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

Dongqin Zhou
Department of Civil and Environmental Engineering
The Pennsylvania State University, University Park, PA, 16802
Email: dongqin.zhou@psu.edu

Vikash V. Gayah*
Department of Civil and Environmental Engineering
The Pennsylvania State University, University Park, PA, 16802
Email: gayah@engr.psu.edu

* Corresponding author.

Word Count: 6868 words + 1 table (250 words per table) = 7118 words

Submitted [*07/07/2022*]

ABSTRACT

Perimeter metering control is a convenient strategy to mitigate urban congestion by manipulating vehicular movements around regional perimeters without modeling the detailed behaviors and interactions. Multi-region perimeter metering control holds promise for efficient management of urban traffic in large-scale networks. However, most existing methods for multi-region control require knowledge of either the traffic dynamics or network properties (i.e., the critical accumulations), and completely model-free techniques are still lacking in the literature. To fill this gap, this paper proposes a novel control scheme based upon model-free multi-agent reinforcement learning. The 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 specialized problem reformulation. The effectiveness of the scheme is demonstrated with numerical experiment conducted on a seven-region urban network.

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

INTRODUCTION

The macroscopic fundamental diagram (MFD) has been recently shown promising in terms of the aggregate traffic dynamics models and network-level control schemes it facilitates (*1–5*), both of which are critical for large-scale urban traffic management. The initial theoretical investigation of the MFD dates back to the 1960s (*6*), but its existence was not verified until recently (*1*, *2*). These seminal works have since inspired a large number of research efforts on the existence analysis (*7–9*) and estimation of MFDs (*10–17*). Other than the derivations, properties of well-defined MFDs have also been examined extensively (*3*, *18–20*). These references have shown that urban networks are subject to instability, hysteresis, and bifurcation phenomena with heterogeneous distribution of vehicle presence. Fortunately, network partitioning can be applied to divide a large heterogeneous area into several smaller regions such that congestion homogeneity is maintained for each region (*21–24*).

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 control schemes, e.g., congestion pricing (*25–27*), route guidance (*28–30*), and others. The most common application utilizing MFDs is perimeter metering control (PMC), which entails regulating the inter-regional transfer flows using traffic signals residing on the boundary of the neighboring regions. The first examination of perimeter control was presented in (*1*) for a single region, which formulated the traffic dynamics with MFDs and proposed the optimal Bang-Bang control policy. Perimeter control for two-region networks, as first formulated in (*31*), has also attracted substantial research interests over the years. For example, analytical and data-driven approaches have been adopted to design solution schemes (*4*, *32–35*), while stability and modeling uncertainty are examined in (*36–40*).

Another line of PMC research pertains to the efficient operations of traffic flows in a multi-region setting. Early endeavors in this vein include (*41*, *42*) where the traffic dynamics are formulated for a multi-reservoir and a mixed network. However, in these works the receiving capacity constraint was neglected; and it was later integrated in (*43*) where a region-based and subregion-based MFD models were proposed. To further enhance traffic dynamics modeling, numerous works have been conducted to consider: boundary queue dynamics (*37*, *44*, *45*), time-delay effects (*46*), demand stochasticity (*47*), and parameter uncertainty in MFDs (*48*). Multi-region PMC embodies great potential for city-level traffic management, for which various solution methods have been proposed in the literature. Examples include linear quadratic regulator (*41*, *44*), model predictive control (*29*, *43*), model-free adaptive control (*49*, *50*), and others. Importantly, (*49*, *50*) proposed solution schemes that are data-driven and model-free, yet the critical accumulation is still explicitly blended into the controller designs. Other mentioned methods, on the other hand, are heavily dependent on knowledge of the environment dynamics. Therefore, it is a high research priority to develop a completely model-free method for multi-region perimeter control.

This paper bridges this gap by proposing a model-free scheme based on 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 PMC problems (*1*, *32*, *44*). The effectiveness of the proposed scheme is demonstrated via comparison with the model predictive control by conducting perimeter control on a seven-region network.

The remainder of the paper is structured as follows. The next section provides the traffic dynamics modeling of multi-region urban networks. Then, the proposed scheme is explained in detail, followed by the numerical experiment results. Finally, concluding remarks are presented in the last section.

TRAFFIC DYNAMICS OF MULTI-REGION URBAN NETWORKS

The general traffic dynamics for an R-region urban network are introduced here; see Fig. 1 for an illustration of a network with $R = 7$. The regions are assumed to be homogenous in terms of congestion distribution;

however, if this assumption does not hold, network partitioning can be applied to maintain homogeneity (*21–23*). As such, a well-defined MFD function $f_i(n_i(t))$ that relates trip completion rate to the regional accumulation $n_i(t)$ is assumed to be able to model each region. The evolution of accumulations in region i can then be expressed as (*28*, *43*):

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

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

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

where $\mathcal{R} = \{1,2,\cdots,R\}$ denotes the network, n_{ij} and q_{ij} are the number of vehicles and traffic demands in R_i destined for R_j, with n_{ii} and q_{ii} defined similarly. u_{ih} is the perimeter controller ($u_{ih} \in [u_{min}, u_{max}]$ with $0 \leq u_{min} < u_{max} \leq 1$) that specifies the allowable ratio of transfer flow from R_i to R_h, 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, as 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 $\theta_{ihj}(t) \in [0,1]$ is the route choice term for vehicles traveling from R_i to R_j via the next region h (hence $\sum_{h \in N_i} \theta_{ihj}(t) = 1$) and is inversely related to the travel time of paths utilizing R_h.

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 (*28*, *43*):

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

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.

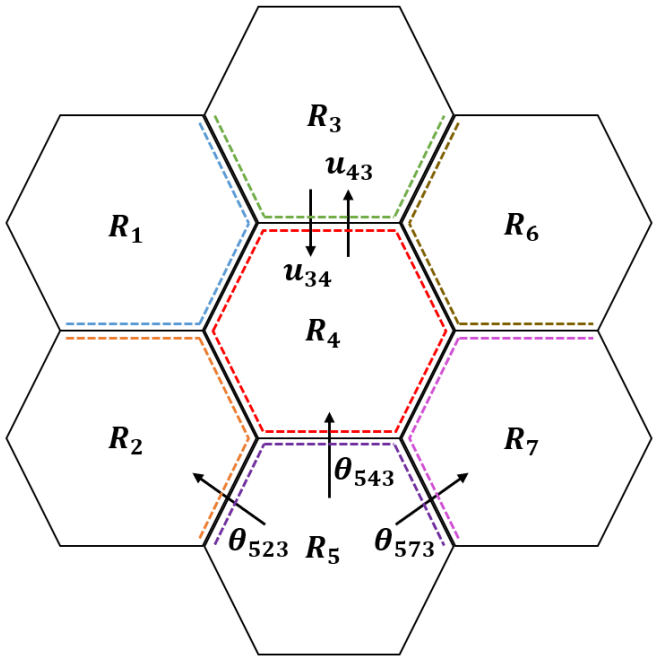

Fig. 1. A seven-region urban network. The dash lines represent the perimeter controllers.

METHODOLOGY

This section reformulates the seven-region control problem in the context of multi-agent reinforcement learning (MARL), followed by detailed explanations of the proposed scheme.

Problem Reformulation

The multi-region PMC problem can be viewed as a cooperative multi-agent task where a group of n agents ($\mathcal{N} = \{1, \cdots, n\}$) learn to achieve a common goal via individualized interactions with the same environment. 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. Formally, the multi-region PMC problem is presented as a decentralized partially observable Markov decision process 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 contains the global information about the entire network. However, due to partial observability, the agents can only observe local instances of the state and act based on these 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. The observation o_t includes similar information for a pair of regions, i.e., two regional accumulations, traffic demands between the two regions, and the congestion indicator.

- **Action space**, $\boldsymbol{\mathcal{U}}$. The Bang-Bang control policy, i.e., to alternate the controller between the minimum and maximum values, has been shown as the optimal form of actions for perimeter control (*1*, *32*, *44*). Motivated by this, the agents assume the Bang-Bang type actions, i.e., either u_{min} or u_{max} for each perimeter controller.

- **Transition dynamics**, $\boldsymbol{\mathcal{P}}$. The selected actions of the individual agents form a joint action, which is executed in the environment and leads a transition to a new state, according to the dynamics

$\mathcal{P}(s_{t+1}|s_t, \boldsymbol{u}_t): \boldsymbol{\mathcal{S}} \times \boldsymbol{\mathcal{U}} \to \boldsymbol{\mathcal{S}}$. 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.

- **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. 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 time steps.

Algorithms

This section first introduces a canonical single-agent deep reinforcement learning method, which provide theoretical background for the proposed scheme that is to be explained subsequently.

Double Deep Q Networks (Double DQN)

As a foundational reinforcement learning technique, Q-learning (*51*) 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 $Q(s_t, u_t)$ that 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 iteratively, 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.

Despite its popularity, the tabular form of Q-learning limits its applicability to large problems that feature an abundance of state-action pairs. To mitigate this, research efforts have long been performed on value function approximation and its stability analysis (*52–54*), with the first success presented in the Deep Q-Networks (DQN) algorithm (*55*). This work has revealed the significant potential of deep reinforcement learning and has since inspired the development of more advanced learning techniques (*56–60*). Despite its success, however, the DQN method is prone to overestimation of the Q values (*56*). 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:

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

where $Q(:,:;\boldsymbol{\theta}_t)$ and $Q(:,:;\boldsymbol{\theta}_t^-)$ represent the Q- and target networks. The target network is a periodic copy of the Q-network, and its utilization provides relatively static targets and helps with learning stability.

Reinforcement learning controller design for multi-region perimeter control (MR-RL)

The success of single-agent RL has significantly boosted its extension to multi-agent systems. However, directly applying single-agent techniques to multi-agent tasks is generally infeasible due to the curse of dimensionality which renders it difficult to estimate the joint Q value. The most common paradigm to alleviate this issue is centralized training with decentralized execution (CTDE, (*61*)), which adopts full centralization conditioning on the global state during learning and decentralization conditioning on local

observations during action taking. Despite notable experimental results, the CTDE paradigm has a major scalability limitation due to fully centralized training. As such, value function decomposition has been proposed (*62*) as a mitigation strategy. Specifically, these methods factorize the centralized Q value as a function of the local Q values estimated by the agents conditioning on the local observations and actions. Importantly, this factorization ensures scalability as the joint action information is no longer required. As a representative method, QMIX (*63*) 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., (*64*).

The multi-region control problem is a fully cooperative task where all agents collaborate to achieve the highest trip completion. Naturally, higher local trip completions lead to higher total trip completion. Hence, the monotonicity assumption holds and the QMIX is adopted to devise the proposed scheme, as denoted by MR-RL that stands for Multi-Region Reinforcement Learning. The learning algorithm for the proposed MR-RL is shown in Fig. 2, and its building components are explained in the following.

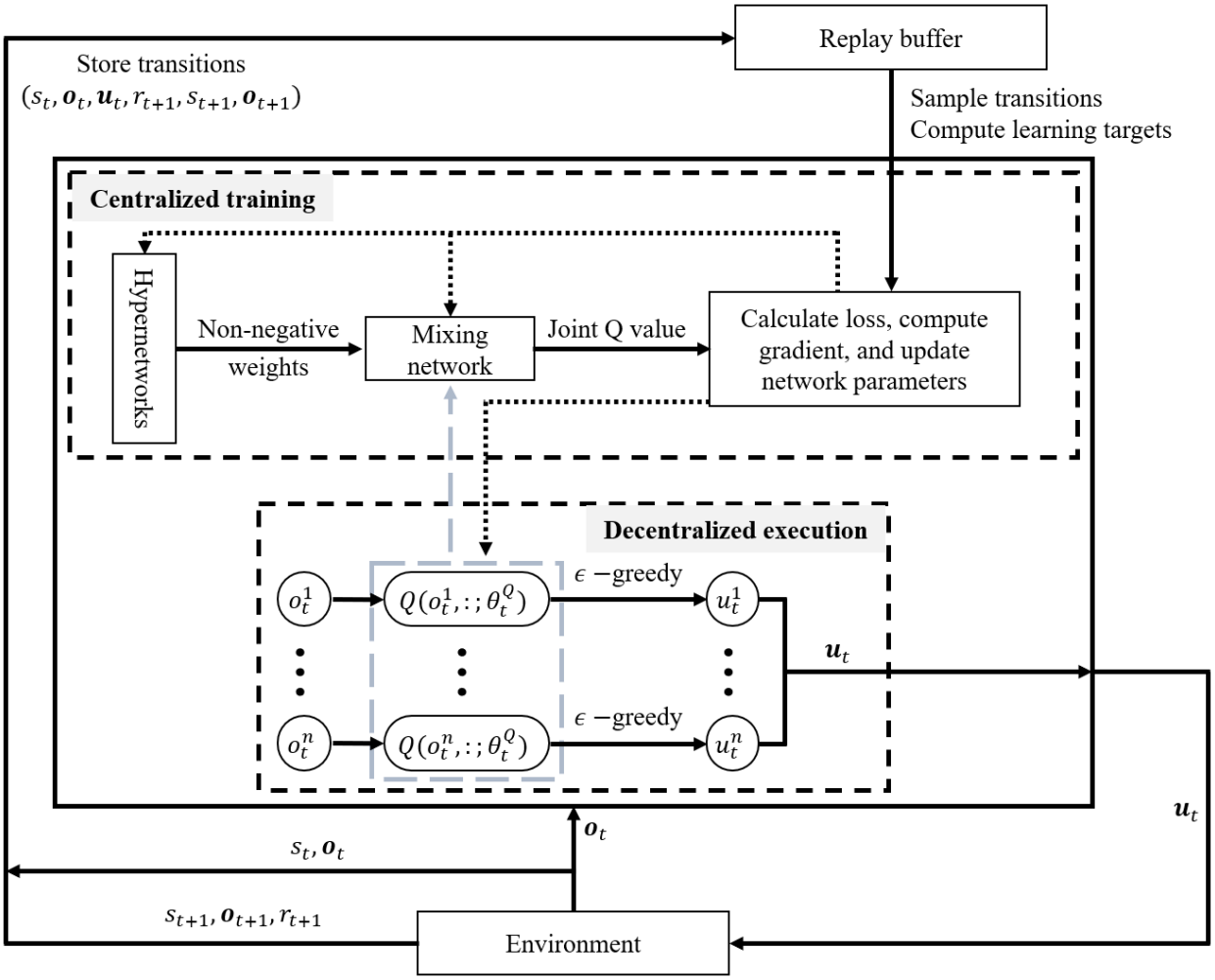

Fig. 2. A diagram of the learning algorithm for the MR-RL scheme.

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 (*59*, *63*, *65*). 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 is the network parameters. The agents receive as input the

local observations o^a and estimate the 4-dimensional local Q values for the two perimeter controllers (each controller has two options, u_{min} and u_{max}). The local actions can then be derived with the $\epsilon-$greedy strategy regarding these 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.

The mixing network, as denoted by $m(\cdot)$, combines the estimated local Q values to provide the joint Q value and is central to value decomposition methods. The QMIX algorithm enforces monotonic decomposition by utilizing non-negative weights for the mixing network, and separate hypernetworks are exploited to produce such weights. Specifically, the hypernetworks take into the global state s_t and generate the weights in a feed-forward structure with 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 (see Fig. 2) 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 whose inputs include the global state for non-negative weights construction. The major distinction between this and the Double DQN target (Eq. (9)) is the mixing network which involves a group of local Q values. Importantly though, this additional complexity precipitates significantly improved scalability to larger multi-agent systems. The network parameters of the MR-RL scheme 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 batch size of sampled transitions from the replay buffer used for network updates, and Y_t^i is the learning target for the i-th transition.

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 (*66*) and later consolidated in (*55*) 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., 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 (*65*) 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 that, 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. Finally, the list of hyperparameters along with their values is presented in Table 1.

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 ϵ −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 ← a batch of b transitions randomly drawn from the buffer
  Periodically 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

Table 1. List of hyperparameters and their values

Hyperparameter	Value	Description
Iteration number (I)	250	The number of training iterations
Generator number (G)	6	The number of experiment 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 ϵ − greedy exploration
ϵ decay	0.98	The exponential decay factor for the ϵ value
Final ϵ	0.01	The final value of ϵ in ϵ − 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 update the target networks

EXPERIMENTS

In this section, the proposed MR-RL is applied for perimeter control on a seven-region network (with configurations shown in Fig. 1), and its performance is compared with two benchmarking methods to evaluate its effectiveness. Note that, there are 24 perimeter controllers for 12 pairs of neighboring regions in the network (see again Fig. 1), hence 12 agents are utilized.

Experiment Setup

In this work, a unit MFD consistent with the one observed in Yokohama (*2*) is utilized, with critical and jam values of 8,240 veh and 34,000 veh, respectively (*35*, *67*). Note that, the unit MFD assumes a piecewise functional form rather than a 3-rd polynomial for the traffic dynamics to be more realistic. For all experiments, each region is modeled with a slightly scaled (within $\pm 10\%$) version of unit MFD, as similarly done in (*29*). In addition, the parameters for the boundary capacity constraints are set to $C_{ih}^{max} = 4.6$ veh/s and $\alpha = 0.48$.

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 and relatively small demands among the periphery regions, which mimics traffic conditions during a morning peak. 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, the boundary values for the perimeter controllers are $u_{min} = 0.1, u_{max} = 0.9$. Finally, region 4 assumes a congested initial state with an accumulation value of 8,750 veh while all other regions are assumed to be uncongested initially with accumulations of 3,850 veh.

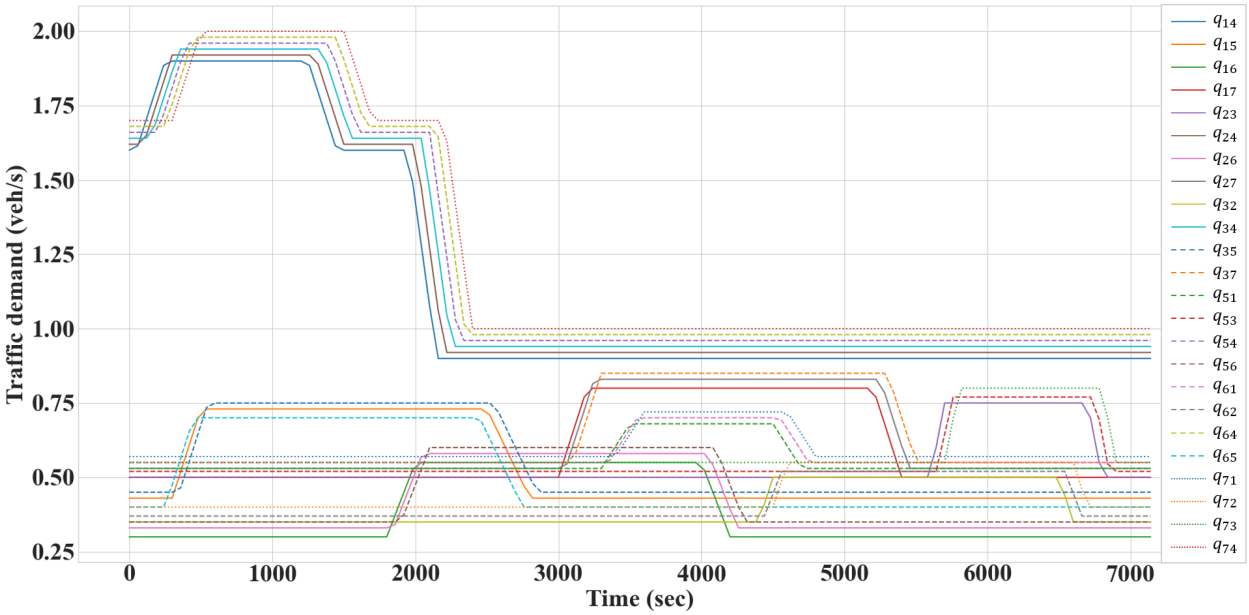

Fig. 3. Traffic demands

Two benchmarking methods, model predictive control (MPC) and no control (NC), are adopted to compare the performances with the MR-RL, in terms of the cumulative trip completion (CTC) they achieve. The NC method does not impose limitations on the transfer flows and instead uses 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 a 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 (*29*) with a control horizon of 2 and a prediction horizon of 3, as similarly adopted in numerous other works (*5*, *49*, *50*). Note that for the seven-region perimeter control problem, a larger prediction horizon does not necessarily lead to better control performance since the solution space of the formulated nonconvex optimization problem becomes so large that finding the global optimum is increasingly difficult.

Experiment Results

For the experiment scenario considered, 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 shad ed area respectively represent the mean and 95% confidence interval of the control gains (in terms of CTC). The MPC and NC are also run five times to report their performance curves, but the curves are relatively invariant as they are not learning-based methods. Note that there is built-in randomness associated with the travel time calculation of different paths used to determine the route choice term (see Eq. (4)); hence these two methods also exhibit a (negligibly small) range of CTC values.

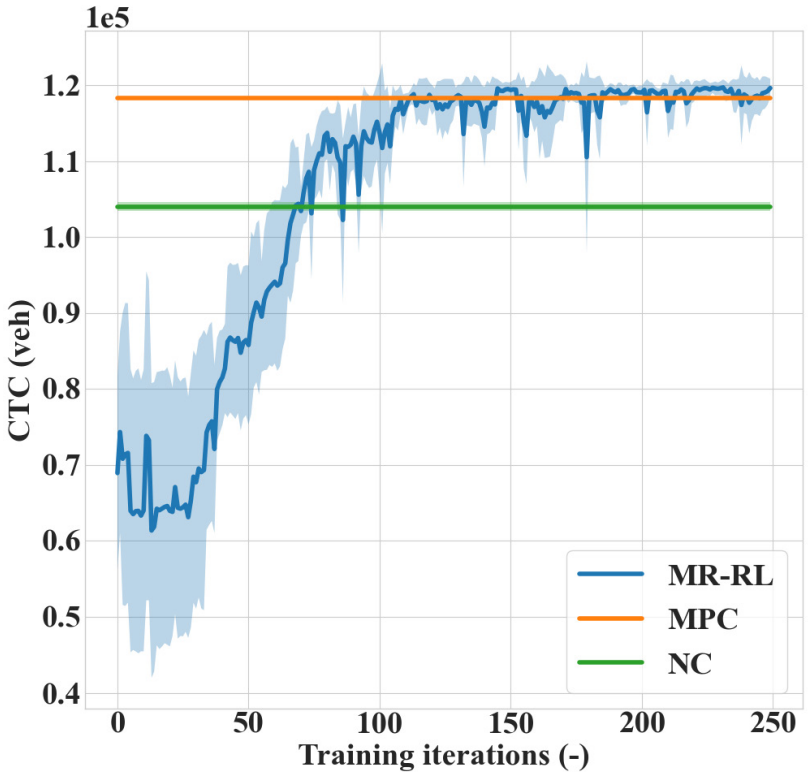

Fig. 4. Performance curves of different methods for the no uncertainty scenario.

As shown in Fig. 4, the NC method realizes the lower CTC value, which 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 better than) the MPC. This showcases the significant

potential of model-free reinforcement learning methods over model-based approaches. The MPC is an optimization-based approach 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 slight underperformance to the proposed scheme, though the MPC could theoretically be the optimal control technique with improved performances via guaranteed global optimum finding. However, implementing 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. Therefore, the initial control performances are not important as only the fully trained MR-RL scheme will be applied.

To further demonstrate the effectiveness of the MR-RL, its control outcomes are examined more carefully in the following. Fig. 5 presents the evolution of accumulations for each region, as achieved by different control methods. The critical accumulations are also provided in dotted lines which helps determine the congestion situation for the regions. As can be observed, under the NC method, region 4 becomes extremely congested in the end 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 flows are much larger than the others. However, notable congestion in region 4 leads to 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 with the lowest CTC. Comparatively, 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 as the most destination-loaded region (i.e., region 4) are protected from over-congestion, which is consistent with the AB strategy proposed in (*1*). Finally, the similarity of accumulations in all regions between the MPC and MR-RL indicates great comparability between them.

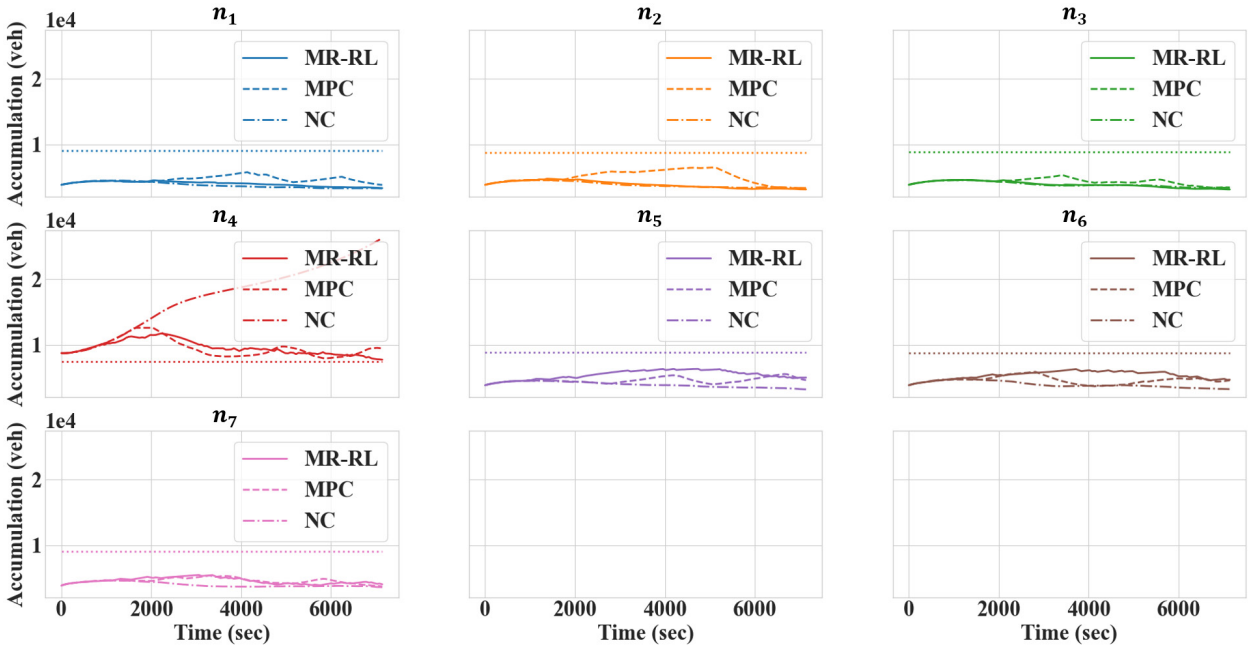

Fig. 5. Accumulation plots for all regions. The dotted lines represent the critical accumulations.

Fig. 6 presents the control actions u_{i4} of the MPC and MR-RL, while all other controllers are omitted from the presentation here. The 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 reveals that the MR-RL 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 MPC actions. Similarly, the accumulations in regions 1, 2, and 3 are smaller than those realized by the MPC since more transfer flows can be completed under the MR-RL policy that exhibits looser control (i.e., more u_{max} values in the actions). In addition, both methods select the maximum value for all controllers in the initial period, which is sensible as there does not exist pronounced congestion within the network (even region 4 is only moderately congested). Overall, these control actions help explain the resulting evolutions of accumulations, which also illustrates how the proposed MR-RL is comparable to the MPC.

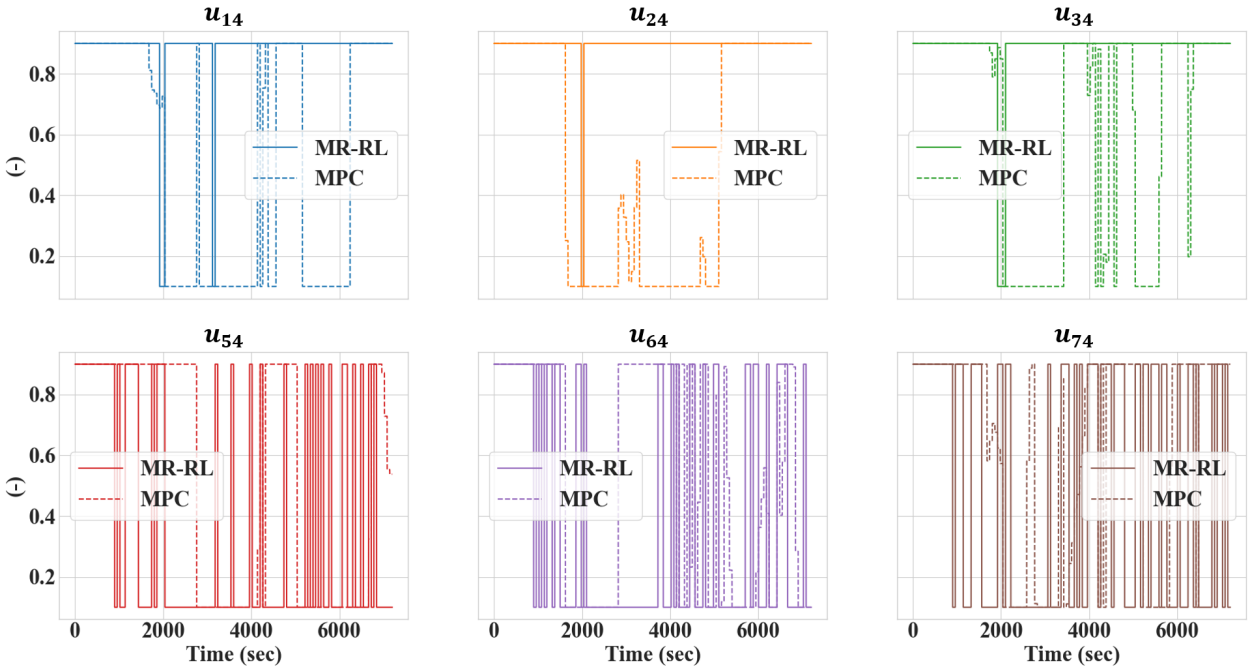

Fig. 6. Control actions u_{i4} of the MPC and proposed MR-RL.

CONCLUDING REMARKS

This paper presents a novel MR-RL scheme for multi-region perimeter control based on model-free multi-agent reinforcement learning. Specifically, the proposed MR-RL features value function decomposition, which significantly helps with scalability, recent breakthrough of single-agent deep reinforcement learning (such as the Ape-X architecture, double Q-learning update rule, experience replay, target networks, etc.), and suitable problem reformulation governed by domain expertise (e.g., the Bang-Bang type action space design). The control efficacy of the MR-RL is demonstrated with numerical experiments on a simulated seven-region urban network, and the results suggest that the scheme can consistently learn and converge to final control performances that are comparable to the MPC method. It is worth reiterating that, the proposed MR-RL is completely model-free which does not require a priori information about the environment. Hence, it is not affected by a wide range of modeling mismatch, e.g., scaling errors and the time-changing feature

of MFDs, to which recent data-driven approaches are liable (*49*, *50*). Further, note that this paper presents the first examination of completely model-free methods on seven-region perimeter control.

Future works to this paper could consider integrating low-level intra-regional signal control, which holds promise for realistic city-wide traffic management. In addition, it would be interesting to investigate whether the pretrained scheme from numerical simulations can be transferred to a microsimulation platform and continue its learning course with real-time interactions. This could help evaluate an extra level of transferability for the MR-RL and its ability to keep learning in a different platform.

ACKNOWLEDGEMENTS

This research was supported by NSF Grant CMMI-1749200.

AUTHOR CONTRIBUTIONS

The authors confirm contribution to the paper as follows: study conception and design: VG, DZ; analysis and interpretation of results: VG, DZ; draft manuscript preparation: VG, DZ. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Daganzo, C. F. Urban Gridlock: Macroscopic Modeling and Mitigation Approaches. *Transportation Research Part B: Methodological*, Vol. 41, No. 1, 2007, pp. 49–62. https://doi.org/10.1016/j.trb.2006.03.001.
2. Geroliminis, N., and C. F. Daganzo. Existence of Urban-Scale Macroscopic Fundamental Diagrams: Some Experimental Findings. *Transportation Research Part B: Methodological*, Vol. 42, No. 9, 2008, pp. 759–770.
3. Daganzo, C. F., V. V. Gayah, and E. J. Gonzales. Macroscopic Relations of Urban Traffic Variables: Bifurcations, Multivaluedness and Instability. *Transportation Research Part B: Methodological*, Vol. 45, No. 1, 2011, pp. 278–288. https://doi.org/10.1016/j.trb.2010.06.006.
4. Geroliminis, N., J. Haddad, and M. Ramezani. Optimal Perimeter Control for Two Urban Regions with Macroscopic Fundamental Diagrams: A Model Predictive Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 14, No. 1, 2013, pp. 348–359. https://doi.org/10.1109/TITS.2012.2216877.
5. Yildirimoglu, M., I. I. Sirmatel, and N. Geroliminis. Hierarchical Control of Heterogeneous Large-Scale Urban Road Networks via Path Assignment and Regional Route Guidance. *Transportation Research Part B: Methodological*, Vol. 118, 2018, pp. 106–123. https://doi.org/10.1016/j.trb.2018.10.007.
6. Godfrey, J. W. The Mechanism of a Road Network. *Traffic Engineering & Control*, Vol. 11, No. 7, 1969, pp. 323–327.
7. Fu, H., Y. Wang, X. Tang, N. Zheng, and N. Geroliminis. Empirical Analysis of Large-Scale Multimodal Traffic with Multi-Sensor Data. *Transportation Research Part C: Emerging Technologies*, Vol. 118, 2020, p. 102725. https://doi.org/10.1016/j.trc.2020.102725.
8. Geroliminis, N., and J. Sun. Properties of a Well-Defined Macroscopic Fundamental Diagram for Urban Traffic. *Transportation Research Part B: Methodological*, Vol. 45, No. 3, 2011, pp. 605–617. https://doi.org/10.1016/j.trb.2010.11.004.
9. Paipuri, M., Y. Xu, M. C. González, and L. Leclercq. Estimating MFDs, Trip Lengths and Path Flow Distributions in a Multi-Region Setting Using Mobile Phone Data. *Transportation Research Part C: Emerging Technologies*, Vol. 118, 2020, p. 102709.

https://doi.org/10.1016/j.trc.2020.102709.
10. Ambühl, L., and M. Menendez. Data Fusion Algorithm for Macroscopic Fundamental Diagram Estimation. *Transportation Research Part C: Emerging Technologies*, Vol. 71, 2016, pp. 184–197. https://doi.org/10.1016/J.TRC.2016.07.013.
11. Buisson, C., and C. Ladier. Exploring the Impact of Homogeneity of Traffic Measurements on the Existence of Macroscopic Fundamental Diagrams. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2124, No. 1, 2009, pp. 127–136. https://doi.org/10.3141/2124-12.
12. Nagle, A. S., and V. V. Gayah. Accuracy of Networkwide Traffic States Estimated from Mobile Probe Data. *Transportation Research Record: Journal of the Transportation Research Board*, No. 2421, 2014, pp. 1–11. https://doi.org/10.3141/2421-01.
13. Du, J., H. Rakha, and V. V Gayah. Deriving Macroscopic Fundamental Diagrams from Probe Data: Issues and Proposed Solutions. *Transportation Research Part C: Emerging Technologies*, Vol. 66, 2016, pp. 136–149.
14. Daganzo, C. F., and L. J. Lehe. Traffic Flow on Signalized Streets. *Transportation Research Part B: Methodological*, Vol. 90, 2016, pp. 56–69. https://doi.org/10.1016/J.TRB.2016.03.010.
15. Laval, J. A., and F. Castrillón. Stochastic Approximations for the Macroscopic Fundamental Diagram of Urban Networks. *Transportation Research Part B: Methodological*, Vol. 81, 2015, pp. 904–916. https://doi.org/10.1016/J.TRB.2015.09.002.
16. Leclercq, L., and N. Geroliminis. Estimating MFDs in Simple Networks with Route Choice. *Procedia - Social and Behavioral Sciences*, Vol. 80, 2013, pp. 99–118. https://doi.org/10.1016/j.sbspro.2013.05.008.
17. Tilg, G., S. Amini, and F. Busch. Evaluation of Analytical Approximation Methods for the Macroscopic Fundamental Diagram. *Transportation Research Part C: Emerging Technologies*, Vol. 114, 2020, pp. 1–19. https://doi.org/10.1016/J.TRC.2020.02.003.
18. Gayah, V. V., and C. F. Daganzo. Clockwise Hysteresis Loops in the Macroscopic Fundamental Diagram: An Effect of Network Instability. *Transportation Research Part B: Methodological*, Vol. 45, No. 4, 2011, pp. 643–655. https://doi.org/10.1016/j.trb.2010.11.006.
19. Mahmassani, H. S., M. Saberi, and A. Zockaie. Urban Network Gridlock: Theory, Characteristics, and Dynamics. *Transportation Research Part C: Emerging Technologies*, Vol. 36, 2013, pp. 480–497. https://doi.org/10.1016/j.trc.2013.07.002.
20. Mazloumian, A., N. Geroliminis, and D. Helbing. The Spatial Variability of Vehicle Densities as Determinant of Urban Network Capacity. Vol. 368, No. 1928, 2010, pp. 4627–4647. https://doi.org/10.1098/rsta.2010.0099.
21. Ji, Y., and N. Geroliminis. On the Spatial Partitioning of Urban Transportation Networks. *Transportation Research Part B: Methodological*, Vol. 46, No. 10, 2012, pp. 1639–1656. https://doi.org/10.1016/j.trb.2012.08.005.
22. Saeedmanesh, M., and N. Geroliminis. Clustering of Heterogeneous Networks with Directional Flows Based on "Snake" Similarities. *Transportation Research Part B: Methodological*, Vol. 91, 2016, pp. 250–269. https://doi.org/10.1016/j.trb.2016.05.008.
23. Saeedmanesh, M., and N. Geroliminis. Dynamic Clustering and Propagation of Congestion in Heterogeneously Congested Urban Traffic Networks. *Transportation Research Part B: Methodological*, Vol. 105, 2017, pp. 193–211.
24. Lopez, C., P. Krishnakumari, L. Leclercq, N. Chiabaut, and H. van Lint. Spatiotemporal Partitioning of Transportation Network Using Travel Time Data. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2623, No. 1, 2017, pp. 98–107. https://doi.org/10.3141/2623-11.
25. Geroliminis, N., and D. M. Levinson. Cordon Pricing Consistent with the Physics of Overcrowding. Transportation and Traffic Theory 2009: Golden Jubilee, 2009, pp. 219–240.
26. Daganzo, C. F., and L. J. Lehe. Distance-Dependent Congestion Pricing for Downtown Zones.

Transportation Research Part B: Methodological, Vol. 75, 2015, pp. 89–99. https://doi.org/10.1016/j.trb.2015.02.010.

27. Zheng, N., R. A. Waraich, K. W. Axhausen, and N. Geroliminis. A Dynamic Cordon Pricing Scheme Combining the Macroscopic Fundamental Diagram and an Agent-Based Traffic Model. *Transportation Research Part A: Policy and Practice*, Vol. 46, No. 8, 2012, pp. 1291–1303. https://doi.org/10.1016/j.tra.2012.05.006.
28. Yildirimoglu, M., M. Ramezani, and N. Geroliminis. Equilibrium Analysis and Route Guidance in Large-Scale Networks with MFD Dynamics. *Transportation Research Part C: Emerging Technologies*, Vol. 59, 2015, pp. 404–420. https://doi.org/10.1016/j.trc.2015.05.009.
29. Sirmatel, I. I., and N. Geroliminis. Economic Model Predictive Control of Large-Scale Urban Road Networks via Perimeter Control and Regional Route Guidance. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 19, No. 4, 2018, pp. 1112–1121. https://doi.org/10.1109/TITS.2017.2716541.
30. Menelaou, C., S. Timotheou, P. Kolios, and C. G. Panayiotou. Joint Route Guidance and Demand Management for Real-Time Control of Multi-Regional Traffic Networks. *IEEE Transactions on Intelligent Transportation Systems*, 2021. https://doi.org/10.1109/TITS.2021.3077870.
31. Haddad, J., and N. Geroliminis. On the Stability of Traffic Perimeter Control in Two-Region Urban Cities. *Transportation Research Part B: Methodological*, Vol. 46, No. 9, 2012, pp. 1159–1176. https://doi.org/10.1016/j.trb.2012.04.004.
32. Aalipour, A., H. Kebriaei, and M. Ramezani. Analytical Optimal Solution of Perimeter Traffic Flow Control Based on MFD Dynamics: A Pontryagin's Maximum Principle Approach. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, No. 9, 2019, pp. 3224–3234. https://doi.org/10.1109/TITS.2018.2873104.
33. Haddad, J. Optimal Perimeter Control Synthesis for Two Urban Regions with Aggregate Boundary Queue Dynamics. *Transportation Research Part B: Methodological*, Vol. 96, 2017, pp. 1–25. https://doi.org/10.1016/j.trb.2016.10.016.
34. Su, Z. C., A. H. F. Chow, N. Zheng, Y. P. Huang, E. M. Liang, and R. X. Zhong. Neuro-Dynamic Programming for Optimal Control of Macroscopic Fundamental Diagram Systems. *Transportation Research Part C: Emerging Technologies*, Vol. 116, 2020, p. 102628. https://doi.org/10.1016/j.trc.2020.102628.
35. Zhou, D., and V. V. Gayah. Model-Free Perimeter Metering Control for Two-Region Urban Networks Using Deep Reinforcement Learning. *Transportation Research Part C: Emerging Technologies*, Vol. 124, 2021, p. 102949.
36. Haddad, J. Robust Constrained Control of Uncertain Macroscopic Fundamental Diagram Networks. *Transportation Research Part C: Emerging Technologies*, Vol. 59, 2015, pp. 323–339. https://doi.org/10.1016/J.TRC.2015.05.014.
37. Li, Y., M. Yildirimoglu, and M. Ramezani. Robust Perimeter Control with Cordon Queues and Heterogeneous Transfer Flows. *Transportation Research Part C: Emerging Technologies*, Vol. 126, 2021, p. 103043. https://doi.org/10.1016/j.trc.2021.103043.
38. Mohajerpoor, R., M. Saberi, H. L. Vu, T. M. Garoni, and M. Ramezani. H∞ Robust Perimeter Flow Control in Urban Networks with Partial Information Feedback. *Transportation Research Part B: Methodological*, Vol. 137, 2020, pp. 47–73. https://doi.org/10.1016/j.trb.2019.03.010.
39. Zhong, R. X., C. Chen, Y. P. Huang, A. Sumalee, W. H. K. Lam, and D. B. Xu. Robust Perimeter Control for Two Urban Regions with Macroscopic Fundamental Diagrams: A Control-Lyapunov Function Approach. *Transportation Research Part B: Methodological*, Vol. 117, 2018, pp. 687–707. https://doi.org/10.1016/j.trb.2017.09.008.
40. Sirmatel, I. I., and N. Geroliminis. Stabilization of City-Scale Road Traffic Networks via Macroscopic Fundamental Diagram-Based Model Predictive Perimeter Control. *Control Engineering Practice*, Vol. 109, 2021, p. 104750. https://doi.org/10.1016/j.conengprac.2021.104750.

41. Aboudolas, K., and N. Geroliminis. Perimeter and Boundary Flow Control in Multi-Reservoir Heterogeneous Networks. *Transportation Research Part B: Methodological*, Vol. 55, 2013, pp. 265–281. https://doi.org/10.1016/j.trb.2013.07.003.
42. Haddad, J., M. Ramezani, and N. Geroliminis. Cooperative Traffic Control of a Mixed Network with Two Urban Regions and a Freeway. *Transportation Research Part B: Methodological*, Vol. 54, 2013, pp. 17–36. https://doi.org/10.1016/j.trb.2013.03.007.
43. Ramezani, M., J. Haddad, and N. Geroliminis. Dynamics of Heterogeneity in Urban Networks: Aggregated Traffic Modeling and Hierarchical Control. *Transportation Research Part B: Methodological*, Vol. 74, 2015, pp. 1–19. https://doi.org/10.1016/j.trb.2014.12.010.
44. Ni, W., and M. Cassidy. City-Wide Traffic Control: Modeling Impacts of Cordon Queues. *Transportation Research Part C: Emerging Technologies*, Vol. 113, 2020, pp. 164–175. https://doi.org/10.1016/j.trc.2019.04.024.
45. Sirmatel, I. I., D. Tsitsokas, A. Kouvelas, and N. Geroliminis. Modeling, Estimation, and Control in Large-Scale Urban Road Networks with Remaining Travel Distance Dynamics. *Transportation Research Part C: Emerging Technologies*, Vol. 128, 2021, p. 103157. https://doi.org/10.1016/J.TRC.2021.103157.
46. Haddad, J., and Z. Zheng. Adaptive Perimeter Control for Multi-Region Accumulation-Based Models with State Delays. *Transportation Research Part B: Methodological*, Vol. 137, 2020, pp. 133–153. https://doi.org/10.1016/J.TRB.2018.05.019.
47. Zhong, R. X., Y. P. Huang, C. Chen, W. H. K. Lam, D. B. Xu, and A. Sumalee. Boundary Conditions and Behavior of the Macroscopic Fundamental Diagram Based Network Traffic Dynamics: A Control Systems Perspective. *Transportation Research Part B: Methodological*, Vol. 111, 2018, pp. 327–355. https://doi.org/10.1016/J.TRB.2018.02.016.
48. Haddad, J., and B. Mirkin. Coordinated Distributed Adaptive Perimeter Control for Large-Scale Urban Road Networks. *Transportation Research Part C: Emerging Technologies*, Vol. 77, 2017, pp. 495–515. https://doi.org/10.1016/j.trc.2016.12.002.
49. Lei, T., Z. Hou, and Y. Ren. Data-Driven Model Free Adaptive Perimeter Control for Multi-Region Urban Traffic Networks With Route Choice. *IEEE Transactions on Intelligent Transportation Systems*, 2019, pp. 1–12. https://doi.org/10.1109/tits.2019.2921381.
50. Ren, Y., Z. Hou, I. I. Sirmatel, and N. Geroliminis. Data Driven Model Free Adaptive Iterative Learning Perimeter Control for Large-Scale Urban Road Networks. *Transportation Research Part C: Emerging Technologies*, Vol. 115, 2020, p. 102618. https://doi.org/10.1016/j.trc.2020.102618.
51. Watkins, C. J. C. H., and P. Dayan. Q-Learning. *Machine Learning*, Vol. 8, No. 3–4, 1992, pp. 279–292. https://doi.org/10.1007/bf00992698.
52. Sutton, R. S., and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2018.
53. Tsitsiklis, J. N., and B. Van Roy. *An Analysis of Temporal-Difference Learning with Function Approximation*. 1997.
54. van Hasselt, H., Y. Doron, F. Strub, M. Hessel, N. Sonnerat, and J. Modayil. Deep Reinforcement Learning and the Deadly Triad. 2018.
55. Mnih, V., K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. Human-Level Control through Deep Reinforcement Learning. *Nature*, Vol. 518, No. 7540, 2015, pp. 529–533. https://doi.org/10.1038/nature14236.
56. van Hasselt, H., A. Guez, and D. Silver. Deep Reinforcement Learning with Double Q-Learning. *30th AAAI Conference on Artificial Intelligence, AAAI 2016*, 2015, pp. 2094–2100.
57. Hessel, M., J. Modayil, H. van Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver. Rainbow: Combining Improvements in Deep Reinforcement Learning. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, 2017, pp. 3215–3222.
58. Schaul, T., J. Quan, I. Antonoglou, and D. Silver. Prioritized Experience Replay. 2016.

59. Lillicrap, T. P., J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous Control with Deep Reinforcement Learning. 2016.
60. Wang, Z., T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot, and N. de Freitas. Dueling Network Architectures for Deep Reinforcement Learning. *33rd International Conference on Machine Learning, ICML 2016*, Vol. 4, 2015, pp. 2939–2947.
61. Oliehoek, F. A., M. T. J. Spaan, and N. Vlassis. Optimal and Approximate Q-Value Functions for Decentralized POMDPs. *Journal Of Artificial Intelligence Research*, Vol. 32, 2008, pp. 289–353. https://doi.org/10.1613/jair.2447.
62. Koller, D., and R. Parr. Computing Factored Value Functions for Policies in Structured MDPs . 1999.
63. Rashid, T., M. Samvelyan, C. S. de Witt, G. Farquhar, J. Foerster, and S. Whiteson. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. 2018.
64. Peng, B., T. Rashid, C. A. S. de Witt, P.-A. Kamienny, P. H. S. Torr, W. Böhmer, and S. Whiteson. FACMAC: Factored Multi-Agent Centralised Policy Gradients. 2021.
65. Horgan, D., J. Quan, D. Budden, G. Barth-Maron, M. Hessel, H. van Hasselt, and D. Silver. Distributed Prioritized Experience Replay. 2018.
66. Lin, L.-J. Self-Improving Reactive Agents Based on Reinforcement Learning, Planning and Teaching. *Machine Learning*, Vol. 8, No. 3–4, 1992, pp. 293–321. https://doi.org/10.1007/bf00992699.
67. Gao, X. (Shirley), and V. V. Gayah. An Analytical Framework to Model Uncertainty in Urban Network Dynamics Using Macroscopic Fundamental Diagrams. *Transportation Research Part B: Methodological*, Vol. 117, 2018, pp. 660–675. https://doi.org/10.1016/j.trb.2017.08.015.
68. Wang, Y., B. Han, T. Wang, H. Dong, and C. Zhang. Off-Policy Multi-Agent Decomposed Policy Gradients. 2020.