Regional Multi-Agent Cooperative Reinforcement Learning for City-Level Traffic Grid Signal Control

Yisha Li, Ya Zhang, *Senior Member, IEEE*, Xinde Li, *Senior Member, IEEE*, and Changyin Sun, *Senior Member, IEEE*

***Abstract*—This article studies the effective traffic signal control problem of multiple intersections in a city-level traffic system. A novel regional multi-agent cooperative reinforcement learning algorithm called RegionSTLight is proposed to improve the traffic efficiency. Firstly a regional multi-agent Q-learning framework is proposed, which can equivalently decompose the global Q value of the traffic system into the local values of several regions. Based on the framework and the idea of human-machine cooperation, a dynamic zoning method is designed to divide the traffic network into several strong-coupled regions according to real-time traffic flow densities. In order to achieve better cooperation inside each region, a lightweight spatio-temporal fusion feature extraction network is designed. The experiments in synthetic, real-world and city-level scenarios show that the proposed RegionSTLight converges more quickly, is more stable, and obtains better asymptotic performance compared to state-of-the-art models.**

***Index Terms*—Human-machine cooperation, mixed domain attention mechanism, multi-agent reinforcement learning, spatio-temporal feature, traffic signal control.**

I. Introduction

NOWADAYS, with rapid urbanization and transportation system modernization, the size of the urban traffic system has increased dramatically. The efficiency of urban traffic system is vital to many aspects such as the economy development and air pollution. From the perspective of traffic management, traffic signal control (TSC) is a widely used and feasible approach [1], [2].

Conventional methods such as FixTime [3], GreenWave [4] and MaxQueue [5] treat TSC as an optimization problem and solve it based on rules under assumptions like having unlimited lane capacity [6]. However, these rule-based methods can hardly handle practical complex traffic systems since these assumptions cannot be satisfied in practice [7].

Reinforcement learning (RL) is a powerful artificial intelligence paradigm for dynamic control and has been widely applied by researchers to TSC in the past few years. In particular, Q-learning, which is one of the most widely used value-based RL algorithms, was first applied to control a single intersection in [8]. With the development of deep learning (DL), deep reinforcement learning (DRL) which is the combination of RL and DL emerges to improve the performance of RL by using DL to enhance the capability of feature extraction [9]. The deep Q network (DQN) is one of the most commonly used value-based DRL algorithms, which uses the target network and memory replay to stabilize the learning process. Li *et al.* [10] used DQN to control traffic signals of a single intersection and achieved better performance compared with conventional rule-based methods and tabular reinforcement learning methods.

When it comes to TSC in multiple intersections, a direct method is to regard the entire network composed of all intersections as an object to control which is known as centralized learning. However there exists a problem where the state and action spaces grow exponentially with the increasing number of intersections, i.e., the curse of dimensionality. To handle this problem, Wu and Lou [11] applied a sequence-to-sequence model with the attention mechanism to decompose the state-action space into sub-spaces and proposed a DRL model based on Meta-learning which decoupled task inference and control to accelerate the learning process. Another method is to use single-agent RL algorithms directly on each intersection which is called an independent RL (IRL). For example, Prashanth and Bhatnagar [12] proposed a Q-learning algorithm with linear function approximation for TSC of a single intersection, which adopted feature-based state representations and was applicable to the environment of multiple intersections by using IRL. Alegre *et al.* [13] further proposed a linear function approximation based on Fourier basis functions in a network of signalized intersections which has the advantage of having error bounds.

Although IRL can handle the curse of dimensionality, it cannot guarantee convergence due to its disregard for environment uncertainty caused by other agents. Multi-agent reinforcement learning algorithms (MARL) are very suitable for TSC of multiple intersections whose foundation is game theory, and the basic multi-agent Q-learning algorithm was first applied to TSC of multiply intersections in [14]. However it is difficult for intersections in real-world traffic systems to

Manuscript received January 22, 2024; revised February 23, 2024; accepted February 29, 2024. This work was supported by the National Science and Technology Major Project (2021ZD0112702), the National Natural Science Foundation (NNSF) of China (62373100, 62233003), and the Natural Science Foundation of Jiangsu Province of China (BK20202006). Recommended by Associate Editor Xiaohua Ge. *(Corresponding authors: Ya Zhang and Xinde Li.)*

Citation: Y. Li, Y. Zhang, X. Li, and C. Sun, "Regional multi-agent cooperative reinforcement learning for city-level traffic grid signal control," *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 9, pp. 1987–1998, Sept. 2024.

The authors are with the School of Automation, Southeast University, Nanjing 210096, and also with the Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Southeast University, Nanjing 210096, China (e-mail: 220211901@seu.edu.cn; yazhang@seu.edu.cn; xindeli@seu.edu.cn; cysun@seu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2024.124365

observe the global state and action, which makes multi-agent Q-learning algorithm not feasible in practice. Therefore, there are several works developing MARL for TSC under the idea of centralized training with decentralized execution [15]–[17]. Under the idea of coordinated learning, multi-agent A2C [15] was developed from IA2C (IRL using A2C) to achieve cooperative actions among agents in which each agent took other agents' policies as part of its state. Wang *et al.* [16] used mean field approximation to model the interactions among agents as neighbors' mean action, which made agents learn a better cooperative strategy. Li *et al.* [17] proposed a knowledge-sharing protocol for interaction agents to communicate with each other.

Considering the characteristics of traffic systems, there exists some research dedicated to zoning the entire traffic network into small regions and then use MARL algorithms to control regions, thus simplifying the problem [18]–[20]. Chu *et al.* [18] proposed a zoning method which only considered horizontal and vertical divisions with size limitation. According to real-time traffic flow density, it obtained several subregions and used a linear approximation Q function to control. It is noted that a set of weight parameters corresponding to a specific sub-region size was used and updated. Tan *et al.* [19] divided the large grid into several subregions with the same topology. Each subregion learned its own RL policy and value function with limited actions. Then a centralized global agent learned to aggregate regional values and formed the global Q-function. However, as the road network grows larger, the global Q-function becomes difficult to learn. Jiang *et al.* [20] proposed a traffic network decomposition approach to divide the large grid into subregions with different degrees of connectivity, and then trained subregions instead of the entire network synchronously.

Although the above works can overcome the partial observation problem, it is not proper to tackle TSC in large-scale traffic systems due to its model complexity. The graph neural network is very suitable for graph modeled problems and it can efficiently capture the relationships between nodes and the global structure of the graph. In [21], a graph convolutional network with an embedded self-attention mechanism was proposed and it utilized dynamic attention of neighbors to help agents achieve more effective collaboration. Wei *et al.* [22] adopted the graph attention network to gather state information of neighbors thus making each intersection agent learn to cooperate with others. Moreover, Wu *et al.* [23] proposed a spatio-temporal graph attention network to extract spatio-temporal features from the local state of each intersection agent and its neighbors thus making it possible to achieve better cooperation performance. However, the spatio-temporal graph attention network proposed in [23] is not lightweight enough to handle TSC in city-level traffic systems.

This paper studies TSC of large-scale traffic systems. First, in terms of MARL, the regional multi-agent Q-learning framework especially for TSC is derived from general multi-agent Q-learning. This framework can make the global Q value of the traffic system equivalent to the sum of local values of regions thus making it scalable to large-scale traffic system. Specifically, the entire traffic network can be divided into several regions which have the characteristics of internal strong coupling and external weak coupling. Then a dynamic zoning approach based on the idea of human-machine cooperation, which divides the entire traffic network into several regions according to the real-time traffic flow density on each road, is designed to make the aforementioned framework applicable. Moreover, in order to achieve better cooperation inside each region, a lightweight spatio-temporal fusion feature extraction network is designed to obtain spatio-temporal fusion features for each intersection from the local states of its neighbors in the region. Specifically, it introduces the LSTM network to extract the temporal feature of intersections and uses the mixed domain attention mechanism to obtain the spatio-temporal fusion features of intersections.

In summary, the major contributions of this paper are as follows.

1) A regional multi-agent Q-learning framework is developed to simplify the overall TSC problem to several regional control problems under the idea of coordinated learning, which is scalable to large-scale multi-agent systems.

2) A dynamic zoning approach is designed to realize human-machine cooperation by which the entire traffic network is divided into several regions according to real-time traffic flow density.

3) A lightweight spatio-temporal fusion feature extraction network is designed to achieve better cooperation inside each region.

The remainder of this paper is organized as follows. Section II introduces the traffic signal control model in a large-scale network and analyses the feasibility of the regional multi-agent reinforcement learning framework for TSC which is the theoretical basis of the proposed RegionSTLight. In Section III, the proposed RegionSTLight is introduced in detail. The numerical experiments are conducted under a synthetic scenario, a real-world scenario and a city-level scenario in Section IV to illustrate the effectiveness of the proposed method. Finally, some conclusions are drawn in Section V.

II. Problem Formulation and Feasibility Analysis

A. Traffic Signal Control Modeling in Large-Scale Network

The traffic system can be modeled as a directed graph $G(I,E)$ where each intersection is seen as a node, the road between two intersections is seen as an edge in the graph, I denotes the collection of intersections and E denotes the collection of roads. $V_i \subseteq I$ represents the set of the neighbors of intersection i, and $e_{ij} \in E$ denotes the road from intersection i to intersection j. It has been proven that the influence between adjacent intersections can be fully described by the traffic flow on the connecting road [24]. Therefore we use the traffic flow as the correlation between intersections to design the zoning method.

A 4-direction intersection shown in Fig. 1 is taken as an example to illustrate the definitions which can be easily generalized to different intersection structures.

1) Road: This is defined as the edge between two intersections. To specify the direction of the road corresponding to intersection, the road can be classified into the input road and

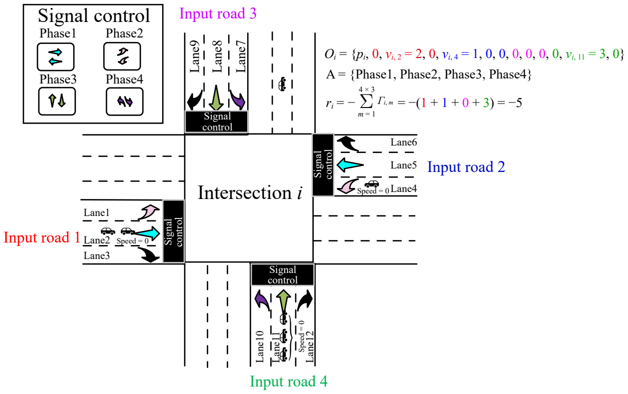


Fig. 1. The definition of observation, action and reward.

output road. The road with the direction entering an intersection is defined as the input road of the intersection and the road with the direction exiting an intersection is defined as the output road. There are 4 input roads in intersection i in Fig. 1.

2) Lane: This is defined as the area where vehicles can drive for a long time; there are 3 lanes on each road in Fig. 1. The lane on the input road is defined as an input lane, and that on the output road is defined as an output lane.

3) Entering Direction: The standard intersection contains four entering directions including north, south, west and east.

4) Flow Direction: This is defined as the direction which traffic flow moves from one intersection to another on the road.

5) Traffic Movement: This is defined as the traffic moving towards a certain direction within an intersection, such as left turns, straight travel and right turns. Typically, right turn traffic can pass regardless of traffic signals but with low priority. Furthermore, the number of lanes one traffic movement can occupy is variable.

6) Link: This is defined as the drivable trajectory from an input lane to an output lane in a traffic movement. For example, for an input lane, there are three output lanes for straight travel in Fig. 1; thus straight travel has three possible links.

7) Phase: This is a combination of traffic signals for different traffic movements. For example, there are four types of phases in Fig. 1 including west-east straight travel, west-east left turn, south-north straight travel and south-north left turn.

Since the traffic system is modeled as a directed graph, the traffic signal control can be defined as a fully cooperative MARL task where each intersection is controlled by an agent, and the whole process is modeled as a decentralized partially observable Markov decision process (Dec-POMDP) [25] represented by $\langle N, \mathbf{S}, \mathbf{O}, \mathbf{A}, \mathbf{R}, P, \boldsymbol{\pi}\rangle$, where N is the number of agents in the game system, $\mathbf{S}$ is the joint state space of the system and the joint state $\boldsymbol{s} = \{s_i, i = 1, \ldots, N\}$ is the collection of the states of all agents, and $\mathbf{O}$ is the joint observation space of the system which is a part of $\mathbf{S}$ and the joint observation is represented by $\boldsymbol{o}$. Similarly $\mathbf{A} = \mathbf{A}_1 \times \cdots \times \mathbf{A}_N$ is the joint action space and the joint action $\boldsymbol{a} = \{a_i, i = 1, \ldots, N\}$ is the collection of the actions of all agents, $\mathbf{R} = \{r_i, i = 1, 2, \ldots, N\}$ is the utility space after all agents take their actions and $r_i : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \to \mathbb{R}$ is the reward function of agent i, $P : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \to [0, 1]$ is the state transition probability distribution of the system, $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_N\}$ is the joint strategy.

At each time step, under the joint strategy $\boldsymbol{\pi}$, the joint action has been taken to interact with the system where each agent chooses its action according to its own strategy and the system observation $\boldsymbol{o}$. After that, the next system observation $\boldsymbol{o}'$ and the total reward $\boldsymbol{r} = \{r_1, \ldots, r_N\}$ can be obtained from the system.

As shown in Fig. 1, consider that each intersection $i \in I$ has Q connected input roads each of which contains M lanes, the local observation of intersection i is designed as

$$o_i = \{p_i, v_{i,m}, m = 1, \ldots, Q \times M\} \tag{1}$$

where p_i represents the current phase at intersection i and $v_{i,m}$ represents the number of vehicles on the input lane m of intersection i.

The local action of intersection i is set to choose the next phase and the local reward function of i is designed as

$$r_i = -\sum_{m=1}^{Q\times M} \Gamma_{i,m} \tag{2}$$

where $\Gamma_{i,m}$ is the queue length on the input lane m of intersection i. The negative value of the sum of the queue lengths is taken as the reward since the goal of RL is to maximize the cumulative rewards.

According to the weak coupling between regions, the queue length between regions is short thus making the reward between regions close to 0, which can in turn prove the theoretical correctness of the proposed regional multi-agent Q-learning framework in Section II-B.

B. Feasibility Analysis of Regional Multi-Agent Q-Learning Framework for Large-Scale TSC

The multi-agent Q-learning framework [26] is a widely-used value-based MARL framework. It sets the target network and samples transitions from replay buffer D to update the Q network to make the training process more stable. Besides, it decouples the selection and evaluation process of action to prevent the problem of high estimation. Based on the Bellman equation [27], the loss function of agent i and its optimization are as follows:

$$\mathcal{L}(\phi_i) = E_{(\boldsymbol{s},\boldsymbol{a},\boldsymbol{r},\boldsymbol{s}')\sim D}\left[\left(Q_{\phi_i}(\boldsymbol{s},\boldsymbol{a}) - y_i\right)^2\right] \tag{3}$$

$$y_i = r_i + \gamma Q_{\phi_i^-}\left(\boldsymbol{s}', \arg\max_{a_i} Q_{\phi_i}(\boldsymbol{s}', \boldsymbol{a})\right) \tag{4}$$

$$\phi_i^* = \arg\min_{\phi_i} \mathcal{L}(\phi_i). \tag{5}$$

where $Q_{\phi_i}(\boldsymbol{s}, \boldsymbol{a})$ represents the value of the state action pair, i.e., the output value of the Q network with ϕ_i as its trainable parameters, y_i indicates the target output value of the Q network, $Q_{\phi_i^-}(\cdot)$ represents the output value of the corresponding target network with ϕ_i^- as its trainable parameters which is often copied from ϕ_i at a certain number of steps, and $\gamma \in [0, 1]$ is the discount factor to limit cumulative rewards. Substituting observations for state, the Q network uses sample estimation and gradient descent to update parameters

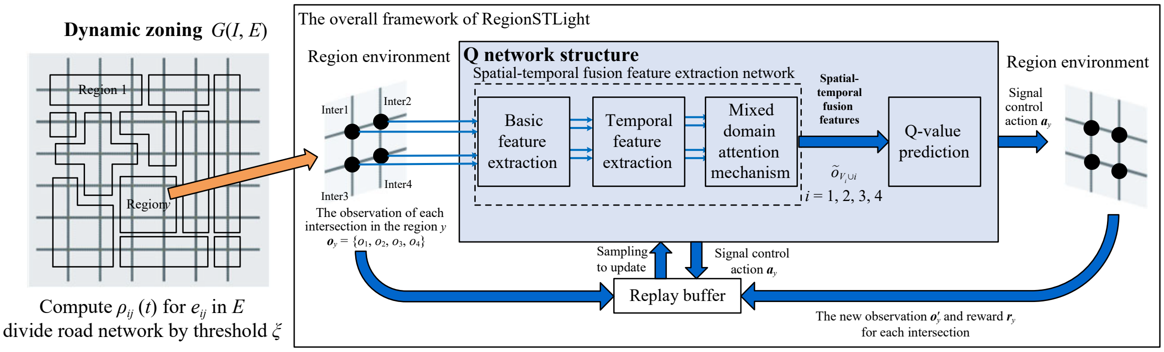

Dynamic zoning $G(I, E)$
Region 1
Region y
Compute $\rho_{ij}(t)$ for e_{ij} in E
divide road network by threshold ξ
The overall framework of RegionSTLight
Region environment
Inter1
Inter2
Inter3
Inter4
The observation of each intersection in the region y
$\boldsymbol{o}_y = \{o_1, o_2, o_3, o_4\}$
Q network structure
Spatial-temporal fusion feature extraction network
Basic feature extraction
Temporal feature extraction
Mixed domain attention mechanism
Spatial-temporal fusion features
$\tilde{o}_{V_i \cup i}$
$i = 1, 2, 3, 4$
Q-value prediction
Region environment
Signal control action $\boldsymbol{a}_y$
Sampling to update
Signal control action $\boldsymbol{a}_y$
Replay buffer
The new observation $\boldsymbol{o}'_y$ and reward $\boldsymbol{r}_y$ for each intersection

Fig. 2. The overall framework of the proposed RegionSTLight.

$$\nabla_{\phi_i}\mathcal{L}(\phi_i) \approx \frac{1}{B}\sum_{j=1}^{B}\nabla_{\phi_i}Q_{\phi_i}\left(\boldsymbol{o}_j,\boldsymbol{a}_j\right)\left(Q_{\phi_i}\left(\boldsymbol{o}_j,\boldsymbol{a}_j\right)-y_{i,j}\right) \tag{6}$$

$$\phi_i \leftarrow \phi_i - \eta\nabla_{\phi_i}\mathcal{L}(\phi_i) \tag{7}$$

where $y_{i,j}$ is the target Q value of agent i in sample j, B is the batch size and η is the learning rate.

The multi-agent Q-learning framework implicitly contains the assumption that $Q(\boldsymbol{o},\boldsymbol{a}) = \sum_{i=1}^{N} Q_{\phi_i}(\boldsymbol{o},\boldsymbol{a})$. To make multi-agent Q-learning framework practicable, the idea of centralized training with decentralized execution is proposed [28]. According to this idea, the total Q value is the linear combination of local Q values, i.e., $Q(\boldsymbol{o},\boldsymbol{a}) = \sum_{i=1}^{N} \psi_i \cdot Q_{\phi_i}(o_i,a_i)$ [29], [30]. More generally, it should satisfy the individual global maximum (IGM) principle to ensure the consistency of the joint and local greedy action selection

$$\begin{aligned}&\forall o \in O,\ \arg\max_{a\in\mathbf{A}} Q(\boldsymbol{o},\boldsymbol{a})\\ &= \left\{\arg\max_{a_1\in\mathbf{A}_1} Q_{\phi_1}(o_1,a_1),\ldots,\arg\max_{a_N\in\mathbf{A}_N} Q_{\phi_N}(o_N,a_N)\right\}.\end{aligned} \tag{8}$$

Under this assumption, each agent can select its action according to its local observation o_i, and the gradients of their parameters are computed by the output of the mixing layer of all the local Q values, that is, the global Q value.

On the other hand, coordinated Q-learning is one MARL trick to balance optimality and scalability by conducting iterative message passing among neighbor agents [31]. The local Q value can be calculated by $Q_{\phi_i}(\boldsymbol{o},\boldsymbol{a}) \approx Q_{\phi_i}(o_i,a_i) + \sum_{j\in V_i} M_j(o_j,a_j,a_{V_j})$ where V_i is the neighbor set of agent i and M_j is the message from neighbor j. Therefore the global Q value is the sum of all local values. There are two implementation methods. One uses the policy of neighbors as the message to pass $Q_{\phi_i}(\boldsymbol{o},\boldsymbol{a}) \approx Q_{\phi_i}(o_i,a_i,\phi_{N_i})$ [15] and the other directly adopts its own observation [32] and action as input $Q_{\phi_i}(\boldsymbol{o},\boldsymbol{a}) \approx Q_{\phi_i}(o_i,a_i)$, that is the independent Q-learning (IQL) [33].

In this paper, we design an MARL framework named as regional multi-agent Q-learning for large-scale TSC. The entire road network is divided into several regions according to the real-time traffic flow on connecting roads of intersections, making the intersections in each region strongly coupled (with enough vehicles on the connecting road) while the coupling between regions is weak. Then inspired by [15], [22], the message from neighbors is designed as the weighted sum of the local features of neighbors in each region to achieve cooperation within the region. The Q value of intersection i satisfies that

$$Q_{\phi_i}(\boldsymbol{o},\boldsymbol{a}) \approx Q_{\phi_i}(\tilde{o}_{V_i\cup i},a_i) \tag{9}$$

where $\tilde{o}_{V_i\cup i}$ represents the weighted sum of the local features of target intersection i and its neighbors in the located region.

Due to weak coupling between regions, the need for interregional cooperation is minimized. The global Q value equals the sum of all regional values, i.e.,

$$Q(\boldsymbol{o},\boldsymbol{a}) = \sum_{y=1}^{Y} Q_y(\boldsymbol{o}_y,\boldsymbol{a}_y) \tag{10}$$

where $Q_y(\boldsymbol{o}_y,\boldsymbol{a}_y) = \sum_{i\in N_y} Q_{\phi_i}(\tilde{o}_{V_i\cup i},a_i)$ is the sum of local values of intersections in region y, Y is the number of regions and N_y is the set of intersections in region y, $\boldsymbol{o}_y = \{o_i, i \in N_y\}$ is the joint observation of region y and $\boldsymbol{a}_y = \{a_i, i \in N_y\}$ is its joint action.

III. The Proposed RegionSTLight

A. Overall Framework of the Proposed RegionSTLight

The overall framework of the proposed RegionSTLight is shown in Fig. 2. Firstly the entire road network is divided into several regions according to real-time traffic flow. Then, the local observations of intersections in each region are input into the spatio-temporal fusion feature extraction network to obtain the spatio-temporal fusion features $\{\tilde{o}_{V_i\cup i}, i \in N_y\}$ of all intersections in this region. Finally the spatio-temporal fusion features are input into the Q-value prediction to output the state-action values of each intersection in region y. Through the *ϵ-greedy* exploration strategy, the joint action of each region is obtained and conducted in the traffic environment thus generating reward and the next state of each intersection in the environment. All of the regional samples $(\boldsymbol{o}_j,\boldsymbol{a}_j,\boldsymbol{r}_j,\boldsymbol{o}'_j)$, $j = 1,\ldots,Y$, are stored in the replay buffer to update the parameters of spatio-temporal fusion feature extraction network and Q-value prediction.

B. Dynamic Zoning of the Traffic Network

In order to make the TSC scheme scalable to urban traffic and utilize expert knowledge, the entire traffic network $G(I,E)$ is divided into several regions at each control step

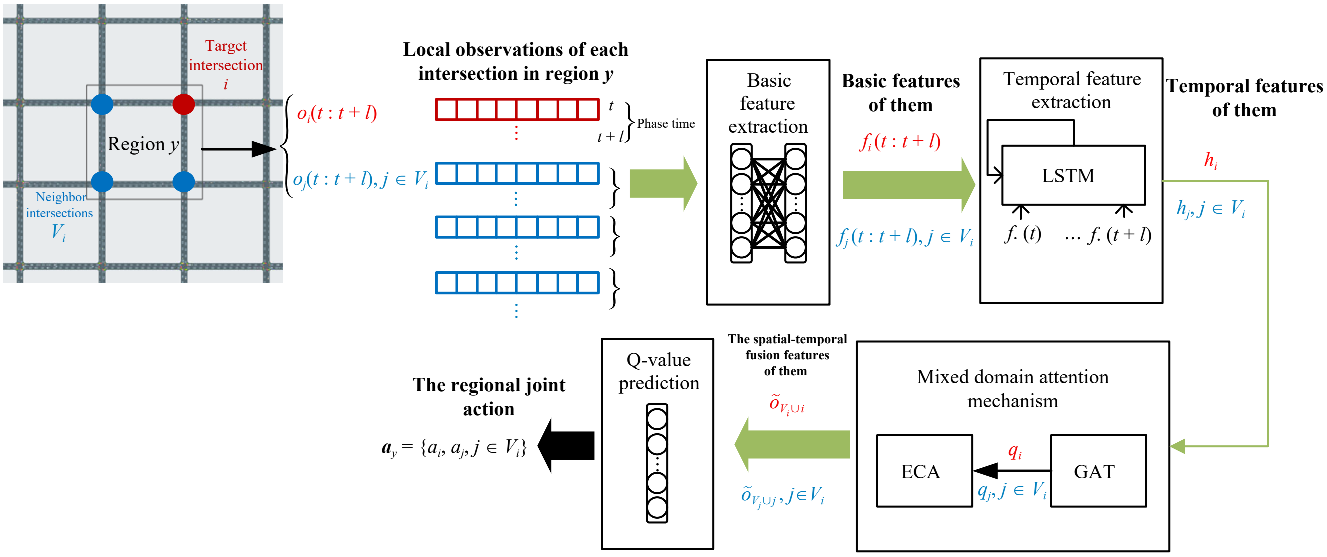

Target intersection i
Region y
Neighbor intersections V_i
Local observations of each intersection in region y
$o_i(t:t+l)$
$o_j(t:t+l), j \in V_i$
t
$t+l$
Phase time
Basic feature extraction
Basic features of them
$f_i(t:t+l)$
$f_j(t:t+l), j \in V_i$
Temporal feature extraction
LSTM
$f.(t)$ … $f.(t+l)$
Temporal features of them
h_i
$h_j, j \in V_i$
Mixed domain attention mechanism
ECA
GAT
q_i
$q_j, j \in V_i$
The spatial-temporal fusion features of them
$\tilde{o}_{V_i \cup i}$
$\tilde{o}_{V_j \cup j}, j \in V_i$
Q-value prediction
The regional joint action
$\boldsymbol{a}_y = \{a_i, a_j, j \in V_i\}$

Fig. 3. The entire Q network structure of RegionSTLight.

according to the real-time traffic flow on the roads by a designed rule explained below.

Firstly, the traffic flow density on road $e_{ij} \in E$ at time t is designed as

$$\rho_{ij}(t) = \frac{\sum_{m \in L_{ij}} x_{ij}^m(t)}{M \times \bar{n}} \tag{11}$$

where L_{ij} is the set of lanes on road e_{ij}, M is the number of lanes on road e_{ij} and $\bar{n}$ is the average lane capacity of the road e_{ij}, $x_{ij}^m(t)$ represents the number of vehicles on lane m of road e_{ij} at time t.

Then, the dynamic zoning principle is as follows. The intersection i and j are assigned into the same region if $\rho_{ij}(t) \geq \xi$, where $\xi \in [0,1)$ is the division threshold and is decided by experiments. The size of each region is set to 5, i.e., the number of intersections in one region is no more than 5, under the assumption that the intersections in a much larger region are not closely connected to each other. Once the size of the divided region is larger than 5, all of its intersections will be viewed as independent regions each of which only contains one intersection. In order to enrich the observation of the independent region, the local observations of geographic neighbor intersections are given for its feature extraction. Moreover, the neighbors of intersection i are other intersections in the region that i belongs to.

Therefore, the traffic flow densities on every road are calculated by (11) at the beginning of each control step and the traffic network is divided into several regions according to the traffic flow densities. After that, the intersections in each region learn how to cooperate with each other by extracting the spatio-temporal fusion features and the optimal global joint action can be obtained by concatenating optimal joint actions of all regions because the IGM principle is satisfied in theory.

C. Q Network Design

The entire Q network structure is shown in Fig. 3 including basic feature extraction, temporal feature extraction, mixed domain attention mechanism and Q-value prediction. Each part is illustrated in detail as follows.

1) Basic Feature Extraction: Firstly the local observations of the intersections in one region will be input into a two-layer multi-layer perceptron (MLP) to obtain their basic features. The basic feature of intersection i is obtained by

$$f_i' = \text{Dense}(o_i) \tag{12}$$

$$f_i = \text{Dense}(f_i') \tag{13}$$

where o_i is the local observation of i, Dense is the full-connected layer and f_i is the basic feature of i. Note that in order to extract temporal feature, during one phase with length l, the local observations $o_i(t:t+l)$ of intersection i at every timestamp should be input into the basic feature extraction as a batch.

2) Temporal Feature Extraction: An LSTM network is adopted to extract temporal features of intersections from their basic features

$$h_i = \text{LSTM}(f_i(t:t+l)) \tag{14}$$

where h_i is the temporal feature of intersection i and the basic features $f_i(t:t+l)$ of intersection i at all timestamp during one phase make up one input of LSTM network while local observation at each timestamp is an input of basic feature extraction.

3) Mixed Domain Attention Mechanism: A mixed domain attention mechanism is proposed to extract the spatial relationship of adjacent intersections and the local relationship of feature dimensions. Firstly, the temporal features of intersection i and its neighbors are input into the graph attention network (GAT) [34] to fuse the spatial information

$$q_i = \text{GAT}(h_i, \{h_j, j \in V_i\}) \tag{15}$$

where q_i is the feature of i which has fused the spatial information from features of its neighbors. The principle of GAT is

$$\varepsilon_{ij} = (h_i W_t) \times (h_j W_s)^T \tag{16}$$

$$\alpha_{ij} = \frac{\exp(\varepsilon_{ij})}{\sum_{j\in V_i}\exp(\varepsilon_{ij})} \tag{17}$$

$$q_i = \sigma(W_q \times \sum_{j\in V_i} \alpha_{ij}(h_j W_c) + b_q) \tag{18}$$

where h_i, h_j are the hidden vectors from the local observations o_i, o_j, ε_{ij} represents the influence of neighbor j on agent i, α_{ij} is the attention that agent i should pay to its neighbor j, and q_i is the spatial-weighted feature vector. Firstly the hidden vectors of the target agent and its neighbors are embedded by different dense layers whose parameters are W_t, W_s respectively and ε_{ij}, α_{ij} can be obtained. After that, the hidden vectors of neighbors are first embedded by another dense layer with parameters W_c and then multiplied by the attention α_{ij}, and finally q_i is obtained through a dense layer whose parameters are W_q, b_q.

Since the intersections in one region are neighbors, the fused features q_j of other intersections $j \in V_i$ in the same region can be similarly obtained with the same input features

$$q_j = \mathrm{GAT}(h_j, \{h_k, k \in V_j\}), \quad j \in V_i. \tag{19}$$

Then, the efficient channel attention network (ECA) [35] is used to obtain the local cross-channel relationship of feature and the spatio-temporal fusion feature of intersection i is

$$\tilde{o}_{V_i \cup i} = \mathrm{ECA}(q_i). \tag{20}$$

The principle of ECA is

$$w_i = GAP(q_i) \tag{21}$$

$$att_i = \sigma(w_i * \kappa) \tag{22}$$

$$\tilde{q}_i = q_i \times att_i \tag{23}$$

where $q_i \in \mathbb{R}^{H\times T\times C}$ is the input feature tensor and C is the number of channels, the features of each channel are aggregated through the global average pooling denoted as GAP without dimensionality reduction, the channel attention att_i is obtained by performing 1D convolution $(*)$ and sigmoid activation(σ) on the aggregated channel features $w_i \in \mathbb{R}^{1\times 1\times C}$ and the kernel size of 1D convolution is adaptively computed by $\kappa = \frac{\log_2(C)+\theta}{\lambda}$, $\theta = 1$, $\lambda = 2$, $\tilde{q}_i$ is the channel-weighted feature tensor. The amount of parameters the ECA increases equals to the kernel size of its 1D convolution.

4) Q-Value Prediction: According to the proposed regional multi-agent Q-learning framework, the Q-values of all actions of intersection i in region y can be obtained by

$$\begin{aligned} Q_{\phi_i}(\boldsymbol{o}, \boldsymbol{a}) &= Q_{\phi_i}(\boldsymbol{o}_y, \boldsymbol{a}_y) \\ &= Q_{\phi_i}(\tilde{o}_{V_i\cup i}, a_i) \\ &= \mathrm{Dense}(\tilde{o}_{V_i\cup i}, a_i). \end{aligned} \tag{24}$$

Therefore, the joint action of region y can be obtained through *ϵ-greedy* exploration strategy, and the global joint action of the traffic system can be obtained by directly concatenating the joint actions of all regions.

D. RegionSTLight Algorithm

Based on the proposed regional multi-agent Q-learning framework and specific RL settings in TSC, the RegionSTLight algorithm is developed in Algorithm 1. All the intersections in the traffic system are parameter-shared for scalability.

Algorithm 1 The RegionSTLight Algorithm

```
Inputs: Initial parameters of the Q network, the division threshold
        $\xi$, learning rate $\eta$, batch size for updating $B$ and the update
        frequency of the target network $K$
replay buffer $D = \emptyset$, $\phi^- = \phi$
for episode = 0 to train_episodes do
    reset the environment and get the initial observation $\boldsymbol{o}$
    for step = 0 to maxstep do
        compute flow densities $\rho_{ij}(\mathrm{step}\times l)$ $\forall i,\ j \in I$ and $i \neq j$
        divide the entire road network into $Y$ regions according to
        new flow densities
        for all regions $y$ do
            select regional joint action $\boldsymbol{a}_y$ by using the spatio-tem-
            poral fusion feature and $\epsilon$-greedy exploration strategy
        end for
        concatenate the regional joint action of all regions and
        execute the total joint action $\boldsymbol{a} = \{a_1, \ldots, a_N\}$
        obtain new observation $\boldsymbol{o}'$ and reward $\boldsymbol{r} = \{r_1, \ldots, r_N\}$
        $D = D \cup (\boldsymbol{o}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{o}')$
        $\boldsymbol{o}' \rightarrow \boldsymbol{o}$
        if the number of samples in $D$ reaches B then
            $y_i = r_i + \gamma Q_{\phi^-}(\tilde{o}_{V_i\cup i}, \arg\max_{a_i} Q_\phi(\tilde{o}_{V_i\cup i}, a_i))$ $\forall$ intersection
            $i$ (belonging to region $y$ in the samples) $\in I$
            $\mathcal{L}(\phi) \approx \frac{1}{B}\sum_{j=1}^{B}\sum_{y=1}^{Y_j}(Q_\phi(\boldsymbol{o}_{j,y}, \boldsymbol{a}_{j,y}) - \sum_{i\in N_y} y_i)^2$
            $\phi \leftarrow \phi - \eta\nabla_\phi\mathcal{L}(\phi)$
        end if
    end for
    if $episodes\%K == 0$ then
        $\phi^- = \phi$
    end if
end for
Return: The final Q network parameters
```

At each control step which is the end of each phase, the flow densities of all roads are computed in real-time. Then, all of intersections are assigned to Y regions by judging if the real-time flow densities are larger than the division threshold ξ. After that, the regional observations are input into the spatio-temporal fusion feature extraction network to obtain the spatio-temporal fusion features and the regional joint action can be obtained through Q-value prediction and *ϵ-greedy* exploration strategy. The global joint action $\boldsymbol{a}$ is conducted in the traffic environment, and the next observation $\boldsymbol{o}'$ and reward $\boldsymbol{r}$ of system can be obtained. The sample $(\boldsymbol{o}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{o}')$ is stored into the buffer D. Finally, the average loss of a batch of samples is computed to update the parameters of the Q network if the number of samples in D reaches the batch size B. Specifically the loss of sample j is computed by the sum of losses of its Y_j regions.

E. Model Complexity Analysis

Assume that the dimension of the local observation of each intersection is d_1, the length of phase time is l, the kernel size of ECA is κ, the number of neurons in each hidden layer is d_2 and the number of optional phases is p. Although RegionSTLight adds some parameters to learn the spatio-temporal fusion features of regions owing to the temporal feature extraction and the mixed domain attention mechanism, both the time and space it requires are approximately equal to $O(d_2^2)$, which are small enough to be scalable to thousands of intersections.

1) Space Complexity: The size of the weight matrices and bias vectors in each component of RegionSTLight is as follows. *a) Basic feature extraction:* $d_1d_2 + d_2 + d_2^2 + d_2$; *b) Temporal feature extraction:* $4(2d_2^2 + d_2)$; *c) Mixed domain attention mechanism:* $3d_2^2 + d_2^2 + d_2 + \kappa$; *d) Q-value prediction:* $d_2p + p$. Hence, the total number of trainable parameters to store is $O(d_2(d_1 + 13d_2 + p + 7) + \kappa + p)$. Normally, the size of hidden layer d_2 is far larger than the kernel size of ECA κ, the number of optional phases p and comparable to the dimension of local observation of intersection d_1. Therefore, the space complexity of RegionSTLight approximately equals $O(d_2^2)$.

2) Time Complexity: The following assumptions are made. a) All the intersections in system can simultaneously utilize RegionSTLight to predict Q values. b) The embeddings for either source or target intersection via W_s, W_c and W_t in GAT can be conducted simultaneously. c) The flow densities on all roads can be computed simultaneously. These assumptions can be satisfied by the computer parallel processing technology. Then, the time complexity which only considers multiplication operation in each component of RegionSTLight is as follows. *a) Basic feature extraction:* $d_1d_2 + d_2^2$; *b) Temporal feature extraction:* ld_2^2; *c) Mixed domain attention mechanism:* $d_2^2 + d_2^2 + \kappa d_2$; *d) Q-value prediction:* d_2p. The size of hidden layer d_2 is normally far larger than the length of phase time l, hence the time complexity is $O(d_2(d_1 + (3 + l)d_2 + \kappa + p))$ and similarly it approximately equals $O(d_2^2)$.

IV. Experiments

The proposed RegionSTLight is applied to a commonly used traffic simulator Cityflow [36] with a synthetic scenario to verify the feasibility and effectiveness of each part. A real-world scenario is also provided to verify its effectiveness in practice and a city-level scenario is given to verify scalability. Moreover, we use tensorflow2.4.0, cuda11.0, cudnn8.0.5 and python3.6.5 on a server with Ubuntu16.04.

A. Experimental Scenarios

In the experiment, a synthetic scenario with 6 × 6 road network and two city-level scenarios with 32 × 32 road network and 16 × 64 road network are designed, and a real-world scenario in Jinan is used.

The optional set of phases for each intersection is {north-south straight, west-east straight, north-south turn left, west-east turn left} with right turns allowed all the time. The length of phase l is 15 s under all the scenarios. Moreover the division threshold ξ is set to 0.5, the number of train episodes is 200 and the queue length $\Gamma_{i,m}$ is acquired by counting the number of vehicles with speed less than 0.1 m/s under all the scenarios.

1) Synthetic Scenarios: Each intersection in the synthetic scenario contains four directions (south-north, north-south, east-west and west-east) and each direction contains an input road and an output road. There are 3 lanes (300 meters in length) on each road, so the capacity of lane $\bar{n}$ equals 40 under the assumption that the average length of vehicles is 5 meters and the minimum gap between vehicles is 2.5 meters.

a) The 6 × 6 synthetic scenario: As shown in Fig. 4, the traffic flows in this scenario are designed to form different strong-coupled regions and the simulation time is set to 3600 seconds (maximum step of one episode is 240). There are straight and turning flows on each road cross the road network and their average arrival rate is 120 vehicles/hour. During the first and last half of the simulation time, different regional traffic flows are generated to simulate the dynamic local traffic states.

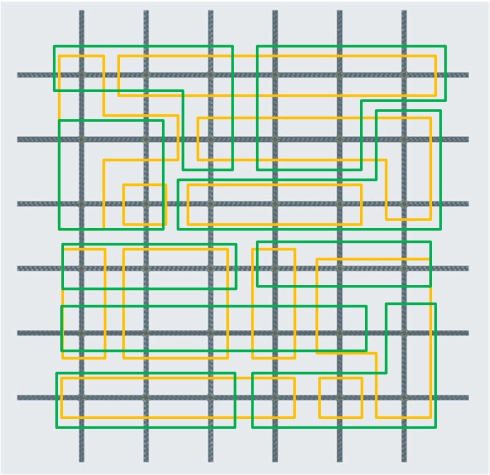


Fig. 4. The traffic flows in the 6 × 6 synthetic scenario: The orange regions are obtained from the traffic flows during the first half of simulation time and the green regions are generated according to the traffic flows during the last half of simulation time.

b) The 32 × 32 synthetic scenario: Similar to the 6 × 6 synthetic scenario, the traffic flows in this scenario are also designed to form different strong-coupled regions to simulate dynamic urban traffic. There are straight and turning flows on each road that cross the road network with their average arrival rate being 120 vehicles/hour. The regional traffic flows are randomly generated, specifically, where the starting and ending intersections are randomly chosen from I and the average arriving rate is uniformly sampled from [90, 240] for each flow. The simulation time is set to 1500 s.

c) The 16 × 64 synthetic scenario: The traffic flows in this scenario are generated in the same manner as the 32 × 32 synthetic scenario. Specifically, there are straight and turning flows on each road that cross the road network with their average arrival rate being 120 vehicles/hour. The regional traffic flows are randomly generated under which the starting and ending intersections are randomly chosen from I and the average arriving rate is uniformly sampled from [90, 240] for each flow. The simulation time is set to 1500 s.

2) Real-World Scenario: As shown in Fig. 5, the traffic flows in the Jinan scenario are collected by roadside cameras near 12 intersections in Dongfeng sub-district, Jinan, China. The simulation time is set to 3600 s. All the captured vehicles are modeled with 5 m length, 2 m width and the maximum speed limited to 11.111 m/s.

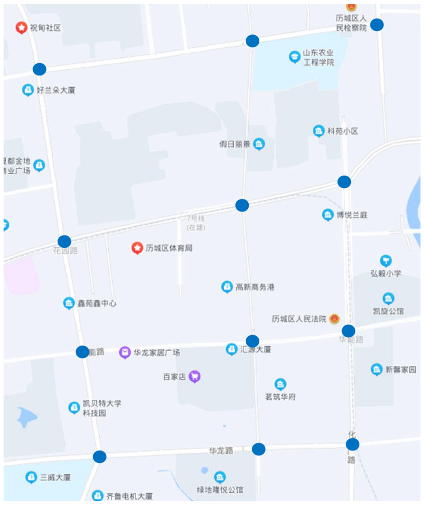


Fig. 5. The road network in Jinan scenario.

B. Different Methods for Comparison and Metrics

1) Different Methods for Comparison: Firstly, in the case study of the real-world scenario, different methods are compared to demonstrate the effectiveness of the proposed RegionSTLight. Then, in the case study of the synthetic scenario, an ablation experiment is conducted in the 6 × 6 synthetic scenario to illustrate the effect of each part of the proposed RegionSTLight and a scalable experiment of different methods is conducted in the 32 × 32 synthetic city-level scenario. All of the methods are listed as follows.

a) FixTime: Choose the next phase in a fixed order for all intersections.

b) MaxPressure [37]: Choose the next phase with the maximum pressure for all intersections.

c) IRL [33]: A MARL model in which an agent corresponds to a Q network and the input of Q network contains local state and action of the agent. In TSC each intersection chooses its next phase according to its own state.

d) GCN [38]: A MARL model for TSC which directly applies GCN to extract spatial features from the geographic neighbors.

e) MADDPG [28]: an MARL model which firstly applies the idea of centralized training with decentralized execution.

f) MA2C [15]: A MARL model for TSC in which each agent takes its neighbor agents' policies as part of its state to achieve cooperative actions.

g) Coder [19]: A MARL model for TSC in which each agent controls a region with the same topology and a centralized global agent is applied to aggregate regional values and form the global Q-function.

h) CoLight [22]: A MARL model for TSC using multi-head GAT to extract spatial features from the geographic neighbors.

i) STGAT [23]: A MARL model for TSC using a spatio-temporal graph attention network to extract spatio-temporal features from the geographic neighbors.

j) RegionSTLight (proposed): A MARL model for TSC based on regional multi-agent Q-learning which uses dynamic zoning and spatio-temporal fusion feature extraction.

k) NoMixed (spatial domain attention mechanism only): A MARL model for TSC which uses the spatial domain attention mechanism to extract spatial features from the geographic neighbors.

l) Single (mixed domain attention mechanism only): A MARL model for TSC which uses the mixed domain attention mechanism to extract spatial features from the geographic neighbors.

m) Region (single with dynamic zoning only): A MARL model for TSC which uses dynamic zoning and the mixed domain attention mechanism to extract spatial features from the regional neighbors.

n) ST-single (single with temporal feature extraction only): A MARL model for TSC which uses the spatio-temporal fusion feature extraction network to extract spatio-temporal fused features from the geographic neighbors.

Note that for fair comparison all the methods except FixTime and MaxPressure adopt the same hyperparameters listed in Table I and all the MARL methods except IRL, MADDPG, MA2C and Coder adopt parameter sharing.

TABLE I
THE HYPERPARAMETERS USED BY DIFFERENT MARL MODELS

Hyperparameter	Value	
	6 × 6 and Jinan	32 × 32
Number of layers in Q-value prediction	1	1
The dimension of hidden vector	32	32
Batch size	32	3
ϵ_{init}	0.8	0.8
Decay factor of ϵ	0.95	0.95
$\epsilon_{\min}$	0.2	0.2
Learning rate	0.001	0.001
The update frequency of the target network K	5	5
Discount factor γ	0.8	0.8
Reward normal factor	250	250
The size of replay buffer	5000	160

2) Metrics: The objective of TSC is to minimize the average travel time of all vehicles entering the traffic system. Therefore we use the average travel time of vehicles and the average queue length of intersections (the sum of queue lengths on all input lanes at each control step) to reflect the efficiency of entire traffic system.

C. Experimental Settings and Real-Time Performance of Dynamic Zoning

1) Experimental Settings: Several different random seeds are used in the experiments of all the scenarios and some train tricks are used to accelerate the learning process of all compared models including taking the entire traffic network as a sample, separating sample generation and network updates, and utilizing early stopping for updates.

2) Real-Time Performance of Dynamic Zoning: Time measurement experiments are conducted in three scenarios to illustrate the real-time performance of dynamic zoning and the results are listed in Table II.

TABLE II
THE REAL-TIME PERFORMANCE OF DYNAMIC ZONING IN THREE SCENARIOS

	Scenario		
	6 × 6	Jinan	32 × 32
Average time required (ms)	0.423	0.143	120.820

It is shown that the average time required by dynamic zoning in the 6 × 6 synthetic scenario and Jinan real-world scenario (3 × 4) is short enough to ignore. While in the 32 × 32 city-level scenario it is close to the practical transition delay [39] which is acceptable.

D. Case Study in The Real-World Scenario

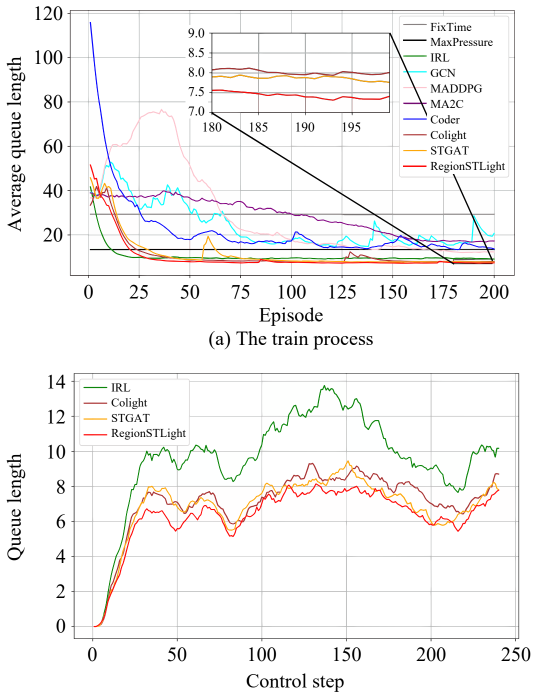
1) Real-Time Performance of Dynamic Zoning: As shown in Fig. 6(a), due to the lightweight structure of the network and the simple design of RL, IRL can converge quickly and finally obtain good performance, while MADDPG and MA2C converge slowly due to the complex design of RL. As for the graph network, GCN can converge faster than those MARL methods but loses some stability. Colight can converge more quickly and obtain better performance due to the GAT, which can reflect the dynamic impact of neighbors. Furthermore, STGAT benefits from the spatio-temporal features and obtains better performance than Colight while the network structure of STGAT is more complex, which makes the learning process slow and hard to learn compared to the proposed RegionSTLight. It is obvious that RegionSTLight has faster convergence and better asymptotic performance in real-world scenario, thus reflecting its effectiveness in practice.

It is noted that the Coder can converge more quickly than MADDPG, MA2C and GCN due to the division of the entire traffic network, while it cannot obtain better asymptotic performance since the global function is difficult to learn.

Moreover, in the same real-world scenario, Fig. 6(b) illustrates that the proposed RegionSTLight achieves consistent performance improvements over other models during the test process for the reason that it can find better cooperation between intersections for real-time dynamic traffic flow.

2) Study of The RegionSTLight: To investigate the impact of the division threshold and the size of region on model performance, we test RegionSTLight in the Jinan scenario. Note that the average queue length is obtained among the last ten episodes and similar results are obtained in other scenarios but not shown due to page limitation.

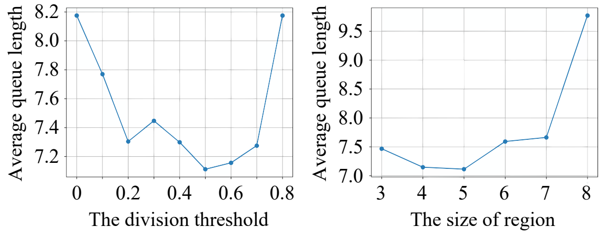
As shown in Fig. 7(a), the best performance is achieved when the division threshold is set to 0.5, and when the threshold increases to 0.8 or decreases to 0, the performance of RegionSTLight is the same as that of ST-single because no region is formed under a high threshold and all intersections forming one large region under a low threshold will be viewed as independent regions.



(a) The train process

(b) The test process

Fig. 6. Comparison of different MARL models in Jinan scenario.



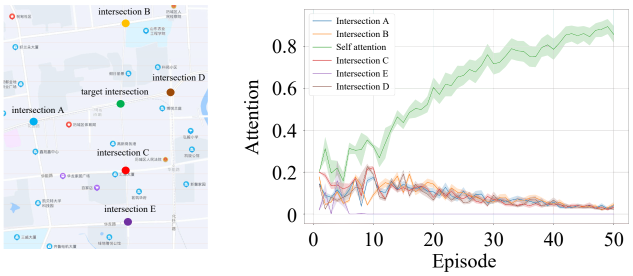
(a) Impact of the division threshold (b) Impact of the size of region

Fig. 7. Performance of RegionSTLight with different division thresholds or different size of region in the Jinan scenario.

Fig. 7(b) illustrates that the best performance is achieved when the size of region is set to 5 which reflects that the intersections in a much larger region are not closely connected to each other. Therefore, considering the influence inside a region containing no more than 5 intersections appears sufficient to guarantee performance for traffic signal control.

3) Attention Study: As is shown in Fig. 8, the target intersection in the Jinan scenario has four geographic neighbor intersections including A, B, C, D, while at the beginning of training, the target intersection is strongly connected to intersection C and intersection E which form a region. Therefore the attention that the target intersection pays to intersection E is large at the beginning of training but equals zero at the rest of training because this region can be avoided under the learned control strategy. It is shown that RegionSTLight can accelerate learning since it dynamically divides the traffic network into several strongly-coupled regions inside which the cooperation is easier to learn.

Besides, since the traffic flow input from each geographic



(a) A typical intersection and its neighbors in Jinan scenario

(b) The attentions learned by RegionSTLight when training

Fig. 8. Attention study of the proposed RegionSTLight.

neighbor intersection is similarly small at the end of training under the learned control strategy, and the attention to four geographic neighbor intersections are almost equal with the self attention largest at the end. It illustrates that the local state of the target intersection is the most important for its signal control in this scenario.

E. Case Study in the Synthetic Scenarios

1) Ablation Experiment in the 6 × 6 Synthetic Scenario: As is shown in Fig. 9, there are six methods to compare including FixTime as a convergence basis. Firstly, Single which uses the mixed domain attention mechanism rather than spatial domain attention mechanism can converge faster than NoMixed. By adding dynamic zoning, Region can further speed up convergence. This is because intersections inside each region are strongly connected which makes the cooperation between them easier to learn. On the other hand, since the spatio-temporal fusion features are useful to learning better cooperation within the region, ST-single can improve asymptotic performance but loses a certain learning speed due to the added parameters. Finally RegionSTLight can achieve the best asymptotic performance with comparable convergence speed which is consistent with the time complexity analyzed in Section III-E. Therefore, the effectiveness of each component in RegionSTLight is verified.

The final performance of these methods are listed in Table III. It is shown that ST-single can evidently improve traffic efficiency by achieving better cooperation between intersections with the proposed spatio-temporal fusion feature extraction network. Moreover, RegionSTLight can further improve the traffic efficiency by dynamically dividing the traffic network into several strong-coupled regions to explore even better cooperation.

2) Scalability to City-Level Traffic: As is shown in Fig. 10, the proposed RegionSTLight can successfully be applied to the city-level scenario with faster convergence speed and better asymptotic performance compared to conventional methods and other state-of-the-art MARL models in TSC. Here the shadow of each curve indicates the boundary of obtained reward. Therefore, it is obvious that the training process of RegionSTLight can be more robust compared to other models. It is noted that the STGAT added with the proposed dynamic zoning method (RegionSTGAT) can obviously accelerate the learning process and improve the stability as well. Traditional MARL models including IRL, MADDPG and MA2C can not be scaled up to the city-level scenario due to model complexity and the increased number of agents.

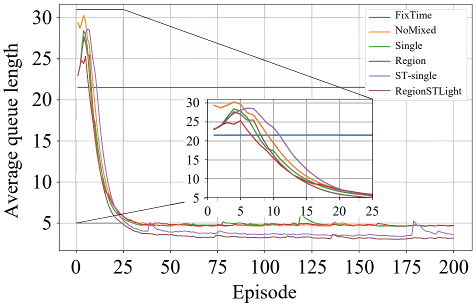


Fig. 9. The train process of different MARL models in the 6 × 6 synthetic scenario.

TABLE III
THE FINAL PERFORMANCE OF SIX METHODS IN THE 6 × 6 SYNTHETIC SCENARIO

	FixTime	NoMixed	Single	Region	ST-single	Region-STLight
Average queue length	21.52	3.87	3.89	3.72	2.32	**2.22**
Average travel time (s)	785.07	265.87	266.74	268.64	237.05	**234.38**

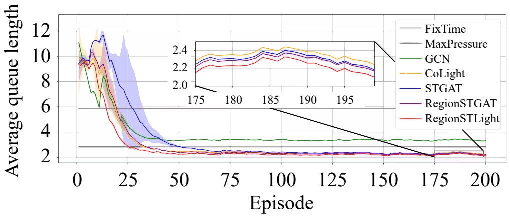


Fig. 10. The train process of different MARL models in 32 × 32 scenario.

The final performance of these methods are listed in Table IV which also illustrates the scalability and effectiveness of RegionSTLight.

Then, Colight, STGAT and RegionSTLight are directly applied to the 16 × 64 sythetic scenario to test their adaptability and the results are listed in Table V. The results show that the proposed RegionSTLight can also achieve the best performance in a different road network.

V. CONCLUSION

In this paper, a novel regional multi-agent cooperative reinforcement learning algorithm named as RegionSTLight is proposed. Firstly, the regional multi-agent Q-learning framework for TSC is derived theoretically which can equivalently decompose the global Q value of traffic system into the local values of several regions to make it scalable to city-level traffic system. Then, based on the proposed framework and the idea of human-machine cooperation, a dynamic zoning method and a more lightweight spatio-temporal fusion feature extraction network are designed to learn better cooperation between intersections in large-scale traffic system. The effective traffic signal control for multiple intersections in city-level traffic system is studied in this paper and determining

TABLE IV
THE FINAL PERFORMANCE OF METHODS IN THE 32 × 32 SYNTHETIC SCENARIO

Model	FixTime	MaxPressure	GCN	CoLight	STGAT	Region STGAT	Region STLight
Average queue length	5.87	2.82	3.39	1.83	1.61	1.57	**1.02**
Average travel time (s)	374.01	247.10	297.94	249.83	247.27	245.05	**237.47**

TABLE V
THE PERFORMANCE OF FOUR METHODS IN THE 16 × 64 SYNTHETIC SCENARIO

Model	Fixtime	CoLight	STGAT	RegionSTLight
Average queue length	8.27	5.40	4.48	**4.34**
Average travel time (s)	737.02	751.00	523.75	**516.68**

how to design the traffic signal control model for heterogeneous intersections in city-level traffic system is one of our research interests in the future.

REFERENCES

[1] K. N. Qureshi and A. H. Abdullah, “A survey on intelligent transportation systems,” *Middle-East J. Scientific Research*, vol. 15, no. 5, pp. 629–642, 2013.

[2] H. Wei, G. Zheng, V. Gayah, and Z. Li, “A survey on traffic signal control methods,” arXiv preprint arXiv: 1904.08117, 2019.

[3] A. J. Miller, “Settings for fixed-cycle traffic signals,” *J. Operational Research Society*, vol. 14, no. 4, pp. 373–386, 1963.

[4] A. Salkham, R. Cunningham, A. Garg, and V. Cahill, “A collaborative reinforcement learning approach to urban traffic control optimization,” in *Proc. IEEE/WIC/ACM Int. Conf. Web Intelligence and Intelligent Agent Tech.*, 2008, vol. 2, pp. 560–566.

[5] G. F. Newell, “Approximation methods for queues with application to the fixed-cycle traffic light,” *Siam Review*, vol. 7, no. 2, pp. 223–240, 1965.

[6] P. Varaiya, “The max-pressure controller for arbitrary networks of signalized intersections,” in *Advances in Dynamic Network Modeling in Complex Transportation Systems*. New York, USA: Springer, 2013, pp. 27–66.

[7] X. Zang, H. Yao, G. Zheng, N. Xu, K. Xu, and Z. Li, “MetaLight: Value-based meta-reinforcement learning for traffic signal control,” in *Proc. AAAI Conf. Artificial Intelligence*, 2020, vol. 34, no. 1, pp. 1153–1160.

[8] B. Abdulhai, R. Pringle, and G. J. Karakoulas, “Reinforcement learning for true adaptive traffic signal control,” *J. Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.

[9] X. Liang, X. Du, G. Wang, and Z. Han, “A deep reinforcement learning network for traffic light cycle control,” *IEEE Trans. Vehicular Technology*, vol. 68, no. 2, pp. 1243–1253, 2019.

[10] L. Li, Y. Lv, and F.-Y. Wang, “Traffic signal timing via deep reinforcement learning,” *IEEE/CAA J. Autom. Sinica*, vol. 3, no. 3, pp. 247–254, 2016.

[11] J. Wu and Y. Lou, “Efficient centralized traffic grid signal control based on meta-reinforcement learning,” *IEEE/CAA J. Autom. Sinica*, 2023. DOI: 10.1109/JAS.2023.123270

[12] L. Prashanth and S. Bhatnagar, “Reinforcement learning with function approximation for traffic signal control,” *IEEE Trans. Intelligent Transportation Systems*, vol. 12, no. 2, pp. 412–421, 2010.

[13] L. N. Alegre, T. Ziemke, and A. L. Bazzan, “Using reinforcement learning to control traffic signals in a real-world scenario: An approach based on linear function approximation,” *IEEE Trans. Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9126–9135, 2021.

[14] Y. Liu, L. Liu, and W.-P. Chen, “Intelligent traffic light control using distributed multi-agent Q learning,” in *Proc. IEEE 20th Int. Conf. Intelligent Transportation Systems*, 2017, pp. 1–8.

[15] T. Chu, J. Wang, L. Codecà, and Z. Li, “Multi-agent deep reinforcement learning for large-scale traffic signal control,” *IEEE Trans. Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1086–1095, 2019.

[16] X. Wang, L. Ke, Z. Qiao, and X. Chai, “Large-scale traffic signal control using a novel multiagent reinforcement learning,” *IEEE Trans. Cybern.*, vol. 51, no. 1, pp. 174–187, 2020.

[17] Z. Li, H. Yu, G. Zhang, S. Dong, and C.-Z. Xu, “Network-wide traffic signal control optimization using a multi-agent deep reinforcement learning,” *Transportation Research Part C: Emerging Technologies*, vol. 125, p. 103059, 2021.

[18] T. Chu, S. Qu, and J. Wang, “Large-scale traffic grid signal control with regional reinforcement learning,” in *Proc. American Control Conf.*, 2016, pp. 815–820.

[19] T. Tan, F. Bao, Y. Deng, A. Jin, Q. Dai, and J. Wang, “Cooperative deep reinforcement learning for large-scale traffic grid signal control,” *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2687–2700, 2019.

[20] S. Jiang, Y. Huang, M. Jafari, and M. Jalayer, “A distributed multi-agent reinforcement learning with graph decomposition approach for large-scale adaptive traffic signal control,” *IEEE Trans. Intelligent Transportation Systems*, vol. 23, no. 9, pp. 14689–14701, 2023.

[21] L. Yan, L. Zhu, K. Song, Z. Yuan, Y. Yan, Y. Tang, and C. Peng, “Graph cooperation deep reinforcement learning for ecological urban traffic signal control,” *Applied Intelligence*, vol. 53, no. 6, pp. 6248–6265, 2023.

[22] H. Wei, N. Xu, H. Zhang, G. Zheng, X. Zang, C. Chen, W. Zhang, Y. Zhu, K. Xu, and Z. Li, “Colight: Learning network-level cooperation for traffic signal control,” in *Proc. 28th ACM Int. Conf. Information and Knowledge Management*, 2019, pp. 1913–1922.

[23] L. Wu, M. Wang, D. Wu, and J. Wu, “DynSTGAT: DynAMIC spatial-temporal graph attention network for traffic signal control,” in *Proc. 30th ACM Int. Conf. Inform. & Knowledge Management*, 2021, pp. 2150–2159.

[24] H. Wei, C. Chen, G. Zheng, K. Wu, V. Gayah, K. Xu, and Z. Li, “PressLight: Learning max pressure control to coordinate traffic signals in arterial network,” in *Proc. 25th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining*, 2019, pp. 1290–1298.

[25] S. Guicheng and W. Yang, “Review on DEC-Pomdp model for MARL algorithms,” in *Proc. Smart Communi., Intelligent Algorithms and Interactive Methods; 4th Int. Conf. Wireless Communi. and Appli.*, 2022, pp. 29–35.

[26] M. L. Littman, “Markov games as a framework for multi-agent reinforcement learning,” in *Machine Learning Proceedings 1994*. New Brunswick, USA: Elsevier, 1994, pp. 157–163.

[27] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. London, UK: MIT Press, 2018.

[28] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, “Multi-agent actor-critic for mixed cooperative-competitive environments,” in *Proc. 31st Int. Conf. Neural Inform. Proc. Syst.*, 2017, vol. 30, pp. 6382–6393.

[29] P. Sunehag, G. Lever, A. Gruslys, *et al.*, “Value-decomposition networks for cooperative multi-agent learning,” arXiv preprint arXiv: 1706.05296, 2017.

[30] T. Rashid, M. Samvelyan, C. Schroeder, G. Farquhar, J. Foerster, and S. Whiteson, “Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning,” in *Proc. Int. Conf. Machine Learning*, 2018, pp. 4295–4304.

[31] C. Guestrin, D. Koller, and R. Parr, “Multiagent planning with factored MDPs,” in *Proc. 14th Int. Conf. Neural Inform. Processing Syst.: Natural and Synthetic*, 2001 vol. 14, pp. 1523–1530.

[32] G. Tesauro, “Extending Q-learning to general adaptive multi-agent

systems," in *Proc. 16th Int. Conf. Neural Inform. Proc. Syst.*, 2003 vol. 16, pp. 871–878.

[33] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *Proc. 10th Int. Conf. Machine Learning*, 1993, pp. 330–337.

[34] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *Stat*, vol. 150, p. 20, 2017.

[35] Q. Wang, B. Wu, P. Zhu, P. Li, and Q. Hu, "ECA-Net: Efficient channel attention for deep convolutional neural networks," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2020.

[36] H. Zhang, S. Feng, C. Liu, Y. Ding, Y. Zhu, Z. Zhou, W. Zhang, Y. Yu, H. Jin, and Z. Li, "CityFlow: A multi-agent reinforcement learning environment for large scale city traffic scenario," in *Proc. World Wide Web Conf.*, 2019, pp. 3620–3624.

[37] P. Varaiya, *The Max-Pressure Controller for Arbitrary Networks of Signalized Intersections*. New York, USA: Springer, 2013.

[38] T. Nishi, K. Otaki, K. Hayakawa, and T. Yoshimura, "Traffic signal control based on reinforcement learning with graph convolutional neural nets," in *Proc. 21st IEEE Int. Conf. Intelligent Transportation Systems*, 2018.

[39] P. Zhou, X. Chen, Z. Liu, T. Braud, P. Hui, and J. Kangasharju, "DRLE: Decentralized reinforcement learning at the edge for traffic light control in the IOV," *IEEE Trans. Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2262–2273, 2020.

Yisha Li received the B.S. degree in automation from Jiangsu University in 2021. She is a master student in electronic information at Southeast University. Her research interests include multiagent systems, reinforcement learning and intelligent transportation systems.

Ya Zhang (Senior Member, IEEE) received the B.S. degree in applied mathematics from China University of Mining and Technology in 2004, and the Ph.D. degree in control engineering from Southeast University in 2010. Since 2010, she has been with Southeast University, where she is currently a Professor with the School of Automation. Her research interests include multiagent systems, reinforcement learning, and network security.

Xinde Li (Senior Member, IEEE) received the Ph.D. degree in control theory and control engineering from the Department of Control Science and Engineering, Huazhong University of Science and Technology (HUST) in 2007. Afterward, he joined the School of Automation, Southeast University where he is currently a Professor and the Ph.D. Supervisor. His research interests include information fusion, object recognition, computer vision, and intelligent robot.

Changyin Sun (Senior Member, IEEE) received the B.S. degree in applied mathematics from the College of Mathematics, Sichuan University in 1996, and the M.S. and Ph.D. degrees in electrical engineering from Southeast University, in 2001 and 2004, respectively. He is currently a Professor with the School of Automation, Southeast University. His current research interests include intelligent control, flight control, pattern recognition, and optimal theory. He is an Associate Editor of the *IEEE Transactions on Neural Networks and Learning Systems*, the *IEEE Neural Processing Letters*, and the *IEEE/CAA Journal of Automatica Sinica*.