Multi-Scale Model-Free Perimeter Control and Local Signal Control in Urban Networks

Dongqin Zhou*
The Institute for Experiential AI
Northeastern University, Boston, MA, 02115
Email: d.zhou@northeastern.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: 7381 words + 0 table (250 words per table) = 7381 words

Submitted [*07/15/2024*]

ABSTRACT

Deep reinforcement learning (DRL) has been shown as an effective paradigm to help solve local signal control or network-wide perimeter metering control problems individually. However, to improve the benefits of DRL applied to urban traffic control, a joint framework that considers both levels is needed as they often have complementary objectives. Early endeavors in such frameworks require the exchange of information between the levels to coordinate the control objectives, demanding increased communication infrastructure. To alleviate such requirements, this work presents a joint framework that features two levels of independent controllers, where both levels are managed by unique reinforcement learning agents that share common goals of throughput maximization. Extensive simulation experiments are conducted to demonstrate the effectiveness of the proposed framework, as well as the robustness of the learned agents against measurement noise of regional accumulation and identification errors of vehicle counts. Further, the framework has been shown capable of learning effectively under a partial local signal control configuration, which highlights its potential practical applicability. The framework holds promise for city-level traffic management driven by reinforcement learning that does not require the need for, often inaccurate, traffic models.

Keywords: Macroscopic fundamental diagram; perimeter control; traffic signal control; multi-scale RL

INTRODUCTION

Perimeter control, built upon aggregate traffic dynamics modeling with network Macroscopic Fundamental Diagrams (MFDs), has been shown effective in congestion mitigation and throughput maximization for urban networks comprised of a single or multiple homogeneous regions. Over the years, various extensions to perimeter control have been investigated, including for example robust control (*1*, *2*) and integration with route guidance or ramp metering (*3–5*). Numerous approaches have been proposed for perimeter control; these include proportional-integral (PI) type feedback controller (*6*, *7*), linear quadratic regulator (*8*, *9*), and the model predictive control (MPC) method (*3*, *10*, *11*). Recent years have also witnessed an increasing trend of data-driven methods such as model free adaptive control and reinforcement learning (*12–18*).

Despite the notable research findings, however, concerns may still arise regarding the practical implementation of perimeter control in urban networks, since the MFD-based dynamics modeling (whence perimeter control is defined) relies on traffic homogeneity. Inevitably, local pockets of congestion are likely to form in dense urban traffic networks (*19–22*), resulting in traffic heterogeneity, which diminishes the effectiveness of perimeter control. To this end, an integrated framework that regulates both the inter-regional exchange flows (viz., perimeter control) and intra-regional traffic signals is needed, wherein the upper-level perimeter control helps maintain regional accumulations around the critical levels while the lower-level signal control combats local congestion to improve traffic homogeneity.

Early endeavors in such integrated control frameworks include (*23–25*), where the lower-level seeks to reduce traffic inhomogeneity at a local scale, facilitating more effective application of perimeter control schemes at the upper level. However, these works require the exchange of information between the levels to coordinate the control objectives, which creates a demand for increased communication infrastructure and may impede its real-world applicability (particularly in dense urban areas). In addition, these works build upon the MPC scheme to formulate the control problems, necessitating accurate modeling of system dynamics that is often intractable in real life. In this regard, note the MPC-based multi-scale perimeter control method in (*26*) also faces high communication requirements with a centralized control paradigm, and the lower-level only considers delay minimization at the perimeter intersections.

To alleviate the modeling inaccuracies and computational costs associated with MPC-based schemes, as well as the requirements for communication infrastructure, hierarchical control frameworks that feature two levels of independent controllers are receiving increasing research interests recently. In (*27*), a volume-based approach and a modified SCATS strategy are used for local signal control, in combination with a PI type feedback perimeter controller for single-region networks. In (*28*), the PI-type regulator is used with the max pressure (MP) signal controller (*29*), where the green times at the perimeter and intra-regional intersections are obtained by solving a set of optimization programs. In contrast, (*30*) solves the upper-level perimeter control problem with reinforcement learning (RL) and adopts an acyclic max pressure signal controller (*31*) at the lower level. Both levels control the green times directly without solving optimization problems, which is more computationally convenient in real applications.

Following these research efforts, this work studies the joint perimeter and signal control problem in urban networks, where both levels are controlled by reinforcement learning agents. While RL has been applied to signal control problems extensively and is also gaining momentum in perimeter control applications, its effectiveness hasn't been investigated for the joint control problem. This work thus extends the frameworks in (*18*, *30*) to consider RL for lower-level signal control. A multi-scale multi-agent training paradigm is presented to realize joint control with RL, and the effectiveness is evaluated using simulated single- and two-region networks. The results show the presented approach is highly comparable (and often times superior) to a baseline comprised of established perimeter controllers and the MP policy (*31*).

The remainder of the paper is structured as follows. The next section explains the methodology adopted in this work. Extensive experiment results are then provided on a single- and two-region network. Finally, conclusions and future work directions are outlined in the last section.

METHODOLOGY

This section explains the joint control framework. The baseline traffic signal and perimeter controllers are first introduced, followed by algorithmic designs of the upper-level and lower-level RL agents. Lastly, the multi-scale training paradigm is presented.

Max Pressure (MP) Controller

This section introduces the Max Pressure (MP) controller in (*31*) along with a few modifications in (*28*). Note, MP is a decentralized signal controller that operates on each intersection, thus the explanations are presented regarding an individual intersection. Also, decentralization renders the MP policy scalable to any urban networks, without requiring any new infrastructure or communication with existing controllers.

Define a roadway segment between two adjacent intersections as a link and a pair of links (l, m) as a movement. Note, only movements that are controllable by traffic signals are considered (uncontrolled movements such as channelized right turns are omitted from the discussion here). A phase refers to a set of movements that can be served together in the same signal duration. Define $x(l, m)$ as the metric to obtain pressure calculations for the MP controller, which amounts to the number of vehicles of movement (l, m) since the point queue model is used to express traffic flows on a link in (*31*). Let $x(l,\cdot)_{max}$ be the maximum number of vehicles (i.e., storage capacity) of link l, $C(l, m)$ be the mean value of saturation flow for movement (l, m), and $\beta(l, m)$ be the turn ratio from link l to m. Further, denote as S_j the movements served by phase j and Out_l the downstream links of link l. Adopting these terminologies, the max pressure control principle works as follows.

First, the weight of a movement is calculated as per (similar to (*28*)):

$$w(l, m) = \frac{x(l, m)}{x(l,\cdot)_{max}} - \sum_{n \in Out_m} \beta(m, n) \cdot \frac{x(m, n)}{x(m,\cdot)_{max}}. \tag{1}$$

The second term of Eq. (1) can be viewed as the average (normalized) number of vehicles weighted by turn ratios at the downstream link. If the link m is an exit, this term will be 0 since there are no further downstream links. As such, the weight of a movement indicates the difference between the (normalized) upstream and downstream number of vehicles. Normalization using the storage capacity is intended to consider link length, as the same number of vehicles may indicate different levels of congestion at links with different lengths. Then, the pressure for each phase is the sum of movement weight times the corresponding saturation flow and computed as:

$$p(S_j) = \sum_{(l,m) \in S_j} C(l, m) \cdot w(l, m). \tag{2}$$

Intuitively, the pressure of a phase indicates its potential of traffic production at the intersection. To see this, notice that large movement weight means the upstream link has more vehicles than the downstream link; that is, there is a large number of vehicles to discharge from the upstream and enough space in the downstream to receive these vehicles. Further, the pressure accounts for the likelihood of serving these vehicles by multiplying the movement weight with the saturation flow. Therefore, high pressure is associated with a larger number of vehicles to serve (i.e., larger movement weight) and greater ability to serve vehicles (i.e., saturation flow). Jointly, the pressure measures the potential of traffic production by each phase at the intersection. Following this logic, the MP controller activates the phase with the highest pressure at each time step, shown as:

$$S^* = \arg\max_j p(S_j). \tag{3}$$

The selected phase is implemented in the traffic environment for a time step, until the next decision is made.

Several modifications are also proposed to ensure real-world applicability. First, the weight of a movement (Eq. (1)) is truncated to be non-negative. Negative weights arise when the (average) number of vehicles at the downstream link is larger than that at the upstream; activating these movements will worsen the imbalanced vehicle presence at the intersection and not contribute to traffic production. Second, the MP control law is executed every time step, contrary to the cycle-based design in (*28*). Denote as Δt the time step length. When the signal phase changes across consecutive time steps, a transition interval is needed (assumed to be $\kappa = 3$s). As such, the pressure of a phase (Eq. (2)) is adjusted by a factor of $\frac{\Delta t-\kappa}{\Delta t}$ if the phase is different from the currently active one; see also (*32*) for this adjustment. Third, to account for practical applications of the MP controller, each signal phase will be activated at least once during a perimeter control interval (to be specified shortly) and therefore have a minimum green time of $\Delta t - \kappa$ (s).

Further, note that calculating the movement weights requires the turn ratio $\beta(m, n)$ which denotes the ratio of vehicle traveling from link m to its downstream link n. In a dynamically congested network, the turn ratio cannot be assumed known a priori. For this reason, a dynamic turn ratio update schedule is presented in (*28*) while a fixed turn ratio is adopted in (*32*). Note, though the MP controller is used for lower-level signal control in (*30*), the information on how the turn ratios are determined is not disclosed. In this work, a simple procedure is devised to estimate the turn ratios for each link. Specifically, in regular intervals of 3 minutes, 50 vehicles (or all vehicles if fewer than 50) are randomly sampled from each link to determine their turning maneuvers. The ratios of turning among these vehicles then provide an estimated turn ratio for the link which is used as constants during the 3-minute intervals. As will be shown in the simulation results, this simple and intuitive estimation procedure renders the MP extremely effective at alleviating local congestion. Also, this procedure is computationally cheap, compared to the update schedule in (*28*). Moreover, estimating the turn ratio is not the focal point in this work, and this procedure is kept the same among all MP implementations to establish fair comparisons. Other estimation procedures that may improve the MP performance further are left as future work directions.

Baseline Perimeter Controllers

In this work, single- and two-region networks are simulated. Hence, the Bang-Bang (BB) policy (*33*) and improved greedy control (I-GC) policy (*18*) are adopted as comparative baselines for perimeter control. These policies are implemented at regular intervals of ΔT (which denotes the perimeter control interval). Other than these, a baseline that simulates the status quo, i.e., no control (NC) policy, will also be used.

The Bang-Bang policy builds upon the notion of MFD-based modeling and alternates its control action by comparing the regional accumulation to the critical value that is associated with maximum traffic production. Specifically, the BB policy chooses the maximum green time for all inbound movements if the regional accumulation is smaller than the critical value and the minimum value otherwise. The BB policy presents a simple and effective way to mitigate urban congestion for single-region networks, and it is real life implementable since it does not require full knowledge of the network dynamics. Though the critical accumulation information is needed, it can still reap sufficient control benefits with estimation inaccuracies.

The I-GC policy extends conventional greedy control to consider three levels of congestion for each region and directly adjusts the inter-regional green times. By introducing a buffer between minimum and maximum green times, the I-GC policy can effectively mitigate regional congestion and significant cordon queues in the event of a region being congested. Concretely, for each region, the I-GC policy selects the maximum green time (g_{max}) for inbound movements if the region operates in free flow (i.e., regional accumulation smaller than the critical value); a smaller (close to minimum, denoted as g_{mid}) green time value is chosen if the region is moderately congested whereas the minimum value (g_{min}) is taken if the region is severely congested. The cutoff points that determine the congestion levels are determined from the regional MFD plots, while the candidate green time values are design parameters.

Algorithmic Designs of RL Agents

The joint perimeter and signal control problem is formulated as a Markov decision process, where the environment represents the simulated single- or two-region networks. At regular intervals of ΔT, an upper-level agent (dubbed U-RL) takes information from the simulation environment and selects perimeter control actions that determine the green times at perimeter intersections. The actions will then be implemented in the environment for the whole interval, and at the end of the interval, it receives a reward back from the environment as an assessment of the action just taken. Similarly, for each intra-regional intersection, a unique lower-level agent (dubbed L-RL) selects actions every $\Delta t \leq \Delta T$ to set the signal timings.

In this work, the information taken by the U-RL includes aggregated regional speed(s) and flow(s), accumulation(s), and standard deviation(s) of lane-level vehicle counts. In single-region networks, the U-RL selects among $\{0,0.2,\cdots,0.8,1.0\}$ as the ratio of green times allocated to entering vehicles during the control interval at perimeter intersections. Intuitively, a larger value means longer green times and thus potentially an increased number of vehicles into the region. To account for practical implementations, the green times are truncated to be between the minimum and maximum values. In two-region networks, the U-RL adopts the same action space as defined for the I-GC (viz, $\{g_{min}, g_{mid}, g_{max}\} \times \{g_{min}, g_{mid}, g_{max}\}$), which directly specifies the green times for each travel direction at the perimeter intersections. The reward for the U-RL is the traveled distance of all vehicles to encourage higher traffic throughput.

The Double DQN algorithm (*34*), as well as the distributed learning architecture (*35*), are utilized to train the U-RL agent. In addition, to consider the possibly delayed impacts of perimeter control, multi-step return is adopted (similar to (*30*)) and the (upper-level) learning targets Y_T^U are computed as:

$$Y_T^U = \sum_{k=0}^{\delta-1} (\gamma^U)^k r_{T+k+1}^U + (\gamma^U)^\delta Q\left(s_{T+\delta}^U, \arg\max_{u'} Q\left(s_{T+\delta}^U, u'; \boldsymbol{\theta}^{UQ}\right); \boldsymbol{\theta}^{UQ-}\right). \quad (4)$$

where δ specifies the number of look-ahead steps (the original single-step return is a special case of Eq. (4) with $\delta = 1$), s_T^U, u_T, r_T^U respectively represent the state, action, and reward at time step T, γ^U is the discount factor that decays the perceived value of future rewards, $Q(:,:;\boldsymbol{\theta}^{UQ})$ and $Q(:,:;\boldsymbol{\theta}^{UQ-})$ respectively denote the Q- and target neural networks parameterized by $\boldsymbol{\theta}^{UQ}$ and $\boldsymbol{\theta}^{UQ-}$.

In a similar fashion, the L-RL agent takes local information around an intersection and selects which phase to activate at regular intervals of Δt. The parameter sharing technique is used to accommodate the (often) large number of intra-regional intersections in urban networks. Specifically, each intersection is controlled by its own reinforcement learning agent, and all these agents share the neural network structure as well as model weights. They receive individualized information at each intersection and select individualized actions as well as obtain individualized rewards. Here, the input information to the L-RL includes the average number of vehicle (weighted by turn ratios) of the four downstream approaches, the upstream vehicle counts grouped by phases (e.g., northbound/southbound left), the current phase, and the regional accumulation(s). Note, the downstream vehicle count is averaged by turn ratios to be similar to the information used by the MP policy, while the upstream vehicle counts indicate the congestion situation that informs where a high potential of traffic production may be reaped. The regional accumulation provides global information about how the region is operating, and it is found this information is beneficial to L-RL's performance. The action specifies which phase to choose from for each intersection, and the reward is the number of discharged vehicles to encourage higher traffic production. The L-RL is also trained with the Double DQN algorithm, but without multi-step return. The (lower-level) learning targets Y_t^L are:

$$Y_t^L = r_{t+1}^L + \gamma^L Q\left(s_{t+1}^L, \arg\max_{u'} Q(s_{t+1}^L, u'; \boldsymbol{\theta}^{LQ}); \boldsymbol{\theta}^{LQ-}\right). \quad (5)$$

where the variables are defined similarly to Eq. (4), but with superscript *L* to denote lower-level.

A Multi-Scale Training Paradigm

To jointly train both the U-RL for upper-level perimeter control and L-RL for lower-level traffic signal control, a multi-scale reinforcement learning approach is adopted; see Algorithm 1. Note that the term "multi-scale" refers to both multi-timescale (with action intervals of ΔT and Δt) and multi-spatial scale (with perimeter control acting at a regional scale while traffic signal control at an intersection scale). The so-constructed joint control agent is denoted as MS-RL that stands for Multi-Scale RL.

As with the convention of value-based deep reinforcement learning methods, both the U-RL and L-RL adopt artificial neural networks to approximate the action value function, and for clarity they are respectively parameterized by $\boldsymbol{\theta}_{iter}^{UQ}$ and $\boldsymbol{\theta}_{iter}^{LQ}$ (where $iter$ is short for iteration). The resulting Q networks can thus be denoted by $Q(:,:;\boldsymbol{\theta}_{iter}^{UQ})$ and $Q(:,:;\boldsymbol{\theta}_{iter}^{LQ})$. Target networks and replay buffers are utilized to improve learning performances, following (*36*) as well as prior adaptations of deep-RL for perimeter control (*14*, *30*, *37*). In particular, target networks help provide relatively static learning targets while the use of replay buffer helps reduce the correlation between training samples. The distributed learning architecture (*35*) is used by the agents such that they can interact with multiple instantiations of the simulation environment concurrently to expand the variety of samples they jointly encounter. Each instantiation is termed a generator (i.e., experience generator) in Algorithm 1. With such architecture, both the L-RL and U-RL agents are updated only once per training iteration. However, such updates can utilize all transitions collected during the iteration which improves the convergence for the agents. To further increase the variability of training samples, the agents perform proactive exploration of the environment with the epsilon-greedy (ϵ-greedy) strategy. To wit, the agents take a greedy action (with respect to the Q-values) with probability $1-\epsilon$ and a random action otherwise.

A few further remarks are provided here regarding the interactions between the agents and the environment. First, the perimeter control interval ΔT is assumed an integer multiple of the signal control interval Δt. After the U-RL agent selects an action, it is executed in the environment for a duration of ΔT. During this time, the L-RL agent interacts with the environment in intervals of Δt. In this way, a state-action-reward transition is obtained every Δt for L-RL but every ΔT for U-RL. Second, notice the parameter sharing technique is used for the L-RL agent, thus a group of transitions will be collected every Δt. The group size is dependent on the number of intersections controlled by the L-RL agent. For the same reason, a vector of states $\boldsymbol{s}_t^L$ is collected for L-RL during environment reset (Line 6) or step (Line 13), whereas in comparison a single state is collected for U-RL (Lines 6 and 17). Third, in Algorithm 1, a represents a local agent that acts for a single intersection while n is the number of controlled intra-regional intersections (hence n local agents). The local actions u_t^{La} taken by the local agents are jointly executed in the environment, which leads to a lower-level environment step (for duration Δt, see Line 13) that yields a vector of lower-level states $\boldsymbol{s}_t^L$ and a vector of lower-level rewards $\boldsymbol{r}_t^L$. The joint transitions are then split and stored in the replay buffer for updating the L-RL parameters. Finally, following the popular independent learning paradigm, the two agents interact with the environment simultaneously (but at different time and spatial scales) and are trained separately, thus reliving the need for increasing communication infrastructure.

Algorithm 1. Multi-Scale RL controller for joint perimeter and signal control (MS-RL).

1: Randomly initialize U-RL network $\boldsymbol{\theta}_0^{UQ}$ and shared L-RL network $\boldsymbol{\theta}_0^{LQ}$
Initialize target networks $\boldsymbol{\theta}_0^{UQ-} = \boldsymbol{\theta}_0^{UQ}, \boldsymbol{\theta}_0^{LQ-} = \boldsymbol{\theta}_0^{LQ}$
Initailize replay buffer D^U, D^L with buffer size B^U, B^L and sample size b^U, b^L for U-RL and L-RL
Specify the number of training iterations I and experience genetaors G

2: **for** *iter* = 1 **to** I **do**
3: Compute the decayed ϵ^U, ϵ^L values for exploration
4: **for** generator = 1 **to** G **do** // concurrent interactions with the environment
5: Load the U-RL and shared L-RL networks $\boldsymbol{\theta}_{iter}^{UQ} = \boldsymbol{\theta}_{iter-1}^{UQ}, \boldsymbol{\theta}_{iter}^{LQ} = \boldsymbol{\theta}_{iter-1}^{LQ}$
6: $s_0^U, \boldsymbol{s}_0^L$ ← Environment.Reset()

7: **for** $T = 1$ **to** N_T **do** // upper-level perimeter control
8: $u_{T-1}^U = \arg\max_u Q(s_{T-1}^U, u; \boldsymbol{\theta}_{iter}^{UQ})$ with probability $1 - \epsilon^U$
a random action with proability ϵ^U
9: Implement u_{T-1}^U into environment // for duration ΔT

10: **for** $t = 1$ **to** N_t **do** // lower-level signal control
11: $u_{t-1}^{La} = \arg\max_u Q(s_{t-1}^{La}, u; \boldsymbol{\theta}_{iter}^{LQ})$ with probability $1 - \epsilon^L$
a random action with proability ϵ^L
12: $\boldsymbol{u}_{t-1}^L = \{u_{t-1}^{La}\}_{a=1}^n$
13: $(\boldsymbol{r}_t^L, \boldsymbol{s}_t^L)$ ← Environment.Step($\boldsymbol{s}_{t-1}^L, \boldsymbol{u}_{t-1}^L$) // for duration Δt
14: Split $(\boldsymbol{s}_{t-1}^L, \boldsymbol{u}_{t-1}^L, \boldsymbol{r}_t^L, \boldsymbol{s}_t^L)$ into $\{s_{t-1}^{La}, u_{t-1}^{La}, r_t^{La}, s_t^{La}\}_{a=1}^n$
15: Store $\{s_{t-1}^{La}, u_{t-1}^{La}, r_t^{La}, s_t^{La}\}_{a=1}^n$ into the replay buffer D^L
16: **end for**

17: (r_T^U, s_T^U) ← Environment.Step(s_{T-1}^U, u_{T-1}^U)
18: Store $(s_{T-1}^U, u_{T-1}^U, r_T^U, s_T^U)$ into replay buffer D^U
19: **end for**
20: **end for**

21: **if** the number of stored transitions exceeds the buffer sizes B^U, B^L **then**
22: Remove outdated transitions
23: **end if**

24: Training samples ← a batch of $b^U(b^L)$ transitions randomly drawn from $D^U(D^L)$
25: Periodically load target networks $\boldsymbol{\theta}_{iter}^{UQ-} = \boldsymbol{\theta}_{iter-1}^{UQ}, \boldsymbol{\theta}_{iter}^{LQ-} = \boldsymbol{\theta}_{iter-1}^{LQ}$ and construct learning targets as per Eq. (4) and (5)
26: $\boldsymbol{\theta}_{iter}^{UQ}, \boldsymbol{\theta}_{iter}^{LQ}$ ← Update the network parameters towards the learning targets
27: **end for**

JOINT CONTROL FOR SINGLE-REGION NETWORKS

This section presents the experiment details of the joint perimeter and signal control framework applied on a single-region network. Several joint control methods are utilized for comparison. In particular, the BB policy is combined with the MP controller (*31*) to benchmark the upper-bound performances, while a non-adaptive fixed time (FT) signal plan is used with the no control (NC) policy at the upper level for the lower-bound performances. Two other baselines are also included for comparison: NC+MP and BB+FT.

Single-Region Network Setup

The single-region network is simulated in SUMO (*38*); see Fig. 1 which shows the protected region (shaded in blue) and the layouts of the intra-regional and perimeter intersections. Each link in the network has a length of 500m and each vehicle is 5m long. The free flow speed of each lane is set to 50 km/h while the saturation flow is 1800 veh/h/lane.

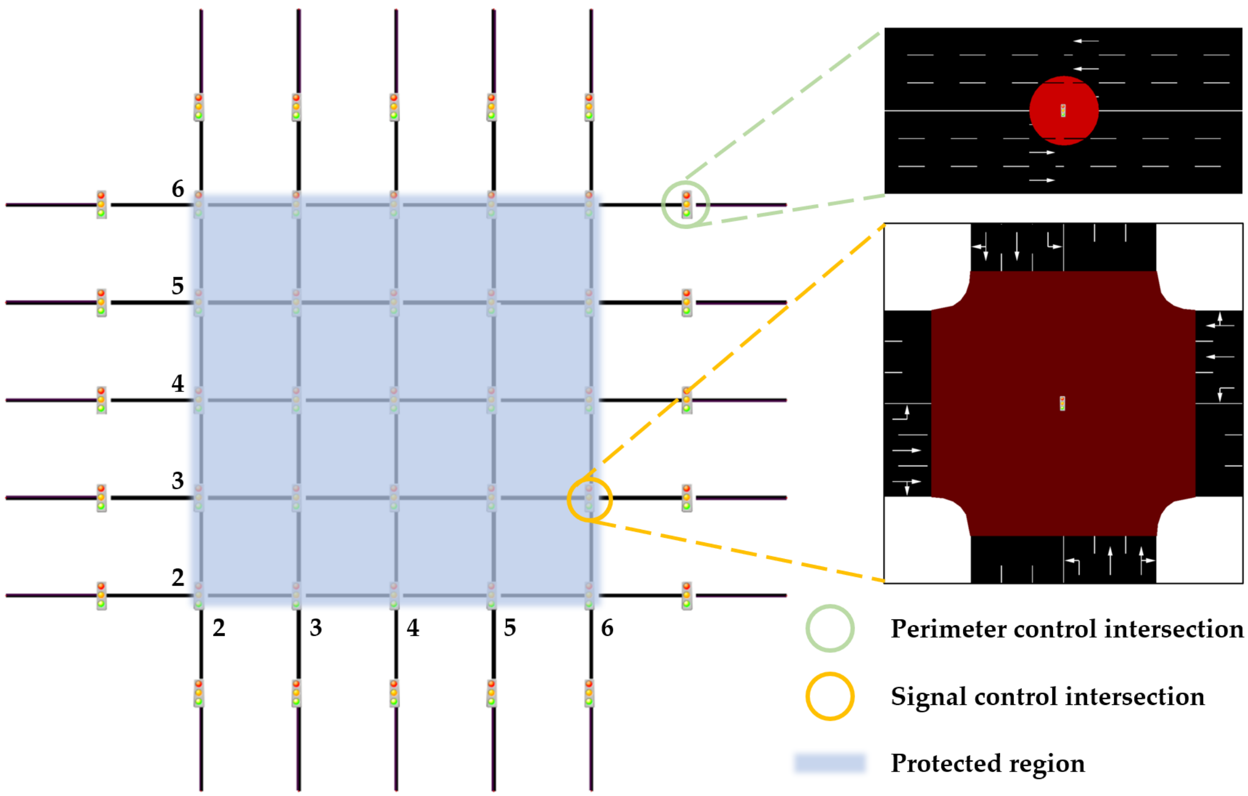

Fig. 1. The simulated single-region urban network.

The simulation step is set to 1s. All intersections in the network are signalized, and the perimeter intersections adopt a shared cycle length of 90s (i.e., $\Delta T = 90$s). Using default signal plans created by SUMO, strong demands were assumed initially to congest the network and to obtain the MFD plot; the critical accumulation for the network was determined to be 3000 veh. The minimum and maximum green times are respectively 5s and 87s. The BB policy alternates the green times between these two values depending on if the region is congested or not (by comparing the accumulation to the critical value). The U-RL sets the green times at the perimeter intersections by choosing the ratio of green time during a signal cycle, with minimum and maximum green times as constraints. In contrast, the NC policy mimics the status quo (no perimeter control) and uses the maximum green time throughout the simulation. The intra-regional intersections are either controlled by the MP policy or the L-RL agent or adopt a fixed-time (FT) signal plan. The FT plan, MP, and L-RL policies all share the same set of phases to make sure the simulation results are comparable, but their sequence order and duration may differ. No offset is assumed as it is inconsequential to the network-level performances in grid networks (*39*). The signal control interval is set to $\Delta t = 10$s for the MP policy or L-RL (hence $N_t = 9$), but the FT plan assumes cycle length of 90s.

Origins are even distributed across the entire network, whereas the destinations are only placed inside the protected region. A strong directional demand is assumed from outside of the region; see Fig. 2. This demand is adopted to simulate scenarios where perimeter control is the most helpful, i.e., to protect destination-loaded regions from over-saturation. The strong demand lasts for 90 minutes as followed by a recovery period of 30 minutes. The total simulation time is 2 hours and thus $N_T = 80$. Each of the total demands (e.g., from outside to inside of the region) is evenly assigned to all associated origin-destination pairs. In microsimulation, the traffic and vehicle behaviors will exhibit variability (e.g., the exact times when vehicles are inserted into the network, the initial routes of the vehicles, and/or the vehicle speeds) during each simulation instance, depending on the random number generation process. For this reason, multiple random seeds were used to enhance simulation realism. The simulated vehicles are initially routed using the stochastic C-logit route choice model (*40*), and a subset of the vehicles (60%) were assumed to be able to adaptively reroute themselves based on prevailing traffic to mimic more realistic driving patterns. This adaptive rerouting has been shown helpful to network-wide operational performances (*20*, *21*), and in this work it happens at regular intervals of 3 minutes.

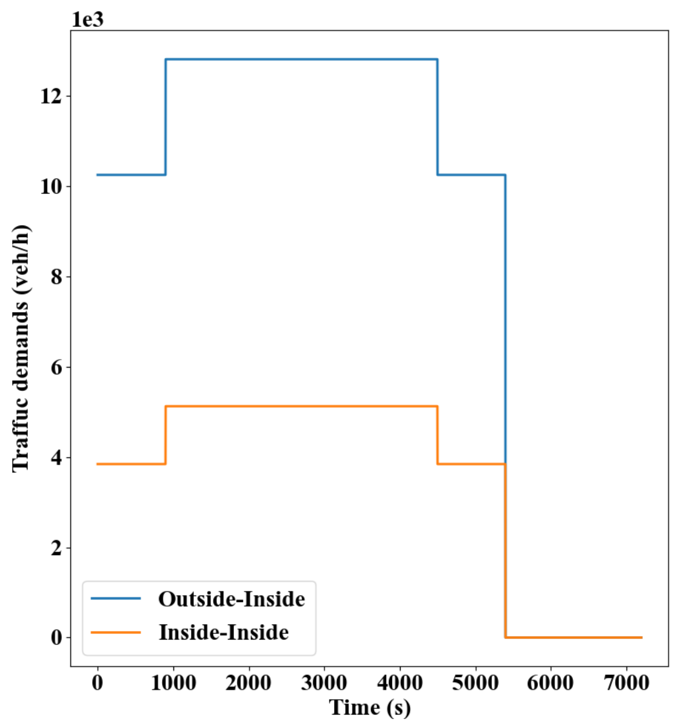

Fig. 2. Traffic demands profile.

Experiment Results

The objective of the joint control framework is to maximize network throughput, as indicated by the cumulative trip completion (CTC). The proposed MS-RL scheme is a learning-based method, and its effectiveness can be evaluated using learning curves that express the evolution of CTC over training iterations; see Fig. 3. The baseline methods have constant CTCs as they do not iteratively update their policies, and the bands reflect the randomness in the simulation. As can be observed, the proposed MS-RL scheme can effectively learn to realize performances directly comparable to the BB+MP policy. This is notable since the BB controller is a proven method for single-region perimeter control whereas the MP policy is an effective decentralized signal controller with proven ability of throughput maximization. Note, though, both the U-RL and L-RL agents in the MS-RL scheme start their learning processes entirely from scratch, with completely random exploration of the environment, so the performances are inferior initially (only slightly better than NC+FT) and improve as the agents learn.

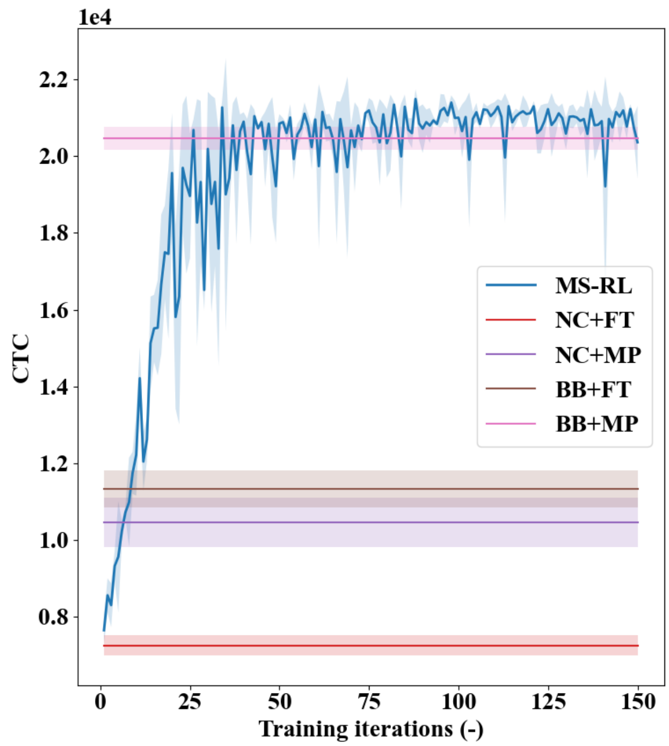

Fig. 3. Learning curve of the MS-RL scheme for single-region networks.

Fig. 4 presents the cumulative count curves by the MS-RL scheme and baseline methods, where "Exit" indicates the cumulative number of exited vehicle (i.e., cumulative trip completion) and "Entry" the cumulative number of vehicles that either are generated within the region from traffic demands or arrive from outside of the region. The vehicle generation from traffic demands is identical across methods, as the demands are the same and those trips are not metered, while the vehicle arrival is influenced by the control policy. As Fig. 4 shows, without perimeter control (NC at the upper level), using the MP policy at the lower level yields much higher cumulative vehicle entry and trip completion than FT, which shows the MP policy is much more effective at reducing traffic congestion than FT. However, the overall trip completion obtained by MP alone is rather modest as it cannot handle oversaturated conditions well, a situation that can be remedied by using BB policy at the upper level. The BB policy acts upon the regional congestion level and limits vehicle arrival from outside of the region as the region gets congested. The delayed vehicle arrival thus mitigates congestion within the region, which allows for a higher trip completion and more vehicle arrival later on. Yet, such effects cannot be realized by the BB policy alone (i.e., BB+FT), and the combination of BB policy at the upper level and MP at the lower level leads to the highest cumulative vehicle entry and trip completion among the baseline methods. Comparisons between the curves by BB+MP and MS-RL imply a great extent of similarity, suggesting the competitiveness of the proposed scheme. The areas between the cumulative entry and exit curves represent the total travel times of all vehicles throughout the simulation. From this, one could conclude the MS-RL scheme realizes the smallest total travel time among all control methods. In a similar fashion, the differences between the cumulative count curves indicate the regional accumulation, and one could see the MS-RL scheme has the steadiest accumulation.

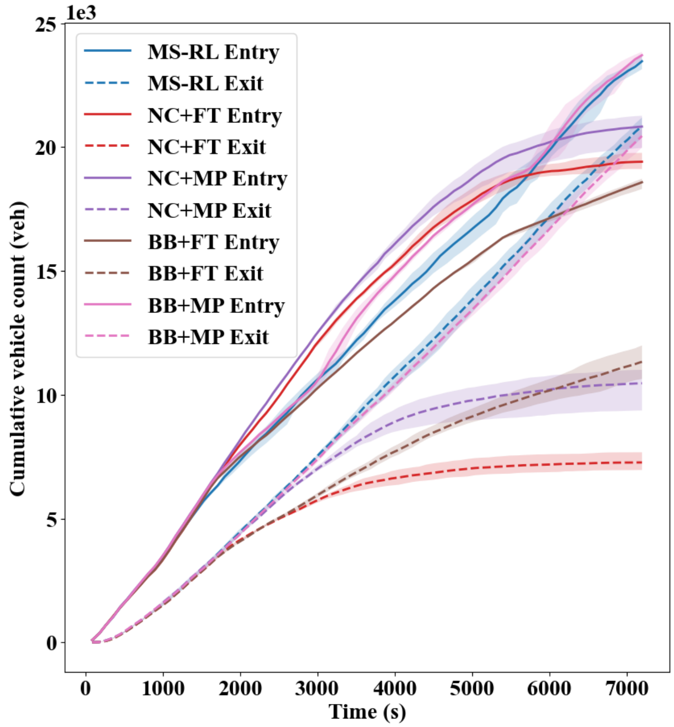

Fig. 4. Cumulative count curves.

To evaluate the learning robustness of the proposed MS-RL scheme, several types of environment uncertainties are considered and infused in the simulation process during each of the training iterations. The uncertainties include measurement noise of the regional accumulation, where the accumulation information perceived by the MS-RL (in its upper- and lower-level state designs) is assumed inaccurate, and such inaccuracy follows a mean-zero normal distribution:

$$\tilde{d}(t) = d(t) + \mathbb{N}(0, \sigma^2) \tag{6}$$

where $\tilde{d}(t)$ is the perceived accumulation, $d(t)$ is the accurate accumulation, and σ is the standard deviation of the normal distribution used to denote the level of measurement noise. The uncertainties also involve identification errors of vehicle count on each lane, which are also in the form of a mean-zero normal distribution and are intended to simulate widespread count discrepancies that happen on each lane of the network. Note that such errors impact vehicle counts not only to a large spatial extent (the whole network) but also at an extremely high frequency (every 10s at which the L-RL agent utilizes the vehicle count information). Hence, the magnitude of the errors is much smaller than measurement noise.

Fig. 5 provides the learning curves for the MS-RL scheme when there are both measurement noise of accumulation and identification errors of vehicle count in the environment. The noise and error are added independently to the respective measurements (accumulation and lane-level vehicle counts), and in Fig. 5 their combinations are indicated in the subplot titles. As shown in the figure, the MS-RL scheme can still realize remarkable control benefits that are comparable to BB+MP, though the learning performances start to deteriorate with higher identification errors. This is even more notable given that the L-RL receives inaccurate accumulation and vehicle counts every $\Delta t = 10$s in the case of combined uncertainties.

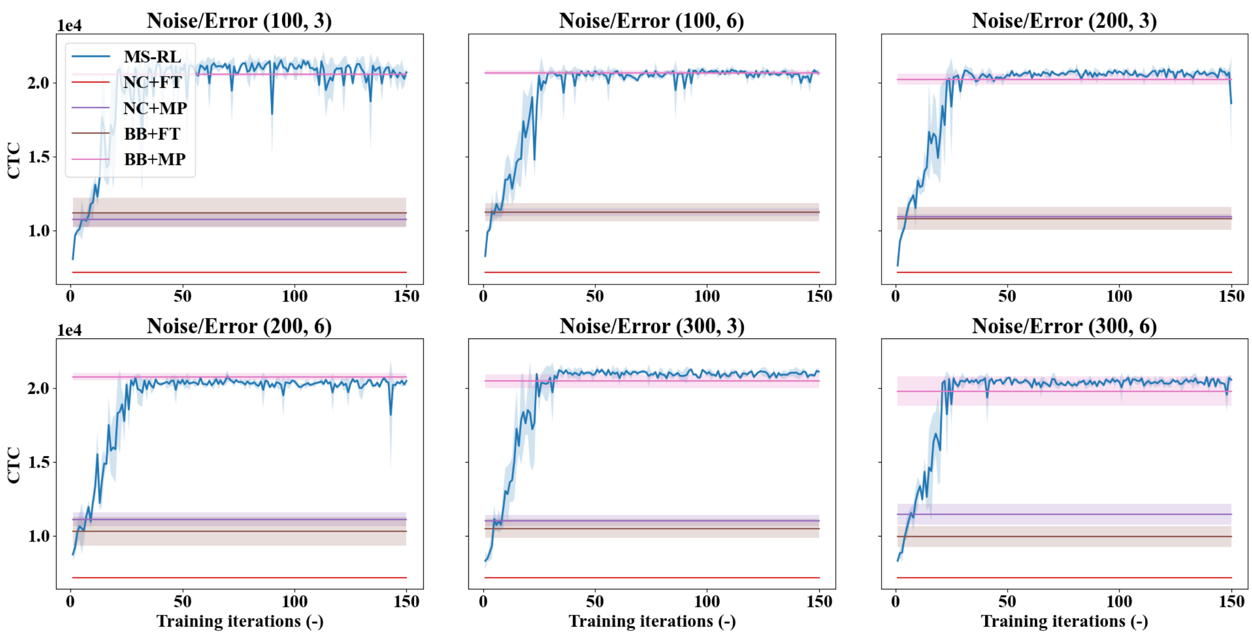

Fig. 5. Learning curves against combined measurement noise and identification errors.

JOINT CONTROL FOR TWO-REGION NETWORKS

In this section, the MS-RL scheme is applied for joint perimeter and signal control on a two-region network. The improved greedy control (I-GC) policy is combined with the MP policy or FT signal plan as baselines.

Two-Region Urban Network Setup

The simulation setup of the two-region network follows that detailed in (*18*). In particular, the network has a larger periphery region R_1 encompassing a smaller city center R_2 (see Fig. 6), and the two regions are connected via two-directional linking roads where perimeter control can be implemented and queued vehicles can be temporarily stored. Initial simulations are run with strong demands to obtain the MFD plots as well as proper values for the I-GC policy parameters (for example the cutoff points used to categorize the congestion levels). The results suggest the trip completion rate peaks at accumulations around 5000 (region 1) and 1500 (region 2) respectively, and region 1(2) is classified as severely congestion if its accumulation is larger than 8000(2000) veh. Moreover, following the same line of reasoning in (*18*), the green times are set to $g_{min} = 0s, g_{mid} = 10s, g_{max} = 87s$ at the perimeter intersections. The I-GC policy, as well as the U-RL agent, selects among combinations of $\{g_{min}, g_{mid}, g_{max}\} \times \{g_{min}, g_{mid}, g_{max}\}$ to determine the green times allocated to vehicles in each direction of travel. For the two-region network, origins and destinations are uniformly distributed inside each region, with inter-regional traffic demands for both directions; see Fig. 7 where traffic demands last for 60 minutes followed by a recovery period of 30 minutes. The total simulation time is 90 minutes hence $N_T = 60$.

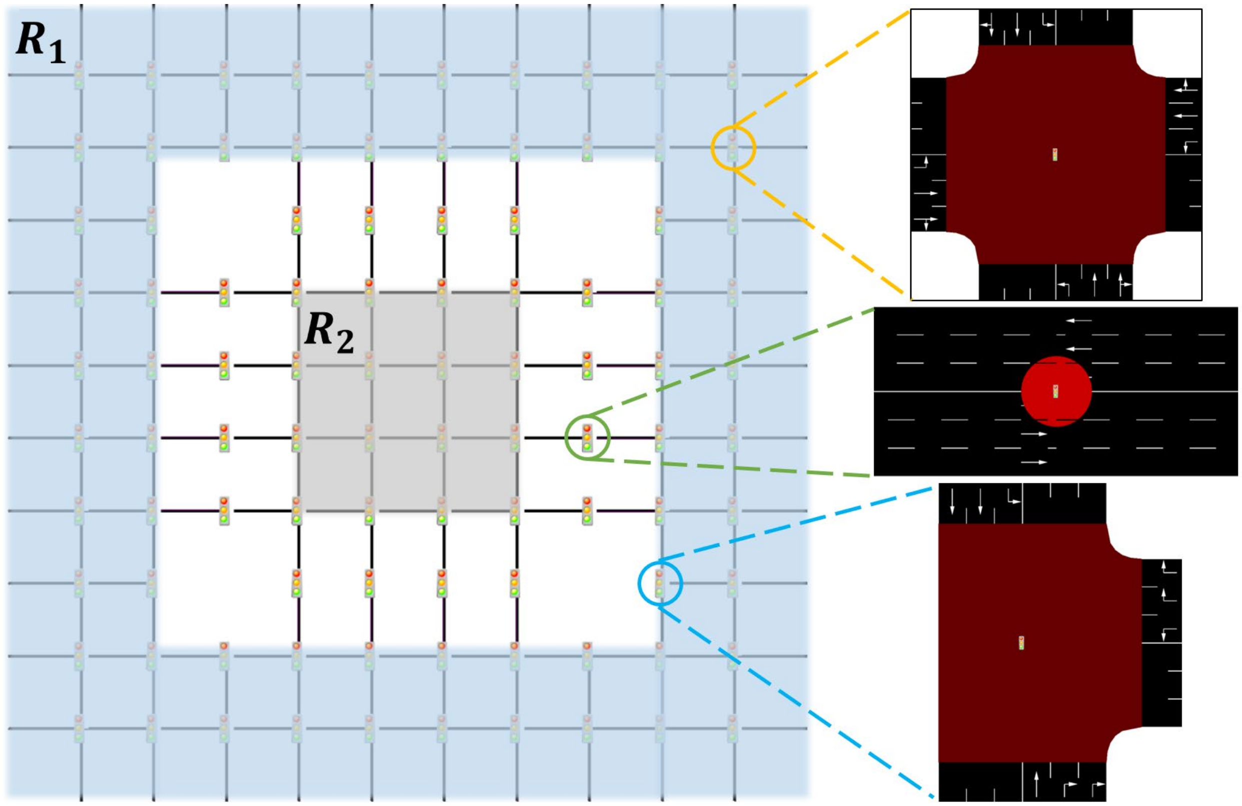

Fig. 6. The simulated two-region urban network.

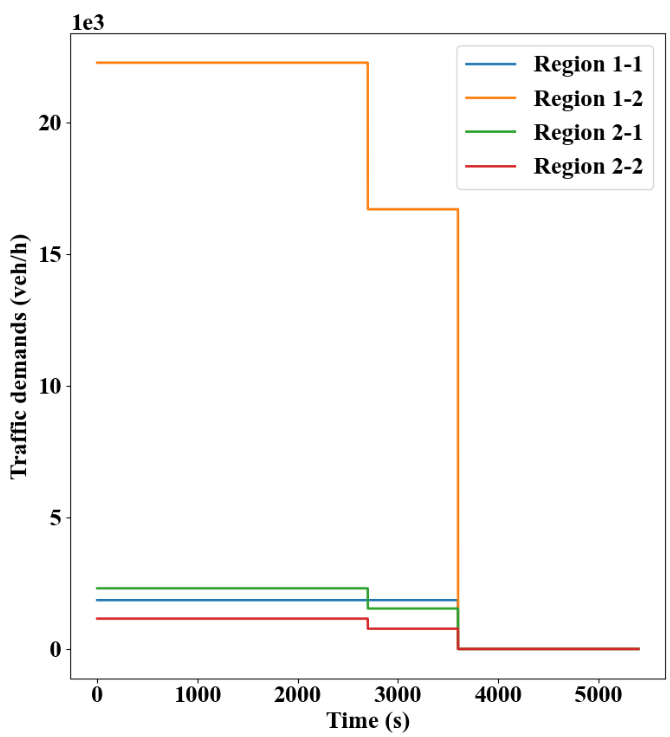

Fig. 7. Demand profiles for two-region networks.

Experiment Results

The learning curve of the MS-RL scheme is provided in Fig. 8, together with the CTC values of the baseline methods. Similar observations and conclusions can be drawn as in Fig. 3. However, the two-region network has significantly more intra-regional intersections and a higher degree of interdependencies between them; thus, the learning curve is more fluctuant compared to the single-region counterpart.

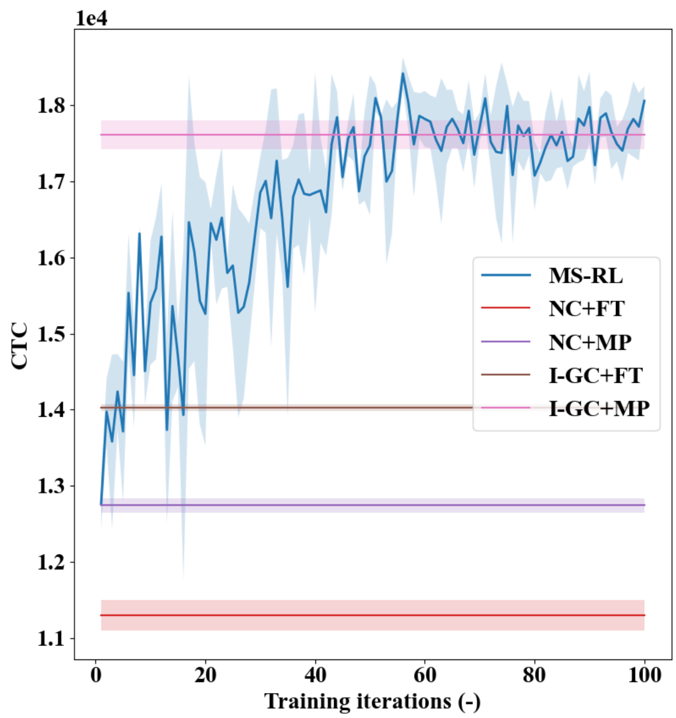

Fig. 8. Learning curve of the MS-RL scheme for two-region networks.

The learning robustness of the MS-RL scheme against combined measurement noise of regional accumulation and identification errors of vehicle counts is also examined here (see Fig. 9). Note, in this set of experiments, the levels of noise and errors are comparatively smaller than in single-region control to account for increased network size (and number of signal control intersections). As shown, despite a lot noisier learning curves (due to more intersections that are prone to identification errors and more accumulation values that are inaccurate), the MS-RL can still compete with (or outperform) the I-GC+MP.

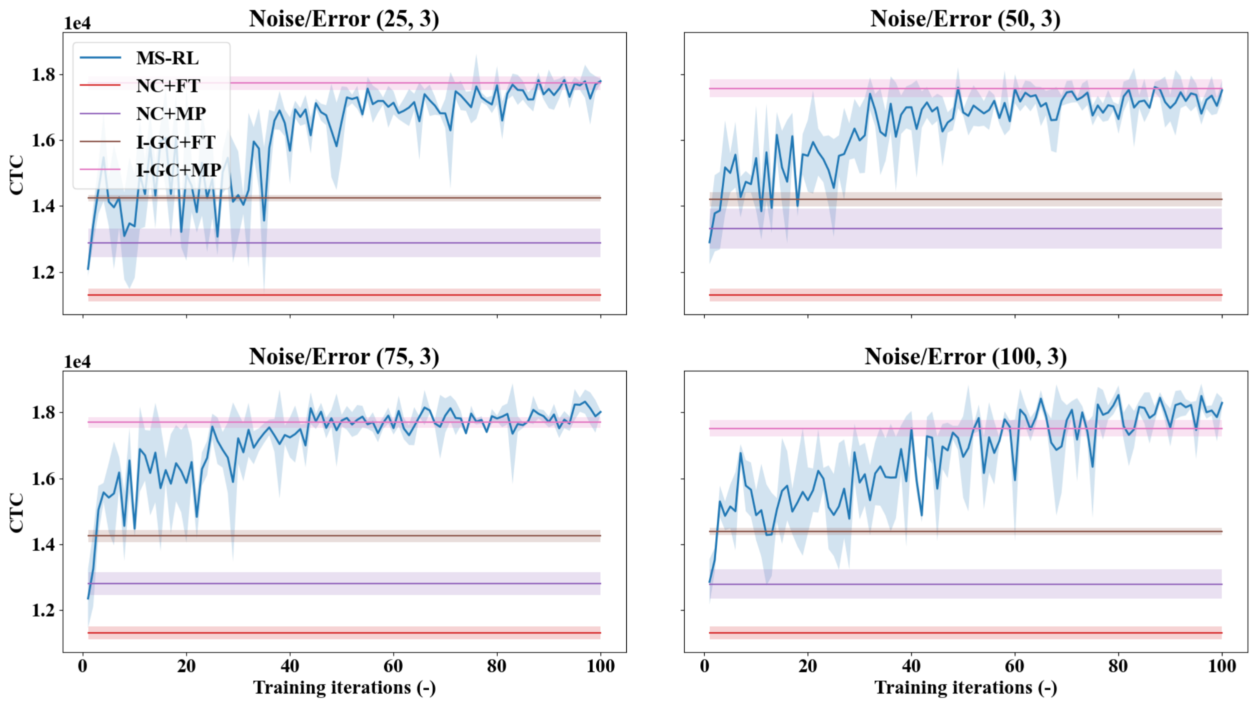

Fig. 9. Learning curves against combined measurement noise and identification errors.

Finally, a test is conducted where a partial implementation of the MP policy or the L-RL agent at the lower level is considered, i.e., only a subset of intersections are controlled by MP or L-RL, as similar to (*28*). Here, the ratio of local intersections managed by MP or L-RL is altered from 10% to 90%, while 0% and 100% respectively denotes FT and MP or L-RL. For each control ratio, 10 random configurations are generated which indicate the random subsets of local intersections to be managed by MP or L-RL. The baseline methods are applied to each configuration under each control ratio and the overall performances are expressed using boxplots (excluding FT methods that are not affected by different MP or L-RL control ratios); see Fig. 10. The learning curves of the MS-RL scheme under different L-RL control ratios are shown in Fig. 11, where the one with 100% L-RL ratio (subplot (f), which is the same as Fig. 8) is also included for perspective. As shown, irrespective of the control ratio, the MS-RL can always learn to compete with the I-GC+MP. Critically, the results show that, in practice one only needs to control a subset of the local intersections (e.g., 50%) using the MP policy or L-RL. In this way, notable control benefits can still be obtained, while enjoying a much lower data collection and computation cost as well as infrastructure requirement. These observations thus emphasize the practical applicability of the MS-RL scheme.

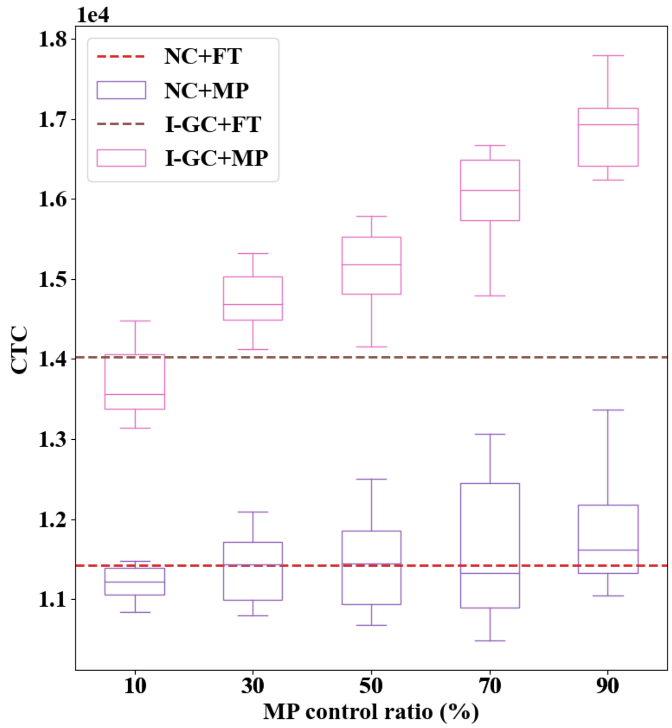

Fig. 10. Performances of the baseline methods under each MP control ratio.

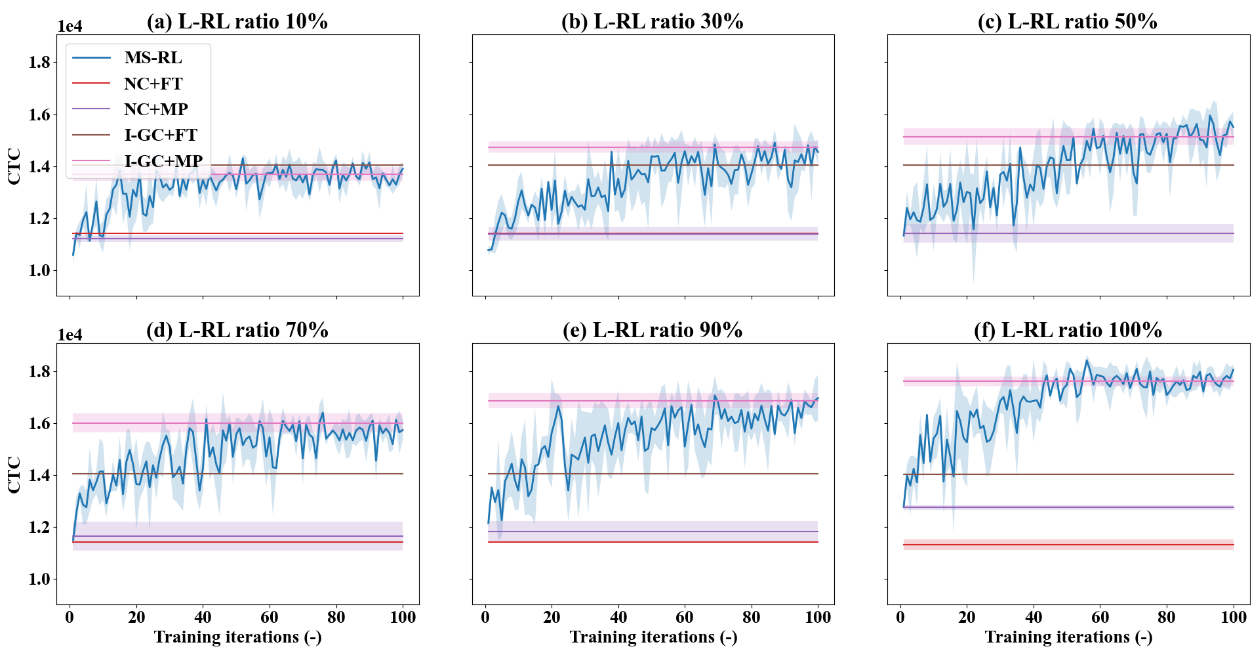

Fig. 11. Learning curves for each L-RL control ratio.

CONCLUDING REMARKS

This paper presents a multi-scale deep reinforcement learning approach for the joint perimeter and signal control problem in urban networks. Using established techniques like parameter sharing and independent learning, the method exhibits excellent control effectiveness and learning robustness. Specifically, the method has been shown effective at regulating traffic and mitigating congestion in simulated single- and two-region networks. It can also readily conduct learning in the presence of environment uncertainties like measurement noise of regional accumulation and identification errors of vehicle counts. Further, to alleviate the data and computation requirements associated with full-scale local signal control in larger urban networks, the method has been shown capable of learning under a partial control configuration at the local intersections. In all cases, the method can compete with and often times outperform a baseline with the max pressure policy at the lower level and established perimeter controllers at the upper level. This joint control framework holds promise for efficient city-level traffic management and contributes ultimately to emerging intelligent transportation systems, with a learning-based design (not built upon heuristic rules) and comprehensive (featuring perimeter and signal control) yet fully implementable policies. Extensions to this work should consider examining the effectiveness of the MS-RL scheme in multi-region networks. Comparisons to methods with networked reinforcement learning agents should also be a research priority.

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: D. Zhou. V. V. Gayah; analysis and interpretation of results: D. Zhou. V. V. Gayah; draft manuscript preparation: D. Zhou. V. V. Gayah. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. 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.
2. Ampountolas, K., N. Zheng, and N. Geroliminis. Macroscopic Modelling and Robust Control of Bi-Modal Multi-Region Urban Road Networks. *Transportation Research Part B: Methodological*, Vol. 104, 2017, pp. 616–637. https://doi.org/10.1016/j.trb.2017.05.007.
3. 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.
4. 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.
5. Yocum, R., and V. V. Gayah. Coordinated Perimeter Flow and Variable Speed Limit Control for Mixed Freeway and Urban Networks. *Transportation Research Record*, Vol. 2676, No. 1, 2022, pp. 596–609. https://doi.org/10.1177/03611981211036677/ASSET/IMAGES/LARGE/10.1177_03611981211036677-FIG8.JPEG.
6. Keyvan-Ekbatani, M., A. Kouvelas, I. Papamichail, and M. Papageorgiou. Exploiting the Fundamental Diagram of Urban Networks for Feedback-Based Gating. *Transportation Research Part B: Methodological*, Vol. 46, No. 10, 2012, pp. 1393–1403. https://doi.org/10.1016/j.trb.2012.06.008.
7. Keyvan-Ekbatani, M., M. Yildirimoglu, N. Geroliminis, and M. Papageorgiou. Multiple Concentric Gating Traffic Control in Large-Scale Urban Networks. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 4, 2015, pp. 2141–2154.

https://doi.org/10.1109/TITS.2015.2399303.

8. 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.
9. 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.
10. 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.
11. Hajiahmadi, M., J. Haddad, B. De Schutter, and N. Geroliminis. Optimal Hybrid Perimeter and Switching Plans Control for Urban Traffic Networks. *IEEE Transactions on Control Systems Technology*, Vol. 23, No. 2, 2015, pp. 464–478. https://doi.org/10.1109/TCST.2014.2330997.
12. 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.
13. Ding, H., J. Zhou, X. Zheng, L. Zhu, H. Bai, and W. Zhang. Perimeter Control for Congested Areas of a Large-Scale Traffic Network: A Method against State Degradation Risk. *Transportation Research Part C: Emerging Technologies*, Vol. 112, 2020, pp. 28–45. https://doi.org/10.1016/J.TRC.2020.01.014.
14. Zhou, D., and V. V. Gayah. Scalable Multi-Region Perimeter Metering Control for Urban Networks: A Multi-Agent Deep Reinforcement Learning Approach. *Transportation Research Part C: Emerging Technologies*, Vol. 148, 2023, p. 104033. https://doi.org/10.1016/J.TRC.2023.104033.
15. Zhou, D., and V. V. Gayah. Improving Deep Reinforcement Learning-Based Perimeter Metering Control Methods With Domain Control Knowledge. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2677, No. 7, 2023, pp. 384–405. https://doi.org/10.1177/03611981231152466.
16. Chen, C., Y. P. Huang, W. H. K. Lam, T. L. Pan, S. C. Hsu, A. Sumalee, and R. X. Zhong. Data Efficient Reinforcement Learning and Adaptive Optimal Perimeter Control of Network Traffic Dynamics. *Transportation Research Part C: Emerging Technologies*, Vol. 142, 2022, p. 103759. https://doi.org/10.1016/J.TRC.2022.103759.
17. Su, Z. C., A. H. F. Chow, and R. X. Zhong. Adaptive Network Traffic Control with an Integrated Model-Based and Data-Driven Approach and a Decentralised Solution Method. *Transportation Research Part C: Emerging Technologies*, Vol. 128, 2021, p. 103154. https://doi.org/10.1016/J.TRC.2021.103154.
18. Zhou, D., and V. V. Gayah. Evaluating the Effectiveness and Transferability of a Data-Driven Two-Region Perimeter Control Method Using Microsimulation. *Transportation Research Record: Journal of the Transportation Research Board*, 2024. https://doi.org/10.1177/03611981241230313.
19. 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.
20. 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.
21. 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.
22. Gayah, V. V, X. S. Gao, and A. S. Nagle. On the Impacts of Locally Adaptive Signal Control on Urban Network Stability and the Macroscopic Fundamental Diagram. *Transportation Research Part*

B: Methodological, Vol. 70, 2014, pp. 255–268.

23. 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.
24. Zhou, Z., B. De Schutter, S. Lin, and Y. Xi. Two-Level Hierarchical Model-Based Predictive Control for Large-Scale Urban Traffic Networks. *IEEE Transactions on Control Systems Technology*, Vol. 25, No. 2, 2017, pp. 496–508. https://doi.org/10.1109/TCST.2016.2572169.
25. Fu, H., N. Liu, and G. Hu. Hierarchical Perimeter Control with Guaranteed Stability for Dynamically Coupled Heterogeneous Urban Traffic. *Transportation Research Part C: Emerging Technologies*, Vol. 83, 2017, pp. 18–38. https://doi.org/10.1016/j.trc.2017.07.007.
26. Yang, K., N. Zheng, and M. Menendez. Multi-Scale Perimeter Control Approach in a Connected-Vehicle Environment. *Transportation Research Part C: Emerging Technologies*, Vol. 94, 2018, pp. 32–49. https://doi.org/10.1016/J.TRC.2017.08.014.
27. Keyvan-Ekbatani, M., X. Gao, V. V. Gayah, and V. L. Knoop. Traffic-Responsive Signals Combined with Perimeter Control: Investigating the Benefits. *Transportmetrica B: Transport Dynamics*, Vol. 7, No. 1, 2019, pp. 1402–1425. https://doi.org/10.1080/21680566.2019.1630688.
28. Tsitsokas, D., A. Kouvelas, and N. Geroliminis. Two-Layer Adaptive Signal Control Framework for Large-Scale Dynamically-Congested Networks: Combining Efficient Max Pressure with Perimeter Control. *Transportation Research Part C: Emerging Technologies*, Vol. 152, 2023, p. 104128. https://doi.org/10.1016/J.TRC.2023.104128.
29. Kouvelas, A., J. Lioris, S. A. Fayazi, and P. Varaiya. Maximum Pressure Controller for Stabilizing Queues in Signalized Arterial Networks. *https://doi.org/10.3141/2421-15*, Vol. 2421, No. 1, 2014, pp. 133–141. https://doi.org/10.3141/2421-15.
30. Su, Z. C., A. H. F. Chow, C. L. Fang, E. M. Liang, and R. X. Zhong. Hierarchical Control for Stochastic Network Traffic with Reinforcement Learning. *Transportation Research Part B: Methodological*, Vol. 167, 2023, pp. 196–216. https://doi.org/10.1016/J.TRB.2022.12.001.
31. Varaiya, P. Max Pressure Control of a Network of Signalized Intersections. *Transportation Research Part C: Emerging Technologies*, Vol. 36, 2013, pp. 177–195. https://doi.org/10.1016/j.trc.2013.08.014.
32. Liu, H., and V. V. Gayah. A Novel Max Pressure Algorithm Based on Traffic Delay. *Transportation Research Part C: Emerging Technologies*, Vol. 143, 2022, p. 103803. https://doi.org/10.1016/J.TRC.2022.103803.
33. 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.
34. 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.
35. Horgan, D., J. Quan, D. Budden, G. Barth-Maron, M. Hessel, H. van Hasselt, and D. Silver. Distributed Prioritized Experience Replay. 2018.
36. 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.
37. 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.
38. Lopez, P. A., M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. P. Flotterod, R. Hilbrich, L. Lucken, J. Rummel, P. Wagner, and E. Wiebner. Microscopic Traffic Simulation Using SUMO. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, Vol. 2018-November, 2018,

pp. 2575–2582. https://doi.org/10.1109/ITSC.2018.8569938.

39. Girault, J. T., V. V. Gayah, S. I. Guler, and M. Menendez. Exploratory Analysis of Signal Coordination Impacts on Macroscopic Fundamental Diagram. *https://doi.org/10.3141/2560-05*, Vol. 2560, 2016, pp. 36–46. https://doi.org/10.3141/2560-05.

40. Cascetta, E., A. Nuzzolo, F. Russo, and A. Vitetta. A Modified Logit Route Choice Model Overcoming Path Overlapping Problems. Specification and Some Calibration Results for Interurban Networks. 1996.