A Scalable Model-Free Deep Reinforcement Learning-Based Perimeter Metering Control Method for Multi-Region Urban Networks

Dongqin Zhou and Vikash V. Gayah*

Department of Civil and Environmental Engineering, The Pennsylvania State University, University Park, PA, 16802

* Corresponding author.

E-mail addresses: dongqin.zhou@psu.edu (D. Zhou), gayah@engr.psu.edu (V. V. Gayah)

ABSTRACT

Perimeter metering control based on macroscopic fundamental diagrams has attracted increasing research interests over the past decade. This strategy provides a convenient way to mitigate urban congestion by manipulating vehicular movements across homogeneous regions without modeling the detailed behaviors and interactions involved with individual vehicle presence. In particular, multi-region perimeter metering control holds promise for efficient traffic management in large-scale urban networks. However, most existing methods for multi-region control require knowledge of either the environment traffic dynamics or network properties (i.e., the critical accumulations), whereas such information is generally difficult to obtain and subject to significant estimation errors. The recently developed model-free techniques, on the other hand, have not yet been shown scalable or applicable to large urban networks. To fill this gap, this paper proposes a novel scalable model-free scheme based on model-free multi-agent deep reinforcement learning. The proposed scheme features value function decomposition in the paradigm of centralized training with decentralized execution, coupled with critical advances of single-agent deep reinforcement learning and problem reformulation guided by domain expertise. Comprehensive experiment results on a seven-region urban network suggest the scheme is: (a) effective, with consistent convergence to final control outcomes that are comparable to the model predictive control method; (b) resilient, with superior learning and control efficacy in the presence of inaccurate input information from the environment; and (c) transferable, with sufficient implementation prospect as well as real time applicability to unencountered environments featuring increased uncertainty.

Keywords: Macroscopic Fundamental Diagram (MFD); multi-region perimeter metering control; model-free multi-agent reinforcement learning (MARL)

1. INTRODUCTION

It has long been challenging to regulate vehicle flows in large-scale urban networks for the purpose of congestion alleviation and throughput maximization. While some pioneering systems have been developed in the past for urban traffic control (e.g., SCOOT (Robertson and Bretherton, 1991), SCATS (Lowrie, 1982), and max pressure (Varaiya, 2013)), they are localized and decentralized approaches that do not consider the network-wide effects. As a result, the control efficacy of these methods might be limited due to network-level phenomena such as congestion propagation. To mitigate this impact and to improve the effectiveness of urban traffic control, there have been much historical efforts to unveil the aggregate relationships between the traffic parameters (Herman and Prigogine, 1979; Williams et al., 1987) and to investigate aggregate modeling of traffic dynamics (Mahmassani and Herman, 1984; Small and Chu, 2003). More recently, the concept of network macroscopic fundamental diagram (MFD) has shown promise to describe urban traffic dynamics at an aggregate level and to facilitate the design of network-level traffic control schemes; see for example (Daganzo, 2007; Daganzo et al., 2011; Geroliminis et al., 2013; Geroliminis and Daganzo, 2008; Yildirimoglu et al., 2018).

The initial theoretical investigation of the MFD dates back to the 1960s (Godfrey, 1969), but its existence was not verified until recently (Daganzo, 2007; Geroliminis and Daganzo, 2008). These seminal works have since inspired sizable research endeavors on the existence analysis (Fu et al., 2020; Geroliminis and Sun, 2011; Paipuri et al., 2020) and estimation of MFDs, e.g., using empirical and microsimulation data (Ambühl and Menendez, 2016; Buisson and Ladier, 2009; Du et al., 2016; Nagle and Gayah, 2014), or with the analytical approaches (Daganzo and Lehe, 2016; Laval and Castrillón, 2015; Leclercq and Geroliminis, 2013; Tilg et al., 2020). Other than the derivations, the properties of well-defined MFDs have also been examined extensively (Daganzo et al., 2011; Gayah and Daganzo, 2011; Mahmassani et al., 2013; Mazloumian et al., 2010). These references have shown that urban networks are subject to instability, hysteresis, and bifurcation phenomena with heterogeneous distribution of vehicle presence. Fortunately, network partitioning strategies can be utilized to divide a large heterogeneous network into several smaller regions such that congestion homogeneity is maintained for each region which can be described by a low-scatter MFD (Ji and Geroliminis, 2012; Lopez et al., 2017; Saeedmanesh and Geroliminis, 2017, 2016).

Well-defined MFDs enable low-complexity modeling of traffic dynamics by focusing on aggregate vehicular movements within and across homogeneous regions. This elegant modeling paradigm has led to the development of numerous regional level control schemes, e.g., congestion pricing (Daganzo and Lehe, 2015; Geroliminis and Levinson, 2009; Li and Ramezani, 2022; Zheng et al., 2012), route guidance (Menelaou et al., 2021; Sirmatel and Geroliminis, 2018; Yildirimoglu et al., 2015), street network and sustainable transit system designs (Amirgholy et al., 2017; DePrator et al., 2017; Gayah and Daganzo, 2012; Gayah et al., 2014; Ortigosa et al., 2017), and others. The most extensively studied control application utilizing the MFDs is perimeter metering control (PMC), which entails regulating the inter-regional vehicle transfer flows using traffic signals residing on the boundaries of neighboring regions. By distributing vehicle presence between distinct regions, PMC aims at maximizing the network throughput i.e., the cumulative trip completion. The first PMC examination was presented in (Daganzo, 2007) for a single region, which formulated the aggregate traffic dynamics modeling using MFDs and proposed the optimal Bang-Bang control policy to manage congestion within the region. Similar approaches have been adopted in (Csikós et al., 2017; Haddad, 2017a), where optimal control and optimization theories

were used to derive model-based or analytical solutions to conduct perimeter control for single-region networks. More commonly, the proportional-integral (PI) type feedback controllers have been designed for single-region perimeter control, e.g., for networks with and without time-delay (Keyvan-Ekbatani et al., 2012, 2015a), with a reduced operational MFD (Keyvan-Ekbatani et al., 2013) or an uncertain MFD (Haddad and Shraiber, 2014). Perimeter control for two-region networks, as first formulated in (Haddad and Geroliminis, 2012), has also attracted substantial research interests over the years. For example, analytical and data-driven approaches have been adopted to design solution schemes (Aalipour et al., 2019; Geroliminis et al., 2013; Haddad, 2017b; Su et al., 2020; Zhou and Gayah, 2021), while stability and modeling uncertainty are examined in (Haddad, 2015; Li et al., 2021; Mohajerpoor et al., 2020; Sirmatel and Geroliminis, 2021; Zhong et al., 2018a).

Another line of PMC research pertains to the efficient operations of traffic flows in a multi-region setting (i.e., for urban networks with more than two regions). Early endeavors in this vein include (Aboudolas and Geroliminis, 2013; Haddad et al., 2013), where the traffic dynamics are formulated for a multi-reservoir and a mixed network. In these efforts, the receiving capacity constraint was neglected; however, this was later rigorously integrated in (Ramezani et al., 2015), which proposed a region-based and subregion-based MFD models. These models are subsequently adopted in (Ren et al., 2020; Sirmatel and Geroliminis, 2018; Yildirimoglu et al., 2018, 2015) to devise path assignment, route guidance, and perimeter control strategies. To further enhance the multi-region traffic dynamics, numerous works have been conducted to consider: boundary queue dynamics (Li et al., 2021; Ni and Cassidy, 2020; Sirmatel et al., 2021), time-delay effects (Haddad and Zheng, 2020), demand stochasticity (Zhong et al., 2018b), trade-off between fairness and efficiency (Moshahedi and Kattan, 2023), parameter uncertainty in MFDs (Haddad and Mirkin, 2017), and others. It is worth noting that, for large-scale multi-region urban networks, traffic dynamics modeling with microscopic approaches becomes increasingly difficult, which manifests the advantage of MFD-based aggregate modeling.

The multi-region PMC problem formulated with these dynamics embodies great potential for city-level traffic management, for which various solution meth+ods have been proposed in the literature. Examples include linear quadratic regulator (Aboudolas and Geroliminis, 2013; Ni and Cassidy, 2020), PI controller (Keyvan-Ekbatani et al., 2015b), model predictive control (Ramezani et al., 2015; Sirmatel and Geroliminis, 2018), model-free adaptive control (Lei et al., 2019; Ren et al., 2020) and reinforcement learning (Chen et al., 2022). Importantly, most solution methods are heavily dependent on knowledge of the environment dynamics, whereas such information is often difficult to acquire in the first place. Additionally, the accuracy of such obtained information is largely prone to estimation errors due to multivaluedness, instability, and hysteresis phenomena that are common in real networks (Daganzo et al., 2011; Gayah and Daganzo, 2011; Mahmassani et al., 2013; Mazloumian et al., 2010). (Lei et al., 2019) and (Ren et al., 2020) are two pioneering works that proposed data-driven and model-free solution schemes, yet the critical accumulation is still explicitly blended into the controller designs. In contrast, (Chen et al., 2022) proposed a truly model-free controller based upon integral reinforcement learning that is also grounded in control theory. While impressive, the devised controller can only conduct perimeter control for relatively small urban networks where drivers do not need to route themselves between the origin and destination regions. As such, the controller may fail to work effectively for city-level urban networks. On this note, it needs to be pointed out that the two deep reinforcement learning-based agents in (Zhou and Gayah, 2021) cannot be directly transplanted here for multi-region perimeter

control either, and the reasons are twofold. First, both agents adopt a centralized control design, and in a multi-region problem setting the dimension of the action space will grow exponentially, which inhibits effective exploration and learning for the agents and thus invalidates their applicability for multi-region perimeter control. Second, the action space designs of both agents are not grounded by transportation theory and lack enough flexibility to cope with fast changing traffic conditions that are likely to arise in multi-region urban networks. Therefore, it is a research priority to develop more scalable model-free control schemes for multi-region perimeter control.

The present paper bridges this gap by proposing a scalable model-free scheme based upon multi-agent reinforcement learning that features centralized training with decentralized execution and value function decomposition. Moreover, the scheme adopts the Bang-Bang type action design, which was corroborated as the optimal action form for perimeter control problems (Aalipour et al., 2019; Daganzo, 2007; Ni and Cassidy, 2020). To demonstrate benefits of the proposed scheme, it is compared with the model predictive control (MPC) method in terms of control effectiveness, resilience to environment uncertainty, and transferability to unseen environments via numerical simulations in a large-scale urban network. It is worth highlighting that, such scalable model-free schemes are particularly helpful and prospective for city-level traffic management and may even constitute the building blocks for an intelligent transportation system in the future. Concretely, the scalable design elevates the applicability of such schemes on macroscopic traffic management from the regional-level to the city-level, which may later be combined with other macro- or micro-level control schemes to form a comprehensive traffic management framework. The model-free design, on the other hand, enables such schemes to learn an effective perimeter control policy from direct interactions with the network sans prior knowledge or detailed modeling of the network. While the abundance of online and archived traffic data might help with the dynamics modeling (e.g., by estimating the MFDs) or even the development of model-based approaches such as PI controller, these approaches may not be flexible or adaptable enough to cope with different traffic conditions without a learning-based component. Instead, these methods may have to formulate and solve a highly nonlinear program every time a new traffic condition is encountered, which is both data and computation intensive. Contrarily, for model-free schemes, these data could help calibrate their learning processes for them to be more adaptive to real-life traffic conditions without detailed modeling and formulation as well as complex solution procedures. Furthermore, note that the term "model-free" refers exclusively to the solution scheme design within which the traffic dynamics are not embedded, whereas the dynamics might still be required to construct the I/O date generator, as in (Lei et al., 2019; Ren et al., 2020). For more discussions on these aspects, the reader is referred to (Chen et al., 2022; Zhou and Gayah, 2021).

The remainder of this paper is outlined as follows. Section 2 provides the general traffic dynamics modeling for multi-region urban networks. Section 3 explains the proposed scheme in detail, and Section 4 presents the comprehensive experiment results. Finally, section 5 summarizes and concludes the paper. Before proceeding further, a list of all symbols and abbreviations used in this paper is compiled in Table 1 to improve readability for the audience, and each notation will also be explained at its first appearance.

Table 1. Symbols and abbreviations

Symbol	Meaning
R	The number of regions of an urban network

R_i	Region i
$\mathcal{R}$	An urban network with R regions
$f_i(\cdot)$	MFD function for region i
$\tilde{f}_i(\cdot)$	MFD function for region i in unseen environments with uncertainty
$n_i(t)$	Total accumulation in region i at time step t
n_{ii}, n_{ij}	Accumulations from region i to region i or j.
n_{ic}	The critical accumulation of region i
n_h^{jam}	The jam accumulation of region h
$\tilde{n}_{ij}$	Measured value of the accumulation with noise
q_{ii}, q_{ij}	Traffic demands from region i to region i or j
$\tilde{q}_{ij}$	Traffic demands in unseen environments with uncertainty
u_{ih}	Perimeter controller between regions i and h
u_{min}, u_{max}	Minimum and maximum values for perimeter controllers
M_{ii}	Internal transfer flow (i.e., exit flow) of region i
M_{ihj}	Transfer flow from region i to j via the next region h
$\widehat{M}_{ihj}(t)$	Capacity-restrained transfer flow from region i to j via the next region h
t_{ihj}	Travel time for vehicles from region i to j via the next region h
θ_{ihj}	Route choice term for vehicles from region i to j via the next region h
N_i	Neighboring regions of R_i
C_{ih}, C_{ih}^{max}	Boundary capacity between regions i and h and its maximum value
α	A parameter associated with the decrease of receiving capacity
$< \boldsymbol{S}, \boldsymbol{O}, \boldsymbol{U}, \boldsymbol{P}, \boldsymbol{r}, \boldsymbol{\pi}, \boldsymbol{\gamma}, \boldsymbol{\mathcal{N}} >$	A tuple that characterizes a Dec-POMDP
n	The number of local agents
$\mathcal{N} = \{1, \cdots, n\}$	A group of n local agents
a	An individual local agent
o_t^a, u_t^a	Observation and action of agent a at time step t
$s_t, \boldsymbol{u}_t, r_t$	State, joint action, and reward at time step t
Δt	The duration of a time step
$\mathcal{P}(s_{t+1}\|s_t, \boldsymbol{u}_t)$	The transition dynamics of the Dec-POMDP
$\pi^a(\cdot)$	The acting policy of agent a
γ	The discount factor
G_t	The return of an episode (i.e., the control period)
T	The total number of time steps
$Q(\cdot)$	Action (Q) value of state-action pairs
κ	The learning rate
$Q(:,:;\theta^Q)$	The shared agent network with parameters θ^Q
ϵ	Probability to take a random action for exploration
$m(:,:;\theta^m)$	Mixing network with parameters θ^m
Y_t	Learning targets
$\mathcal{L}(\cdot)$	The loss function of the proposed scheme
b	Batch size of sampled transitions
B	The replay buffer size
I	The total number of iterations
G	The total number of generators
$\mathbb{N}(\cdot)$	Normal distribution
δ	Standard deviation of measurement noise
$\omega(t)$	Random error of the MFDs in unseen environments
$\mathbb{U}(-\lambda, \lambda)$	A uniform distribution with parameter λ

$\upsilon(t)$	Random error of the traffic demands in unseen environments
σ	Standard deviation of the random error in unencountered demands

Abbreviation	Meaning
MFD	Macroscopic fundamental diagram
PMC	Perimeter metering control
PI	Proportional-integral
MPC	Model predictive control
CTC	Cumulative trip completion
MARL	Multi-agent reinforcement learning
Dec-POMDP	Decentralized partially observable Markov decision process
DQN	Deep Q-Networks
CTDE	Centralized training with decentralized execution
MR-RL	Multi-Region Reinforcement Learning, i.e., the proposed scheme
NC	No control
EE	Estimation error (of the regional accumulations)

2. TRAFFIC DYNAMICS OF MULTI-REGION URBAN NETWORKS

The general traffic dynamics for an R-region urban network are introduced here. An illustration of a network with seven regions (i.e., $R = 7$) is presented in Fig. 1 but note the traffic dynamics are applicable to networks with both more and fewer regions. Each region in the R-region network is assumed to be homogenous in terms of congestion distribution; however, if this assumption does not hold, network partitioning can be applied to maintain homogeneity (Ji and Geroliminis, 2012; Saeedmanesh and Geroliminis, 2017, 2016). As such, a well-defined MFD $f_i(n_i(t))$ that relates trip completion rate to the regional accumulation $n_i(t)$ could be used to model each region. Note further that, the notion of MFD is used interchangeably with the notion of network exit function herein, as consistent with the convention of perimeter control related studies (Aboudolas and Geroliminis, 2013; Chen et al., 2022; Sirmatel and Geroliminis, 2018; Su et al., 2020; Zhou and Gayah, 2021). The dynamic evolution of accumulations in region i can be expressed as follows (Ramezani et al., 2015; Yildirimoglu et al., 2015):

$$n_i(t) = \sum_{j \in \mathcal{R}} n_{ij}(t) \tag{1}$$

$$\dot{n}_{ii}(t) = q_{ii}(t) - M_{ii}(t) + \sum_{h \in N_i} u_{hi}(t) \cdot M_{hii}(t) \tag{2}$$

$$\dot{n}_{ij}(t) = q_{ij}(t) + \sum_{h \in N_i; h \neq j} u_{hi}(t) \cdot M_{hij}(t) - \sum_{h \in N_i} u_{ih}(t) \cdot M_{ihj}(t) \tag{3}$$

where $\mathcal{R}$ denotes the network with $\mathcal{R} = \{1,2,\cdots,R\}$, n_{ij} and q_{ij} are respectively the number of vehicles and traffic demands in R_i destined for R_j, and n_{ii} and q_{ii} are defined similarly. u_{ih} is the perimeter controller (bounded by $[u_{min}, u_{max}]$ with $0 \leq u_{min} < u_{max} \leq 1$) that specifies the allowable ratio of transfer flow from R_i to R_h (see the dash lines in Fig. 1), with h belonging to

the neighboring regions of R_i, N_i. $M_{ihj}(t)$ represents the transfer flow from R_i to R_j via the next region h, while $M_{ii}(t)$ is the exit flow of region i. These two terms are calculated by:

$$M_{ihj}(t) = \theta_{ihj}(t) \cdot \frac{n_{ij}(t)}{n_i(t)} \cdot f_i\big(n_i(t)\big), i \neq j, h \in N_i \tag{4}$$

$$M_{ii}(t) = \frac{n_{ii}(t)}{n_i(t)} \cdot f_i\big(n_i(t)\big) \tag{5}$$

where $\theta_{ihj}(t) \in [0,1]$ denotes the route choice that expresses the ratio of transfer flows from R_i to R_j utilizing the next immediate region h; hence $\sum_{h \in N_i} \theta_{ihj}(t) = 1$ (see again Fig. 1 the θ_{ihj} terms for R_5). In this paper, the route choice term is inversely related to the travel time of paths utilizing R_h. Concretely, a predefined set of shortest paths connecting regions i and j is obtained using the Dijkstra's algorithm. The travel times of these shortest paths t_{ihj} are then calculated so as to compute the route choice via a Softmax operation, i.e., $\theta_{ihj} = \exp\big(-t_{ihj}\big) / \sum_{k \in N_i} \exp\,(-t_{ikj})$. Note that, the Softmax operation is executed for all control methods employed in this work, hence they all share the same route choice modeling process as well as environment dynamics. As such, a fair comparison of all methods can be realized to evaluate their respective control efficacy.

The receiving capacity of regions with high accumulations might be insufficient to contain all inflow vehicles, thus restraining the full penetration of transfer flows. As such, the capacity-restrained transfer flows $\widehat{M}_{ihj}(t)$ are defined as (Ramezani et al., 2015; Yildirimoglu et al., 2015):

$$\widehat{M}_{ihj}(t) = \min\left(M_{ihj}(t), C_{ih}\big(n_h(t)\big) \cdot \frac{M_{ihj}(t)}{\sum_{k \in \mathcal{R}, k \neq i} M_{ihk}(t)}\right) \tag{6}$$

where $C_{ih}\big(n_h(t)\big)$ is the boundary capacity between R_i and R_h and is a function of $n_h(t)$ as in:

$$C_{ih}\big(n_h(t)\big) = \begin{cases} C_{ih}^{max}, & 0 \leq n_h(t) \leq \alpha \cdot n_h^{jam} \\ \dfrac{C_{ih}^{max}}{1-\alpha} \cdot \left(1 - \dfrac{n_h(t)}{n_h^{jam}}\right), & \alpha \cdot n_h^{jam} \leq n_h(t) \leq n_h^{jam} \end{cases} \tag{7}$$

where C_{ih}^{max} is the maximum boundary capacity between region i and h, n_h^{jam} is the accumulation value of region h where gridlock arises, and $\alpha \in (0,1)$ is a parameter that signals the decrease of receiving capacity with the increase of accumulation. Note that, it is customary to model large-scale urban networks using the MFD-based traffic dynamics presented in this section, as widely seen in the existing literature. For example, see (Ramezani et al., 2015; Yildirimoglu et al., 2015) for theoretical analyses of these dynamics as well as control scheme designs. Additional control applications can also be found in (Genser and Kouvelas, 2022) for congestion pricing, (Ren et al., 2020) for perimeter control, (Sirmatel and Geroliminis, 2018; Yildirimoglu et al., 2018) for integrated route guidance, and others. The authors thus do not repeat the discussions herein.

With these multi-region traffic dynamics, different techniques can be utilized for perimeter metering control, and their performances are evaluated in terms of the control objective, i.e., to maximize the cumulative trip completion (CTC) of the network.

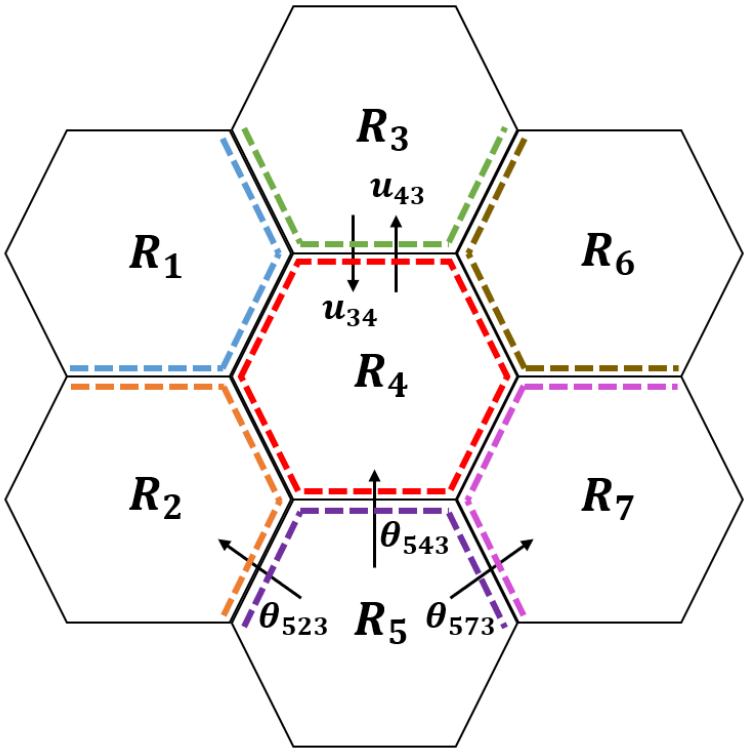


Fig. 1. A seven-region urban network. The dash lines represent the perimeter controllers.

3. METHODOLOGY

This section first reformulates the multi-region perimeter control problem in the context of multi-agent reinforcement learning (MARL). Then detailed explanations of the proposed scheme are provided, as well as its formalization and implementation details. It should be pointed out that, the proposed scheme (as well as the traffic dynamics presented in the previous section) is applicable to general multi-region networks with any number of regions. However, for realistic considerations, the proposed scheme will be evaluated in a seven-region network that has also been examined in (Sirmatel and Geroliminis, 2018). This plan of action, i.e., to propose a generic data-driven method for perimeter metering control and demonstrate it on a realistic network, has been widely adopted in the literature (Chen et al., 2022; Lei et al., 2019; Ren et al., 2020; Su et al., 2020).

3.1 Problem reformulation

The multi-region perimeter control problem can be viewed as a cooperative multi-agent task where a group of n agents ($\mathcal{N} = \{1, \cdots, n\}$) learn collaboratively to achieve a common control objective via individualized interactions with the same environment. Specifically, at time step t, each agent $a \in \mathcal{N}$ receives an individual local observation o_t^a from the environment and chooses an action u_t^a based on the observation, thus forming a joint action $\boldsymbol{u}_t$. The environment implements the joint action and transitions to a new state at the next time step, while in the meantime returning a reward r_{t+1} back to the agents. In this work, each agent is supposed to regulate two inter-regional vehicle movements by selecting values for a pair of perimeter controller on a regional boundary. For

instance, a certain agent needs to determine proper values for the controllers u_{34}, u_{43} that reside on the boundary between R_3 and R_4 (see Fig. 1). As such, the number of required agents could increase rapidly with the number of regions considered, depending on the network configurations. Formally, the multi-region perimeter control problem is presented as a decentralized partially observable Markov decision process (Dec-POMDP) defined by a tuple $< \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{O}}, \boldsymbol{\mathcal{U}}, \boldsymbol{\mathcal{P}}, \boldsymbol{r}, \boldsymbol{\pi}, \boldsymbol{\gamma}, \boldsymbol{\mathcal{N}} >$.

- **State space, $\boldsymbol{\mathcal{S}}$**, and **observation space, $\boldsymbol{\mathcal{O}}$**. The state of the environment contains the global information about the entire network. However, due to partial observability that is common in multi-agent tasks, the agents can only observe local instances of the state and act based on the observations. In this work, the state s_t consists of all regional accumulations, traffic demands, and a binary congestion indicator that denotes whether the regions are congested or not. Since each agent selects actions for a pair of neighboring regions, the observation o_t^a includes only information about this pair of regions, i.e., the accumulations and traffic demands concerning the two regions, together with the related congestion indicator. The regional accumulations can be obtained from the environment with relative ease, e.g., with proper instrumentation like loop detectors. These detectors could also help evaluate the congestion condition of the regions. The traffic demands, on the other hand, can be readily estimated from historical observations. Note that there might be measurement or estimation errors in the state and/or observation information, and these errors will be comprehensively examined in the experiments; see Section 4.3.

- **Action space**, $\boldsymbol{\mathcal{U}}$. The optimal policy for perimeter control problems has been shown in the form of Bang-Bang in the literature (Aalipour et al., 2019; Daganzo, 2007; Ni and Cassidy, 2020). Control policies that build upon the Bang-Bang form will alternate the perimeter controller between the minimum and maximum values, depending on the congestion status of the regions. Note that, different policies exist that are based on the Bang-Bang form, for example the IOA approach (Aalipour et al., 2019) and greedy control. In the present work, a control scheme will be devised whose policy adopts the Bang-Bang form to leverage its optimality. Each agent chooses either u_{min} or u_{max} for the two perimeter controllers and thus will have a 4-dimensional action space (two options for the two controllers). After selection, the actions are held constant for the duration of a time step, Δt.

- **Transition dynamics**, $\boldsymbol{\mathcal{P}}$. The selected actions of the individual agents form a joint action, $\boldsymbol{u}_t$, which is executed in the environment and leads a transition to a new state, according to the dynamics $\mathcal{P}(s_{t+1}|s_t, \boldsymbol{u}_t)$: $\boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{U}} \rightarrow \boldsymbol{\mathcal{S}}$. Note that, the proposed scheme is model-free and thus internalizes such dynamics through the learning process without explicit modeling.

- **Reward function**, $\boldsymbol{r}$. After executing the joint action, the environment returns a real-time scalar reward back to the agents as a quality assessment. The reward $r(s_t, \boldsymbol{u}_t)$ helps guide the agents to achieve the control objective, i.e., to maximize the cumulative trip completion; and therefore, it is defined as the trip completion in a time step. To facilitate more effective learning, the reward is normalized into $[0, 1]$ by a large constant (Henderson et al., 2017). Further, a large negative penalty is appended to the reward if undesirable situations (e.g., gridlock) should arise as a result of the selected actions. Note that the reward defined above is provided for all agents to evaluate their collective control gains, thus avoiding the need to explicitly deduce their individual contributions, a problem known as multi-agent credit assignment (Chang et al., 2003) and often challenging in a cooperative task.

- **Policy, $\boldsymbol{\pi}$**, and **discount factor**, $\boldsymbol{\gamma}$. The agents select actions for the perimeter controllers based upon the local observation o_t^a, according to the policy $\pi^a(u^a|o^a)$. To differentiate immediate rewards from delayed ones, a discount factor $\gamma \in [0,1]$ is utilized, which also implicitly determines the number of future time steps accounted. Intuitively, $\gamma = 1$ implies equal importance for all rewards regardless of when the rewards are obtained (i.e., infinite future steps are considered); on the other extreme, $\gamma = 0$ means that only the immediate reward matters. The discount factor is a user-defined hyperparameter with its value often derived via a tuning procedure to properly balance the importance of short-term and long-term rewards. Collectively, the agents learn via trial and error to maximize the expected total discounted reward, i.e., the return, as calculated by $G_t = \sum_{\tau=t}^{T} \gamma^{\tau-t} r_{\tau+1}$ where T is the total number of steps in the control period. With the above reward definition, maximizing the return amounts to maximizing the cumulative trip completion for the control period.

3.2 Algorithm

This section first introduces a canonical single-agent deep reinforcement learning method and then presents an overview of multi-agent reinforcement learning, both of which help provide theoretical background for the proposed scheme to be explained subsequently. Note that, both algorithms to be introduced are inherently value-based, and this decision has two major considerations. First, the previous efforts of the authors suggest that policy-based methods can only generate control actions that change gradually across consecutive time steps, which are unable to cope with the complex changeable traffic conditions in multi-region urban networks. Second, value-based methods can facilitate adopting the previously mentioned action space design (i.e., the Bang-Bang form) that is grounded in transportation theory; thus, this type of method is more prospective than the policy-based counterparts.

3.2.1 Double Deep Q Networks (Double DQN)

As a foundational reinforcement learning technique for discrete control tasks, Q-learning (Watkins and Dayan, 1992) has received sustained interests over the years. Using a tabular form, it stores the long-term quality measurements of distinct state-action pairs, i.e., the Q value $Q(s_t, u_t)$ which denotes the expected return from the environment after taking action u_t at state s_t. During the learning process, the Q values are updated with each visit to a state-action pair, according to:

$$Q(s_t, u_t) \leftarrow Q(s_t, u_t) + \kappa \cdot \left(r_{t+1} + \gamma \cdot \max_u Q(s_{t+1}, u) - Q(s_t, u_t) \right) \tag{8}$$

where κ is the learning rate. With sufficient learning updates, the Q values tend towards invariant, and the final learned policy can be derived in a greedy manner with respect to the Q values, i.e., $u_t = \pi(s_t) = \arg\max_u Q(s_t, u)$.

With a simple update rule and a tabular structure, Q-learning has attracted research interests both on the engineering applications and theoretical investigations; see for example (Araghi et al., 2013; Jin et al., 2018). However, the tabular form limits its applicability to large problems that feature an abundance of state-action pairs. To mitigate this issue, research efforts have long been performed on value function approximation and its stability analysis (Sutton and Barto, 2018;

Tsitsiklis and Roy, 1997; van Hasselt et al., 2018), with the first success presented in the seminal Deep Q-Networks (DQN) algorithm (Mnih et al., 2015). This work has demonstrated the potential of deep reinforcement learning to realize human-level control performances and has since inspired the development of more advanced learning techniques (Hessel et al., 2017; Lillicrap et al., 2016; Schaul et al., 2016; van Hasselt et al., 2015; Wang et al., 2015). Despite its success, however, the DQN method is prone to overestimation of the Q values as the quantity used for action selection (i.e., $\max_u Q(s_t, u)$ as in Eq. (8)) is also used to evaluate the action. In double Q-learning (van Hasselt, 2010), separate sets of values are used for action selection and evaluation, and this has been shown helpful to alleviate the overestimation issue (van Hasselt et al., 2015). In the latter reference, an improved algorithm named Double DQN is proposed, which revises the learning target of DQN by using the Q-network for action selection and target network for evaluation, as follows:

$$Y_t = r_{t+1} + \gamma Q\left(s_{t+1}, \arg\max_u Q(s_{t+1}, u; \boldsymbol{\theta}_t); \boldsymbol{\theta}_t^-\right) \quad (9)$$

where $Q(\cdot,\cdot;\boldsymbol{\theta}_t)$ and $Q(\cdot,\cdot;\boldsymbol{\theta}_t^-)$ respectively represent the Q- and target networks. Note that, the target network is a periodic copy of the Q-network, and its utilization helps provide relatively static learning targets which is beneficial to the learning stability. Note further that the notations θ with subscript t (θ_t, θ_t^-) refer to the weight and bias parameters of the neural networks in this and subsequent sections, which is different from those with subscripts that denote the regions (i.e., θ_{ihj} in Eq. (4)). The latter is the route choice term which will not be optimized by the method (but will be updated with travel times). While training, samples (i.e., state-action-reward pairs) are collected to construct learning targets (according to Eq. (9)) for the Double DQN, which performs its learning by adjusting the Q-network predictions towards these targets.

3.2.2 Multi-Agent Reinforcement Learning (MARL)

This section provides an overview of multi-agent reinforcement learning (MARL), which presents the evolution of various learning paradigms that lays the foundation for the learning algorithm adopted in this work. This section may be skipped without loss of continuity.

The success of single-agent reinforcement learning has significantly boosted its extension to multi-agent systems. However, directly applying single-agent techniques to multi-agent tasks is generally not feasible, and the reasons are multifold. First and foremost, single-agent methods face the curse of dimensionality as the joint action space increases exponentially with the number of agents, which renders it difficult to fully explore the solution space. In addition, the expanded action space also raises scalability concerns for estimating the joint Q-value, thus hindering the acquisition of the optimal policy. Second, in multi-agent systems, the global state information is often not available to the single-agent methods during action taking, which thus impedes obtaining the joint action and further the estimation of the joint Q-values. Moreover, the multi-agent system becomes vulnerable when controlled by single-agent methods as even slight information loss could result in drastically undesirable actions that disrupt its normal operation.

The most intuitive approach to address the aforementioned issues is to utilize a group of independent agents for control where each agent acts solely based on its local observations without regard to the behaviors of other agents. In this manner, the other agents are considered as part of the environment and single-agent training procedure is readily applicable. The initial formalization

of this idea was presented in independent Q-learning (Tan, 1993), with extensions to actor-critic methods (Foerster et al., 2017), distributed learning (Lauer and Riedmiller, 2000), and others. This type of algorithm is fully scalable to large problems as each agent acts locally and requires minimal information; for this reason, training each agent can be done efficiently. However, these methods may encounter convergence issues due to non-stationarity (Choi et al., 1999), which refers to the phenomenon that the actions taken by one agent could impact the state and rewards received by the other agents. In other words, the environment is dynamic rather than static for each agent, and this invalidates the Markov property and the naïve use of experience replay (Lin, 1992) that are critical to single-agent methods. Further, independent approaches lack communication between the agents, thus making it difficult to achieve coordination between the learned policies.

Another approach to improving scalability of single-agent methods to multi-agent systems is parameter sharing (Chu and Ye, 2017; Gupta et al., 2017; Terry et al., 2020), where numerous agents are adopted for control with shared network parameters. In this setting, all agents share the same policy, but each agent can produce specialized actions with different local observations that are often appended with agent identification (Gupta et al., 2017; Terry et al., 2020). During the training process, the samples collected by each agent are pooled together to update the shared network, which is beneficial to scalability as the size of the shared network does not expand with the number of agents. In addition, these methods are much more efficient than the independent approaches as only a single set of learning parameters needs to be updated for all agents. However, parameter sharing methods suffer from lack of theoretical support and are still susceptible to non-stationarity. Moreover, most effective application of parameter sharing requires the definition of local rewards, which is a complex multi-agent credit assignment problem. Further, indiscriminate sharing of parameters for all agents, as typically implemented in the literature, has been shown detrimental to the final convergence and control performances (Christianos et al., 2021).

As suggested in (Terry et al., 2020), increased centralization during learning helps mitigate the non-stationarity issue, whereas decentralization is required during execution as the agents do not have access to the global information and can only act upon the local observations. Fortunately, extra global state information can often be utilized to help train the decentralized policies, which yields the paradigm of centralized training with decentralized execution (CTDE, (Oliehoek et al., 2008)) that is considered the most common or even default paradigm of MARL. Concretely, this paradigm adopts full centralization conditioning on the global state to resolve the non-stationarity issue and decentralization conditioning on local observations to ensure scalable action taking and to mitigate partial observability. Representative works in this vein include MADDPG (Lowe et al., 2017) and COMA (Foerster et al., 2017). MADDPG extends the established single-agent Deep Deterministic Policy Gradient (Lillicrap et al., 2016) algorithm to the multi-agent setting, and by maintaining a centralized critic for each actor, it is compatible with cooperative, competitive, or even mixed scenarios. COMA, on the other hand, has a single centralized critic for all decentralized actors. Using a counterfactual baseline, it can explicitly address the multi-agent credit assignment. There are also some subsequent improvements to this paradigm, e.g., with the attention mechanism (Iqbal and Sha, 2019) or recursive reasoning (Wen et al., 2019).

Despite notable experimental results, the CTDE paradigm has a major scalability limitation due to fully centralized training, which is exacerbated with multiple centralized critics as in (Lowe et al., 2017). In between full centralization with scalability constraints and full decentralization with non-stationarity concerns, value function decomposition has been proposed (Koller and Parr,

1999). Specifically, these methods factorize the centralized Q value as a parameterized function of the local Q values that are estimated by the agents conditioning on the local observations and actions. Based on the local Q values, decentralized policies can be derived in a greedy manner. The centralized Q value is used to calculate the temporal difference error (Sutton and Barto, 2018), for which the gradient can be computed and used to update the network parameters. Importantly, factorization of the centralized Q value ensures scalability as its estimation does not require the joint action information. Moreover, these approaches can implicitly address multi-agent credit assignment (Wang et al., 2021) and thus obtain more coordinated control policies. For further discussions on how value factorization may enhance scalable learning in the CTDE paradigm, the reader is referred to (Peng et al., 2021; Wang et al., 2021).

Pioneering works in the vein of value factorization include value decomposition networks (Sunehag et al., 2018) and QMIX (Rashid et al., 2018). Value decomposition networks present a linear factorization of the centralized Q value under the CTDE paradigm. While effective, this form of factorization lacks enough representational complexity for more complicated tasks. In comparison, the QMIX method decomposes the joint Q value as a nonlinear but monotonic composition of the local Q values, and this decomposition has been widely adopted in later efforts, e.g., (Peng et al., 2021). Other novel improvements over these methods have also been proposed; see (Rashid et al., 2020; Son et al., 2019; Wang et al., 2021). Additionally, see (Hernandez-Leal et al., 2018; OroojlooyJadid and Hajinezhad, 2019) for more discussions on these methods as well as more reviews of multi-agent reinforcement learning. In this work, value decomposition methods will be adopted to devise the learning algorithm for the proposed control scheme.

3.2.3 Reinforcement Learning controller design for Multi-Region perimeter control (MR-RL)

The multi-region perimeter control problem considered in this paper is a fully cooperative multi-agent task where all agents work collaboratively to achieve the highest cumulative trip completion. In this paper, the QMIX method is adopted as the learning algorithm for the proposed scheme, as denoted by MR-RL that stands for Multi-Region Reinforcement Learning. In particular, the proposed MR-RL scheme features a group of decentralized agents, which act upon their local observations and estimate the local Q values, a mixing network, which provides the collective estimate of the centralized Q value from the local Q values, and separate hypernetworks, which generate weights for the parameterized mixing network. Moreover, the MR-RL integrates into its design the Double DQN update rule and the Ape-X distributed learning architecture (Horgan et al., 2018). The learning algorithm for the proposed MR-RL scheme is shown in Fig. 2, and in the following, these building components are explained in greater detail. Before proceeding further, please note that the "networks" in this section (e.g., the mixing network and hypernetworks) are not related to the traffic networks mentioned previously in the MFD-based dynamics modeling. Instead, they refer to neural networks in the context of deep learning and reinforcement learning. Further note that, compared with (Zhou and Gayah, 2021) which also adopts deep reinforcement learning methods for perimeter control, the scheme presented in this work differs in a variety of ways. Most importantly, the training and execution processes are decoupled utilizing the CTDE paradigm, and value function decomposition approaches are employed for enhanced scalability of the MR-RL scheme. These components are critical to the effective perimeter control of a large-scale urban network with involved traffic dynamics, which is otherwise not achievable using the single-agent methods in (Zhou and Gayah, 2021).

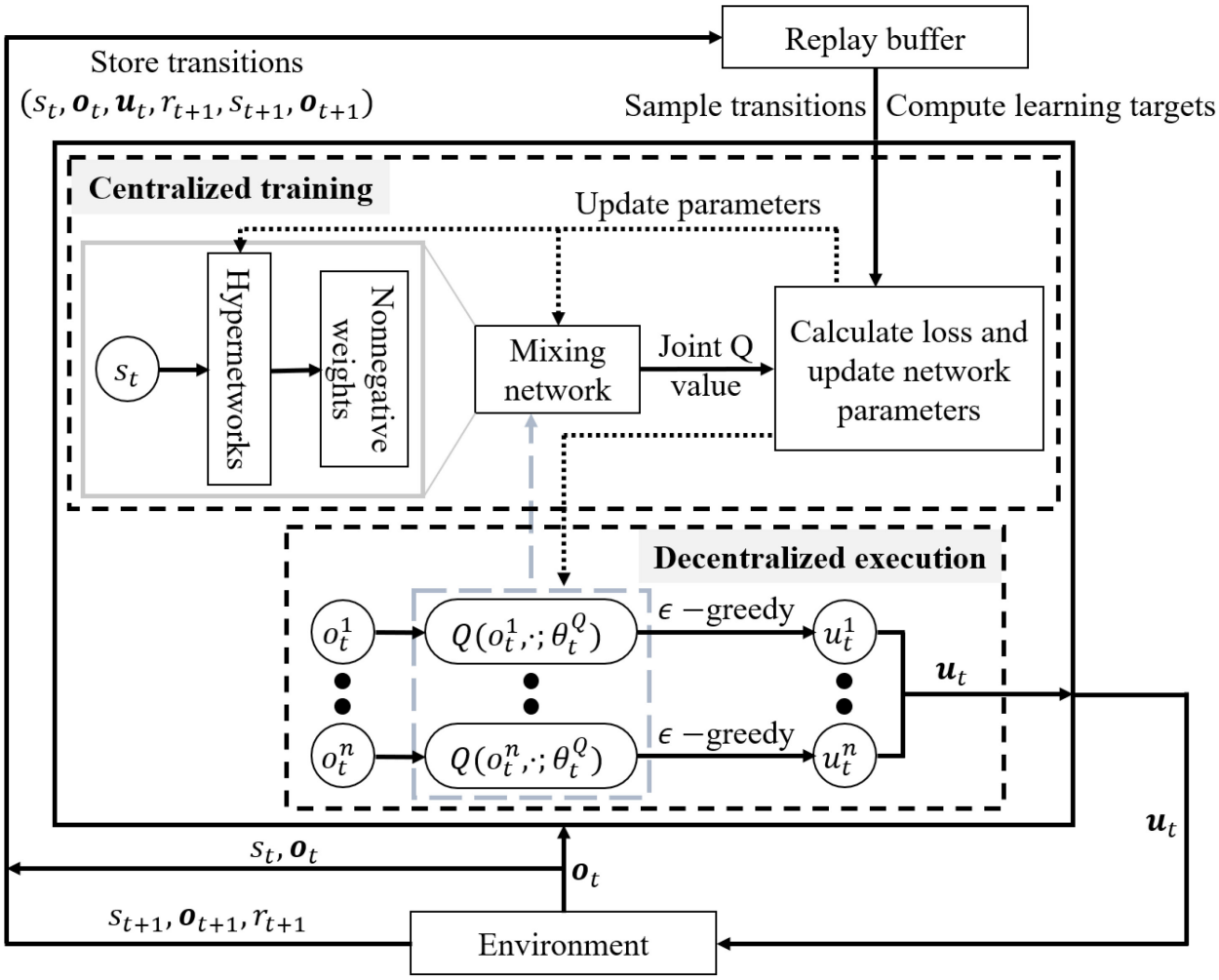


Fig. 2. A diagram of the learning algorithm for the MR-RL scheme. Inputs needed for the scheme to select perimeter control actions only include the local observations $\boldsymbol{o_t}$ (decentralized execution). Inputs needed to train the scheme include the states and local observations at the current and next steps, as well the joint action and reward, i.e., $(\boldsymbol{s_t}, \boldsymbol{o_t}, \boldsymbol{u_t}, \boldsymbol{r_{t+1}}, \boldsymbol{s_{t+1}}, \boldsymbol{o_{t+1}})$ (centralized training).

The MR-RL scheme holds a group of agents for multi-region perimeter control, and each agent is constructed as a multi-layer perceptron, a structure widely used in the literature (Horgan et al., 2018; Lillicrap et al., 2016; Rashid et al., 2018). To improve training efficiency, parameters of the agent network are shared. Hence, the agents with shared parameters can be represented as $Q(o^a, u^a; \theta^Q)$, where θ^Q represents the weight and bias of the agent neural networks. Each agent a receives as input the local observation o^a and estimates the 4-dimensional local Q values for the two associated perimeter controllers (each controller has two options, u_{min} and u_{max}); see the box titled "Decentralized execution" in Fig. 2. The local action can then be derived with the ϵ −greedy strategy regarding the local Q values, i.e., the greedy action $\arg\max_{u^a} Q(o^a, u^a; \theta^Q)$ is chosen with probability $1 - \epsilon$ and a random action otherwise. To better balance exploration and exploitation, the ϵ value is decayed through time, with the decay schedule to be presented shortly. It is worth reiterating that the local observation o^a includes a congestion indicator for a pair of neighboring

regions, as well as information about the accumulations and traffic demands. While obtaining the congestion indicator does require estimates of the critical accumulation, the critical accumulation itself (as well as the system traffic dynamics) is not embedded in the design of the proposed scheme. Instead, the scheme only acts upon the congestion indicator it receives from the environment, regardless of whether such information is accurate or not. Such a strategy, i.e., system dynamics not involved in the controller design, is called "model-free". See (Chen et al., 2022; Ren et al., 2020; Zhou and Gayah, 2021) for more discussions on this. Further, in practice, the congestion information can be readily estimated with proper instrumentation (e.g., loop detectors), and in this work its inaccuracies will be systematically investigated in Section 4.3.

The mixing network, as denoted by $m(\cdot)$, adopts a feed-forward neural network structure and outputs the joint Q value using the local Q values estimated by the local agents; see the box titled "Centralized training" in Fig. 2. This network is central to the notion of value decomposition. The QMIX algorithm uses non-negative weights for the mixing network to realize monotonic value factorization, and separate hypernetworks are exploited to produce such weights. Specifically, the hypernetworks take the global state s_t as input and generate weights for the mixing network with non-negativity ensured by an absolute activation function. The hypernetworks also create biases for the mixing network, but these are not restricted to be non-negative.

The Double DQN update rule, along with the QMIX type value decomposition, is used to construct learning targets for the proposed MR-RL scheme, as follows:

$$Y_t = r_{t+1} + \gamma \cdot m\left(s_{t+1}, \left\{Q\left(o_{t+1}^a, \arg\max_{u^a} Q\left(o_{t+1}^a, u^a; \theta_t^Q\right); \theta_t^{Q-}\right)\right\}_{a=1}^{n}; \theta_t^{m-}\right) \tag{10}$$

where $\arg\max Q\left(\cdot,\cdot;\theta_t^Q\right)$ is the local action selection using the shared agent network, $Q(\cdot,\cdot;\theta_t^{Q-})$ is the action evaluation with the target agent network, and $m(\cdot,\cdot;\theta_t^{m-})$ represents the target mixing network. Note that, inputs of the mixing network include the global state for the hypernetworks to generate non-negative weights, and $\theta_t^m(\theta_t^{m-})$ also includes parameters for the hypernetworks. Therefore, the hypernetworks can be viewed as a component of the mixing network. The major distinction between this target and that of the Double DQN in Eq. (9) is the mixing network which involves a group of local Q values. Importantly though, this additional complexity significantly improves the scalability of reinforcement learning to larger multi-agent systems that is otherwise absent in single-agent methods. The parameters of the MR-RL scheme (i.e., weights and/or biases of the agent and mixing networks) can be updated by minimizing the following loss:

$$\mathcal{L}\left(\theta_t^Q, \theta_t^m\right) = \sum_{i=1}^{b}\left[Y_t^i - m\left(s_t^i, \left\{Q\left(o_t^{a,i}, u_t^{a,i}; \theta_t^Q\right)\right\}_{a=1}^{n}; \theta_t^m\right)\right]^2 \tag{11}$$

where b is the number of transitions sampled from the replay buffer used for updating the network parameters, Y_t^i is the learning target for the i-th transition, and $Q\left(o_t^{a,i}, u_t^{a,i}; \theta_t^Q\right)$ is the i-th local Q value estimated by agent a at observation $o_t^{a,i}$ and action $u_t^{a,i}$. Again, note that θ_t^Q and θ_t^m do not represent the route choice term, which will not be optimized (but will be updated with travel times).

The proposed MR-RL scheme is model-free in that it does not require a priori knowledge of the environment dynamics. Instead, it learns the control policy from pure interactions with the environment, and the interactions are stored in a replay buffer in the form of state-action-reward

pairs, i.e., the transitions in Fig. 2. The use of a replay buffer was initially presented in (Lin, 1992) and later consolidated in (Mnih et al., 2015) as a critical component of deep reinforcement learning. Specifically, the replay buffer is first utilized to store the collected transitions; then during training, minibatches of transitions are randomly sampled from the buffer to update the network parameters i.e., weights and/or biases of the shared agent network, hypernetworks, and the mixing network. The replay buffer has been shown helpful to stabilize the learning process as the random sampling helps remove correlations between the transitions. Further, to guarantee effective learning for the MR-RL scheme, the Ape-X distributed architecture (Horgan et al., 2018) is adopted. Concretely, the architecture maintains numerous instantiations of the environment in parallel, with which the MR-RL interacts to collect an increased number of transitions. These derived transitions are then pooled together in the replay buffer for future updates of the network parameters. With enough training updates, the final learned control strategy can be obtained by applying the greedy policy on the fully trained agent network, i.e., $u_t^a = \pi(o_t^a) = \arg\max_u Q(o_t^a, u; \theta_t^Q)$.

With these expositions, the proposed MR-RL scheme built with the learning algorithm and the Ape-X architecture is formalized in Algorithm 1. Note again, θ_t^m expresses the weights of the mixing network which include weights of the hypernetworks as a constituent element. In addition, the generator refers to the instantiated environment, i.e., a transition generator. By design, each generator will produce a complete sequence of state-action-reward pairs during the control period, which corresponds to an episode in the context of reinforcement learning.

Algorithm 1. Reinforcement Learning controller for Multi-Region perimeter control (MR-RL)

Randomly initialize shared agent network $\boldsymbol{\theta}_0^Q$ and mixing network $\boldsymbol{\theta}_0^m$ (hypernetworks included)
Initialize target agent and mixing networks $\boldsymbol{\theta}_0^{Q-} = \boldsymbol{\theta}_0^Q, \boldsymbol{\theta}_0^{m-} = \boldsymbol{\theta}_0^m$
Initailize replay buffer, buffer size B, sample size b, iteration number I, and genetaor number G
for *iter* = 1 **to** I **do**
  Compute the decayed ϵ value for $\epsilon-$greedy exploration
  for generator = 1 **to** G **do**
    Load the shared agent network $\boldsymbol{\theta}_{iter}^Q = \boldsymbol{\theta}_{iter-1}^Q$
    $s_0, \boldsymbol{o}_0 \leftarrow$ Environment.Reset()
    for $t = 1$ **to** T **do**
      $u_{t-1}^a = \arg\max_u Q(o_{t-1}^a, u; \theta_{iter}^Q)$ with probability $1-\epsilon$
        a random action with proability ϵ
      $\boldsymbol{u}_{t-1} = \{u_{t-1}^a\}_{a=1}^n$
      $(r_t, s_t, \boldsymbol{o}_t) \leftarrow$ Environment.Step$(s_{t-1}, \boldsymbol{o}_{t-1}, \boldsymbol{u}_{t-1})$
      Store $(s_{t-1}, \boldsymbol{o}_{t-1}, \boldsymbol{u}_{t-1}, r_t, s_t, \boldsymbol{o}_t)$ into the replay buffer
    end for
  end for
  if the number of stored transitions exceeds the buffer size B **then**
    Remove outdated transitions
  end if
  Training samples $\leftarrow$ a batch of b transitions randomly drawn from the replay buffer
  Periodically load target networks $\boldsymbol{\theta}_{iter}^{Q-} = \boldsymbol{\theta}_{iter-1}^Q, \boldsymbol{\theta}_{iter}^{m-} = \boldsymbol{\theta}_{iter-1}^m$
  $\boldsymbol{\theta}_{iter}^Q, \boldsymbol{\theta}_{iter}^m \leftarrow$ Update the network parameters by minimizing the loss as in Eq. (11)
end for

To conclude this section, implementation details of the MR-RL scheme are provided in the following[1]. First, on the collected transitions, reward clipping (Mnih et al., 2015) is not applied so that the rewards are roughly on the same scale as the centralized Q value (see Eq. (10)). This is helpful since otherwise the feedback signals from the rewards are either over- or under-weighted. In addition, only transitions with rewards > 0.1 are stored and later used to update the networks. This sample selection strategy is adopted so that the MR-RL can learn mostly from well-rewarded control actions. Also, this helps eliminate the undesirable learning updates that might be otherwise performed with negative-reward samples which feature either gridlock or invalid accumulations. Second, on the network architectures, the shared agent network is built with a 64-unit dense ReLU layer and an output 4-unit dense linear layer. The mixing network has the same structure as in the QMIX method (Rashid et al., 2018), and the hypernetworks only assume a single linear layer. The weights of all networks are randomly initialized according to a normal distribution with default parameterization. The target networks share the same structures as the original networks, whose weights are periodically used to update the weights of the former. Third, on the training procedure, gradient clipping (Goodfellow et al., 2016) is not employed as otherwise the learning updates would be nearly negligible. The learning updates are performed by the default-setting RMSprop

[1] Upon acceptance of this manuscript, the code will be available at: https://github.com/DongqinZhou/MR-RL

optimizer (Tieleman and Hinton, 2012), with the learning rate to be specified shortly. Critically, the network parameters are only updated if the update networks yield reduced loss (as in Eq. (11)). This helps avoid unwanted updates to the networks that may disrupt the training process and affect the subsequent transitions the MR-RL scheme encounters. Moreover, the learning process would subside if convergence were reached early, as indicated by nearly invariant control outcomes the MR-RL realizes. Fourth, on the computational software, all experiments considered in this paper are conducted on standalone Linux machines with Python 3.9 and Tensorflow 2.8.0. The external hardware settings (e.g., CPU/GPU capability, RAM) are not impactful to the final performances. Finally, the list of hyperparameters along with their values is presented in Table 2. The values are obtained via a random search of all candidate values, which improves the learning performances but does not cause overfitting to the scenarios specified. Contrarily, a systematic grid search of all hyperparameters will be extremely computationally intensive and leads to reduced transferability for the proposed scheme.

Table 2. List of hyperparameters and the selected values

Hyperparameter	Value	Description
Iteration number (I)	250	The number of training iterations
Generator number (G)	6	The number of environment instantiations to collect transitions
Replay buffer size (B)	10000	The storage capacity of the replay buffer
Sample size (b)	1000	The number of transitions sampled for network updates
Initial ϵ	0.90	The initial value of ϵ in $\epsilon-$ greedy exploration
ϵ decay	0.98	The exponential decay factor for the ϵ value
Final ϵ	0.01	The final value of ϵ in $\epsilon-$ greedy exploration
Update epoch	5	The times to update the network parameters at each iteration
Initial learning rate	0.003	The initial learning rate used by RMSprop for the network updates
Learning rate decay	0.95	The exponential learning rate decay factor at each iteration
Minimum learning rate	0.0001	The minimum learning rate used by RMSprop
Discount factor	0.8	The discount factor used to compute the learning targets (Eq. (10))
Target networks lifetime	10	The number of iterations to periodically update the target networks

4. EXPERIMENTS

In this section, six experiment scenarios with different types of uncertainties are considered and simulated on a seven-region urban network (see Fig. 1 for the configurations) to comprehensively evaluate the control effectiveness, resilience, and transferability of the MR-RL scheme, as detailed in Table 3. Note that, there are 24 perimeter controllers for 12 pairs of neighboring regions in the network (see again Fig. 1), hence 12 local agents are utilized.

It might be worth pointing out that, to the best knowledge of the authors, the seven-region network simulated here is the largest one that has ever been examined in perimeter control related works. Previous efforts that adopt similar dynamics modeling are either investigating a different control application (Yildirimoglu et al., 2018, 2015) or considering perimeter control only in smaller urban networks (Chen et al., 2022; Lei et al., 2019; Ramezani et al., 2015; Ren et al., 2020), with (Moshahedi and Kattan, 2023; Yildirimoglu et al., 2018) being the only exceptions that directly study perimeter control in seven-region networks. Note in particular that the perimeter metering control problem is formulated for two regions using the regional model in (Ramezani et

al., 2015) though in total 19 subregions are considered in the plant using the subregional model. Traffic management in large urban networks is inherently more challenging due to complex user behaviors such as routing and difficulties in optimization associated with significantly more control variables. Thus, such a large network could better demonstrate the advantage of model-free data-driven approaches over model-based ones. Also, a large network could better gauge the scalability as well as applicability of the proposed scheme to city-level traffic management. It is, however, surely interesting to see if the proposed scheme, as well as previous model-free data-driven approaches (Chen et al., 2022; Lei et al., 2019; Ren et al., 2020), could be applied to urban networks with even more regions, yet this is a question that cannot be answered in the present work. The authors believe that a seven-region network is sufficiently representative of city-level traffic networks and hope to answer this question in an extension of this study.

Table 3. Descriptions of experiment scenarios

Scenario No.	Uncertainty types	Description
1	No uncertainty	A benchmark scenario to illustrate the learning processes and control effectiveness
2	Measurement noise	A scenario to test the resilience to noise in accumulation measurements due to potential sensor malfunction
3	Varying traffic demands	A scenario to test the resilience to temporally and spatially changeable traffic demands
4	Estimation errors	A scenario to test the resilience to inaccurate estimation of regional production and congestion information
5	MFDs and demands	A scenario to test transferability to unencountered environments with uncertainty in MFDs and demands
6	Accumulations	A scenario to test transferability to unencountered environments with uncertainty in accumulations

4.1 Experiment setup

In this work, a unit MFD consistent with the one observed in Yokohama (Geroliminis and Daganzo, 2008) is utilized, with critical and jam accumulations being respectively 8,240 veh and 34,000 veh (Gao and Gayah, 2018; Zhou and Gayah, 2021). Note that, the unit MFD assumes a piecewise functional form (linear for extreme congestion and third-order polynomial otherwise) rather than a solitary third-order polynomial form, the former of which renders the traffic dynamics to be more realistic (e.g., the trip completion drops to 0 at jam accumulation). For all experiments, each of the seven regions is modeled with a slightly scaled (within $\pm 10\%$) version of unit MFD, as similarly done in (Sirmatel and Geroliminis, 2018). In addition, the parameters for the boundary capacity constraints are set to $C_{ih}^{max} = 4.6$ veh/s and $\alpha = 0.48$; see Eq. (7).

The traffic demand profiles adopted for the numerical experiments are shown in Fig. 3. A two-hour control period is simulated with high inflows to region 4 (i.e., the "city center") and relatively small demands among the periphery regions. Note these demand profiles are intended to mimic traffic conditions during a morning peak, and in this period the traffic demands to peripherical regions are expected to be low. The adopted traffic demands might appear overly light at first glance, but in fact such demands could lead to a nearly gridlocked condition in region 4, as will be presented shortly. The duration of a time step is set as $\Delta t = 60s$, which is a realistic cycle

length for the signalized intersections on the regional boundaries that implement perimeter control. In addition, to account for more realistic implementation of perimeter control, the boundary values are $u_{min} = 0.1, u_{max} = 0.9$, indicating the transfer flows will neither be completely prohibited or accommodated. Finally, region 4 assumes a congested initial state with an accumulation value of 8,750 veh while all other regions are uncongested initially with accumulations of 3,850 veh.

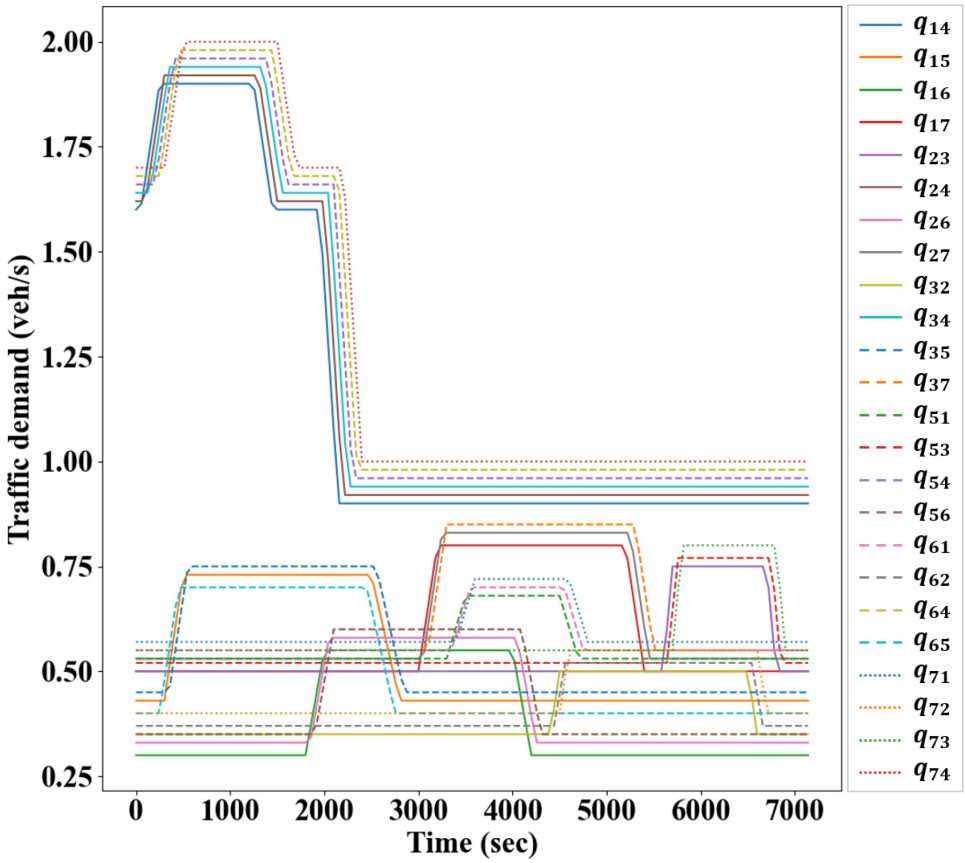


Fig. 3. Traffic demands with high inflows into region 4.

The environment (or equivalently referred to as the I/O data generator in (Chen et al., 2022; Lei et al., 2019; Ren et al., 2020)) utilized in this work (see Fig. 2) is constructed with the numerical equations presented in Section 2, along with the information on the MFDs, traffic demands, and initial states specified in the above. The environment also serves as the plant model for comparative control strategies, i.e., model predictive control (MPC) and no control (NC). These comparative methods, together with the proposed MR-RL scheme, are applied to conduct perimeter control via interactions with the plant model or environment, and their performances are compared in terms of the achieved cumulative trip completion (CTC). The NC method does not impose limitations on the transfer flows and instead used the maximum value for all perimeter controllers; it is usually adopted as a baseline method that provides the lower-bound control performances. In contrast, the MPC is an advanced model-based rolling horizon optimization scheme that has achieved state-of-the-art control performances. However, one major disadvantage of the MPC is that it builds upon full knowledge of the environment dynamics (i.e., the MFDs and dynamic equations governing vehicle movement between regions), which are generally difficult to obtain in the first place. In

this paper, the MPC is implemented as per the perimeter control-only scheme in (Sirmatel and Geroliminis, 2018) with a control horizon of 2 and a prediction horizon of 3. Reasons for selecting this prediction horizon are mainly twofold. First, for the seven-region perimeter control problem considered in this work, a longer prediction horizon does not necessarily lead to improved control performances. This is partly because the solution space of the formulated nonlinear nonconvex optimization program becomes significantly expanded, and as a result it is increasingly difficult for the MPC method to find the global optimum, which thus diminishes its control effectiveness. It is certainly promising (yet extremely challenging) to ensure global optimum finding for the MPC method, but this is beyond the scope of this paper. Second, a longer prediction horizon would also dramatically increase the computation burden for the MPC as it needs to conduct the optimization procedure in a considerably larger solution space. Moreover, note that adopting a short prediction horizon is not atypical in the literature, especially for large networks; for example see (Lei et al., 2019; Yildirimoglu et al., 2018). Importantly, a prediction horizon of 3 is used in (Lei et al., 2019) for the MPC to conduct perimeter control in a five-region network. Hence, the authors believe that it is reasonable to set the prediction horizon to 3 in the current work. The selection of the control horizon, on the other hand, is consistent with the settings in numerous previous works (Geroliminis et al., 2013; Hajiahmadi et al., 2015; Ren et al., 2020; Sirmatel and Geroliminis, 2018).

4.2 Effectiveness of the MR-RL scheme

The no uncertainty scenario is examined closely in this section to demonstrate the effectiveness of the proposed MR-RL scheme. Here, the traffic dynamics assumed by the MPC in the prediction model are the same as those in the plant. The MR-RL is trained with five fixed random seeds and its performance curves are shown in Fig. 4, where the darker line and shaded area respectively represent the mean and 95% confidence interval of the control gains (in terms of CTC). For clarity of presentation, the control gains achieved by the MR-RL scheme are reported every five iterations. The MPC and NC are also run five times to report their performance curves, but these curves are relatively invariant as they are not learning-based methods. Note that, the learning objective of the MR-RL is to select proper perimeter control actions such that the CTC of the network is maximized, and it does so by interacting with the environment and internalizing the traffic dynamics. However, the seven-region traffic dynamics are rather involved (see Section 2 and compare with two-region dynamics in (Haddad et al., 2012)), and learning in such an environment is prone to perturbations due to complex user behaviors. Hence, the learning trajectories of the MR-RL tend to be fluctuant. A possible way of mitigation is to disable the learning process altogether once the control gains reach a certain threshold and start to stabilize, but this then would not truthfully reflect how the scheme learns in the environment. Regardless, these performance curves are intended to convey that the scheme can consistently learn and effectively improve trip completion in the network, while fluctuations in the learning processes are allowed. Seemingly fluctuating learning curves are not unusual in the literature; for example see (Horgan et al., 2018; Mnih et al., 2015; Rashid et al., 2018; van Hasselt et al., 2015). Note further that, to be consistent with value factorization studies (Peng et al., 2021; Rashid et al., 2018; Wang et al., 2021), the mean rather than the median values of the control gains are reported in Fig. 4. However, the mean is more sensitive to randomness and extreme values in the learning process, thus the darker line in Fig. 4 would appear more fluctuant than reported using the median values. For the reasons discussed above, more analytical focus is placed on the general trend of the performance curves instead of the detailed learning fluctuations henceforth in the present work.

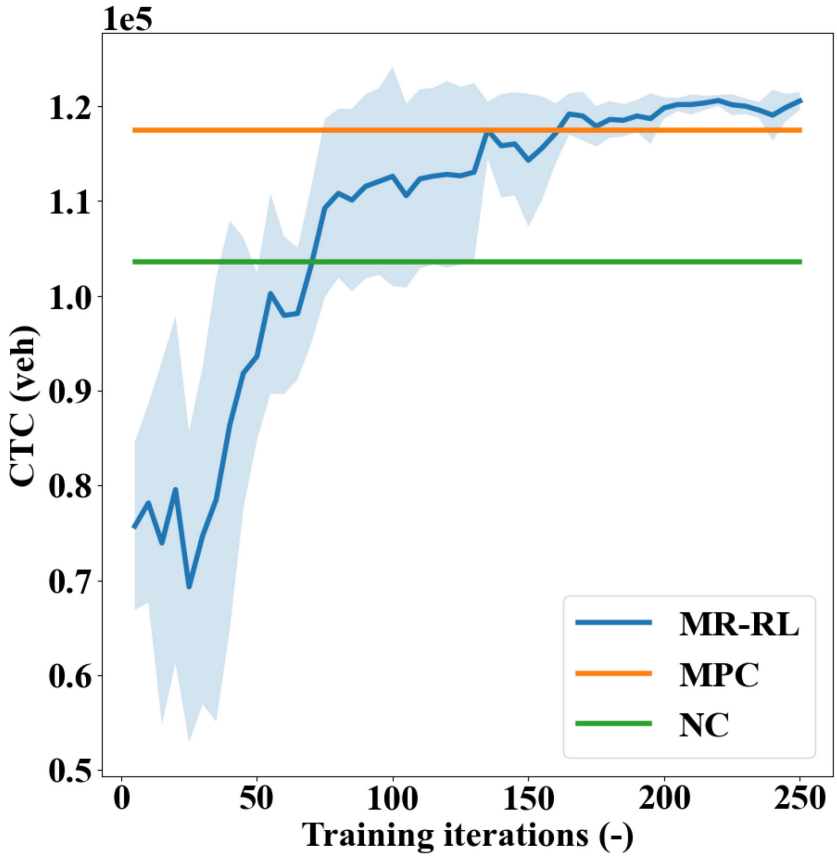


Fig. 4. Performance curves of different methods for the no uncertainty scenario.

As Fig. 4 shows, when training of the MR-RL scheme is completed (at the 250th iteration), the NC method realizes the lowest CTC value for the network. This is expected since unlimited vehicle inflow into region 4 aggravates the congestion therein and adversely impacts other inter-region vehicular movements. More importantly, the proposed MR-RL can consistently learn and achieve control gains that are commensurate with (sometimes even slightly better than) the MPC. This showcases the significant potential of model-free data-driven approaches over model-based ones, as similarly presented in (Chen et al., 2022; Lei et al., 2019; Ren et al., 2020). The MPC is an optimization-based method and derives control actions by solving a large nonlinear nonconvex program that features a sizable solution space. As such, it may fail to find the global optimum, which leads to the underperformance to the proposed scheme. Though the MPC could theoretically be the optimal control technique with improved performances via guaranteed global optimum finding, the implementation of this is not conceivably straightforward. Comparatively, the MR-RL learns the control policy via trial and error, and through this process it can encounter better acting strategy than the MPC. Finally, note that training performances of the MR-RL in the early period are noticeably worse than the NC method. This is reasonable since during this period the MR-RL is principally exploring the environment. In this paper, the training process is presumed to be completed with numerical simulations. Thus, the poor control performances initially are not concerning as only the fully trained MR-RL scheme will be applied to control with advantageous gains at the last iteration.

To further demonstrate the effectiveness of the MR-RL, its control outcomes are examined more carefully in the following. Fig. 5 presents the control actions u_{i4} of the MR-RL and the MPC, while all other controllers are omitted from the presentation. This selective presentation is done intentionally since other controllers are nearly inactive, i.e., they all adopt the maximum value u_{max}. This is expected since the implementation of perimeter control here is mostly designed at protecting region 4 from severe congestion, for which u_{i4} being active is sufficient. Likewise, the NC actions are not included for comparison either since they are all equal to the maximum value. Fig. 6 presents the resulting evolutions of accumulations for each region, as achieved by different control methods. The critical accumulations are also provided in dash lines which help determine the congestion situation for the regions.

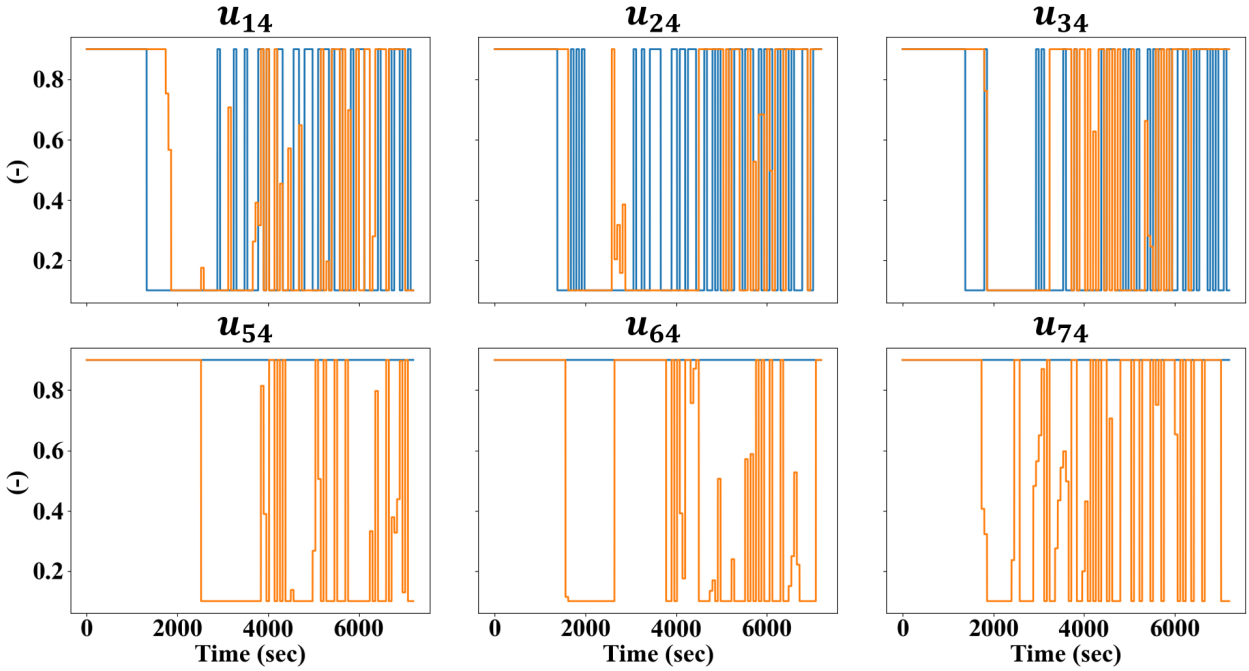


Fig. 5. Control actions u_{i4} of the proposed MR-RL (in blue) and the MPC (in orange).

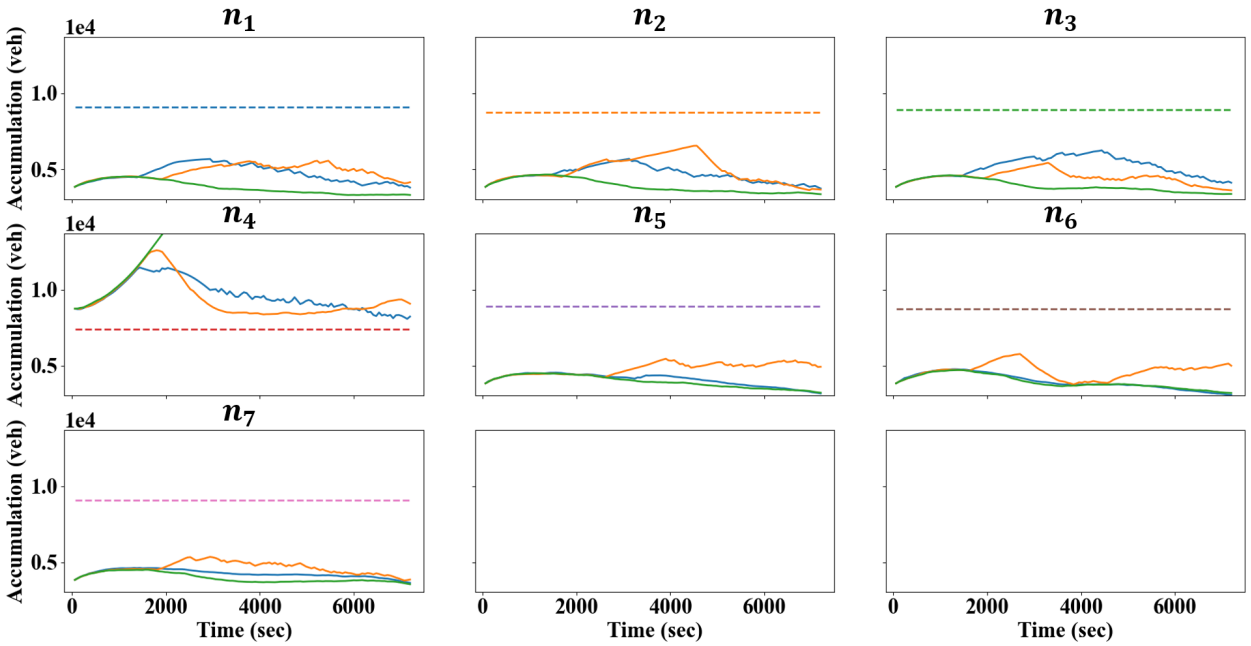


Fig. 6. Accumulation plots for all regions. The dash lines represent the critical accumulations. Blue: MR-RL; Orange: MPC; Green: NC. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

A few notable observations can be made from these plots. First and foremost, under the NC method (i.e., no perimeter control), region 4 becomes extremely congested (in fact, nearly gridlocked, but this is not shown in Fig. 6 for the other subplots to be more readable) at the end of the control period while the accumulations in other regions are generally smaller than realized by the MPC or the MR-RL. This is understandable as the region 4-bounded traffic flows are much larger than the others. However, severe congestion in region 4 leads to a small trip completion therein, which also makes inter-regional travel time-consuming. For example, region 6-bounded vehicles in region 2 that normally would travel via region 4 might need to take a longer route to reach their destinations. Consequently, the trip completions in other regions will be negatively influenced and the NC method ends up achieving the lowest CTC. In comparison, both the MPC and MR-RL can significantly reduce the congestion in region 4, while in the meantime keeping the accumulations in other regions under the critical values. This implies that these methods can indeed perform effective perimeter control since the most destination-loaded region (i.e., region 4) are protected from over-congestion, as consistent with the AB strategy proposed in (Daganzo, 2007). Second, both the MPC and MR-RL select the maximum value for all perimeter controllers in the initial period, which is sensible as there does not exist pronounced congestion within the network (e.g., even region 4 is only moderately congested). On this note, mind that the MR-RL chooses either u_{min} or u_{max} for the perimeter controllers, as grounded in the action space design of Bang-Bang form. While the optimal perimeter control policy has been shown in the form of Bang-Bang (Aalipour et al., 2019; Daganzo, 2007; Ni and Cassidy, 2020), in practice the policy is difficult to implement and may cause abrupt fluctuations of traffic conditions in the network that could further increase congestion heterogeneity (Geroliminis et al., 2013). On the other hand, the Bang-Bang form allows for the design of control policies that can better adapt to fast-changing traffic situations which is otherwise not achievable by smooth control policies. With an on-or-off

rule, it has the potential to better regulate regional accumulations around the critical level such that the network throughput is maximized. Also, applications of the Bang-Bang form mainly depend on the regional congestion level, which is often attainable with proper instrumentation. Therefore, as a middle ground, moving average can be applied to smooth out the control actions for easiness of practical implementations; further discussions on this aspect as well as more solutions techniques can be found in (Geroliminis et al., 2013). Third, notice that the MPC imposes stricter limitation on the transfer flows to region 4 from regions 5, 6, and 7; hence the accumulations in these three regions are generally larger than those resulted from the MR-RL actions. It also appears from the MR-RL actions that regulating the transfer flows from regions 1, 2, and 3 is sufficient to curb the congestion in region 4 and improve the trip completion. Importantly though, despite the differences in the control actions, the resulting evolutions of accumulations by the MPC and MR-RL exhibit a high level of similarity, which indicates great comparability between the two methods and showcases the effectiveness of the MR-RL. Finally, it should be acknowledged that, while the actions and accumulations plots can indeed help establish the effectiveness of the MR-RL scheme, these plots are specific to the scenario under consideration and may not be interpreted as a universal outcome of the MR-RL when applied to multi-region perimeter control. Instead, the scheme will learn to adopt different courses of actions based on the scenario it is trained on.

4.3 Resilience of the MR-RL scheme

This section evaluates the learning resilience of the proposed MR-RL against inexact inputs from the environment, i.e., inaccurate accumulations, demands, and congestion indicator information.

4.3.1 Measurement noise of accumulations

This scenario tests the learning ability of the MR-RL scheme in the presence of measurement noise on the accumulations, which simulates potential sensor malfunction that could lead to inaccurate vehicle identifications. Concretely, the measurement noise considered here is defined as (similar to (Ren et al., 2020)):

$$\tilde{n}_{ij}(t) = n_{ij}(t) + \mathbb{N}(0, \delta^2) \tag{12}$$

where $\tilde{n}_{ij}(t)$ is the measured value of the accumulation $n_{ij}(t)$ from the environment and $\mathbb{N}(0, \delta^2)$ represents a mean-zero normal distribution with variance δ^2.

In this scenario, measurement noise with the δ value ranging from 0 to 60 is tested. Note that, the measurement noise presented in Eq. (12) is imposed on the detailed accumulations n_{ij}; thus, the noise experienced at the regional level is seven times larger than specified by the δ value (for the seven-region network under study). For example, with a δ value of 40, the measurement noise at the regional level follows a normal distribution with variance $7 \times 40^2 = 11{,}200$, which is apparently significant given the critical accumulation is around 8,240 veh. Foreseeably, a control method without a feedback mechanism (e.g., NC) would be extremely sensitive to this noise and yield control gains that are rather fluctuant. Moreover, under the NC policy, gridlock would often arise in region 4 (which is already severely congested without the measurement noise) as it cannot meter the vehicle entries. Thus, for the NC method, numerous simulations are run for each δ value using distinct random seeds, and the mean values of the CTC realized when the network is not

gridlocked are reported. In contrast, control schemes with a feedback mechanism (e.g., the MPC and proposed MR-RL) can readily cope with the measurement noise and effectively improve trip completion; see Fig. 7 for the performance curves where one training instance is provided for each δ value. The MR-RL is a learning-based scheme, and with increasing measurement noise in the environment its learning trajectories tend to be noisier; see the curves with $\delta \geq 40$ for example. However, as explained previously, the learning fluctuations are less informative than the general learning trend. Critically, despite learning fluctuations, the MR-RL scheme can consistently produce final perimeter control policies that are comparable to the MPC, regardless of the level of uncertainty in the accumulation measurements. This manifests the resilience of the MR-RL scheme against measurement noise, which is not surprising as the scheme is not subject to the modeling inaccuracies and instead adjusts its course of actions based on the measured accumulations.

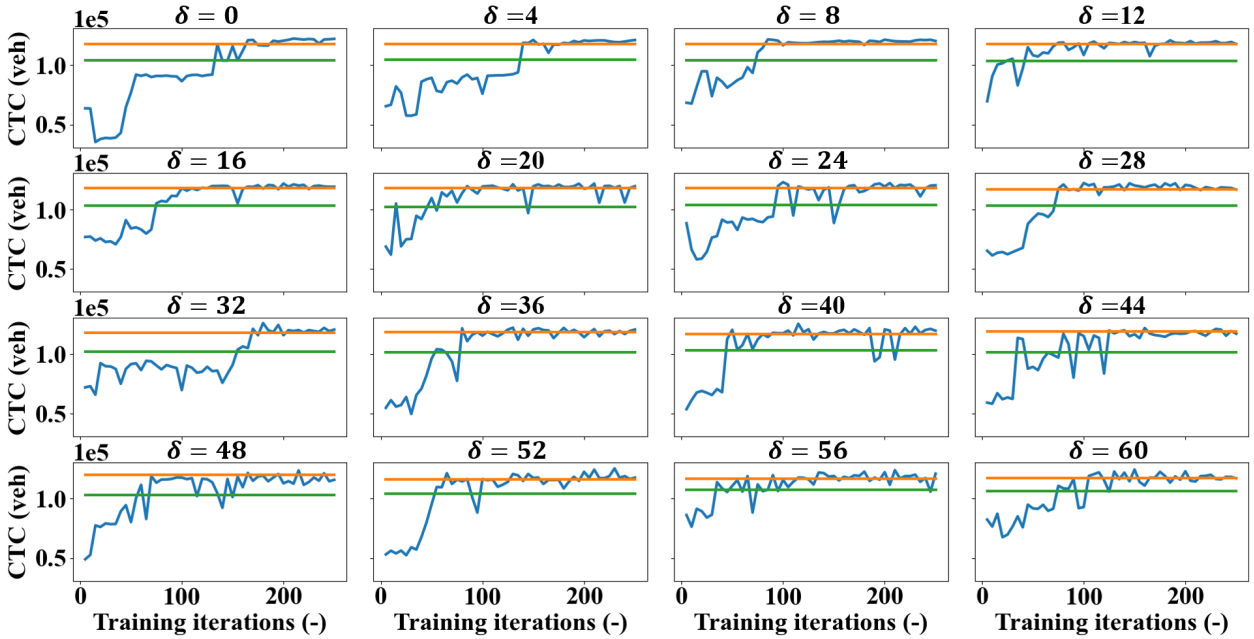


Fig. 7. Performance curves of different methods under measurement noise. Blue: MR-RL; Orange: MPC; Green: NC. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.3.2 Iteration- and spatially-varying traffic demands

This scenario first tests the learning resilience of the MR-RL when confronted with temporally changeable demand patterns. Specifically, the traffic demands are assumed to be non-repetitive over different training iterations, as in (Ren et al., 2020), which mimics the temporal variation of traffic demands during different days. After training with the iteration-varying traffic demands, the MR-RL scheme is evaluated on the no uncertainty scenario, on which the MPC and NC methods are also applied for comparison. A total of 10 distinct profiles are adopted for each demand function in Fig. 3, and the representative iteration-varying traffic demands are provided in Fig. 8 for q_{14}, q_{15}, whereas the varying profiles for other demands are omitted for clarity of

presentation. Note that, the MR-RL is trained for 250 iterations, hence the traffic demands would alter every 25 iterations.

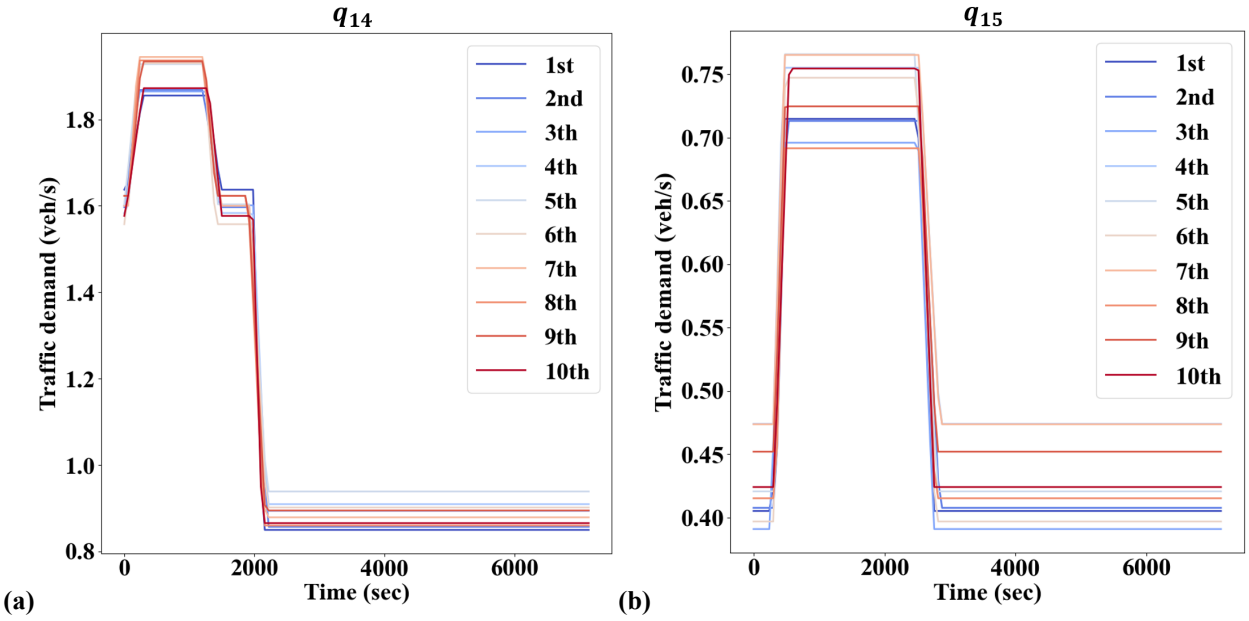


Fig. 8. Iteration-varying traffic demands for: (a) q_{14}; (b) q_{15}.

Fig. 9 presents the cumulative trip completions realized over time in the individual regions as well as the network altogether by the MPC and the MR-RL schemes, where the scheme trained with the iteration-varying traffic demands is denoted by "(I)" to differentiate from the one directly trained for the no uncertainty scenario (as in Section 4.2). Note that, the CTCs shown in Fig. 9 are expressed as differences from the NC method for better readability and comparison (y axis dubbed as "CTC Diff."). Also note, though the MR-RL is trained on the iteration-varying demands, it is evaluated on the no uncertainty scenario along with the MPC and NC, so the traffic demands are the same for them and thus the trip completion curves can fairly represent how each method works. As can be observed, the trip completions in the peripherical regions are almost identical across different methods, i.e., the CTC Diff. is around 0. This is expected as the simulated scenario, despite with iteration-varying traffic demands, mimics a morning peak when most vehicles are destined for the city center (i.e., region 4). Only a small portion of vehicles travel across the peripherical regions, and such travel is not metered by any method. As a result, traffic conditions in the periphery do not significantly differ across the methods (see also the accumulation plots in Fig. 6), which renders the trip completions similar. Comparatively, the trip completion in region 4 is substantially improved with the enforcement of perimeter control, i.e., the CTC Diff. keeps increasing over time for the MPC and the MR-RL schemes. This is also not surprising as the congestion in region 4 is alleviated with restrained vehicle entries. In addition, the improved regional trip completion could further lead to a higher CTC for the whole network; see Fig. 9(h). Importantly, albeit trained with iteration-varying traffic demands, the MR-RL scheme can still achieve cumulative trip completions for the network that is even slightly higher than the MPC. This implies that minor perturbations in the traffic demands due to day-to-day variations can be accommodated by the MR-RL, which demonstrates its superior learning resilience, even more so considering that the MPC has full knowledge of the environment dynamics. From a practical

standpoint, this indicates that accurate demand information does not need to be known beforehand by the scheme to perform effective perimeter control; instead, it can be trained on a set of estimated demand profiles for a target scenario with ensured control benefits on those scenarios.

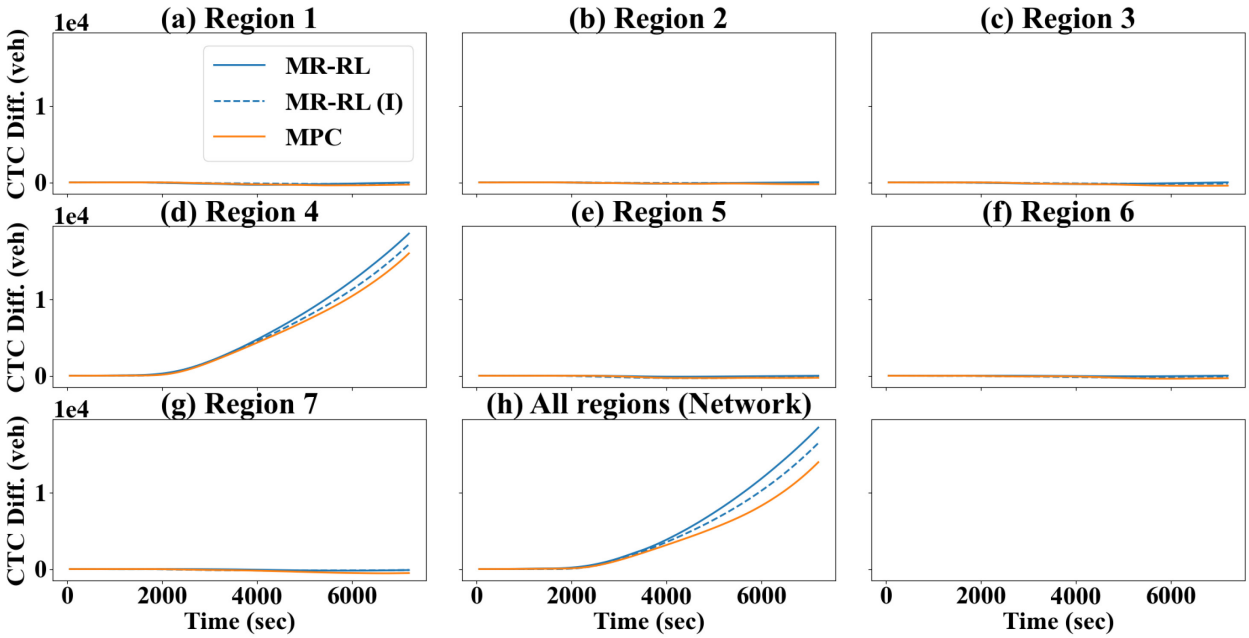


Fig. 9. Cumulative trip completions in individual and all regions (i.e., the network) by the MPC and MR-RL schemes, where "(I)" denotes the scheme is trained with the iteration-varying traffic demands. The CTCs are expressed as differences to the NC method.

Similar to the above, the learning resilience of the MR-RL against spatially-varying traffic demands is tested. In particular, a total of 10 spatially changeable demand profiles is considered, and within each profile the traffic flows destined for region 4 are randomly shuffled. This random shuffling also applies to the demands between the peripherical regions, but separately. Note that, it is not realistic to shuffle all demands at once irrespective of their destination regions. This is because the resulting traffic demands would lack a clear trend towards the city center (i.e., region 4) and thus not be representative of traffic conditions during a morning peak.

Through the training course of the MR-RL, the traffic demands are varied spatially every 25 iterations, and after training the scheme is evaluated on the no uncertainty scenario, together with the MPC and NC. The cumulative trip completions realized in the individual regions and the network altogether are shown in Fig. 10, where the MR-RL trained with the spatially-varying demands is denoted by "(S)". The analyses regarding the differential curves in Fig. 10 are largely similar to those on Fig. 9, thus the authors do not repeat the discussions here. However, it is worth noting that the tests conducted in this section indicate that the proposed MR-RL scheme is resilient to both temporal and spatial variations in the distributions of the traffic demands. This is important to the practical applications of perimeter control. Particularly, despite relatively accurate estimates of traffic demands can be obtained with the abundance of online and archived traffic data, the real-time traffic conditions during perimeter control implementations will always be different from the historical estimates. And because of this it is crucial for the control scheme to be able to adapt to

variations in the demands. In this work, such adaptability is enabled for the MR-RL scheme with a feedback-based learning process. Similar control approaches with ensured adaptability may also be found in (Chen et al., 2022; Lei et al., 2019; Ren et al., 2020).

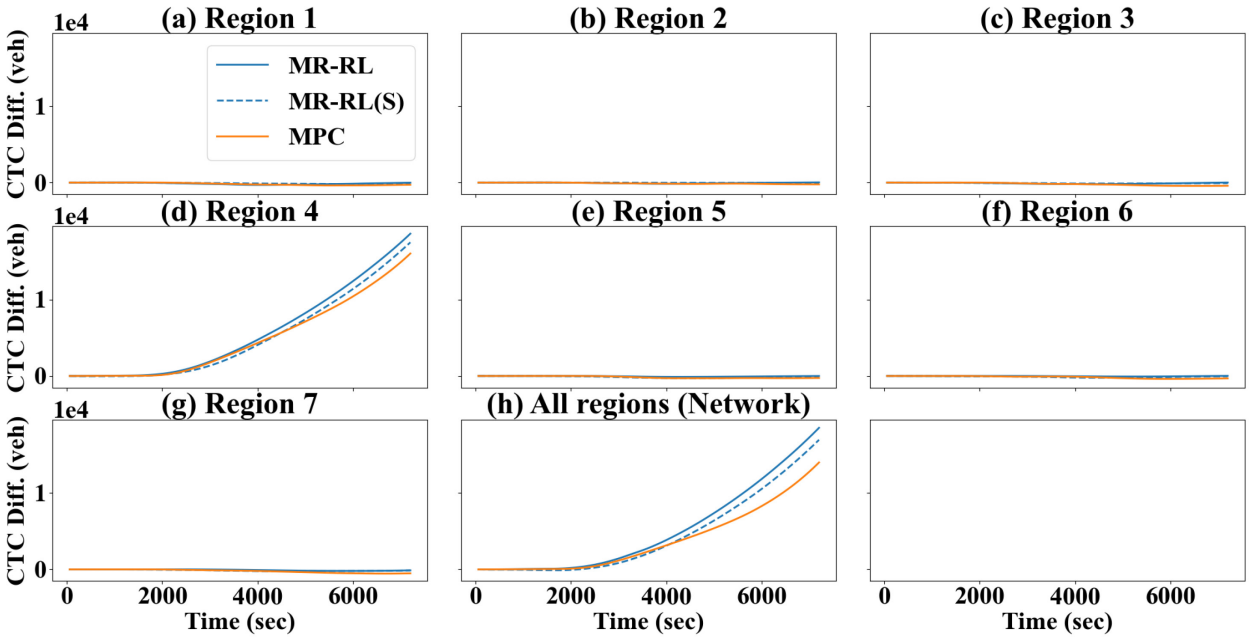


Fig. 10. Cumulative trip completions realized by the MPC and MR-RL schemes, where "(S)" denotes the scheme is trained with the spatially-varying traffic demands. The CTCs are expressed as differences to the NC method.

4.3.3 Estimation errors of the critical accumulations

The proposed MR-RL takes the congestion indicator in its local observation to select decentralized actions and in its state to conduct centralized training. However, the acquisition of the congestion indicator necessitates the estimation of the critical accumulations. Therefore, the congestion indicator received from the environment might be inaccurate due to estimation errors of the critical accumulations that are common in urban networks (Daganzo et al., 2011; Gayah and Daganzo, 2011; Mahmassani et al., 2013; Mazloumian et al., 2010). Importantly, note that the environment is assumed to have access to the critical accumulation information, though such information might be prone to errors. In comparison, the MR-RL does not have access (nor does it require access) to the critical accumulations. Instead, it acts upon the congestion indicator it receives, regardless of whether the indicator can correctly reflect the congestion status in the environment. Foreseeably, however, more accurate congestion information could be beneficial to the learning performances. To this end, this scenario tests the learning ability of the MR-RL when provided with imprecise classifications of regional congestion levels due to estimation errors of the critical accumulations.

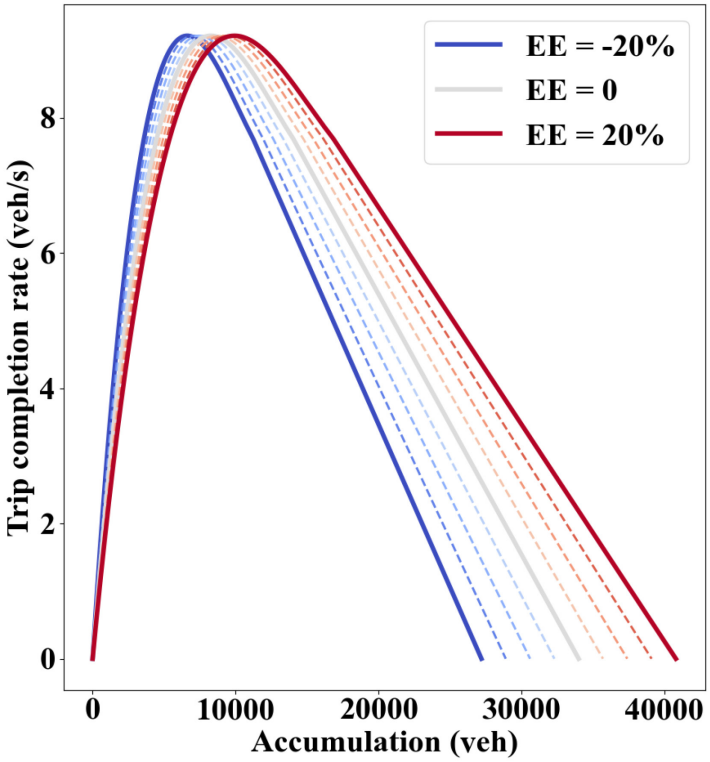
It is worth highlighting that, while the MR-RL scheme does not embed into its design the system dynamics or MFD information (hence, "model-free"), it is still prone to inaccurate inputs from the environment, i.e., the learning and control efficacy of the MR-RL might be hampered by misleading information received from the environment. This resembles the potential mismatch of

traffic dynamics in the prediction model and plant faced by the MPC method. The inaccuracies associated with the accumulation measurements and traffic demands have been examined in the previous two scenarios, and this scenario focuses more on the inaccurate information concerning the MFDs, or more specifically the inexact congestion indicator as a result of imprecisely estimated critical accumulations in the environment. Note that, special attention has been paid to the errors related to the critical accumulations as these errors could directly impact the inputs received by the MR-RL. In addition, the critical accumulation is arguably the most important piece of information about the MFD, for which model linearization has been extensively applied around it (Aboudolas and Geroliminis, 2013; Haddad, 2015; Haddad and Shraiber, 2014; Keyvan-Ekbatani et al., 2012). In comparison, other features regarding the MFD (e.g., functional form, maximum trip completion, and jam accumulation) are principally specific to the environment and not immediately perceivable by the MR-RL; thus, the relevant errors are not tested herein and left as future work of this study. Note further that, the inaccuracies in the congestion indicator are assumed to result from estimation errors on the critical accumulations but not from classification errors of regional congestion with correct critical accumulations. Reasons for this are twofold. First, the classification errors with correct information might cause unrealistic identifications of congestion status. For example, under this error, even an empty region may be categorized as congested, which is clearly not reasonable. Second, the classification errors would often arise as a result of the estimation errors, but in a more realistic manner. In one case, with significant under (over) estimation errors, regions that are in fact quite uncongested (congested) may be treated as congested (uncongested). In another case, with moderate estimation errors, regions that are operating at or near the best conditions (i.e., the accumulations are at or around the critical values) might still have congestion indicators that are mis-classified. Notice that this is also the scenario under which determining the congestion status is particularly difficult due to the proximity between the actual accumulations and the desired ones.

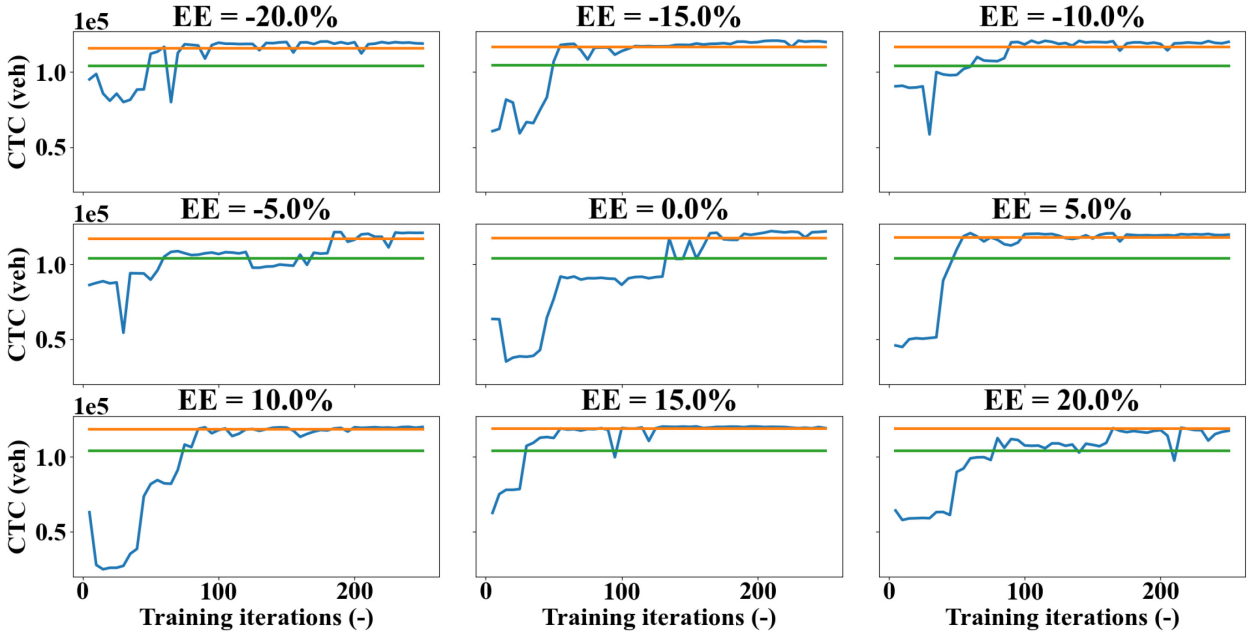
In this scenario, estimation errors ranging from -20% to 20% are considered, and each level of error applies to all regions at the same time. For instance, an estimation error of +5% indicates that region i ($i = 1, \cdots, 7$) with accumulation values of $\leq 1.05 \cdot n_{ic}$ is classified as uncongested, so the environment is perceived to be more productive than it really is. Conversely, negative estimation errors suggest that the environment is more productive than perceived by the MR-RL. For a fair comparison, the MPC method is also subject to this estimation error. However, the MPC does not explicitly utilize the critical accumulation information in its optimization-based solution scheme; hence the estimation error is imposed on the MFD functions of the MPC prediction model. Particularly, the MFD functions in the prediction model are shifted to the left (right) to simulate negative (positive) estimation errors, with the maximum trip completion not altered; see Fig. 11(a) for the shifted unit MFD function with estimation errors. It is worth noting that, the MFD errors considered in (Ren et al., 2020) are in the form of scaling uncertainty, which changes not only the critical accumulations but also the maximum trip completions. As a consequence, the MFD shape is significantly different across the prediction model and plant for the MPC. In contrast, the proposed iterative learning scheme therein only perceives a different critical accumulation from the environment. Therefore, the scaling errors might cause worse performances for the MPC, in an unfair fashion. The shifting process adopted in the current work would thus lead to a more impartial comparison between the MPC and the proposed MR-RL, as the shifting error for the former is more commensurate with the estimation error for the latter. As a side note, the NC method is not impacted by the estimation error as its policy is not dependent on any information from the environment. Finally, notice in Fig. 11(a) that the jam accumulations of the MFD function

have also been shifted resulting from the estimation errors, but this is practically inconsequential as the MPC control inputs would not allow the accumulations to approach the jam value (Haddad et al., 2013; Sirmatel and Geroliminis, 2018).

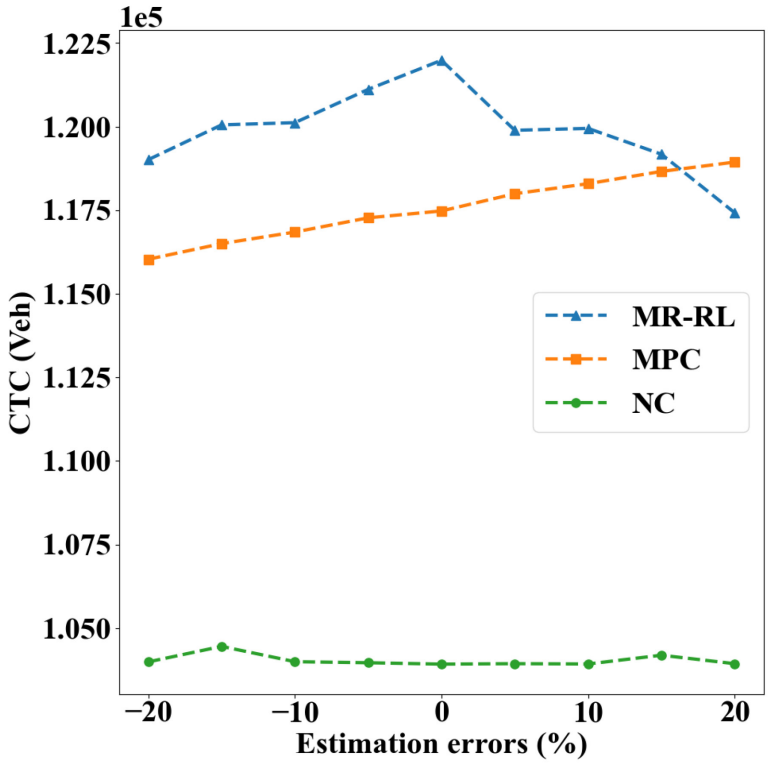
Building upon the no uncertainty scenario, performance curves of the MR-RL under each level of estimation error (EE) are shown in Fig. 11(b). For better comparison of the three methods, the realized CTCs by the MPC, NC, and the MR-RL at the last iteration are also presented in Fig. 11(c). The standard errors of the realized CTCs are negligibly small and therefore not included in the plot. Note that, all three methods share the same plant or environment, so their effectiveness can be easily compared with the control outcomes. As Fig. 11(c) reveals, under the NC policy, the total number of trips completed during the control period is largely invariant against estimation errors, which is expected since the NC policy (as well as the plant) does not change with the estimation errors. In contrast, under the MPC policy, the cumulative trip completion consistently increases with the estimation errors. While this may appear counterintuitive, it is foreseeable to a certain extent. Concretely, with negative estimation errors, the MPC might deem the network more congested than it really is and impose stricter limitations on the transfer flows, thus hampering trip completions. On the other hand, lessened restrictions are enforced with positive estimation errors, which allows more transfer flows to the city center. This would cause more pronounced congestion within region 4 but in the meantime yield a higher trip completion for the network. In addition, the control efficacy of the MPC is impacted by its optimization process. Without guarantees of finding the global optimal solution, a seemingly undesirable solution under overestimation might in fact be superior to the solution found without estimation error; see Fig. 11(a) in (Ren et al., 2020) for another instance of this, where the MPC realizes improved performances with overestimation of the network productions, though such improvement diminishes as the error further increases. The theoretical analyses of these phenomena are certainty worthy of deeper investigations, but they are beyond the scope of the present study and thus left as future works.



(a) Shifted unit MFD function with estimation errors ranging from -20% to 20%; the gray curve denotes the un-shifted unit MFD function.



(b) Performance curves under different levels of estimation errors (EE). Blue: MR-RL; Orange: MPC; Green: NC. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(c) CTCs realized by the MPC, NC, and the MR-RL at the last iteration.

Fig. 11. Setup and results for the resilience test on estimation errors of the critical accumulations.

The results in Fig. 11(b) and (c) also indicate that the effectiveness of the MR-RL is prone to the estimation errors of the critical accumulations (especially when the environment production is over-estimated). This is reasonable as the received congestion indicator with such errors cannot truthfully reflect the congestion conditions of the regions, which is thus detrimental to the learning process of the MR-RL scheme. And as one would expect, the MR-RL achieves the best control outcome when it receives accurate congestion information from the environment, i.e., without estimation errors. Importantly, despite being impacted by these errors, the MR-RL can consistently learn final control policies that are far superior to NC and most of the time even superior to the MPC. This showcases the learning ability of the MR-RL and its resilience to inaccuracies of the congestion indicator. While the MR-RL may fail to perform comparably to the MPC with large over-estimation errors (i.e., $\geq 20\%$), this is hardly an issue in reality as whether or not a region is congested can be conveniently obtained with high accuracy in an instrumented network.

In summary, the experiment results in this section show that the proposed MR-RL can learn to conduct perimeter control effectively and compete with (often times outperform) the MPC even with imprecise input information from the environment. Concretely, the MR-RL can accommodate measurement noise on the accumulations, temporal and spatial variations in the distributions of traffic demands, and inaccurate congestion information due to estimation errors concerning the critical accumulations. These results manifest the learning resilience of the MR-RL scheme against environment uncertainties as well as its control effectiveness. Note that, the proposed MR-RL is model-free in that it does not embed in its design any knowledge about the environment, whereas such information is necessary to the MPC for it to be applicable to perimeter control. This contrast

thus further highlights the learning ability and resilience of the MR-RL as the model-free MR-RL with inaccurate inputs can achieve control outcomes that are comparable or even superior to those of the model-based MPC with full access to the environment.

4.4 Transferability of the MR-RL scheme

The transferability of the MR-RL is examined in this section by applying a pretrained scheme to unencountered environments with unknown uncertainty in the MFDs, traffic demands, and/or accumulations. Note that, the environment uncertainty examined in Section 4.3 can be internalized by the MR-RL during training; however, here the uncertainty is built in the new environments and not perceivable by the MR-RL. For all experiments conducted in this section, the pretrained MR-RL from the no uncertainty scenario is utilized for control without additional training.

4.4.1 Unknown uncertainty in MFDs and traffic demands

This scenario tests the transferability of the MR-RL to environments with unknown uncertainty in the MFDs and traffic demands, as follows (Geroliminis et al., 2013; Zhou and Gayah, 2021):

$$\tilde{f}_i\big(n_i(t)\big) = f_i\big(n_i(t)\big) + \omega(t) \cdot n_i(t) \tag{13}$$

$$\tilde{q}_{ij}(t) = \max\big(q_{ij}(t) \cdot \big(1 + \upsilon(t)\big), 0\big) \tag{14}$$

where $\omega(t)$ follows a mean-zero uniform distribution with parameter λ (i.e., $\mathbb{U}(-\lambda, \lambda)$) and $\upsilon(t)$ a mean-zero normal distribution with scale σ (i.e., $\mathbb{N}(0, \sigma^2)$). These uncertainties could represent random modeling errors that result from imperfect knowledge of either environment production or demand allocation. The new environments can then be obtained by replacing the MFD and demand terms in Section 2 with Eqs. (13)-(14). These uncertainties are also embedded in the MFDs plant (but not in the prediction model) for the MPC to establish a fair comparison with the MR-RL.

This scenario considers parameter values of $\sigma, \lambda = 0, 0.1, 0.2$, and new environments are constructed with each combination. The pretrained MR-RL is utilized to conduct perimeter control in the new environments, along with the MPC and NC methods. Note that, the MPC method, when applied to the new environments, still formulates and solves nonlinear optimization programs to determine the control actions, whereas the MR-RL directly applies its policy learnt from the no uncertainty scenario. Further, to minimize the effects of randomness, 10 distinct random seeds are adopted for each method, and the realized CTCs in each environment are presented in Fig. 12 using box plots. As can be observed, the pretrained MR-RL scheme, when transferred to unencountered environments with unknown uncertainties in MFDs and traffic demands, could still achieve control gains that are superior to the MPC. This indicates the traffic dynamics internalized by the MR-RL during the training process are transferable to new environments governed by the same modeling principles (for example, the conservation equations). Therefore, the pretrained MR-RL can select sensible actions for perimeter control even with unseen observations. In comparison, the MPC is liable to the extra uncertainties in the environment and may fail to act optimally. Furthermore, these results suggest that the MR-RL is amenable to additional uncertainty in the MFDs and traffic demands from the environment it was trained on. Practically speaking, this implies the MR-RL could be first trained in a relatively deterministic environment and then applied in a more realistic (noisier) environment, while ensuring sufficient control advantage over the MPC.

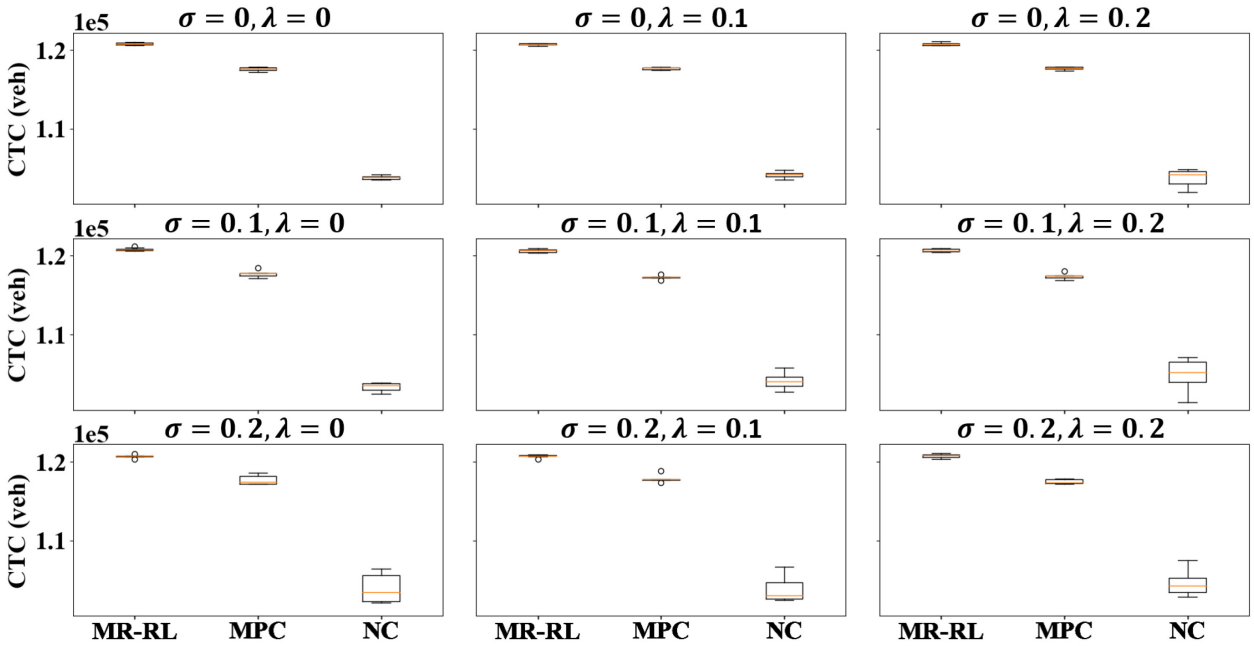


Fig. 12. CTCs achieved by three methods in new environments.

4.4.2 Unknown uncertainty in accumulations

In this scenario, the measurement noise is adopted again to evaluate the transferability of the MR-RL. Specifically, the applied scheme is trained on an environment without measurement noise but tested on environments with such uncertainty. Similar to Section 4.3.1, δ values ranging from 0 to 60 are considered, and the control gains achieved by three methods over 10 runs are provided in Fig. 13. As shown, the CTC values realized by the pretrained MR-RL are generally comparable to those by the MPC, and under low measurement noise (i.e., ≤ 28) the pretrained MR-RL can often outperform the MPC. This again demonstrates the superior transferability of the MR-RL, more so considering the MPC has full knowledge of the environment dynamics and can adjust its policy with the uncertainty whereas the MR-RL has not perceived the measurement noise in its course of learning. As the level of measurement noise increases, the control advantage of the MR-RL over the MPC diminishes, which is expected as the MPC adopts a closed-loop structure with feedbacks from the plant and can thus counter the considerable measurement noise to some extent. In contrast, the pretrained MR-RL is applying a fixed policy without feedback-based adjustments; hence it is prone to the large errors in accumulation measurements. However, the feedback mechanism of the MPC comes with high computational cost since it needs to solve a sizable nonlinear program at each time step of the control period for every level of noise. Comparatively, the computation time needed to apply the pretrained MR-RL to different levels of measurement noise is nearly negligible, as the actions are derived from a direct forward pass through the agent network. In this regard, the MR-RL is significantly more real-time applicable than the MPC. Finally, notice that the CTC differences between perimeter control methods (both the MR-RL and MPC) and no control tend to decrease under high measurement noise (see $\delta = 56$ for example). This is largely a result of the substantial variations in the CTC values under the NC policy. More specifically, the NC method

can neither meter the congestion in the network or regulate regional accumulations with feedbacks. On the other hand, the measurement noise with large δ values is rather notable in the environment. Hence, in the absence of perimeter control, the accumulations in the network tend to vary greatly, thus rendering the CTC values extremely sensitive to the high measurement noise. The reduced CTC differences might make the NC method appear misleadingly effective, but this is not the case. Quite the contrary, this suggests that perimeter control methods with feedback mechanisms are needed to curb the network congestion and to cope with such high levels of uncertainty from the environment.

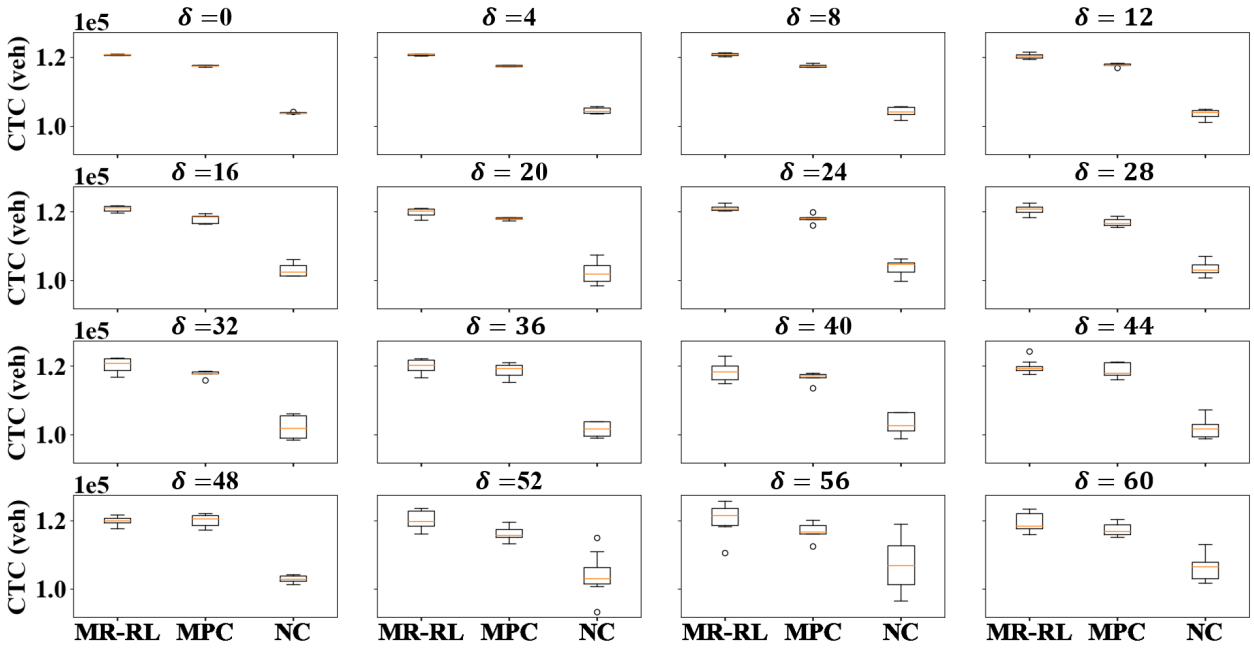


Fig. 13. The achieved CTCs under measurement noise in the unencountered environments.

In summary, the performance contrasts presented above highlight the application prospect of the MR-RL scheme since it can transfer the knowledge it internalized during training to conduct perimeter control effectively in unencountered environments with more uncertainty. In addition, the MR-RL does so in a real-time fashion without additional training in the new environments, yet the MPC needs to formulate and solve numerous nonlinear nonconvex optimization programs to derive a control policy for these environments. Note that, in line with the reinforcement learning literature, the transferability of the MR-RL scheme is examined on environments with similar base settings (i.e., same dynamics modeling principles utilizing the MFDs and similar traffic conditions mimicking a morning peak scenario). Transferring the MR-RL to environments with considerably different settings (e.g., a network with distinct number of regions, different demand patterns like an evening peak, or sudden fluctuations in the regional traffic conditions such as road closures) is not what reinforcement learning is intended for and may warrant investigations in another learning paradigm (e.g., transfer learning). This is also true for other learning-based data-driven approaches (Chen et al., 2022; Ren et al., 2020). On this note, it should be pointed out that the MPC method is not transferable or even applicable to new environments if the dynamics are unknown. By design, the MPC can only conduct perimeter control with full knowledge of the environment dynamics, and it solves for a control policy by formulating the perimeter control problem from scratch, which

is challenging on its own as a significant amount of information is needed. This goes without saying that the traffic dynamics might often be unknown for distinctively different scenarios. Note further that, this paper conducts all experiments with numerical simulations, and transferring the pretrained scheme to a microsimulation platform is left as future work.

5. CONCLUDING REMARKS

This paper presents a novel scheme (termed MR-RL) for large-scale multi-region perimeter control building upon model-free multi-agent deep reinforcement learning. The proposed MR-RL features value function decomposition that significantly improves learning scalability to problem settings with numerous agents, recent breakthroughs of single-agent deep reinforcement learning (such as the Ape-X architecture, double Q-learning update rule, experience replay, and target networks), and problem reformulation governed by domain expertise (e.g., the Bang-Bang form action design). To evaluate the control applicability of the MR-RL, comprehensive numerical experiments are conducted on a simulated seven-region urban network, and the results suggest that the scheme is: (a) effective, with consistent learning behaviors and convergence to final control outcomes that are comparable to the MPC method; (b) resilient, with sufficient learning and control efficacy even in the presence of inaccurate input information from the environment; and (c) transferable, with superior application prospect to unencountered environments characterizing increased uncertainty.

The proposed MR-RL has several distinct advantages over existing model-based or model-free data-driven perimeter control approaches (Chen et al., 2022; Geroliminis et al., 2013; Lei et al., 2019; Ren et al., 2020; Sirmatel and Geroliminis, 2018; Zhou and Gayah, 2021). First, the MR-RL is model-free in that knowledge of environment dynamics is not embedded in the design of the scheme, whereas model-based methods (Geroliminis et al., 2013; Sirmatel and Geroliminis, 2018) necessitate such knowledge to determine a perimeter control policy. The data-driven approaches in (Lei et al., 2019; Ren et al., 2020) also build into the controller designs the critical accumulations of the network. The model-free design is essential for a controller to cope with the complex traffic conditions that may often arise in multi-region networks, under which circumstances model-based methods such as model predictive control may not even be applicable as explicit modeling of the environment dynamics is extremely difficult. And because of the model-free design, the proposed MR-RL scheme is remarkably resilient to a wide range of modeling uncertainties associated with the accumulation measurements, traffic demand variations, and MFD functions, to which the MPC is susceptible. This highlights the prospect of practical applications for the MR-RL as it can learn effectively regardless of inaccurate information from the environment and compete with (or even outperform) the MPC with full access to the environment dynamics. Second, the MR-RL has been shown scalable via extensive numerical experiments on a seven-region urban network, which is the largest one ever considered in perimeter control studies. In contrast, previous model-free strategies (Chen et al., 2022; Zhou and Gayah, 2021) have only been shown applicable to smaller networks. The scalable design of the MR-RL as well as its verification is critical as the scheme may later be combined with other macroscopic or microscopic control applications to form a comprehensive city-level traffic management framework. Third, the MR-RL features a learning-based design, and with such design it can internalize knowledge about the environment during training and transfer this knowledge to unencountered environments to perform effective perimeter control. The transferability aspect of the perimeter control methods (particularly the model-free data-driven ones) is important as traffic conditions of the environment on which the methods are

applied may often times be different from those of the environment the methods are trained on. In addition, the dynamics modeling of the applied environment may not be straightforward but instead rather data and computation intensive; for occasions like these the ability to transfer the learned knowledge is especially crucial. However, the transferability aspect is largely overlooked in the literature, and most model-free methods are trained specifically for certain scenarios. Also note that, after training of the MR-RL scheme is completed, applying it for control in the new environments is real time applicable, as the perimeter control actions can be directly obtained via a forward pass through the fully trained agent network. In comparison, the MPC is not transferable and faces a significant computation cost due to formulation and solution of the control problem. While the training process does take some time (about three times longer than applying the MPC), this is not concerning as the scheme can be first trained offline and then applied online with real time applicability and control advantage over the MPC, as indicated in Section 4.4. For more discussions in this regard, the reader may also refer to (Zhou and Gayah, 2021).

To conclude this paper, the limitations and future research directions are pointed out here. First, in the present study the convergence consistence of the MR-RL has been demonstrated in the form of performance curves (Fig. 4). However, in-depth theoretical analyses of the training processes may be needed to shed more light on how and why the scheme can consistently learn from direct interactions with the environment. The authors intend to further look into this, along the lines of (Chen et al., 2022) where control stability of the system and convergence to optimality have been guaranteed by utilizing the Lyapunov theory. Second, numerical simulations have been adopted in this paper to evaluate the MR-RL, as consistent with plentiful previous works. However, a more realistic assessment may be established with microsimulation. On this note, it is worthy of further investigation to see if the MR-RL can transfer to a microsimulation platform and adapt to such environment with continued data feeding and training. It is also a research priority to design a control scheme that can transfer the internalized knowledge to distinctively different settings. Moreover, as alluded in Section 4.3.3, other types of uncertainty relevant to the MFD (e.g., scaling errors, functional form, time-changing feature) might also impact the learning performance of the MR-RL. However, this work cannot inspect all environment uncertainties exhaustively, and thus additional sensitivity analyses might be needed to fully demonstrate the resilience of the MR-RL. Finally, future works should also consider examining the equity issue in the context of perimeter control, perhaps by following the initial steps in (Moshahedi and Kattan, 2023).

ACKNOWLEDGEMENTS

This research was supported by NSF Grant CMMI-1749200.

REFERENCES

Aalipour, A., Kebriaei, H., Ramezani, M., 2019. Analytical Optimal Solution of Perimeter Traffic Flow Control Based on MFD Dynamics: A Pontryagin's Maximum Principle Approach. IEEE Trans. Intell. Transp. Syst. 20, 3224–3234. https://doi.org/10.1109/TITS.2018.2873104

Aboudolas, K., Geroliminis, N., 2013. Perimeter and boundary flow control in multi-reservoir heterogeneous networks. Transp. Res. Part B Methodol. 55, 265–281. https://doi.org/10.1016/j.trb.2013.07.003

Ambühl, L., Menendez, M., 2016. Data fusion algorithm for macroscopic fundamental diagram estimation. Transp. Res. Part C Emerg. Technol. 71, 184–197. https://doi.org/10.1016/J.TRC.2016.07.013

Amirgholy, M., Shahabi, M., Gao, H.O., 2017. Optimal design of sustainable transit systems in congested urban networks: A macroscopic approach. Transp. Res. Part E Logist. Transp. Rev. 103, 261–285. https://doi.org/10.1016/J.TRE.2017.03.006

Araghi, S., Khosravi, A., Johnstone, M., Creighton, D., 2013. Q-learning method for controlling traffic signal phase time in a single intersection. IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC 1261–1265. https://doi.org/10.1109/ITSC.2013.6728404

Buisson, C., Ladier, C., 2009. Exploring the Impact of Homogeneity of Traffic Measurements on the Existence of Macroscopic Fundamental Diagrams. Transp. Res. Rec. J. Transp. Res. Board 2124, 127–136. https://doi.org/10.3141/2124-12

Chang, Y.H., Ho, T., Kaelbling, L., 2003. All learning is local: Multi-agent learning in global reward games, in: Advances in Neural Information Processing Systems 16.

Chen, C., Huang, Y.P., Lam, W.H.K., Pan, T.L., Hsu, S.C., Sumalee, A., Zhong, R.X., 2022. Data efficient reinforcement learning and adaptive optimal perimeter control of network traffic dynamics. Transp. Res. Part C Emerg. Technol. 142, 103759. https://doi.org/10.1016/J.TRC.2022.103759

Choi, S., Yeung, D.Y., Zhang, N., 1999. An Environment Model for Nonstationary Reinforcement Learning, in: Advances in Neural Information Processing Systems 12.

Christianos, F., Papoudakis, G., Rahman, A., Albrecht, S. V., 2021. Scaling Multi-Agent Reinforcement Learning with Selective Parameter Sharing, in: 38th International Conference on Machine Learning. https://doi.org/10.48550/arxiv.2102.07475

Chu, X., Ye, H., 2017. Parameter Sharing Deep Deterministic Policy Gradient for Cooperative Multi-agent Reinforcement Learning.

Csikós, A., Charalambous, T., Farhadi, H., Kulcsár, B., Wymeersch, H., 2017. Network traffic flow optimization under performance constraints. Transp. Res. Part C Emerg. Technol. 83, 120–133. https://doi.org/10.1016/J.TRC.2017.08.002

Daganzo, C.F., 2007. Urban gridlock: Macroscopic modeling and mitigation approaches. Transp. Res. Part B Methodol. 41, 49–62. https://doi.org/10.1016/j.trb.2006.03.001

Daganzo, C.F., Gayah, V. V., Gonzales, E.J., 2011. Macroscopic relations of urban traffic variables: Bifurcations, multivaluedness and instability. Transp. Res. Part B Methodol. 45, 278–288. https://doi.org/10.1016/j.trb.2010.06.006

Daganzo, C.F., Lehe, L.J., 2016. Traffic flow on signalized streets. Transp. Res. Part B Methodol. 90, 56–69. https://doi.org/10.1016/J.TRB.2016.03.010

Daganzo, C.F., Lehe, L.J., 2015. Distance-dependent congestion pricing for downtown zones. Transp. Res. Part B Methodol. 75, 89–99. https://doi.org/10.1016/j.trb.2015.02.010

DePrator, A.J., Hitchcock, O., Gayah, V. V, 2017. Improving urban street network efficiency by prohibiting conflicting left turns at signalized intersections. Transp. Res. Rec. 2622, 58–69.

Du, J., Rakha, H., Gayah, V. V, 2016. Deriving macroscopic fundamental diagrams from probe data: Issues and proposed solutions. Transp. Res. Part C Emerg. Technol. 66, 136–149.

Foerster, J.N., Farquhar, G., Afouras, T., Nardelli, N., Whiteson, S., 2017. Counterfactual Multi-Agent Policy Gradients. 32nd AAAI Conf. Artif. Intell. AAAI 2018 2974–2982. https://doi.org/10.48550/arxiv.1705.08926

Fu, H., Wang, Y., Tang, X., Zheng, N., Geroliminis, N., 2020. Empirical analysis of large-scale multimodal traffic with multi-sensor data. Transp. Res. Part C Emerg. Technol. 118, 102725.

https://doi.org/10.1016/j.trc.2020.102725
Gao, X. (Shirley), Gayah, V. V., 2018. An analytical framework to model uncertainty in urban network dynamics using Macroscopic Fundamental Diagrams. Transp. Res. Part B Methodol. 117, 660–675. https://doi.org/10.1016/j.trb.2017.08.015
Gayah, V., Daganzo, C., 2012. Analytical Capacity Comparison of One-Way and Two-Way Signalized Street Networks: https://doi.org/10.3141/2301-09 2301, 76–85. https://doi.org/10.3141/2301-09
Gayah, V. V., Daganzo, C.F., 2011. Clockwise hysteresis loops in the Macroscopic Fundamental Diagram: An effect of network instability. Transp. Res. Part B Methodol. 45, 643–655. https://doi.org/10.1016/j.trb.2010.11.006
Gayah, V. V, Gao, X.S., Nagle, A.S., 2014. On the impacts of locally adaptive signal control on urban network stability and the macroscopic fundamental diagram. Transp. Res. Part B Methodol. 70, 255–268.
Genser, A., Kouvelas, A., 2022. Dynamic optimal congestion pricing in multi-region urban networks by application of a Multi-Layer-Neural network. Transp. Res. Part C Emerg. Technol. 134, 103485. https://doi.org/10.1016/J.TRC.2021.103485
Geroliminis, N., Daganzo, C.F., 2008. Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. Transp. Res. Part B Methodol. 42, 759–770.
Geroliminis, N., Haddad, J., Ramezani, M., 2013. Optimal perimeter control for two urban regions with macroscopic fundamental diagrams: A model predictive approach. IEEE Trans. Intell. Transp. Syst. 14, 348–359. https://doi.org/10.1109/TITS.2012.2216877
Geroliminis, N., Levinson, D.M., 2009. Cordon Pricing Consistent with the Physics of Overcrowding, Transportation and Traffic Theory 2009: Golden Jubilee. https://doi.org/10.1007/978-1-4419-0820-9_11
Geroliminis, N., Sun, J., 2011. Properties of a well-defined macroscopic fundamental diagram for urban traffic. Transp. Res. Part B Methodol. 45, 605–617. https://doi.org/10.1016/j.trb.2010.11.004
Godfrey, J.W., 1969. The mechanism of a road network. Traffic Eng. Control 11, 323–327.
Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.
Gupta, J.K., Egorov, M., Kochenderfer, M., 2017. Cooperative Multi-agent Control Using Deep Reinforcement Learning. Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 10642 LNAI, 66–83. https://doi.org/10.1007/978-3-319-71682-4_5/COVER/
Haddad, J., 2017a. Optimal coupled and decoupled perimeter control in one-region cities. Control Eng. Pract. 61, 134–148. https://doi.org/10.1016/j.conengprac.2017.01.010
Haddad, J., 2017b. Optimal perimeter control synthesis for two urban regions with aggregate boundary queue dynamics. Transp. Res. Part B Methodol. 96, 1–25. https://doi.org/10.1016/j.trb.2016.10.016
Haddad, J., 2015. Robust constrained control of uncertain macroscopic fundamental diagram networks. Transp. Res. Part C Emerg. Technol. 59, 323–339. https://doi.org/10.1016/J.TRC.2015.05.014
Haddad, J., Geroliminis, N., 2012. On the stability of traffic perimeter control in two-region urban cities. Transp. Res. Part B Methodol. 46, 1159–1176. https://doi.org/10.1016/j.trb.2012.04.004
Haddad, J., Mirkin, B., 2017. Coordinated distributed adaptive perimeter control for large-scale urban road networks. Transp. Res. Part C Emerg. Technol. 77, 495–515.

https://doi.org/10.1016/j.trc.2016.12.002
Haddad, J., Ramezani, M., Geroliminis, N., 2013. Cooperative traffic control of a mixed network with two urban regions and a freeway. Transp. Res. Part B Methodol. 54, 17–36. https://doi.org/10.1016/j.trb.2013.03.007
Haddad, J., Ramezani, M., Geroliminis, N., 2012. Model predictive perimeter control for urban areas with macroscopic fundamental diagrams, in: Proceedings of the American Control Conference. pp. 5757–5762. https://doi.org/10.1109/acc.2012.6314693
Haddad, J., Shraiber, A., 2014. Robust perimeter control design for an urban region. Transp. Res. Part B Methodol. 68, 315–332. https://doi.org/10.1016/j.trb.2014.06.010
Haddad, J., Zheng, Z., 2020. Adaptive perimeter control for multi-region accumulation-based models with state delays. Transp. Res. Part B Methodol. 137, 133–153. https://doi.org/10.1016/J.TRB.2018.05.019
Hajiahmadi, M., Haddad, J., De Schutter, B., Geroliminis, N., 2015. Optimal hybrid perimeter and switching plans control for urban traffic networks. IEEE Trans. Control Syst. Technol. 23, 464–478. https://doi.org/10.1109/TCST.2014.2330997
Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., Meger, D., 2017. Deep Reinforcement Learning that Matters. 32nd AAAI Conf. Artif. Intell. AAAI 2018 3207–3214.
Herman, R., Prigogine, I., 1979. A two-fluid approach to town traffic. Science (80-.). 204 (4389), 148–151. https://doi.org/10.1126/SCIENCE.204.4389.148
Hernandez-Leal, P., Kartal, B., Taylor, M.E., 2018. A Survey and Critique of Multiagent Deep Reinforcement Learning. Auton. Agent. Multi. Agent. Syst. 33, 750–797. https://doi.org/10.1007/s10458-019-09421-1
Hessel, M., Modayil, J., van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., Silver, D., 2017. Rainbow: Combining Improvements in Deep Reinforcement Learning. 32nd AAAI Conf. Artif. Intell. AAAI 2018 3215–3222.
Horgan, D., Quan, J., Budden, D., Barth-Maron, G., Hessel, M., van Hasselt, H., Silver, D., 2018. Distributed Prioritized Experience Replay.
Iqbal, S., Sha, F., 2019. Actor-attention-critic for multi-agent reinforcement learning, in: 36th International Conference on Machine Learning, ICML 2019. International Machine Learning Society (IMLS), pp. 5261–5270.
Ji, Y., Geroliminis, N., 2012. On the spatial partitioning of urban transportation networks. Transp. Res. Part B Methodol. 46, 1639–1656. https://doi.org/10.1016/j.trb.2012.08.005
Jin, C., Allen-Zhu, Z., Bubeck, S., Jordan, M.I., 2018. Is Q-Learning Provably Efficient?, in: Advances in Neural Information Processing Systems, 31.
Keyvan-Ekbatani, M., Kouvelas, A., Papamichail, I., Papageorgiou, M., 2012. Exploiting the fundamental diagram of urban networks for feedback-based gating. Transp. Res. Part B Methodol. 46, 1393–1403. https://doi.org/10.1016/j.trb.2012.06.008
Keyvan-Ekbatani, M., Papageorgiou, M., Knoop, V.L., 2015a. Controller design for gating traffic control in presence of time-delay in urban road networks. Transp. Res. Part C Emerg. Technol. 59, 308–322. https://doi.org/10.1016/j.trc.2015.04.031
Keyvan-Ekbatani, M., Papageorgiou, M., Papamichail, I., 2013. Urban congestion gating control based on reduced operational network fundamental diagrams. Transp. Res. Part C Emerg. Technol. 33, 74–87. https://doi.org/10.1016/J.TRC.2013.04.010
Keyvan-Ekbatani, M., Yildirimoglu, M., Geroliminis, N., Papageorgiou, M., 2015b. Multiple concentric gating traffic control in large-scale urban networks. IEEE Trans. Intell. Transp. Syst. 16, 2141–2154. https://doi.org/10.1109/TITS.2015.2399303

Koller, D., Parr, R., 1999. Computing factored value functions for policies in structured MDPs , in: 16th International Joint Conference on Artificial Intelligence. pp. 1332–1339.

Lauer, M., Riedmiller, M.A., 2000. An Algorithm for Distributed Reinforcement Learning in Cooperative Multi-Agent Systems , in: 17th International Conference on Machine Learning. pp. 535–542.

Laval, J.A., Castrillón, F., 2015. Stochastic approximations for the macroscopic fundamental diagram of urban networks. Transp. Res. Part B Methodol. 81, 904–916. https://doi.org/10.1016/J.TRB.2015.09.002

Leclercq, L., Geroliminis, N., 2013. Estimating MFDs in Simple Networks with Route Choice. Procedia - Soc. Behav. Sci. 80, 99–118. https://doi.org/10.1016/j.sbspro.2013.05.008

Lei, T., Hou, Z., Ren, Y., 2019. Data-Driven Model Free Adaptive Perimeter Control for Multi-Region Urban Traffic Networks With Route Choice. IEEE Trans. Intell. Transp. Syst. 1–12. https://doi.org/10.1109/tits.2019.2921381

Li, Y., Ramezani, M., 2022. Quasi revenue-neutral congestion pricing in cities: Crediting drivers to avoid city centers. Transp. Res. Part C Emerg. Technol. 145, 103932. https://doi.org/10.1016/J.TRC.2022.103932

Li, Y., Yildirimoglu, M., Ramezani, M., 2021. Robust perimeter control with cordon queues and heterogeneous transfer flows. Transp. Res. Part C Emerg. Technol. 126, 103043. https://doi.org/10.1016/j.trc.2021.103043

Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D., 2016. Continuous control with deep reinforcement learning, in: 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings. International Conference on Learning Representations, ICLR.

Lin, L.-J., 1992. Self-improving reactive agents based on reinforcement learning, planning and teaching. Mach. Learn. 8, 293–321. https://doi.org/10.1007/bf00992699

Lopez, C., Krishnakumari, P., Leclercq, L., Chiabaut, N., van Lint, H., 2017. Spatiotemporal Partitioning of Transportation Network Using Travel Time Data. Transp. Res. Rec. J. Transp. Res. Board 2623, 98–107. https://doi.org/10.3141/2623-11

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., Mordatch, I., 2017. Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments. Adv. Neural Inf. Process. Syst. 2017-Decem, 6380–6391.

Lowrie, P.R., 1982. Scats: The Sydney coordinated adaptive traffic system - principles, methodology, algorithms, in: International Conference of Road Traffic Signal. pp. 67–70.

Mahmassani, H., Herman, R., 1984. Dynamic User Equilibrium Departure Time and Route Choice on Idealized Traffic Arterials. https://doi.org/10.1287/trsc.18.4.362 18, 362–384. https://doi.org/10.1287/TRSC.18.4.362

Mahmassani, H.S., Saberi, M., Zockaie, A., 2013. Urban network gridlock: Theory, characteristics, and dynamics. Transp. Res. Part C Emerg. Technol. 36, 480–497. https://doi.org/10.1016/j.trc.2013.07.002

Mazloumian, A., Geroliminis, N., Helbing, D., 2010. The spatial variability of vehicle densities as determinant of urban network capacity 368, 4627–4647. https://doi.org/10.1098/rsta.2010.0099

Menelaou, C., Timotheou, S., Kolios, P., Panayiotou, C.G., 2021. Joint Route Guidance and Demand Management for Real-Time Control of Multi-Regional Traffic Networks. IEEE Trans. Intell. Transp. Syst. https://doi.org/10.1109/TITS.2021.3077870

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A.,

Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D., 2015. Human-level control through deep reinforcement learning. Nature 518, 529–533. https://doi.org/10.1038/nature14236

Mohajerpoor, R., Saberi, M., Vu, H.L., Garoni, T.M., Ramezani, M., 2020. H∞ robust perimeter flow control in urban networks with partial information feedback. Transp. Res. Part B Methodol. 137, 47–73. https://doi.org/10.1016/j.trb.2019.03.010

Moshahedi, N., Kattan, L., 2023. Alpha-fair large-scale urban network control: A perimeter control based on a macroscopic fundamental diagram. Transp. Res. Part C Emerg. Technol. 146, 103961. https://doi.org/10.1016/J.TRC.2022.103961

Nagle, A.S., Gayah, V. V., 2014. Accuracy of Networkwide Traffic States Estimated from Mobile Probe Data. Transp. Res. Rec. J. Transp. Res. Board 1–11. https://doi.org/10.3141/2421-01

Ni, W., Cassidy, M., 2020. City-wide traffic control: Modeling impacts of cordon queues. Transp. Res. Part C Emerg. Technol. 113, 164–175. https://doi.org/10.1016/j.trc.2019.04.024

Oliehoek, F.A., Spaan, M.T.J., Vlassis, N., 2008. Optimal and Approximate Q-value Functions for Decentralized POMDPs. J. Artif. Intell. Res. 32, 289–353. https://doi.org/10.1613/jair.2447

OroojlooyJadid, A., Hajinezhad, D., 2019. A Review of Cooperative Multi-Agent Deep Reinforcement Learning.

Ortigosa, J., Gayah, V. V., Menendez, M., 2017. Analysis of one-way and two-way street configurations on urban grid networks. https://doi.org/10.1080/21680566.2017.1337528 7, 61–81. https://doi.org/10.1080/21680566.2017.1337528

Paipuri, M., Xu, Y., González, M.C., Leclercq, L., 2020. Estimating MFDs, trip lengths and path flow distributions in a multi-region setting using mobile phone data. Transp. Res. Part C Emerg. Technol. 118, 102709. https://doi.org/10.1016/j.trc.2020.102709

Peng, B., Rashid, T., de Witt, C.A.S., Kamienny, P.-A., Torr, P.H.S., Böhmer, W., Whiteson, S., 2021. FACMAC: Factored Multi-Agent Centralised Policy Gradients, in: The 35th Conference on Neural Information Processing Systems.

Ramezani, M., Haddad, J., Geroliminis, N., 2015. Dynamics of heterogeneity in urban networks: Aggregated traffic modeling and hierarchical control. Transp. Res. Part B Methodol. 74, 1–19. https://doi.org/10.1016/j.trb.2014.12.010

Rashid, T., Farquhar, G., Peng, B., Whiteson, S., 2020. Weighted QMIX: Expanding Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning, in: Advances in Neural Information Processing Systems. Neural information processing systems foundation, pp. 10199–10210. https://doi.org/10.48550/arxiv.2006.10800

Rashid, T., Samvelyan, M., de Witt, C.S., Farquhar, G., Foerster, J., Whiteson, S., 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning, in: International Conference of Machine Learning.

Ren, Y., Hou, Z., Sirmatel, I.I., Geroliminis, N., 2020. Data driven model free adaptive iterative learning perimeter control for large-scale urban road networks. Transp. Res. Part C Emerg. Technol. 115, 102618. https://doi.org/10.1016/j.trc.2020.102618

Robertson, D.I., Bretherton, R.D., 1991. Optimizing Networks of Traffic Signals in Real Time—The SCOOT Method. IEEE Trans. Veh. Technol. 40, 11–15. https://doi.org/10.1109/25.69966

Saeedmanesh, M., Geroliminis, N., 2017. Dynamic clustering and propagation of congestion in heterogeneously congested urban traffic networks. Transp. Res. Part B Methodol. 105, 193–211.

Saeedmanesh, M., Geroliminis, N., 2016. Clustering of heterogeneous networks with directional flows based on "Snake" similarities. Transp. Res. Part B Methodol. 91, 250–269. https://doi.org/10.1016/j.trb.2016.05.008

Schaul, T., Quan, J., Antonoglou, I., Silver, D., 2016. Prioritized experience replay, in: 4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings. International Conference on Learning Representations, ICLR.

Sirmatel, I.I., Geroliminis, N., 2021. Stabilization of city-scale road traffic networks via macroscopic fundamental diagram-based model predictive perimeter control. Control Eng. Pract. 109, 104750. https://doi.org/10.1016/j.conengprac.2021.104750

Sirmatel, I.I., Geroliminis, N., 2018. Economic Model Predictive Control of Large-Scale Urban Road Networks via Perimeter Control and Regional Route Guidance. IEEE Trans. Intell. Transp. Syst. 19, 1112–1121. https://doi.org/10.1109/TITS.2017.2716541

Sirmatel, I.I., Tsitsokas, D., Kouvelas, A., Geroliminis, N., 2021. Modeling, estimation, and control in large-scale urban road networks with remaining travel distance dynamics. Transp. Res. Part C Emerg. Technol. 128, 103157. https://doi.org/10.1016/J.TRC.2021.103157

Small, K.A., Chu, X., 2003. Hypercongestion. J. Transp. Econ. Policy 37 (3), 319–352.

Son, K., Kim, D., Kang, W.J., Hostallero, D., Yi, Y., 2019. QTRAN: Learning to Factorize with Transformation for Cooperative Multi-Agent Reinforcement Learning, in: 36th International Conference on Machine Learning. International Machine Learning Society (IMLS), pp. 5887–5896. https://doi.org/10.48550/arxiv.1905.05408

Su, Z.C., Chow, A.H.F., Zheng, N., Huang, Y.P., Liang, E.M., Zhong, R.X., 2020. Neuro-dynamic programming for optimal control of macroscopic fundamental diagram systems. Transp. Res. Part C Emerg. Technol. 116, 102628. https://doi.org/10.1016/j.trc.2020.102628

Sunehag, P., Lever, G., Gruslys, A., Marian Czarnecki, W., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J.Z., Tuyls, K., Graepel, T., 2018. Value-Decomposition Networks For Cooperative Multi-Agent Learning Based On Team Reward, in: 17th International Conference on Autonomous Agents and MultiAgent Systems. pp. 2085–2087. https://doi.org/10.5555/3237383.3238080

Sutton, R.S., Barto, A.G., 2018. Reinforcement learning: An introduction. MIT Press.

Tan, M., 1993. Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents, in: 10th International Conference on Machine Learning Proceedings. Elsevier, pp. 330–337. https://doi.org/10.1016/B978-1-55860-307-3.50049-6

Terry, J.K., Grammel, N., Hari, A., Santos, L., 2020. Parameter Sharing is Surprisingly Useful for Multi-Agent Deep Reinforcement Learning.

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop Divide the Gradient by a Running Average of Its Recent Magnitude. COURSERA Neural Networks Mach. Learn. 4, 26–31.

Tilg, G., Amini, S., Busch, F., 2020. Evaluation of analytical approximation methods for the macroscopic fundamental diagram. Transp. Res. Part C Emerg. Technol. 114, 1–19. https://doi.org/10.1016/J.TRC.2020.02.003

Tsitsiklis, J.N., Roy, B. Van, 1997. An Analysis of Temporal-Difference Learning with Function Approximation, IEEE Transactions on Automatic Control.

van Hasselt, H., 2010. Double Q-learning, in: Advances in Neural Information Processing Systems. pp. 2613–2621.

van Hasselt, H., Doron, Y., Strub, F., Hessel, M., Sonnerat, N., Modayil, J., 2018. Deep Reinforcement Learning and the Deadly Triad.

van Hasselt, H., Guez, A., Silver, D., 2015. Deep Reinforcement Learning with Double Q-learning.

30th AAAI Conf. Artif. Intell. AAAI 2016 2094–2100.
Varaiya, P., 2013. Max pressure control of a network of signalized intersections. Transp. Res. Part C Emerg. Technol. 36, 177–195. https://doi.org/10.1016/j.trc.2013.08.014
Wang, Y., Han, B., Wang, T., Dong, H., Zhang, C., 2021. Off-Policy Multi-Agent Decomposed Policy Gradients, in: International Conference on Learning Representations.
Wang, Z., Schaul, T., Hessel, M., van Hasselt, H., Lanctot, M., de Freitas, N., 2015. Dueling Network Architectures for Deep Reinforcement Learning. 33rd Int. Conf. Mach. Learn. ICML 2016 4, 2939–2947.
Watkins, C.J.C.H., Dayan, P., 1992. Q-learning. Mach. Learn. 8, 279–292. https://doi.org/10.1007/bf00992698
Wen, Y., Yang, Y., Luo, R., Wang, J., Pan, W., 2019. Probabilistic Recursive Reasoning for Multi-Agent Reinforcement Learning. 7th Int. Conf. Learn. Represent. https://doi.org/10.48550/arxiv.1901.09207
Williams, J.C., Mahmassani, H.S., Herman, R., 1987. Urban traffic network flow models. Transp. Res. Rec. 1112, 78–88.
Yildirimoglu, M., Ramezani, M., Geroliminis, N., 2015. Equilibrium analysis and route guidance in large-scale networks with MFD dynamics. Transp. Res. Part C Emerg. Technol. 59, 404–420. https://doi.org/10.1016/j.trc.2015.05.009
Yildirimoglu, M., Sirmatel, I.I., Geroliminis, N., 2018. Hierarchical control of heterogeneous large-scale urban road networks via path assignment and regional route guidance. Transp. Res. Part B Methodol. 118, 106–123. https://doi.org/10.1016/j.trb.2018.10.007
Zheng, N., Waraich, R.A., Axhausen, K.W., Geroliminis, N., 2012. A dynamic cordon pricing scheme combining the Macroscopic Fundamental Diagram and an agent-based traffic model. Transp. Res. Part A Policy Pract. 46, 1291–1303. https://doi.org/10.1016/j.tra.2012.05.006
Zhong, R.X., Chen, C., Huang, Y.P., Sumalee, A., Lam, W.H.K., Xu, D.B., 2018a. Robust perimeter control for two urban regions with macroscopic fundamental diagrams: A control-Lyapunov function approach. Transp. Res. Part B Methodol. 117, 687–707. https://doi.org/10.1016/j.trb.2017.09.008
Zhong, R.X., Huang, Y.P., Chen, C., Lam, W.H.K., Xu, D.B., Sumalee, A., 2018b. Boundary conditions and behavior of the macroscopic fundamental diagram based network traffic dynamics: A control systems perspective. Transp. Res. Part B Methodol. 111, 327–355. https://doi.org/10.1016/J.TRB.2018.02.016
Zhou, D., Gayah, V. V., 2021. Model-free perimeter metering control for two-region urban networks using deep reinforcement learning. Transp. Res. Part C Emerg. Technol. 124, 102949.