^*$, and $K_p > K_p^*$, then

1) $|\hat{V}^k(x) - V^k(x)| \leq \xi$,

2) $\|\hat{u}^{k+1}(x) - u^{k+1}(x)\| \leq \xi$, $\|\hat{u}_{on}^{k+1}(x) - u_{on}^{k+1}(x)\| \leq \xi$, $\|\hat{\theta}^{k+1}(x) - \theta^{k+1}(x)\| \leq \xi$,

3) $|\hat{V}^k(x) - V^*(x)| \leq \xi$, $\|\hat{u}^{k+1}(x) - u^*(x)\| \leq \xi$, $\|\hat{u}_{on}^{k+1}(x) - u_{on}^*(x)\| \leq \xi$, $\|\hat{\theta}^{k+1}(x) - \theta^*(x)\| \leq \xi$

hold for all $x \in \Omega$. ∎

Proof 6.3.1 Proposition 6.3.1 can be easily proved by extending Proof 5.3.1. ∎

Proposition 6.3.1 indicates that the proposed CADP approach implemented with the MACNN framework can approximate the optimal value function $V^*(x)$ and control policy $\{u^*(x), u_{on}^*(x), \theta^*(x)\}$, which aims to minimize the TTS of the whole mixed urban-freeway network.

Remark 6.3.2 It should be noted that the implementation of the critic and actor NNs requires that the vectors Ψ, Φ, Ξ, and Θ in (6.19) and (6.20) must satisfy $\text{rank}(\Psi) = K_c$, $\text{rank}(\Phi) = D_1 \cdot K_d$, $\text{rank}(\Xi) = D_2 \cdot K_r$, and $\text{rank}(\Theta) = D_3 \cdot K_p$.

These rank conditions are equivalent to the conditions for persistence of excitation. Under such circumstances, the GLS-based methods can be applied to approximate the iterative value functions and the iterative control policies. ∎

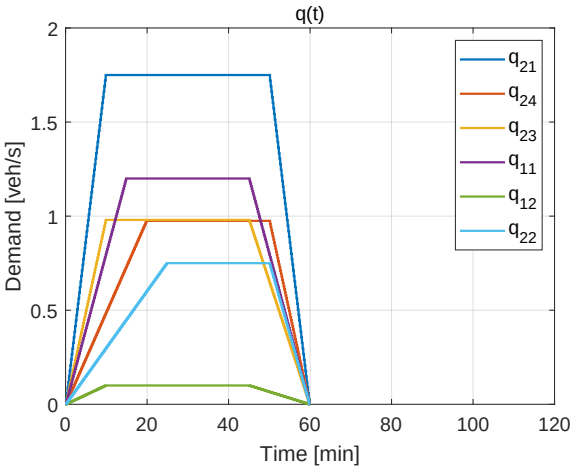
6.4 Numerical experiments

Simulation experiments are conducted on an urban network connected by a ring expressway, see Figure 6.1(a). The urban network is partitioned into five homogeneous regions with each region admitting a well-defined MFD of the form $G_i(n_i) = (1.4877 \times 10^{-7} \times n_i^3 - 2.9815 \times 10^{-3} \times n_i^2 + 15.0912 \times n_i)/3600$. The critical accumulation state is $n_{cr} = 3400$ veh and the jam state is $n_{jam} = 10000$ veh, as shown in Figure 6.2(b). The ring expressway (both inside and outside rings) is divided into 24 cells, and the length of each cell is 500 m. All the cells in the ACTM model share the same fundamental diagram (FD) as shown in Figure 6.2(c) (Muralidharan et al., 2009). The sampling time interval is $\Delta t = 10$ s.

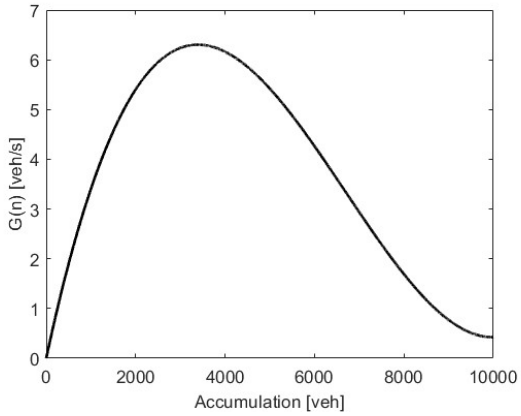
Figure 6.2(a) plots the O-D travel demand in a peak period with one peak hour followed by one off-peak hour for congestion dissolving. As the central business district, $R1$ attracts more trips than the periphery regions. While for periphery regions, take $R2$ as an example, most of the travel demand of $R2$ goes to the city center $R1$, followed by travel demand to its opposite region $R4$ and its neighbors $R3$ and $R5$. Demand patterns of $R3$, $R4$, and $R5$ follow a similar trend as $R2$. The demand is also associated with a 10% coefficient of variation to represent the underlying stochasticity.

The performance of the proposed CADP approach is compared with the PIAL strategy. The PIAL strategy is a decentralized strategy where 'PI' stands for Proportional-Integral perimeter control (Keyvan-Ekbatani et al., 2012), 'A' stands for ALINEA ramp metering (Ramezani et al., 2015), and 'L' stands for Logit-based route choice modeling (Ramezani et al., 2015).

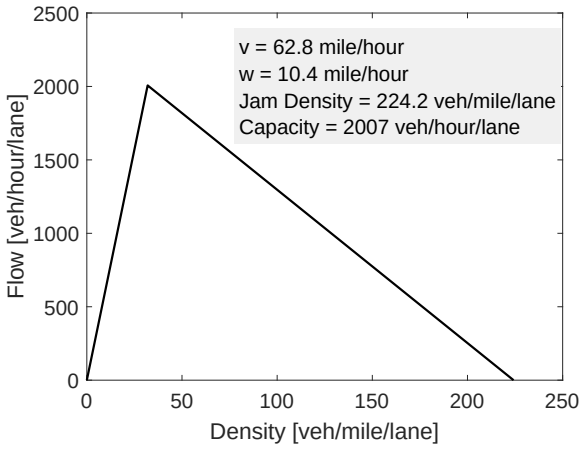
The CADP policy is trained in the environment of a fully-cooperative game. Figure 6.3 shows the training process of the CADP algorithm. The objective function converges after around 70 iterations, which indicates that the training algorithm performs well in solving the CCP. Table 6.2 summarizes the performance delivered by the two control strategies. The percentage numbers in columns two to four present the decreases in TTS achieved by the CADP approach compared with the baseline PIAL. Under the PIAL strategy, the total travel time (TTS) for the entire network is



(a)



(b)



(c)

Figure 6.2 Simulation environment: (a) Travel demand profile, (b) MFD function, and (c) FD for ACTM model.

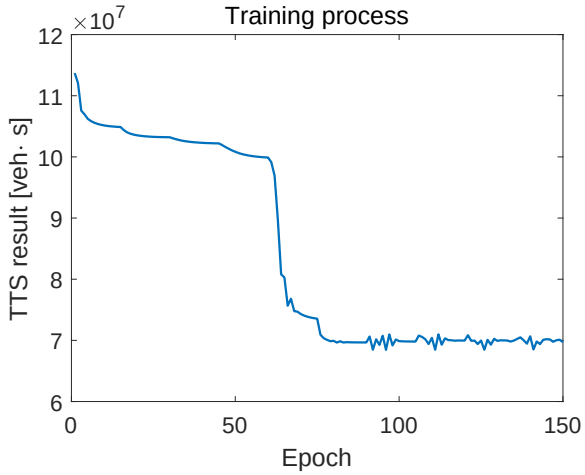


Figure 6.3 Train process of the CADP algorithm

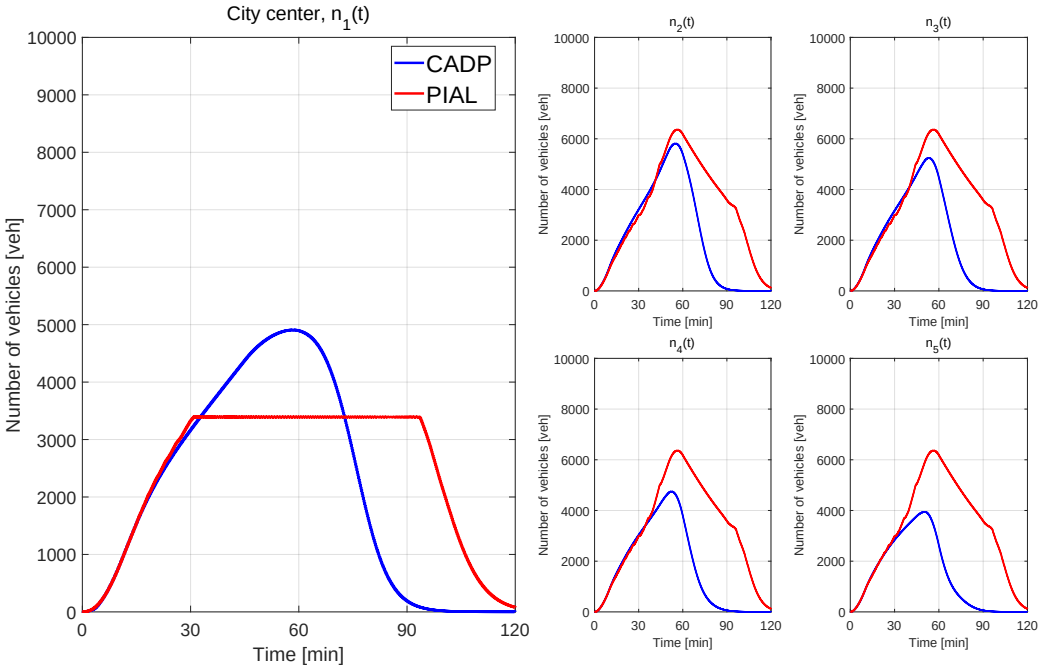
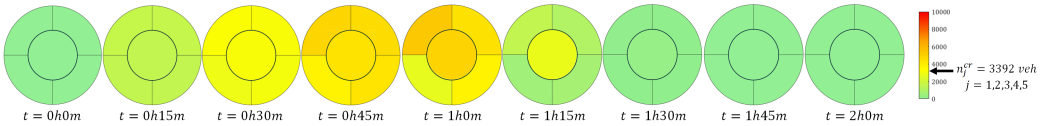
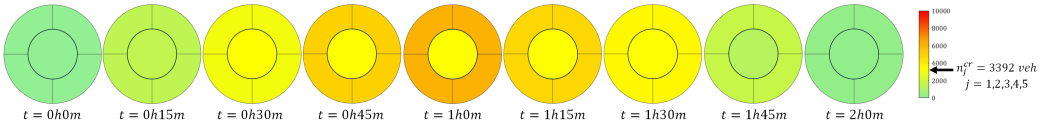


Figure 6.4 Urban accumulation state evolution

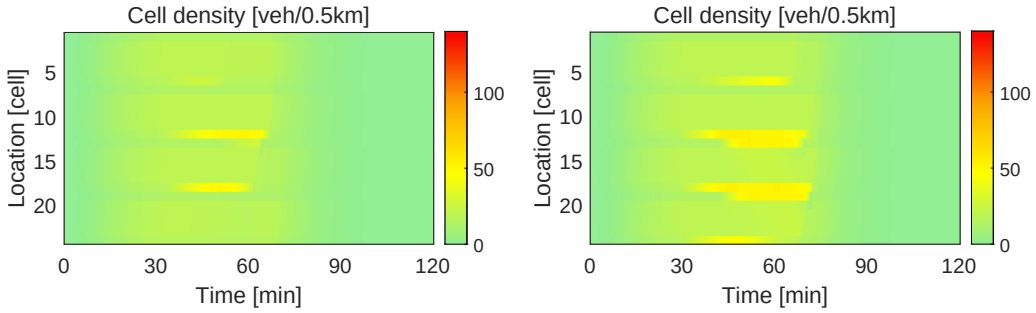


(a) CADP



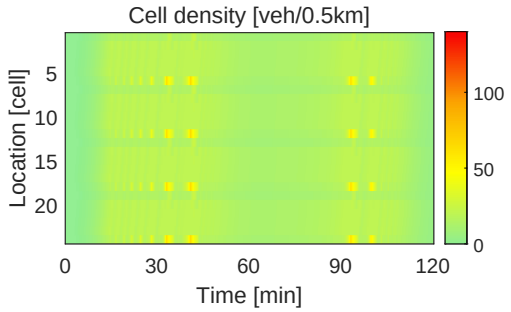
(b) PIAL

Figure 6.5 Snapshots of the urban accumulation state evolution

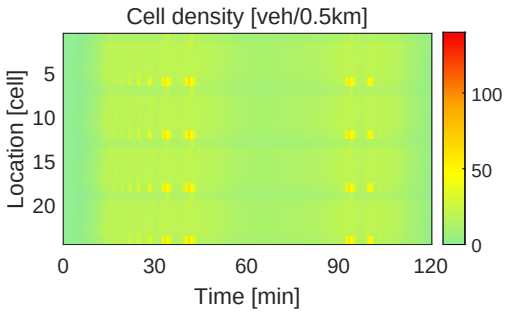


(a) CADP, Outside lanes

(b) CADP, Inside lanes



(c) PIAL, Outside lanes



(d) PIAL, Inside lanes

Figure 6.6 Ring expressway state evolution

$11.570 \times 10^7(veh \cdot s)$, with the urban network contributing to the majority of this value. The proposed CADP strategy reduces the TTS to $6.980 \times 10^7(veh \cdot s)$, a 39.7% improvement compared with the PIAL strategy.

Table 6.2 Performance comparison in TTS ($\times 10^7$ veh·s)

	Urban network	Ring expressway	Whole network
CADP	6.660 (-39.8%)	0.384 (-24.9%)	6.980 (-39.7%)
PIAL	11.059 (-)	0.511 (-)	11.570 (-)

Figure 6.4 shows the accumulation state evolution of the urban road network and Figure 6.5 presents some snapshots of the simulation process. After the peak hour, the travel demand decreases and the congestion dissolves. However, the congestion of the city center $R1$ and the periphery regions cannot be fully dissipated under PIAL. The central state even starts to recover nearly one hour after the peak. In contrast, the moderate congestion under CADP is soon dissipated to the initial empty state, for both $R1$ and the periphery regions. The PIAL results in much more severe traffic congestion in the periphery regions than the CADP. By the end of the simulation, congestion in periphery regions is dissipated under CADP but not under PIAL.

Figure 6.7 details the perimeter control evolution. At the first hour, full access is allowed for cross-boundary flows for most of the time under both CADP and PIAL control schemes. However, the PIAL strategy starts to limit the cross-boundary flow from the central to periphery regions but no limit for the transfer flow from outside to the central after the first hour. The proportional-integral perimeter controller attempts to maintain the accumulation state around the critical point of the MFD so that the maximal throughput can be achieved. This explains why using the PIAL policy, the traffic dissipation in the central area is slower than in the periphery regions. On the other hand, there are a few limitations for the cross-boundary flows under the CADP-based perimeter control scheme. This implies that the CADP policy can have a better balance of traffic loads for each urban region.

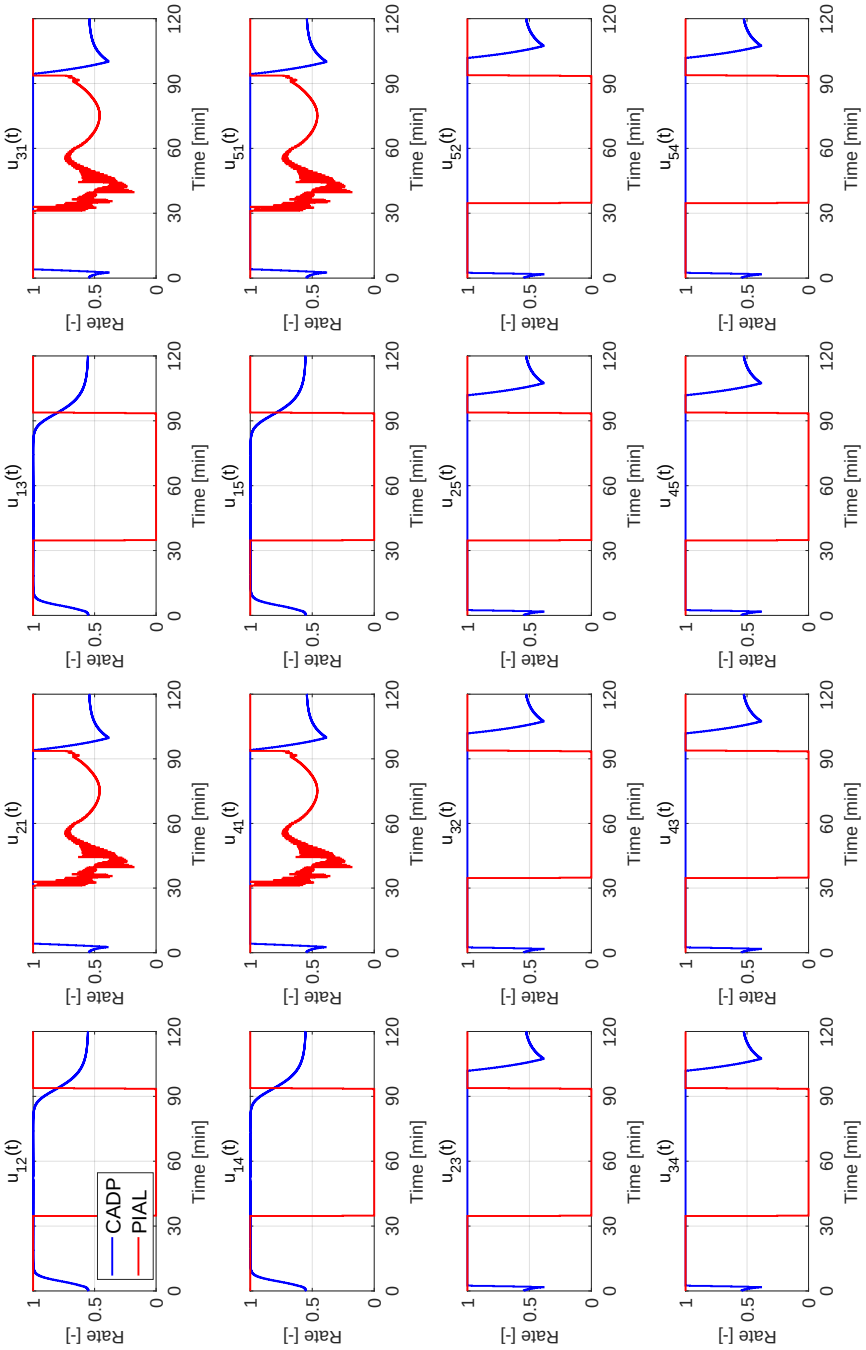


Figure 6.7 Perimeter control sequences

Figure 6.8 and Figure 6.9 depict the route guidance strategy under PIAL and CADP, respectively. The route guidance strategy is represented by the O-D demand split ratio using different paths. For O-D demand between neighboring regions, i.e., q_{23} and q_{25}, the PIAL policy reflects that drivers intend to use the arterial roads rather than the ring expressway. On the other hand, the CADP approach splits the flows more equally. This leaves space for the expressway to accommodate the demand from opposite regions. For O-D demand between opposite regions, i.e., q_{24} and q_{35}, PIAL outputs the even distribution strategy, see Figure 6.8. CADP intends to guide transfer flows to cross the periphery regions and take advantage of the ring expressway, see Figure 6.9. This keeps the cross-border traffic from going through the city center $R1$. In fact, this is the main purpose of building the ring expressway in reality. The ring expressway not only reduces the congestion in the city center but also saves travel time for travelers since they can travel at a higher speed. When the ring expressway is congested, using periphery regions as a detour is a good alternative. However, directing too much traffic to the ring expressway might lead to congestion in the expressway. As shown in Figure 6.6, both CADP and PIAL strategies can keep the ring expressway in a moderate congestion state.

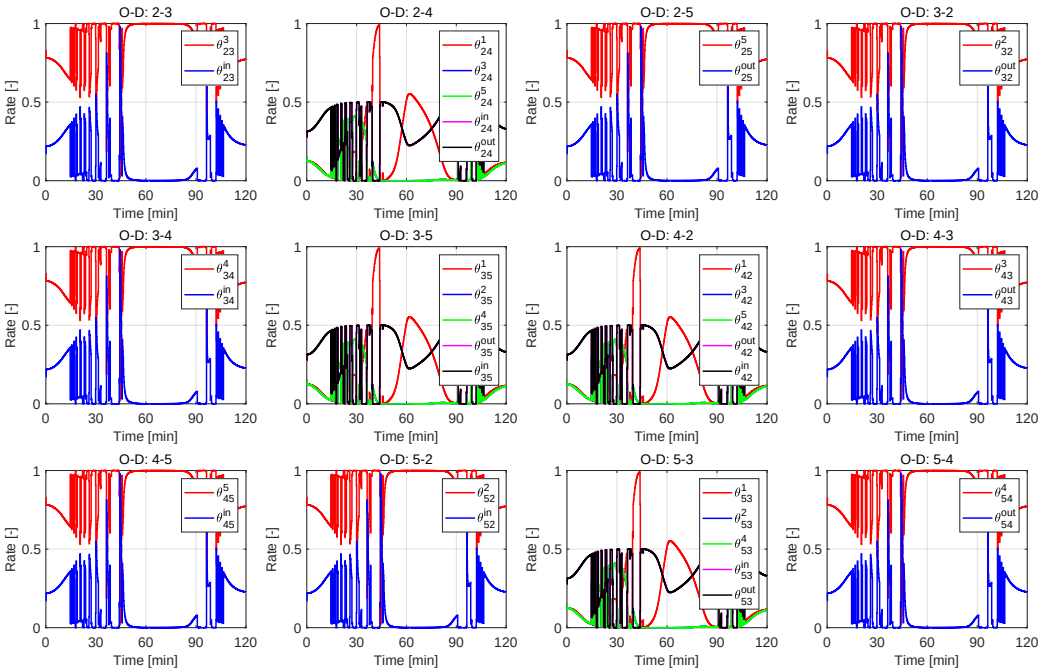


Figure 6.8 PIAL route guidance strategy

Figure 6.10 depicts the ramp metering control sequences. For the PIAL strategy, as an uncoordinated control, each control system aims to maximize its own benefit. PIAL prohibits travelers from entering the ring expressway from urban regions right after the ring expressway shows signs of congestion. There is no limitation on the

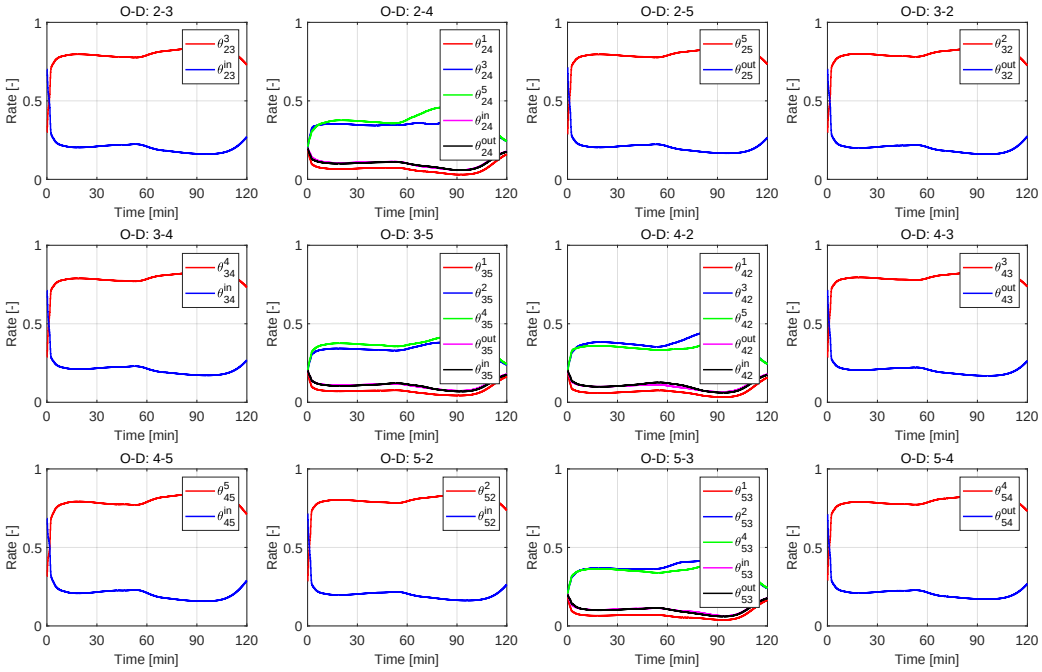


Figure 6.9 CADP route guidance strategy

off-ramp from the expressway to urban regions. Since the ring expressway is in a moderate congestion state, as a cooperative strategy, CADP can always guarantee the access of travelers to use the ring expressway. This achieves a full utilization of the capacity of the ring expressway, leading to a 24.9% reduction in the expressway total travel time compared with the PIAL. This demonstrates that a cooperative control strategy can achieve better performance not only for the whole network but also for the competing agents.

6.5 Conclusions

This study contributes to the field of traffic control in large-scale mixed urban-expressway networks by proposing a cooperative control model for a multi-region urban network with a ring expressway. The integration of perimeter control, route guidance, and ramp metering allows for improved network performance, reduced congestion, and minimized travel times.

A cooperative adaptive dynamic programming (CADP) approach was proposed to optimize the cooperative control of the mixed urban-expressway system, taking into account the interactions and dynamics between different agents. The numerical

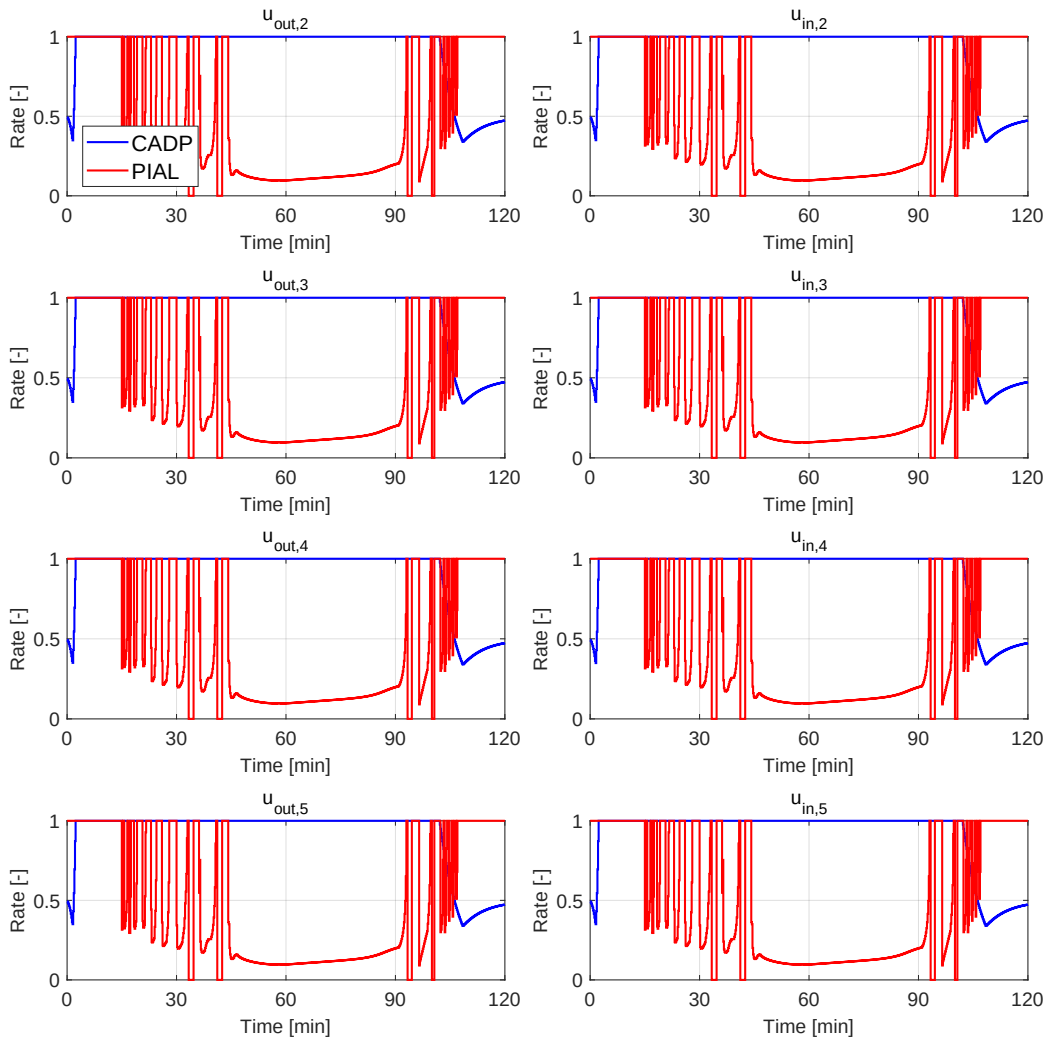


Figure 6.10 Ramp metering control sequences

studies demonstrated the effectiveness of the proposed cooperative control strategy. It was observed that the coordinated CADP approach led to a significant reduction in total travel time and better protection against over-saturation when compared to uncoordinated strategies. The cooperation among the different control mechanisms enabled a more efficient utilization of the network and expressway capacity and improved traffic flow.

The findings of this study highlight the potential benefits of employing cooperative control strategies in large-scale urban networks with expressways. It also explains the phenomenon in reality that cross-region travelers prefer expressways over local roads in the city center since they can travel faster. And when the expressway is congested, it is beneficial to use periphery regions as a detour. Future efforts will be

dedicated to extending the proposed CADP cooperative control strategy to a traffic environment mixed with human-driven vehicles and autonomous vehicles.

7 Summary of the thesis and future research topics

7.1 Summary of thesis

This dissertation was to study adaptive traffic control of large-scale heterogeneous urban networks based on the macroscopic fundamental diagram (MFD) framework. We thoroughly explored three key research themes: data-efficient "model-free" perimeter control (Chapter 3 and Chapter 4), adaptive perimeter control integrated with regional route guidance (Chapter 5), and cooperative control of mixed urban-expressway networks (Chapter 6). This section briefly summarizes the main contributions and findings of the thesis.

"Model-free" optimal perimeter control

Existing data-driven perimeter control strategies do not consider the effect of heterogeneous real-time data resolution. Besides, perfect information on the system dynamics is the prerequisite for traditional (model-based) optimal perimeter controllers, making them fragile to model calibration errors and external disturbances. To overcome these challenges, Chapter 3 proposed an integral reinforcement learning (IRL) method to learn the MFD system dynamics and devise an adaptive optimal perimeter controller. This study mainly contributes in the following aspects:

- A continuous-time control with time-varying reinforcement interval to adapt to the heterogeneous real-time resolution of data measurements
- An experience replay technique to reduce the sampling complexity and enhance the efficiency of available data
- A "model-free" integral reinforcement learning (IRL) method to relax the requirement of exact knowledge on system dynamics
- The Lyapunov theory to guarantee convergence of the IRL-based algorithms and the stability of the controlled traffic dynamics

The set-point control problem of perimeter-controlled MFD systems was investigated in Chapter 3. Note that both travel demand and supply generally vary with time. Traffic networks are usually subject to various uncertainties such as model errors and demand measurement noises. Thus, defining a proper set point as the control target might not be a trivial task. Considering the time-varying nature of the travel demand pattern and supply function, Chapter 4 proposed a novel trajectory stability concept in the MFD framework. This study mainly contributes in the following aspects:

- Reformulation of the conventional set-point perimeter control problem into an optimal tracking perimeter control problem
- Trajectory stability under the proposed tracking perimeter control guaranteed by Lyapunov theory
- Improvement in reducing total travel time and enhancement in cumulative trip completion by applying the tracking perimeter control

In Chapter 3 and Chapter 4, approximate optimization methods were carried out to address the curse of dimensionality of the optimal control problem. The optimal perimeter controller was parameterized and then approximated by neural networks (NN), which moderates the computational complexity. Both state and input constraints are considered while no model linearization is required. The major finding of these two studies was an easy-to-check rank condition used to verify the data richness, i.e., whether the sampled data are sufficient to ensure that the NNs are well-trained. Combined with the Lyapunov theory, this finding revealed that if the rank condition was satisfied, the approximated perimeter controller using the NNs could stabilize the accumulation state at the desired equilibrium and achieve satisfactory performance in minimizing the total network delay.

Iterative adaptive perimeter control and regional route guidance

Coupling perimeter control and regional route guidance (PCRG) is a promising strategy to decrease congestion heterogeneity and reduce delays in large-scale MFD-based urban networks. With the increase in urban region size, previous studies found that one needs to distinguish the model used for optimization and the plant that replicates the real traffic system. The differences in traffic network structures and input data between the model and the plant are known as the so-called model-plant mismatch. The heterogeneous congestion distribution and uncertain MFD parameters make the plant dynamics unavailable for optimal control design. Existing data-driven methods (e.g., reinforcement learning) do not consider the model-plant

mismatch and the limited access to plant-generated data, e.g., subregional OD-specific accumulations. To fill the research gap, Chapter 5 developed an iterative adaptive dynamic programming (IADP) based method to address the limited data source induced by the model-plant mismatch and approximate the optimal PCRG strategy without knowing the system dynamics. This study mainly contributes in the following aspects:

- An actor-critic neural network structure developed to circumvent the requirement of complete information on plant dynamics
- Efficiency in data use despite limited access to available plant-generated data
- Robustness against various uncertainties (demand noise, MFD error, trip distance heterogeneity) when minimizing the total time spent in the urban network
- Outperforming the "benchmark" model predictive control (MPC) approach in improving network mobility and computational efficiency

Performance comparisons with other PCRG schemes under various scenarios indicated that the IADP controller trained with a limited data source achieved comparable performance with the MPC approach using perfect measurements from the plant. When more detailed plant data were available, the IADP approach could even outperform the MPC controller. This is the major finding of the study, which demonstrates the great potential of the proposed scheme in improving the efficiency of multi-region MFD systems.

Cooperative control of mixed urban-expressway networks

Traffic control and management of large-scale urban networks involves not only the regulation of traffic flows on arterial roads, but also those on highways/expressways that connect different parts of the city. Ring expressways built in many megacities (e.g., Beijing) are playing an important role in the traffic management of urban networks. With on- and off-ramps to connect the city's periphery areas, ramp metering is usually desired to protect the expressways from over-congestion. Few studies have explored the cooperation of perimeter control, route guidance, and ramp metering strategies in improving the whole network mobility. To fill this gap, Chapter 6 proposed a cooperative adaptive dynamic programming (CADP) approach to solve the cooperative control problem for a mixed urban-expressway network. This study mainly features the following aspects:

- A mixed traffic network composed of a multi-region urban network modeled by the MFD and a ring expressway going through the periphery regions modeled by the asymmetric cell transmission model (ACTM)
- A multi-agent actor-critic neural network (MACNN) framework to train the agents of perimeter control, route guidance, and ramp metering to fully cooperate
- Achievement in both central urban congestion alleviation and whole network mobility improvement

Numerical studies demonstrated that the CADP could reduce the total travel delay by 48.1% compared with the model-based decentralized strategies and by 39.0% compared with the D-ADP strategy. In addition, the city center was well protected from over-congestion by applying the CADP approach. This finding of the study sheds light on the potential benefits of employing cooperative control strategies in large-scale urban networks with expressways. It also explains why travelers driving from one region to another prioritize expressways with longer distances over shorter arterial roads in the city center, i.e., they can travel faster and do not have to suffer from traffic congestion. When the expressway is congested, it might be beneficial to use arterial roads in periphery regions as a detour.

7.2 Future works

Based on the findings of this dissertation, this section elaborates on the potential field applications and outlines the directions for future research. Here are some research topics that are worth future efforts:

1. Learning the traffic dynamics in a controlled environment:

   Conventional machine learning methods for learning traffic dynamics are regression-oriented and prioritize fitting input-output data via supervised learning with powerful function approximators. The dynamics are learned using the data collected before the controller is devised and deployed. However, the deployment of controllers may change the characteristics of the traffic system conversely. For instance, a new traffic signal control (actuating the perimeter control) scheme can not only change the network capacity (the MFD shape) but also induce a demand pattern variation. Learning the traffic dynamics in a controlled urban network deserves further exploration.

2. Event-triggered control of large-scale urban networks:

   An urban traffic network in a megacity can be regarded as a system of systems (SoS). In such an SoS system, various traffic control and management systems are deployed, e.g., perimeter control, regional route guidance, ramp metering, and variable speed limit. In operating this SoS system, coordinating all these control systems to work as a team to improve the network performance requires considerable computational effort in addressing the communications between these systems and calculating their optimal outputs. Event-triggered control is a promising way to resolve this difficulty. Event-triggered control is reactive and generates sensor sampling and control actuation when a triggering condition is violated, e.g., the plant state deviates more than a certain threshold from a desired value. Doing so can significantly reduce the computational burden without degrading the control performance. How to devise an efficient event-triggered control mechanism for large-scale urban networks is an unanswered question.

3. Fault tolerance in MFD-based regional route guidance systems:

   Most existing works on MFD-based regional route guidance assume that travelers will always obey the guidance of the optimal strategy output by the system. However, such an assumption usually cannot be met in the real world. Besides, it is very difficult to estimate the driver compliance rate in real time when the route guidance system is in operation. That is, a failure of the devised optimal strategy to reach its 100% control performance is inevitable. Therefore, it is necessary to improve the ability of a controlled MFD system to maintain control objectives in spite of the occurrence of a fault. This appeals to introducing a fault-tolerant mechanism in the regional route guidance system of the MFD framework. Note that the "fault-tolerant control" is different from the "robust control". Fault tolerance can be obtained through fault accommodation, which requires changes in controller parameters and even structure to avoid the consequences of a fault.

Other potential applications of MFDs in traffic control and management, including but not limited to ride-sourcing (Ramezani and Nourinejad, 2018; Nourinejad and Ramezani, 2020; Beojone and Geroliminis, 2023; Huang et al., 2023), static demand management integrated with dynamic supply control (Yildirimoglu and Ramezani, 2020; Kumarage et al., 2021), urban air mobility operation (Haddad et al., 2021; Safadi et al., 2023b; Safadi et al., 2023a), are yet to be further explored.

A.1 Flow conservation of the two- and three-region MFD systems

This appendix presents the conservation equations and dynamics in the affine form of two cases investigated in the literature, i.e., the two-region and the three-region MFD systems, as shown in Figure 3.4(a) and Figure 3.4(b), respectively. To begin with, let $M_{ii}(t) = \frac{n_{ii}(t)}{n_i(t)} G_i(n_i(t))$ and $M_{ij}(t) = \frac{n_{ij}(t)}{n_i(t)} G_i(n_i(t))$ denote the within-region flow and cross-boundary flow at time t, respectively.

Case 1: The two-region MFD system

Let $L = 2$ (i.e., the two-region MFD dynamics as shown by Figure 3.4(a) defined in Geroliminis et al., 2013), $n = [n_{11}, n_{12}, n_{21}, n_{22}]^T \in \mathbb{R}^4$ and $u = [u_{12}, u_{21}]^T \in \mathbb{R}^2$. The flow conservation equations are given as

$$\begin{aligned}
\frac{\mathrm{d}n_{11}(t)}{\mathrm{d}t} &= -M_{11}(t) + M_{21}(t)u_{21}(t) + q_{11}(t) \\
\frac{\mathrm{d}n_{12}(t)}{\mathrm{d}t} &= -M_{12}(t)u_{12}(t) + q_{12}(t) \\
\frac{\mathrm{d}n_{21}(t)}{\mathrm{d}t} &= -M_{21}(t)u_{21}(t) + q_{21}(t) \\
\frac{\mathrm{d}n_{22}(t)}{\mathrm{d}t} &= -M_{22}(t) + M_{12}(t)u_{12}(t) + q_{22}(t)
\end{aligned}$$

For this case, the new state and control are $\tilde{n} = [\tilde{n}_1, \tilde{n}_2, \tilde{n}_3, \tilde{n}_4]^T \in \mathbb{R}^4$ and $\tilde{u} = [\tilde{u}_1, \tilde{u}_2]^T \in \mathbb{R}^2$, respectively. The drift dynamics $\mathbf{F} \in \mathbb{R}^4$ and input dynamics $\mathbf{S} \in \mathbb{R}^{4\times 2}$ of their affine-form traffic dynamics are

$$\mathbf{F}(\tilde{n}) \triangleq \begin{bmatrix} -M_{11} + M_{21}u_{21}^* + q_{11} \\ -M_{12}u_{12}^* + q_{12} \\ -M_{21}u_{21}^* + q_{21} \\ -M_{22} + M_{12}u_{12}^* + q_{22} \end{bmatrix}, \ \mathbf{S}(\tilde{n}) \triangleq \begin{bmatrix} 0 & M_{21} \\ -M_{12} & 0 \\ 0 & -M_{21} \\ M_{12} & 0 \end{bmatrix}$$

Case 2: The three-region MFD system

Let $L = 3$ (i.e., three-region MFD dynamics as shown by Figure 3.4(b), see example in Zhong et al., 2018b), $n = [n_{11}, n_{12}, n_{21}, n_{22}, n_{23}, n_{32}, n_{33}]^T \in \mathbb{R}^7$ and $u = [u_{12}, u_{21}, u_{23}, u_{32}]^T \in \mathbb{R}^4$. The flow conservation equations are given as

$$\begin{aligned}
\frac{\mathrm{d}n_{11}(t)}{\mathrm{d}t} &= -M_{11}(t) + M_{21}(t)u_{21}(t) + q_{11}(t) \\
\frac{\mathrm{d}n_{12}(t)}{\mathrm{d}t} &= -M_{12}(t)u_{12}(t) + q_{12}(t) \\
\frac{\mathrm{d}n_{21}(t)}{\mathrm{d}t} &= -M_{21}(t)u_{21}(t) + q_{21}(t) \\
\frac{\mathrm{d}n_{22}(t)}{\mathrm{d}t} &= -M_{22}(t) + M_{12}(t)u_{12}(t) + M_{32}(t)u_{32}(t) + q_{22}(t) \\
\frac{\mathrm{d}n_{23}(t)}{\mathrm{d}t} &= -M_{23}(t)u_{23}(t) + q_{23}(t) \\
\frac{\mathrm{d}n_{32}(t)}{\mathrm{d}t} &= -M_{32}(t)u_{32}(t) + q_{32}(t) \\
\frac{\mathrm{d}n_{33}(t)}{\mathrm{d}t} &= -M_{33}(t) + M_{23}(t)u_{23}(t) + q_{33}(t)
\end{aligned}$$

For this case, $\tilde{n} = [\tilde{n}_1, \ldots, \tilde{n}_7]^T \in \mathbb{R}^7$ and $\tilde{u} = [\tilde{u}_1, \ldots, \tilde{u}_4]^T \in \mathbb{R}^4$. $\mathbf{F} \in \mathbb{R}^7$ and $\mathbf{S} \in \mathbb{R}^{7\times 4}$ of their affine-form traffic dynamics are

$$\mathbf{F}(\tilde{n}) \triangleq \begin{bmatrix} -M_{11} + M_{21}u_{21}^* + q_{11} \\ -M_{12}u_{12}^* + q_{12} \\ -M_{21}u_{21}^* + q_{21} \\ -M_{22} + M_{12}u_{12}^* + M_{32}u_{32}^* + q_{22} \\ -M_{23}u_{23}^* + q_{23} \\ -M_{32}u_{32}^* + q_{32} \\ -M_{33} + M_{23}u_{23}^* + q_{33} \end{bmatrix}, \mathbf{S}(\tilde{n}) \triangleq \begin{bmatrix} 0 & M_{21} & 0 & 0 \\ -M_{12} & 0 & 0 & 0 \\ 0 & -M_{21} & 0 & 0 \\ M_{12} & 0 & 0 & M_{32} \\ 0 & 0 & -M_{23} & 0 \\ 0 & 0 & 0 & -M_{32} \\ 0 & 0 & M_{23} & 0 \end{bmatrix}$$

A.2 A four-region case with no model-plant mismatch in Study 3

This case considers a network of four regions, see Figure A2.1, wherein the model and the plant share the identical model structure. The proposed IADP approach is first compared to several existing schemes in a case of no model error and trip

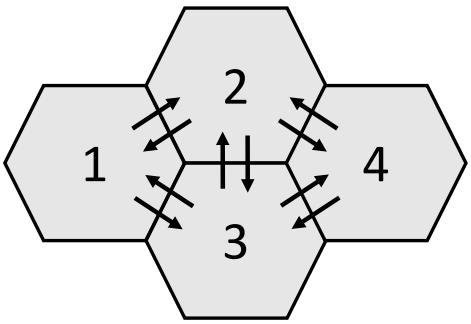


Figure A2.1 The four-region network

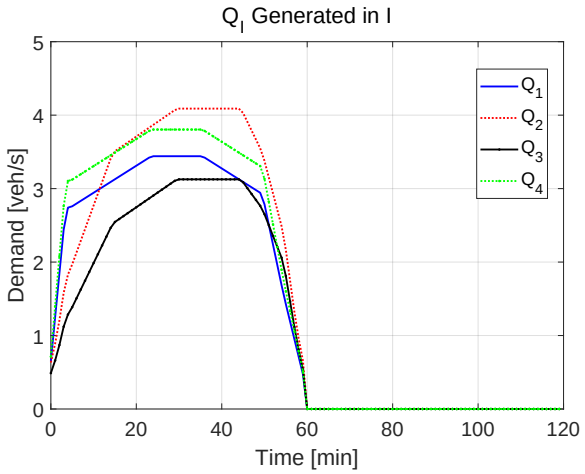
distance heterogeneity. Apart from the methods mentioned in Section 4, the online learning-based IRL approach proposed by Chen et al. (2022) is investigated. The IRL approach is also model-free. Different from the IADP scenario, $\{n_{ij}(t), u_{ij}(t), \theta^h_{ij}(t)\}$ are available for training the IRL algorithm. Besides, the IRL approach is trained in an online manner, i.e., the parameters of the implemented controller (i.e., the actor NN weights) are updated at any time step with new data fed from the environment during the simulation.

In this numerical example, all regions share the same MFD given by $G_I(N_I(t)) \triangleq P_I(N_I(t))/L_I(t) = (1.4877^{-7}N_I^3(t) - 2.9815^{-3}N_I^2(t) + 15.0912N_I(t))/3600$ (veh/s) where $L_I = 3600$ (m), $I = 1, \ldots, 4$. Figure A2.2(a) and Figure A2.2(b) depict the regional and OD-specific travel demand, respectively. Performance comparison is made among IADP, MPC-PM, IRL, and PIL.

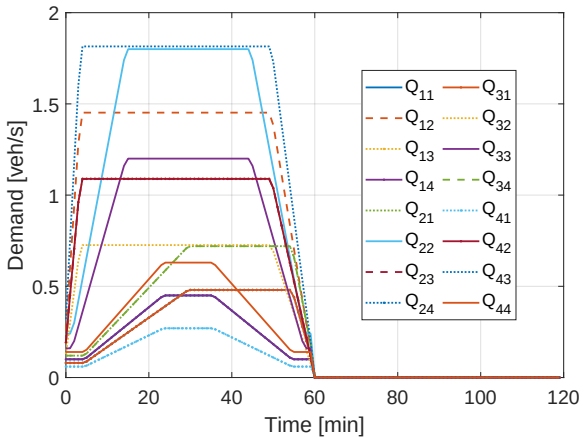
Figure A2.3(a)-Figure A2.3(d) depict the evolution of accumulation states N_I. The accumulation states under IADP and MPC-PM follow a similar trend and manage to dissolve the congestion in all regions. This indicates that the well-trained IADP can achieve comparable performance with the MPC-PM approach. Specifically, vehicles traveling from Region 1 to Region 4 (or vice versa) have to pass through either Region 2 or Region 3. Therefore, Region 2 and Region 3 experience more severe congestion while IADP and MPC-PM can protect Region 1 and Region 4 from over-saturation. In contrast, the PIL and IRL fail to clear the network by the end of the simulation.

Table A2.1 Performance comparison among various PCRG schemes of the four-region case

	IADP	MPC-PM	IRL	PIL
TTS ($\times$ 1e7 veh·s)	3.861 (-54.2%)	3.823 (-54.6%)	6.259 (-25.7%)	8.424 (-)
Avg. CPU time/step (s)	5.188e-4	9.471e-1	3.276	5.655e-3



(a) Demand generated in Region I



(b) OD-specific demand pattern

Figure A2.2 Demand profile for the four-region case.

Table A2.1 summarizes the performance comparison among various PCRG schemes of the four-region case. The MPC-PM approach significantly improves the TTS minimization over the baseline PIL strategy and exhibits the best performance when there are not any uncertainties. Despite not having complete knowledge of the system dynamics, IADP demonstrates comparable performance to the MPC-PM. In contrast to the online learning-based IRL, the off-line iterative learning-based IADP is capable of effectively learning network dynamics using extensive collected data before implementing the control law to regulate network traffic. As a result, the IADP can well operate the network transportation under recurrent traffic conditions once it is trained with sufficient historical data. In addition, the IADP achieves this performance at less computational time than the MPC approach for implementation.

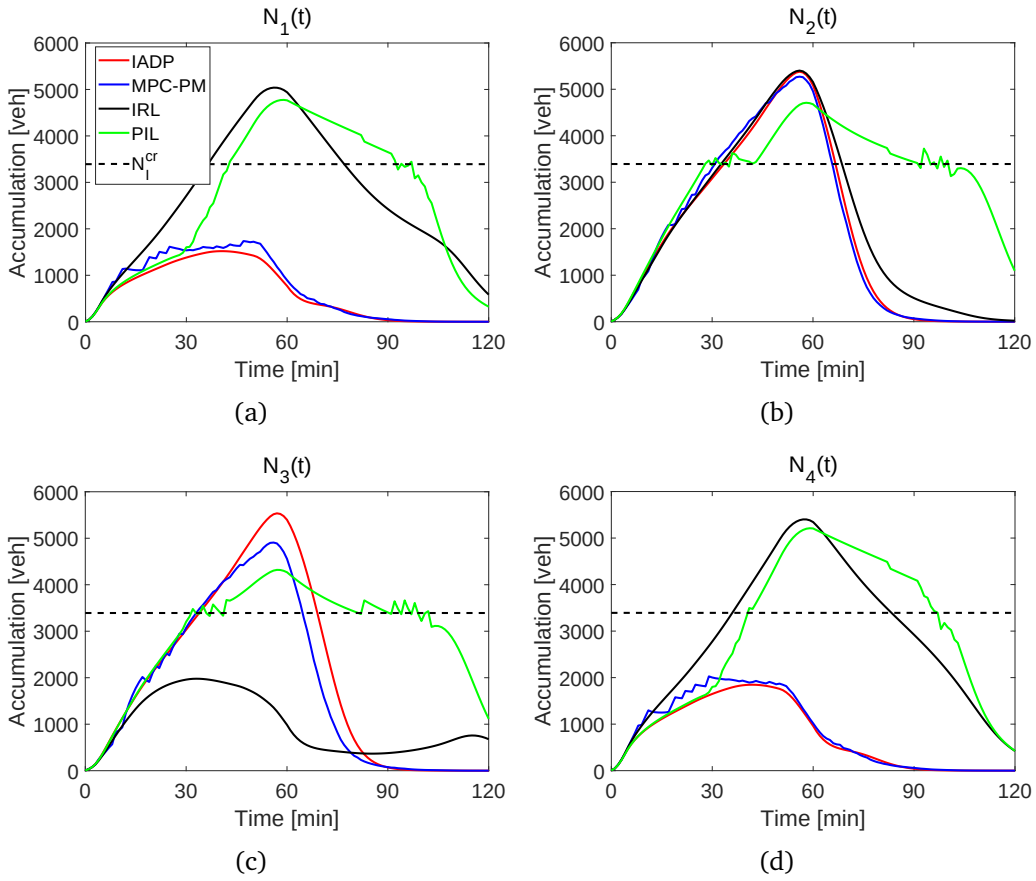


Figure A2.3 Accumulation state evolution of the four-region case. (a) N_1, (b) N_2, (c) N_3, and (d) N_4.

Figure A2.4(a), Figure A2.4(b) and Figure A2.4(c) show the perimeter control inputs U_{IJ} over time devised by IADP, MPC-PM and IRL, respectively. By comparing Figure A2.4(a) and Figure A2.4(b), the IADP and MPC-PM strategies have a similar control pattern despite that the IADP depicts a smoother control. Under IRL, Figure A2.4(c) restricts the cross-boundary flow from Region 1 and Region 4 to Region 3, protecting Region 3 from saturation, however, at the price of the highest congestion in Region 1 and 4.

Figure A2.5(a), Figure A2.5(b) and Figure A2.5(c) present the evolutions of route guidance schemes devised by IADP, MPC-PM and IRL, respectively. Generally, more travelers in adjacent regions take direct paths between the neighboring areas instead of detouring through third-party regions under all schemes. For non-adjacent OD demand q_{14}, q_{41}, more travelers are suggested to pass through Region 3 instead of Region 2. This is because Region 2 attracts more travel demand than Region 3. This helps to balance the traffic load in the network. Only when the perimeter control

between Region 4 and Region 2 is activated and to protect Region 3 under IRL, travelers change their route preference.

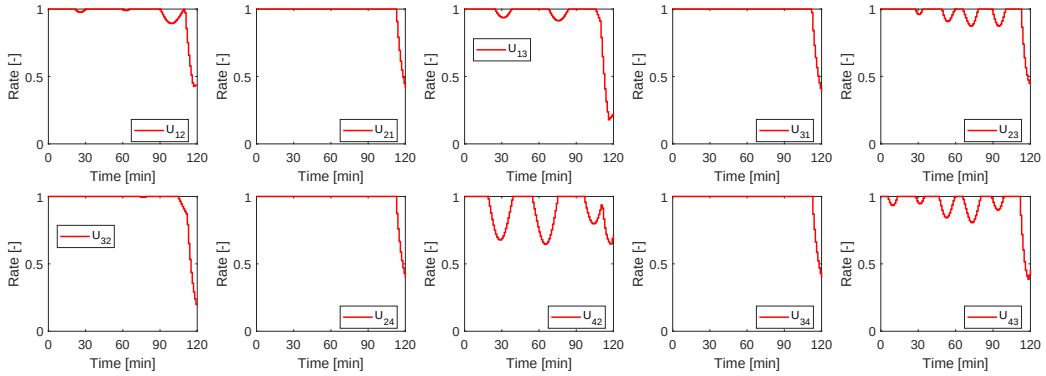
A.3 Supplementary results of Case 1-Example 1 in Study 3

A.3.1 Accumulation state and PCRG evolution of Case 1-Example 1

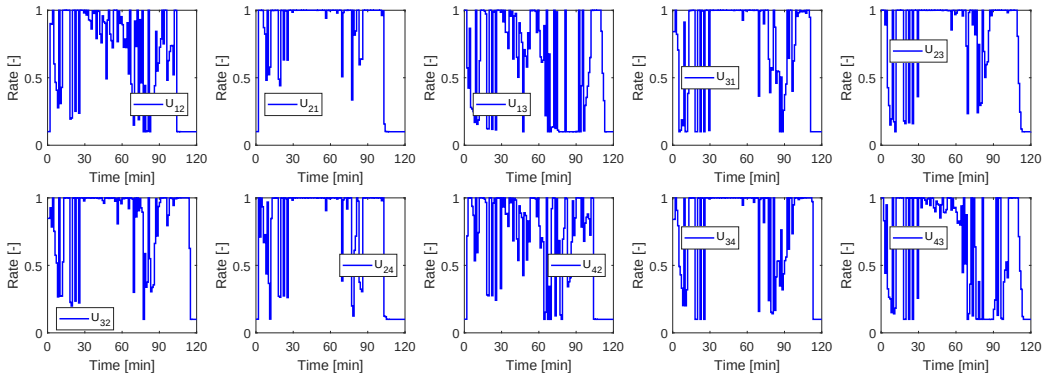
Figure A3.6 and Figure A3.7 show the subregional and regional accumulation state evolution of Case 1-Example 1, respectively. As observed, the PIL scheme maintains Region 1's accumulation state close to the critical point during the stationary congestion period (during the 40th-70th min). The MPC-UKF and MPC-PM schemes both achieve a lower average congestion level in both regions than the PIL. The control performance of the MPC-PM is slightly better than the MPC-UKF because the former uses the exact measurements from the plant. It is interesting to note that the IADP-PT scheme significantly decreases the average congestion level in Region 1 compared to the two MPC schemes. In contrast, the IADP-PT scheme results in a higher congestion level in Region 2 than the MPC schemes. The proposed IADP approach achieves the lowest average congestion level in the city center at the cost of the highest congestion level in the periphery.

Figure A3.8(a), Figure A3.9(a), Figure A3.10(a), Figure A3.11(a), and Figure A3.12(a) show the perimeter control inputs u_{ij} over time devised by IADP, IADP-PT, MPC-PM, MPC-UKF and PIL, respectively. Nearly no restrictions are enforced for cross-boundary flows by IADP, especially for the flows from the city center to the periphery, see Figure A3.8(a). When the accumulation state of Region 1 exceeds the critical point during the 40th-70th min, the PIL restricts the inflow for the benefit of individual regions, which obstructs the cross-boundary flows, see Figure A3.12(a). MPC-PM and MPC-UKF only occasionally restrict the flows from the central to the periphery to balance the traffic load, see Figure A3.10(a) and Figure A3.11(a).

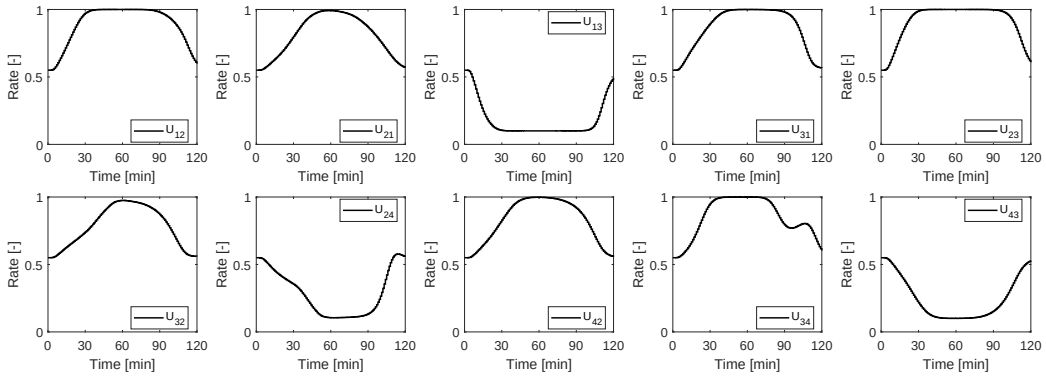
Figure A3.8(b), Figure A3.9(b), Figure A3.10(b), Figure A3.11(b), and Figure A3.12(b) present the evolutions of route guidance schemes devised by IADP, IADP-PT, MPC-PM, MPC-UKF and PIL, respectively. Generally, more travelers in adjacent regions take direct paths between the neighboring areas instead of detouring through



(a) U_{IJ}^{iadp}

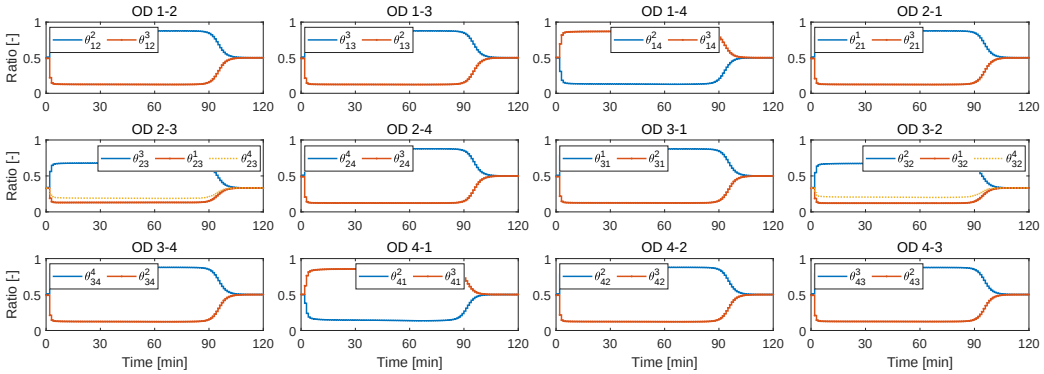


(b) U_{IJ}^{mpc}

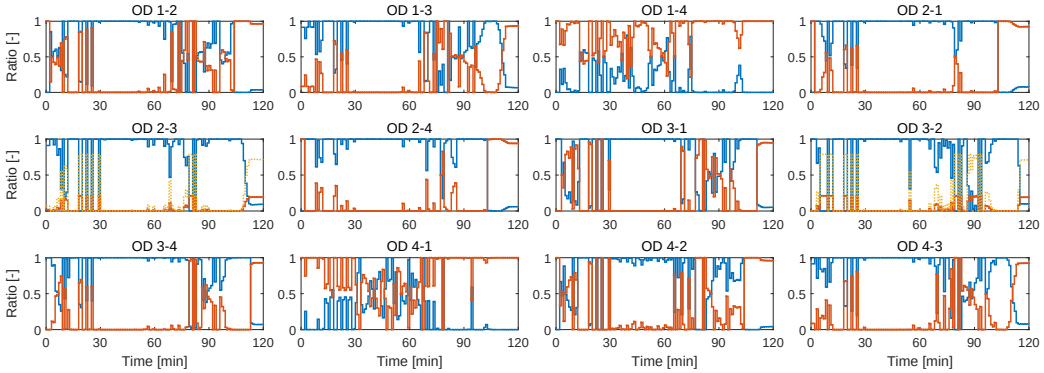


(c) U_{IJ}^{irl}

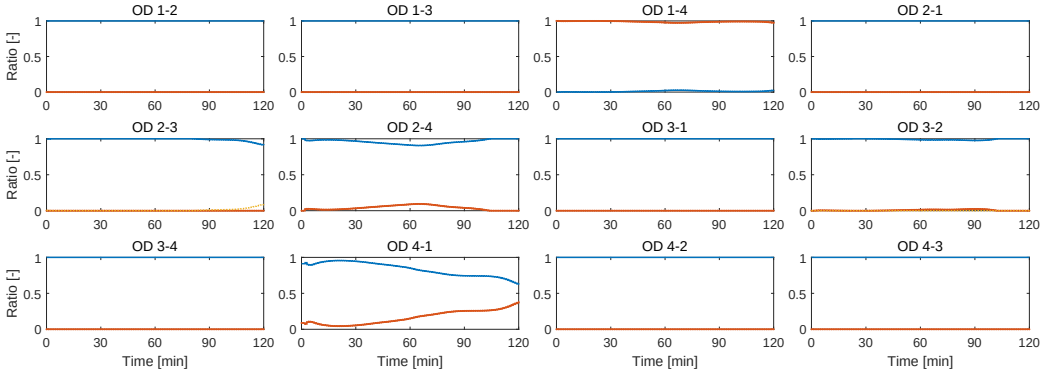
Figure A2.4 Perimeter control inputs of the four-region case. (a) IADP, (b) MPC-PM, and (c) IRL.



(a) Θ_{iadp}



(b) Θ_{mpc}



(c) Θ_{irl}

Figure A2.5 Route guidance input evolution of the four-region case. (a) IADP, (b) MPC-PM, and (c) IRL.

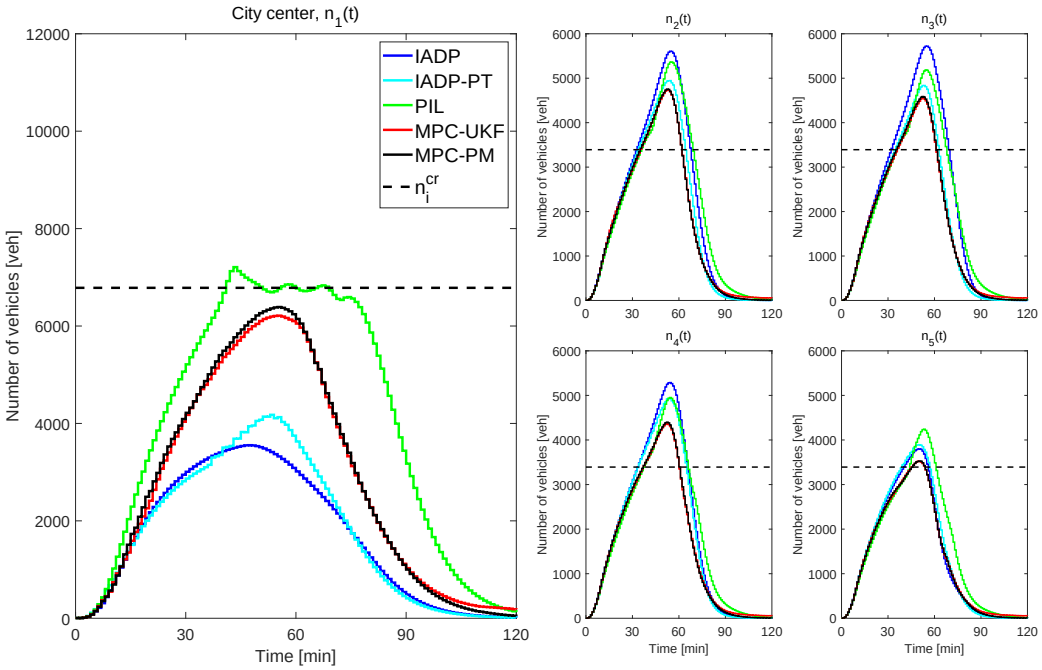


Figure A3.6 Subregional accumulation evolution of Case 1-Example 1.

third-party regions under all schemes. Nevertheless, PIL has a higher probability of detouring via the central region compared with the other schemes. For non-adjacent OD demand q_{24}, q_{35}, travelers can choose to circle the periphery region or pass through the central region. More travelers are suggested to use the periphery under IADP, see Figure A3.8(b) and to go through the city center under MPC-PM and MPC-UKF, see Figure A3.10(b) and Figure A3.11(b). This explains why IADP achieves the lowest accumulation state in the central region, see Figure A3.6.

A.3.2 MPC controller tuning

Figure A3.13 shows the control performances and computational efforts of MPC-PM controllers with different prediction horizons. Figure A3.13(c) presents the TTS performance and the average CPU times. The results indicate that: (a) TTS performance would be fairly insensitive to the choice of the prediction horizon when it is greater than 15 min, and (b) average CPU times increase with the increase in prediction horizons. Therefore, considering the tradeoff between control performance and computational effort, we set the prediction horizon to 15 min for the MPC-based controllers in our case studies.

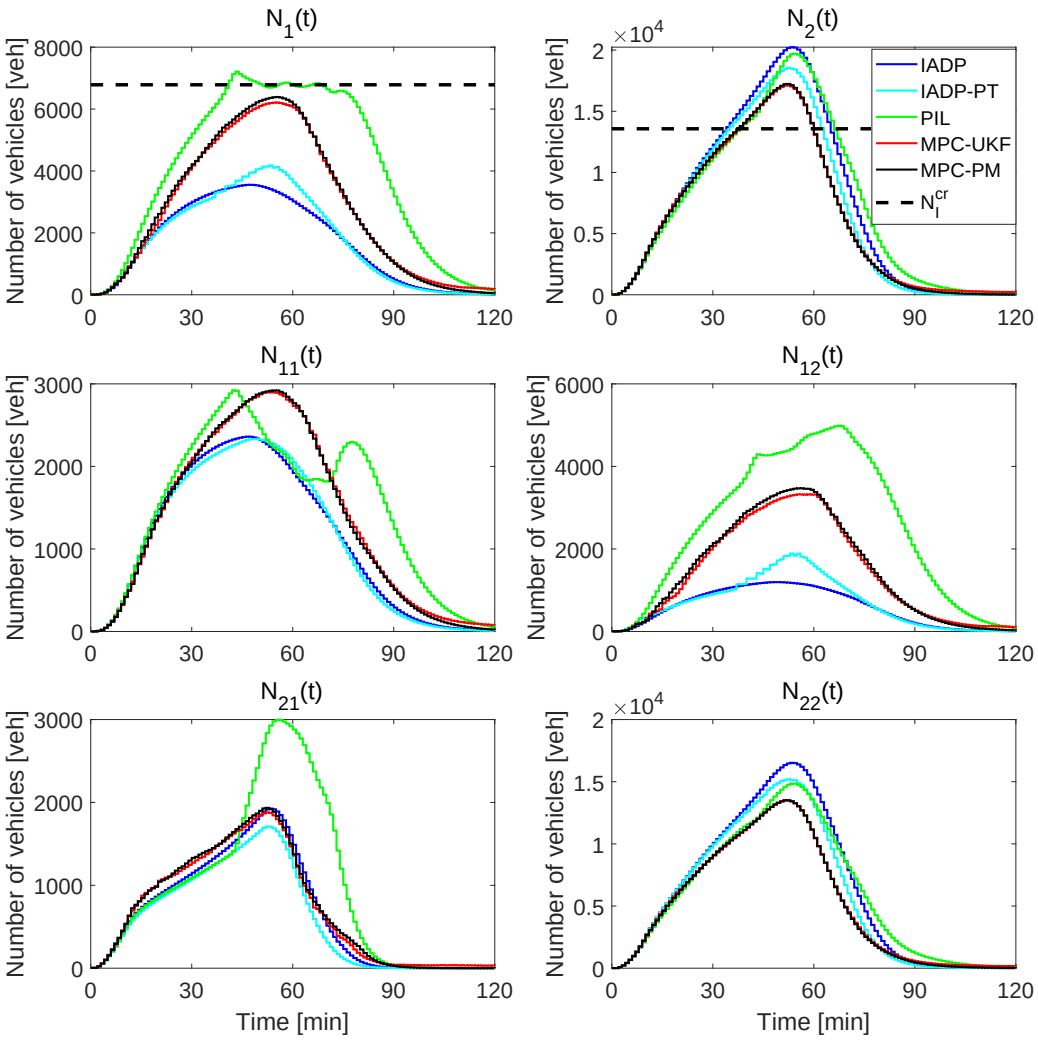
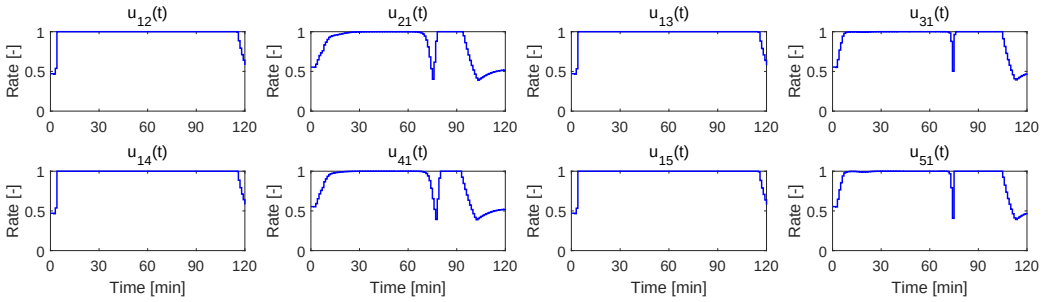
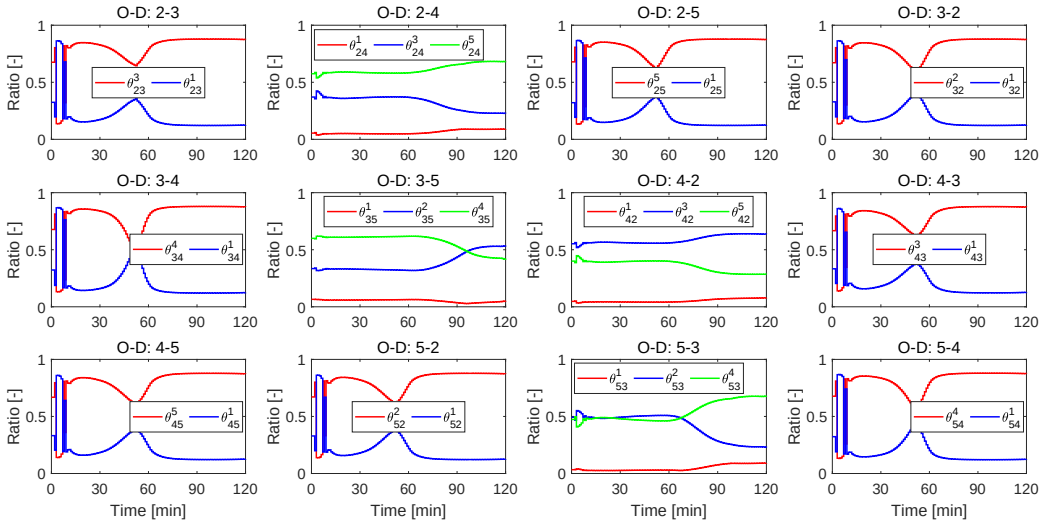


Figure A3.7 Regional accumulation evolution of Case 1-Example 1.

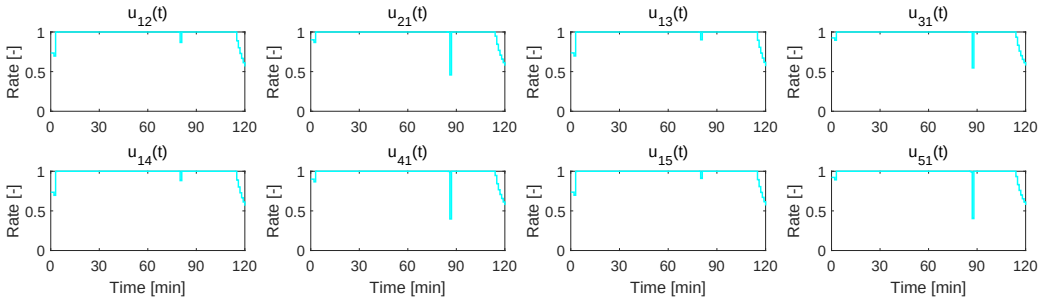


(a)

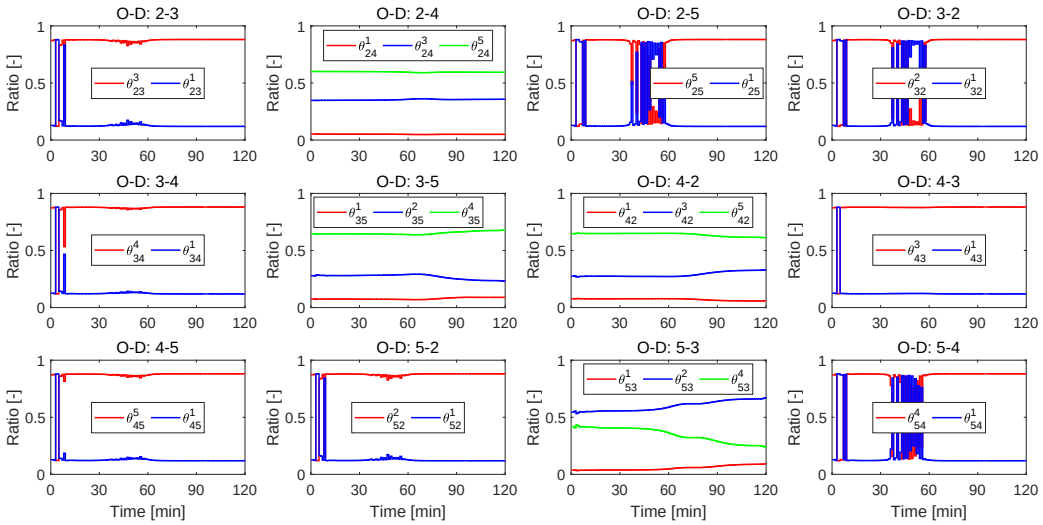


(b)

Figure A3.8 IADP PCRG of Case 1-Example 1. (a) PC, and (b) RG.

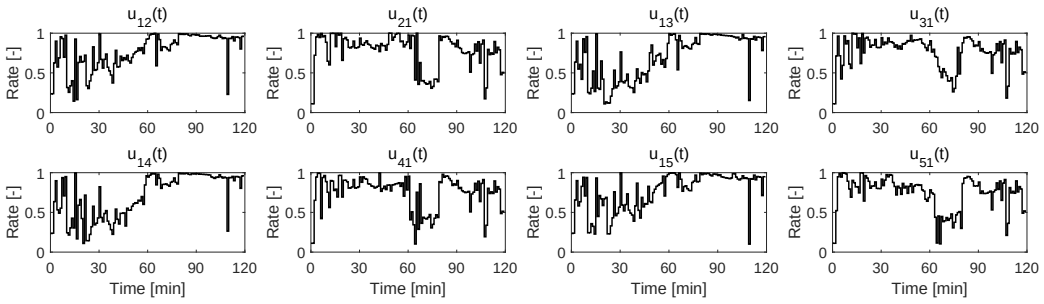


(a)

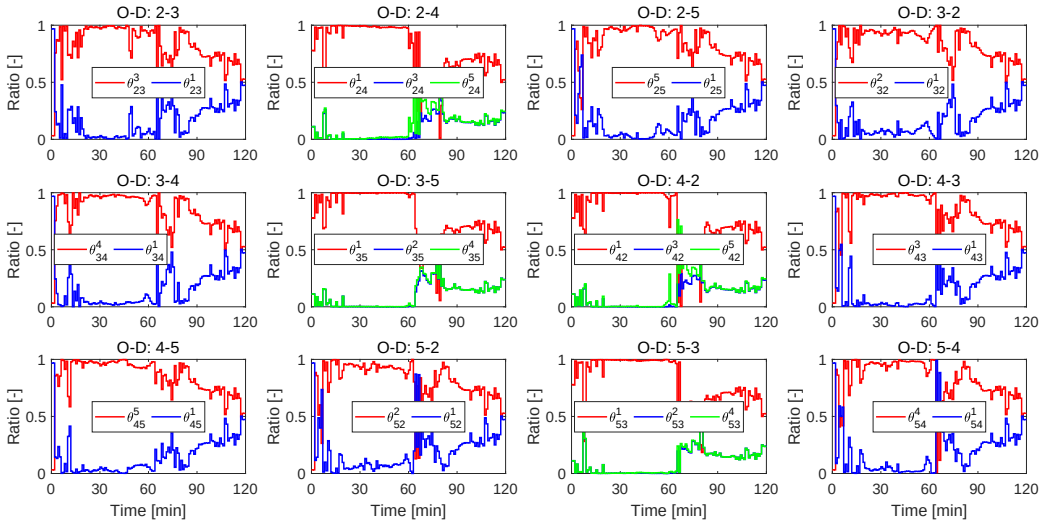


(b)

Figure A3.9 IADP-PT PCRG of Case 1-Example 1. (a) PC, and (b) RG.



(a)



(b)

Figure A3.10 MPC-PM PCRG of Case 1-Example 1. (a) PC, and (b) RG.

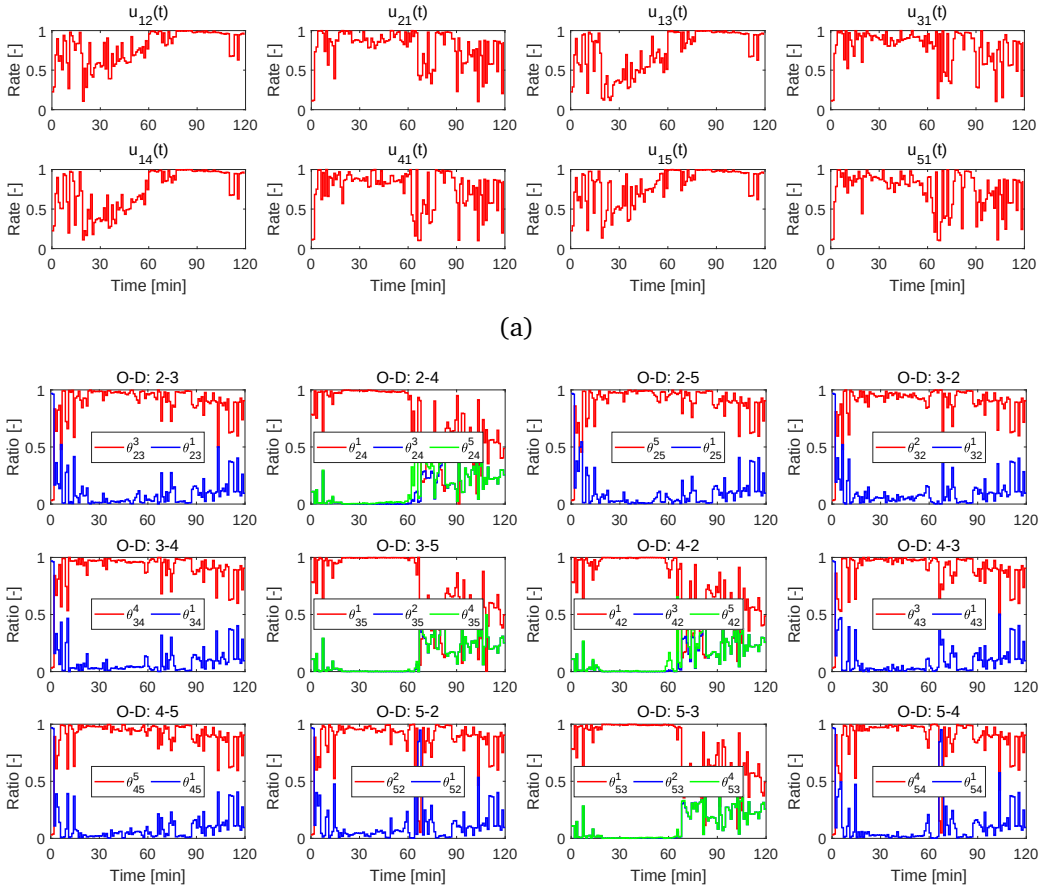


Figure A3.11 MPC-UKF PCRG of Case 1-Example 1. (a) PC, and (b) RG.

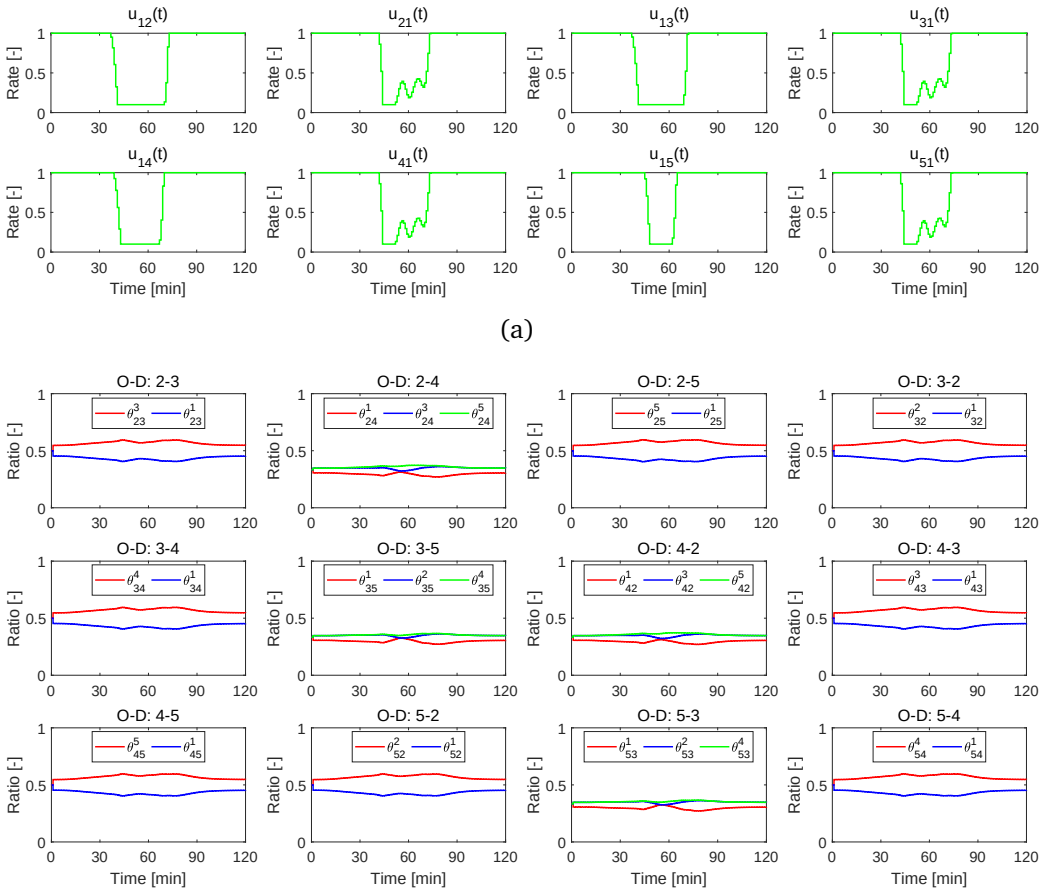
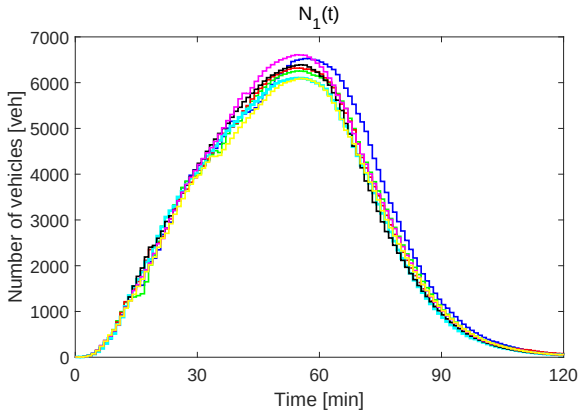
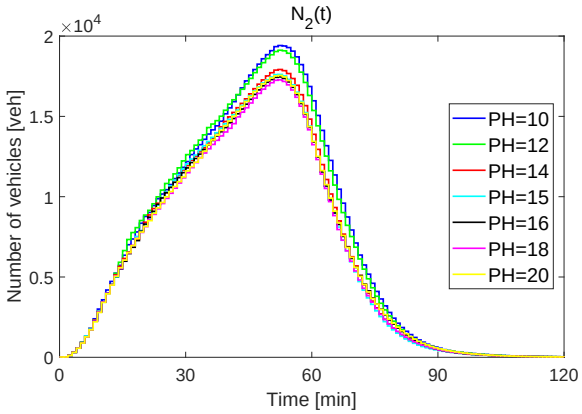


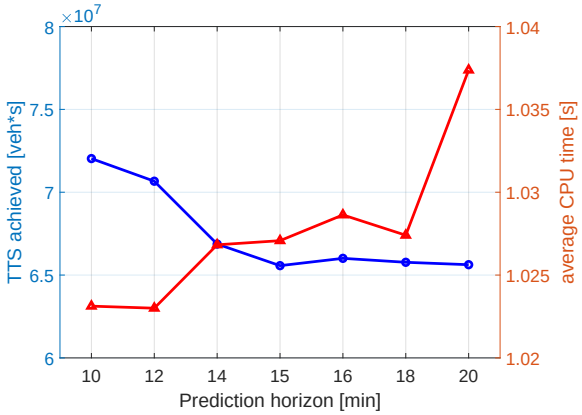
Figure A3.12 PIL PCRG of Case 1-Example 1. (a) PC, and (b) RG.



(a)



(b)



(c)

Figure A3.13 Performance comparison among MPC controllers with different prediction horizons. (a) State N_1, (b) State N_2, and (c) TTS and average CPU times.

Bibliography

Aalipour, Ali, Hamed Kebriaei, and Mohsen Ramezani (2018). "Analytical Optimal Solution of Perimeter Traffic Flow Control Based on MFD Dynamics: A Pontryagin's Maximum Principle Approach". In: *IEEE Transactions on Intelligent Transportation Systems* (cit. on pp. 13, 80).

Aboudolas, Konstantinos and Nikolas Geroliminis (2013). "Perimeter and boundary flow control in multi-reservoir heterogeneous networks". In: *Transportation Research Part B: Methodological* 55, pp. 265–281 (cit. on pp. 12, 80).

Abu-Khalaf, Murad, Jie Huang, and Frank L Lewis (2006). *Nonlinear H2/H-Infinity Constrained Feedback Control: A Practical Design Approach Using Neural Networks*. Springer Science & Business Media (cit. on p. 34).

Abu-Khalaf, Murad and Frank L Lewis (2005). "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach". In: *Automatica* 41.5, pp. 779–791 (cit. on p. 55).

Abu-Khalaf, Murad, Frank L Lewis, and Jie Huang (2008). "Neurodynamic programming and zero-sum games for constrained control systems". In: *IEEE Transactions on Neural Networks* 19.7, pp. 1243–1252 (cit. on p. 20).

Ambühl, Lukas and Monica Menendez (2016). "Data fusion algorithm for macroscopic fundamental diagram estimation". In: *Transportation Research Part C: Emerging Technologies* 71, pp. 184–197 (cit. on pp. 2, 11).

Ameli, Mostafa, Mohamad Sadegh Shirani Faradonbeh, Jean-Patrick Lebacque, Hossein Abouee-Mehrizi, and Ludovic Leclercq (2022). "Departure time choice models in urban transportation systems based on mean field games". In: *Transportation Science* 56.6, pp. 1483–1504 (cit. on pp. 2, 11).

Ampountolas, Konstantinos, Nan Zheng, and Nikolas Geroliminis (2017). "Macroscopic modelling and robust control of bi-modal multi-region urban road networks". In: *Transportation Research Part B: Methodological* 104, pp. 616–637 (cit. on pp. 2, 11).

Baldi, Simone, Iakovos Michailidis, Vasiliki Ntampasi, et al. (2019). "A simulation-based traffic signal control for congested urban traffic networks". In: *Transportation Science* 53.1, pp. 6–20 (cit. on pp. 3, 14, 15).

Batista, Sérgio FA, Deepak Ingole, Ludovic Leclercq, and Mónica Menéndez (2021). "The role of trip lengths calibration in model-based perimeter control strategies". In: *IEEE Transactions on Intelligent Transportation Systems* 23.6, pp. 5176–5186 (cit. on pp. 101, 121, 130, 150).

Batista, Sérgio FA and Ludovic Leclercq (2019). "Regional dynamic traffic assignment framework for macroscopic fundamental diagram multi-regions models". In: *Transportation Science* 53.6, pp. 1563–1590 (cit. on p. 16).

Batista, SFA, Ludovic Leclercq, and Nikolas Geroliminis (2019). "Estimation of regional trip length distributions for the calibration of the aggregated network traffic models". In: *Transportation Research Part B: Methodological* 122, pp. 192–217 (cit. on pp. 16, 17, 150).

Beojone, Caio Vitor and Nikolas Geroliminis (2023). "A dynamic multi-region MFD model for ride-sourcing with ridesplitting". In: *Transportation Research Part B: Methodological* 177, p. 102821 (cit. on p. 169).

Bertsekas, D. and J.N. Tsitsiklis (1996). *Neuro-Dynamic Programming*. Athena Scientific (cit. on p. 151).

Cao, Jin and Monica Menendez (2015). "System dynamics of urban traffic based on its parking-related-states". In: *Transportation Research Part B: Methodological* 81, pp. 718–736 (cit. on pp. 2, 11).

Chen, C, YP Huang, WHK Lam, et al. (2022). "Data efficient reinforcement learning and adaptive optimal perimeter control of network traffic dynamics". In: *Transportation Research Part C: Emerging Technologies* 142, p. 103759 (cit. on pp. 91, 102, 111–113, 119, 142, 150, 153, 173).

Chen, Can, Nikolas Geroliminis, and Renxin Zhong (2024). "An iterative adaptive dynamic programming approach for macroscopic fundamental diagram-based perimeter control and route guidance". In: *Transportation Science* 58.4, pp. 896–918 (cit. on p. 150).

Daganzo, Carlos F (1994). "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory". In: *Transportation research part B: methodological* 28.4, pp. 269–287 (cit. on p. 4).

– (2007). "Urban gridlock: Macroscopic modeling and mitigation approaches". In: *Transportation Research Part B: Methodological* 41.1, pp. 49–62 (cit. on pp. 1, 12, 80).

Daganzo, Carlos F, Vikash V Gayah, and Eric J Gonzales (2011). "Macroscopic relations of urban traffic variables: Bifurcations, multivaluedness and instability". In: *Transportation Research Part B: Methodological* 45.1, pp. 278–288 (cit. on p. 2).

Daganzo, Carlos F and Nikolas Geroliminis (2008). “An analytical approximation for the macroscopic fundamental diagram of urban traffic”. In: *Transportation Research Part B: Methodological* 42.9, pp. 771–781 (cit. on p. 2).

Ding, Heng, Yunran Di, Zhongxiang Feng, et al. (2022). “A perimeter control method for a congested urban road network with dynamic and variable ranges”. In: *Transportation Research Part B: Methodological* 155, pp. 160–187 (cit. on p. 13).

Ding, Heng, Hanyu Yuan, Xiaoyan Zheng, et al. (2020a). “Integrated control for a large-scale mixed network of arterials and freeways”. In: *IEEE Intelligent Transportation Systems Magazine* 13.3, pp. 131–145 (cit. on pp. 18, 141).

Ding, Heng, Jingwen Zhou, Xiaoyan Zheng, et al. (2020b). “Perimeter control for congested areas of a large-scale traffic network: A method against state degradation risk”. In: *Transportation Research Part C: Emerging Technologies* 112, pp. 28–45 (cit. on p. 13).

Franklin, Gene F, J David Powell, and Abbas Emami-Naeini (2015). *Feedback control of dynamic systems*. Pearson London (cit. on p. 59).

Fu, Hui, Saifei Chen, Kaiyu Chen, Anastasios Kouvelas, and Nikolaos Geroliminis (2021). “Perimeter control and route guidance of multi-region MFD systems with boundary queues using colored Petri Nets”. In: *IEEE Transactions on Intelligent Transportation Systems* (cit. on p. 17).

Fu, Hui, Na Liu, and Gang Hu (2017). “Hierarchical perimeter control with guaranteed stability for dynamically coupled heterogeneous urban traffic”. In: *Transportation Research Part C: Emerging Technologies* 83, pp. 18–38 (cit. on pp. 13, 80).

Gao, Shengling, Daqing Li, Nan Zheng, Ruiqi Hu, and Zhikun She (2022). “Resilient perimeter control for hyper-congested two-region networks with MFD dynamics”. In: *Transportation Research Part B: Methodological* 156, pp. 50–75 (cit. on p. 13).

Gartner, Nathan H and Peter Wagner (2004). “Analysis of traffic flow characteristics on signalized arterials”. In: *Transportation Research Record* 1883.1, pp. 94–100 (cit. on p. 2).

Gayah, Vikash V and Carlos F Daganzo (2011). “Clockwise hysteresis loops in the macroscopic fundamental diagram: an effect of network instability”. In: *Transportation Research Part B: Methodological* 45.4, pp. 643–655 (cit. on p. 2).

Gayah, Vikash V, Xueyu Shirley Gao, and Andrew S Nagle (2014). “On the impacts of locally adaptive signal control on urban network stability and the macroscopic fundamental diagram”. In: *Transportation Research Part B: Methodological* 70, pp. 255–268 (cit. on p. 1).

Geroliminis, Nikolas and Burak Boyacı (2012). “The effect of variability of urban systems characteristics in the network capacity”. In: *Transportation Research Part B: Methodological* 46.10, pp. 1607–1623 (cit. on p. 14).

Geroliminis, Nikolas and Carlos F Daganzo (2008). “Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings”. In: *Transportation Research Part B: Methodological* 42.9, pp. 759–770 (cit. on p. 2).

Geroliminis, Nikolas, Jack Haddad, and Mohsen Ramezani (2013). “Optimal perimeter control for two urban regions with macroscopic fundamental diagrams: A model predictive approach”. In: *IEEE Transactions on Intelligent Transportation Systems* 14.1, pp. 348–359 (cit. on pp. 3, 12, 56, 57, 80, 83, 101, 171).

Geroliminis, Nikolas and Jie Sun (2011). “Properties of a well-defined macroscopic fundamental diagram for urban traffic”. In: *Transportation Research Part B: Methodological* 45.3, pp. 605–617 (cit. on p. 107).

Godfrey, JW (1969). “The mechanism of a road network”. In: *Traffic Engineering & Control* 8.8 (cit. on p. 1).

Gomes, Gabriel and Roberto Horowitz (2006). “Optimal freeway ramp metering using the asymmetric cell transmission model”. In: *Transportation Research Part C: Emerging Technologies* 14.4, pp. 244–262 (cit. on pp. 142, 147).

Gu, Ziyuan, Sajjad Shafiei, Zhiyuan Liu, and Meead Saberi (2018). “Optimal distance- and time-dependent area-based pricing with the Network Fundamental Diagram”. In: *Transportation Research Part C: Emerging Technologies* 95, pp. 1–28 (cit. on pp. 2, 11).

Haddad, Jack (2015). “Robust constrained control of uncertain macroscopic fundamental diagram networks”. In: *Transportation Research Part C* 59, pp. 323–339 (cit. on pp. 13, 31, 55, 59, 80, 92, 150).

– (2017a). “Optimal coupled and decoupled perimeter control in one-region cities”. In: *Control Engineering Practice* 61, pp. 134–148 (cit. on p. 13).

– (2017b). “Optimal perimeter control synthesis for two urban regions with aggregate boundary queue dynamics”. In: *Transportation Research Part B: Methodological* 96, pp. 1–25 (cit. on p. 13).

Haddad, Jack and Nikolas Geroliminis (2012). “On the stability of traffic perimeter control in two-region urban cities”. In: *Transportation Research Part B: Methodological* 46.9, pp. 1159–1176 (cit. on pp. 1, 79).

Haddad, Jack and Boris Mirkin (2016). “Adaptive perimeter traffic control of urban road networks based on MFD model with time delays”. In: *International Journal of Robust and Nonlinear Control* 26.6, pp. 1267–1285 (cit. on pp. 12, 80).

– (2017). “Coordinated distributed adaptive perimeter control for large-scale urban road networks”. In: *Transportation Research Part C: Emerging Technologies* 77, pp. 495–515 (cit. on p. 81).

– (2020). “Resilient perimeter control of macroscopic fundamental diagram networks under cyberattacks”. In: *Transportation research part B: methodological* 132, pp. 44–59 (cit. on pp. 2, 11).

Haddad, Jack, Boris Mirkin, Kfir Assor, et al. (2021). “Traffic flow modeling and feedback control for future Low-Altitude Air city Transport: An MFD-based approach”. In: *Transportation Research Part C: Emerging Technologies* 133, p. 103380 (cit. on p. 169).

Haddad, Jack, Mohsen Ramezani, and Nikolas Geroliminis (2013). “Cooperative traffic control of a mixed network with two urban regions and a freeway”. In: *Transportation Research Part B: Methodological* 54, pp. 17–36 (cit. on pp. 1, 3, 4, 13, 16, 18, 79, 80, 141, 142, 144, 147).

Haddad, Jack and Arie Shraiber (2014). “Robust perimeter control design for an urban region”. In: *Transportation Research Part B: Methodological* 68, pp. 315–332 (cit. on pp. 13, 26, 32, 80, 100).

Haddad, Jack and Zhengfei Zheng (2020). “Adaptive perimeter control for multi-region accumulation-based models with state delays”. In: *Transportation Research Part B: Methodological* 137, pp. 133–153 (cit. on pp. 13, 81, 84).

Hamedmoghadam, Homayoun, Nan Zheng, Daqing Li, and Hai L Vu (2022). “Percolation-based dynamic perimeter control for mitigating congestion propagation in urban road networks”. In: *Transportation Research Part C: Emerging Technologies* 145, p. 103922 (cit. on p. 13).

Han, Yu, Mohsen Ramezani, Andreas Hegyi, Yufei Yuan, and Serge Hoogendoorn (2020). “Hierarchical ramp metering in freeways: An aggregated modeling and control approach”. In: *Transportation research part C: emerging technologies* 110, pp. 1–19 (cit. on p. 18).

Helbing, Dirk (2009). “Derivation of a fundamental diagram for urban traffic flow”. In: *The European Physical Journal B* 70.2, pp. 229–241 (cit. on p. 2).

Hou, Zhongsheng and Ting Lei (2020). “Constrained model free adaptive predictive perimeter control and route guidance for multi-region urban traffic systems”. In: *IEEE Transactions on Intelligent Transportation Systems* 23.2, pp. 912–924 (cit. on pp. 2, 11, 17).

Hu, Zijian and Wei Ma (2024). “Demonstration-guided deep reinforcement learning for coordinated ramp metering and perimeter control in large scale networks”. In: *Transportation Research Part C: Emerging Technologies* 159, p. 104461 (cit. on pp. 2–4, 11, 18).

Huang, YP, JH Xiong, A Sumalee, et al. (2020). “A dynamic user equilibrium model for multi-region macroscopic fundamental diagram systems with time-varying delays”. In: *Transportation Research Part B: Methodological* 131, pp. 1–25 (cit. on pp. 2, 11, 16, 100).

Huang, Yunping, Jianhui Xiong, Shu-Chien Hsu, et al. (2024). “A comparison of the accumulation-based, trip-based and time delay macroscopic fundamental diagram models”. In: *Transportmetrica A: Transport Science*, pp. 1–37 (cit. on p. 139).

Huang, Yunping, Nan Zheng, Enming Liang, Shu-Chien Hsu, and Renxin Zhong (2023). “An Approximate Dynamic Programming Approach to Vehicle Dispatching and Relocation Using Time-Dependent Travel Times”. In: *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 2652–2657 (cit. on pp. 2, 169).

Hunt, PB, DI Robertson, RD Bretherton, and M Cr Royle (1982). “The SCOOT on-line traffic signal optimisation technique”. In: *Traffic Engineering & Control* 23.4 (cit. on p. 1).

Ji, Yangbeibei, Winnie Daamen, Serge Hoogendoorn, Sascha Hoogendoorn-Lanser, and Xiaoyu Qian (2010). “Investigating the shape of the macroscopic fundamental diagram using simulation data”. In: *Transportation Research Record* 2161.1, pp. 40–48 (cit. on p. 2).

Ji, Yuxuan and Nikolas Geroliminis (2012). “On the spatial partitioning of urban transportation networks”. In: *Transportation Research Part B: Methodological* 46.10, pp. 1639–1656 (cit. on p. 2).

Jiang, Shang, Cong Quoc Tran, and Mehdi Keyvan-Ekbatani (2024). “Regional route guidance with realistic compliance patterns: Application of deep reinforcement learning and MPC”. In: *Transportation Research Part C: Emerging Technologies* 158, p. 104440 (cit. on pp. 2, 11).

Keyvan-Ekbatani, Mehdi, Anastasios Kouvelas, Ioannis Papamichail, and Markos Papageorgiou (2012). “Exploiting the fundamental diagram of urban networks for feedback-based gating”. In: *Transportation Research Part B: Methodological* 46.10, pp. 1393–1403 (cit. on pp. 12, 80, 122, 155).

Keyvan-Ekbatani, Mehdi, Markos Papageorgiou, and Victor L Knoop (2015a). “Controller design for gating traffic control in presence of time-delay in urban road networks”. In: *Transportation Research Procedia* 7, pp. 651–668 (cit. on pp. 12, 80).

Keyvan-Ekbatani, Mehdi, Markos Papageorgiou, and Ioannis Papamichail (2013). "Urban congestion gating control based on reduced operational network fundamental diagrams". In: *Transportation Research Part C: Emerging Technologies* 33, pp. 74–87 (cit. on pp. 1, 12, 79, 80).

Keyvan-Ekbatani, Mehdi, Mehmet Yildirimoglu, Nikolas Geroliminis, and Markos Papageorgiou (2015b). "Multiple concentric gating traffic control in large-scale urban networks". In: *IEEE Transactions on Intelligent Transportation Systems* 16.4, pp. 2141–2154 (cit. on pp. 12, 80).

Kheterpal, Nishant, Kanaad Parvate, Cathy Wu, et al. (2018). "Flow: Deep reinforcement learning for control in sumo". In: *EPiC Series in Engineering* 2, pp. 134–151 (cit. on p. 15).

Knoop, VL, SP Hoogendoorn, and JWC Van Lint (2012). "Routing strategies based on macroscopic fundamental diagram". In: *Transportation Research Record* 2315.1, pp. 1–10 (cit. on pp. 2, 11, 16).

Kouvelas, Anastasios, Mohammadreza Saeedmanesh, and Nikolas Geroliminis (2017). "Enhancing model-based feedback perimeter control with data-driven online adaptive optimization". In: *Transportation Research Part B: Methodological* 96, pp. 26–45 (cit. on pp. 1, 13, 14, 80).

– (2023). "A Linear-Parameter-Varying Formulation for Model Predictive Perimeter Control in Multi-Region MFD Urban Networks". In: *Transportation Science* (cit. on p. 121).

Kumarage, Sakitha, Mehmet Yildirimoglu, Mohsen Ramezani, and Zuduo Zheng (2021). "Schedule-constrained demand management in two-region urban networks". In: *Transportation Science* 55.4, pp. 857–882 (cit. on pp. 2, 11, 169).

Kutadinata, Ronny, Will Moase, Chris Manzie, Lele Zhang, and Tim Garoni (2016). "Enhancing the performance of existing urban traffic light control through extremum-seeking". In: *Transportation Research Part C: Emerging Technologies* 62, pp. 1–20 (cit. on p. 80).

Leclercq, Ludovic, Nicolas Chiabaut, and Béatrice Trinquier (2014). "Macroscopic fundamental diagrams: A cross-comparison of estimation methods". In: *Transportation Research Part B: Methodological* 62, pp. 1–12 (cit. on pp. 1, 79).

Leclercq, Ludovic and Nikolas Geroliminis (2013). "Estimating MFDs in simple networks with route choice". In: *Transportation Research Part B: Methodological* 57, pp. 468–484 (cit. on pp. 16, 100).

Leclercq, Ludovic, Alméria Sénécat, and Guilhem Mariotte (2017). "Dynamic macroscopic simulation of on-street parking search: A trip-based approach". In: *Transportation Research Part B: Methodological* 101, pp. 268–282 (cit. on pp. 2, 11).

Lee, Jaeyoung and Richard S Sutton (2021). “Policy iterations for reinforcement learning problems in continuous time and space - Fundamental theory and methods”. In: *Automatica* 126, p. 109421 (cit. on p. 111).

Lei, Ting, Zhongsheng Hou, and Ye Ren (2019). “Data-Driven Model Free Adaptive Perimeter Control for Multi-Region Urban Traffic Networks With Route Choice”. In: *IEEE Transactions on Intelligent Transportation Systems* (cit. on pp. 14, 16, 81).

Lewis, Frank L and Draguna Vrabie (2009). “Reinforcement learning and adaptive dynamic programming for feedback control”. In: *IEEE Circuits and Systems Magazine* 9.3, pp. 32–50 (cit. on p. 90).

Lewis, Frank L, Draguna Vrabie, and Vassilis L Syrmos (2012). *Optimal control*. John Wiley & Sons (cit. on p. 45).

Li, Dai and Bart De Schutter (2022). “Distributed model-free adaptive predictive control for urban traffic networks”. In: *IEEE Transactions on Control Systems Technology* 30.1, pp. 180–192 (cit. on p. 81).

Li, Ye, Reza Mohajerpoor, and Mohsen Ramezani (2021a). “Perimeter control with real-time location-varying cordon”. In: *Transportation Research Part B: Methodological* 150, pp. 101–120 (cit. on p. 13).

Li, Ye, Mehmet Yildirimoglu, and Mohsen Ramezani (2021b). “Robust perimeter control with cordon queues and heterogeneous transfer flows”. In: *Transportation Research Part C: Emerging Technologies* 126, p. 103043 (cit. on p. 13).

Liu, Derong, Shan Xue, Bo Zhao, Biao Luo, and Qinglai Wei (2020). “Adaptive dynamic programming for control: A survey and recent advances”. In: *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 51.1, pp. 142–160 (cit. on p. 23).

Loder, Allister, Lukas Ambühl, Monica Menendez, and Kay W Axhausen (2019). “Understanding traffic capacity of urban networks”. In: *Scientific reports* 9.1, pp. 1–10 (cit. on p. 2).

Lopez, Pablo Alvarez, Michael Behrisch, Laura Bieker-Walz, et al. (2018). “Microscopic traffic simulation using sumo”. In: *2018 21st international conference on intelligent transportation systems (ITSC)*. IEEE, pp. 2575–2582 (cit. on p. 74).

Lowrie, PR (1982). “The Sydney cooridinated adaptive traffic (SCAT) system-principles, methodology, algorithm”. In: *Proc. of International Conference on Road Traffic Signaling*. IEE, pp. 67–70 (cit. on p. 1).

Lyashevskiy, Sergey (1996). “Constrained optimization and control of nonlinear systems: new results in optimal control”. In: *Proceedings of 35th ieee conference on decision and control*. Vol. 1. IEEE, pp. 541–546 (cit. on p. 45).

Lyshevski, S Edward (1998). “Optimal control of nonlinear continuous-time systems: design of bounded controllers via generalized nonquadratic functionals”. In: *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207)*. Vol. 1. IEEE, pp. 205–209 (cit. on p. 34).

Ma, Wenfei, Yunping Huang, Xiao Jin, and Renxin Zhong (2024). “Functional form selection and calibration of macroscopic fundamental diagrams”. In: *Physica A: Statistical Mechanics and its Applications* 640, p. 129691 (cit. on p. 2).

Mariotte, Guilhem, Ludovic Leclercq, SFA Batista, Jean Krug, and Mahendra Paipuri (2020). “Calibration and validation of multi-reservoir MFD models: A case study in Lyon”. In: *Transportation Research Part B: Methodological* 136, pp. 62–86 (cit. on pp. 2, 11).

Mercader, Pedro and Jack Haddad (2021). “Resilient multivariable perimeter control of urban road networks under cyberattacks”. In: *Control Engineering Practice* 109, p. 104718 (cit. on p. 13).

Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, et al. (2015). “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540, p. 529 (cit. on pp. 3, 14, 142).

Modares, Hamidreza and Frank L Lewis (2014). “Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning”. In: *Automatica* 50.7, pp. 1780–1792 (cit. on pp. 84, 87).

Modares, Hamidreza, Frank L. Lewis, and Mohammad-Bagher Naghibi-Sistani (2014). “Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems”. In: *Automatica* 50, pp. 193–202 (cit. on pp. 14, 27, 47, 51, 52, 61, 116).

Mohajerpoor, Reza, Meead Saberi, Hai L Vu, Timothy M Garoni, and Mohsen Ramezani (2020). “H ∞ robust perimeter flow control in urban networks with partial information feedback”. In: *Transportation Research Part B: Methodological* 137, pp. 47–73 (cit. on pp. 2, 11, 80).

Moshahedi, Nadia and Lina Kattan (2023). “Alpha-fair large-scale urban network control: A perimeter control based on a macroscopic fundamental diagram”. In: *Transportation Research Part C: Emerging Technologies* 146, p. 103961 (cit. on pp. 2, 11).

Muralidharan, Ajith, Gunes Dervisoglu, and Roberto Horowitz (2009). “Freeway traffic flow simulation using the link node cell transmission model”. In: *2009 American Control Conference*. IEEE, pp. 2916–2921 (cit. on p. 155).

Murray, John J, Chadwick J Cox, George G Lendaris, and Richard Saeks (2002). "Adaptive dynamic programming". In: *IEEE transactions on systems, man, and cybernetics, Part C (Applications and Reviews)* 32.2, pp. 140–153 (cit. on p. 111).

Nourinejad, Mehdi and Mohsen Ramezani (2020). "Ride-sourcing modeling and pricing in non-equilibrium two-sided markets". In: *Transportation Research Part B: Methodological* 132, pp. 340–357 (cit. on p. 169).

Ogata, Katsuhiko et al. (2010). *Modern control engineering*. Vol. 5. Prentice hall Upper Saddle River, NJ (cit. on p. 111).

Papageorgiou, Markos, Christina Diakaki, Vaya Dinopoulou, Apostolos Kotsialos, and Yibing Wang (2003). "Review of road traffic control strategies". In: *Proceedings of the IEEE* 91.12, pp. 2043–2067 (cit. on p. 1).

Ramezani, Mohsen, Jack Haddad, and Nikolas Geroliminis (2015). "Dynamics of heterogeneity in urban networks: aggregated traffic modeling and hierarchical control". In: *Transportation Research Part B: Methodological* 74, pp. 1–19 (cit. on pp. xiii, 12, 16, 80, 100, 101, 105–107, 147, 150, 155).

Ramezani, Mohsen and Mehdi Nourinejad (2018). "Dynamic modeling and control of taxi services in large-scale urban networks: A macroscopic approach". In: *Transportation Research Part C: Emerging Technologies* 94, pp. 203–219 (cit. on p. 169).

Ramezani, Mohsen and Amir Hosein Valadkhani (2023). "Dynamic ride-sourcing systems for city-scale networks-Part I: Matching design and model formulation and validation". In: *Transportation Research Part C: Emerging Technologies* 152, p. 104158 (cit. on pp. 2, 11).

Ren, Ye, Zhongsheng Hou, Isik Ilber Sirmatel, and Nikolas Geroliminis (2020). "Data driven model free adaptive iterative learning perimeter control for large-scale urban road networks". In: *Transportation Research Part C: Emerging Technologies* 115, p. 102618 (cit. on pp. 14, 26, 81).

Saeedmanesh, Mohammadreza and Nikolas Geroliminis (2017). "Dynamic clustering and propagation of congestion in heterogeneously congested urban traffic networks". In: *Transportation research procedia* 23, pp. 962–979 (cit. on p. 1).

Saeedmanesh, Mohammadreza, Anastasios Kouvelas, and Nikolas Geroliminis (2021). "An extended Kalman filter approach for real-time state estimation in multi-region MFD urban networks". In: *Transportation Research Part C: Emerging Technologies* 132, p. 103384 (cit. on p. 13).

Safadi, Yazan, Rao Fu, Quan Quan, and Jack Haddad (2023a). "Macroscopic fundamental diagrams for low-altitude air city transport". In: *Transportation Research Part C: Emerging Technologies* 152, p. 104141 (cit. on p. 169).

Safadi, Yazan, Nikolas Geroliminis, Jack Haddad, et al. (2023b). “Aircraft Departures Management for Low Altitude Air City Transport based on Macroscopic Fundamental Diagram”. In: *2023 American Control Conference (ACC)*. IEEE, pp. 4393–4398 (cit. on p. 169).

Silver, David, Aja Huang, Chris J Maddison, et al. (2016). “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529.7587, p. 484 (cit. on pp. 3, 14, 142).

Sirmatel, Isik Ilber and Nikolas Geroliminis (2018). “Economic model predictive control of large-scale urban road networks via perimeter control and regional route guidance”. In: *IEEE Transactions on Intelligent Transportation Systems* 19.4, pp. 1112–1121 (cit. on pp. 2, 11, 17, 26, 100, 101, 103, 106, 107, 132, 135, 147).

– (2019). “Nonlinear moving horizon estimation for large-scale urban road networks”. In: *IEEE Transactions on Intelligent Transportation Systems* 21.12, pp. 4983–4994 (cit. on p. 13).

– (2021). “Stabilization of city-scale road traffic networks via macroscopic fundamental diagram-based model predictive perimeter control”. In: *Control Engineering Practice* 109, p. 104750 (cit. on p. 13).

Sirmatel, Isik Ilber, Dimitrios Tsitsokas, Anastasios Kouvelas, and Nikolas Geroliminis (2021). “Modeling, estimation, and control in large-scale urban road networks with remaining travel distance dynamics”. In: *Transportation Research Part C: Emerging Technologies* 128, p. 103157 (cit. on p. 13).

Su, ZC, Andy HF Chow, CL Fang, EM Liang, and RX Zhong (2023). “Hierarchical control for stochastic network traffic with reinforcement learning”. In: *Transportation Research Part B: Methodological* 167, pp. 196–216 (cit. on pp. 2, 11).

Su, Z.C., Andy H.F. Chow, N. Zheng, et al. (2020). “Neuro-dynamic programming for optimal control of macroscopic fundamental diagram systems”. In: *Transportation Research Part C: Emerging Technologies* 116, p. 102628 (cit. on pp. 15, 26, 31, 33, 57, 59, 81, 102, 150).

Sutton, Richard S and Andrew G Barto (2018). *Reinforcement learning: An introduction*. MIT press (cit. on pp. 3, 14).

Tsitsokas, Dimitrios, Anastasios Kouvelas, and Nikolas Geroliminis (2023). “Two-layer adaptive signal control framework for large-scale dynamically-congested networks: Combining efficient Max Pressure with Perimeter Control”. In: *Transportation Research Part C: Emerging Technologies* 152, p. 104128 (cit. on pp. 2, 11).

Valadkhani, Amir Hosein and Mohsen Ramezani (2023). “Dynamic ride-sourcing systems for city-scale networks, Part II: Proactive vehicle repositioning”. In: *Transportation Research Part C: Emerging Technologies* 152, p. 104159 (cit. on pp. 2, 11).

Vrabie, D., O. Pastravanu, M. Abu-Khalaf, and F.L. Lewis (2009). “Adaptive optimal control for continuous-time linear systems based on policy iteration”. In: *Automatica* 45.2, pp. 477 –484 (cit. on p. 27).

Wang, Jiawen, Xiaozheng He, Srinivas Peeta, and Xiaoguang Yang (2021). “Feedback perimeter control with online estimation of maximum throughput for an incident-affected road network”. In: *Journal of Intelligent Transportation Systems* 26.1, pp. 81–99 (cit. on p. 80).

Wei, Bangyang, Meead Saberi, Fangni Zhang, Wei Liu, and S Travis Waller (2020). “Modeling and managing ridesharing in a multi-modal network with an aggregate traffic representation: A doubly dynamical approach”. In: *Transportation Research Part C: Emerging Technologies* 117, p. 102670 (cit. on pp. 2, 11).

Yildirimoglu, Mehmet and Nikolas Geroliminis (2014). “Approximating dynamic equilibrium conditions with macroscopic fundamental diagrams”. In: *Transportation Research Part B: Methodological* 70, pp. 186–200 (cit. on pp. 1, 2, 11, 16, 17, 79).

Yildirimoglu, Mehmet and Mohsen Ramezani (2020). “Demand management with limited cooperation among travellers: A doubly dynamic approach”. In: *Transportation Research Part B: Methodological* 132, pp. 267–284 (cit. on pp. 2, 11, 139, 169).

Yildirimoglu, Mehmet, Mohsen Ramezani, and Nikolas Geroliminis (2015). “Equilibrium analysis and route guidance in large-scale networks with MFD dynamics”. In: *Transportation Research Procedia* 9, pp. 185–204 (cit. on pp. 16, 100, 103, 104, 106, 107, 147).

Yildirimoglu, Mehmet, Isik Ilber Sirmatel, and Nikolas Geroliminis (2018). “Hierarchical control of heterogeneous large-scale urban road networks via path assignment and regional route guidance”. In: *Transportation Research Part B: Methodological* 118, pp. 106–123 (cit. on pp. 16, 100, 103, 135, 150).

Yocum, Rebeka and Vikash V Gayah (2022). “Coordinated perimeter flow and variable speed limit control for mixed freeway and urban networks”. In: *Transportation research record* 2676.1, pp. 596–609 (cit. on pp. 18, 141).

Yu, Hansong and Zhongsheng Hou (2020). “Two-level hierarchical optimal control for urban traffic networks”. In: *Transportmetrica A: Transport Science*, pp. 1–22 (cit. on p. 80).

Zhang, Kun, Huaguang Zhang, He Jiang, and Yingchun Wang (2018). “Near-optimal output tracking controller design for nonlinear systems using an event-driven ADP approach”. In: *Neurocomputing* 309, pp. 168–178 (cit. on p. 87).

Zhang, Kun, Huaguang Zhang, Geyang Xiao, and Hanguang Su (2017). “Tracking control optimization scheme of continuous-time nonlinear system via online single network adaptive critic design method”. In: *Neurocomputing* 251, pp. 127–135 (cit. on p. 84).

Zheng, Nan and Nikolas Geroliminis (2020). “Area-based equitable pricing strategies for multimodal urban networks with heterogeneous users”. In: *Transportation Research Part A: Policy and Practice* 136, pp. 357–374 (cit. on pp. 2, 11).

Zheng, Nan, Guillaume Rérat, and Nikolas Geroliminis (2016). “Time-dependent area-based pricing for multimodal systems with heterogeneous users in an agent-based environment”. In: *Transportation Research Part C: Emerging Technologies* 62, pp. 133–148 (cit. on p. 85).

Zhong, Renxin, Agachai Sumalee, Tianlu Pan, and WHK Lam (2014). “Optimal and robust strategies for freeway traffic management under demand and supply uncertainties: an overview and general theory”. In: *Transportmetrica A: Transport Science* 10.10, pp. 849–877 (cit. on p. 14).

Zhong, Renxin, Jianhui Xiong, Yunping Huang, et al. (2020). “Dynamic system optimum analysis of multi-region macroscopic fundamental diagram systems with state-dependent time-varying delays”. In: *IEEE Transactions on Intelligent Transportation Systems* 21 (9), pp. 4000–4016 (cit. on pp. 2, 11, 16, 100).

Zhong, Renxin, Jianhui Xiong, Yunping Huang, et al. (2021). “Dynamic user equilibrium for departure time choice in the basic trip-based model”. In: *Transportation Research Part C: Emerging Technologies* 128, p. 103190 (cit. on pp. 2, 11, 16).

Zhong, RX, C Chen, YP Huang, et al. (2018a). “Robust perimeter control for two urban regions with macroscopic fundamental diagrams: a control-lyapunov function approach”. In: *Transportation Research Part B: Methodological* 117, pp. 687–707 (cit. on pp. 1, 2, 11, 13, 31, 32, 69, 80, 84, 86, 150).

Zhong, R.X., Y.P. Huang, C. Chen, et al. (2018b). “Boundary conditions and behavior of the macroscopic fundamental diagram based network traffic dynamics: A control systems perspective”. In: *Transportation Research Part B: Methodological* 111, pp. 327 –355 (cit. on pp. 1, 26, 31, 32, 45, 55, 69, 80, 85, 100, 111, 150, 172).

Zhou, Dongqin and Vikash V Gayah (2021). “Model-free perimeter metering control for two-region urban networks using deep reinforcement learning”. In: *Transportation Research Part C: Emerging Technologies* 124, p. 102949 (cit. on pp. 15, 26, 81, 102, 142).

Zhou, Zhao, Bart De Schutter, Shu Lin, and Yugeng Xi (2016). “Two-level hierarchical model-based predictive control for large-scale urban traffic networks”. In: *IEEE Transactions on Control Systems Technology* 25.2, pp. 496–508 (cit. on pp. 13, 80).