A Collaborative Communication-Qmix Approach for Large-scale Networked Traffic Signal Control

Xiaoyu Chen, Gang Xiong, Yisheng Lv[†], Yuanyuan Chen, Bing Song, and Fei-Yue Wang

Abstract— **Networked Traffic Signal Control (NTSC) has become an essential component in Intelligent Transportation Systems (ITS). To satisfy both scale and coordination challenging requirements for large-scale networked traffic signal control, this paper proposes a Communication-Qmix (CQmix) approach based on Qmix and Long Short-Term Memory (LSTM) communication module correspondingly. Firstly, we apply Qmix as the foundation for balancing large-scale and effective optimization, benefiting from its centralized-training and decentralized-execution mechanism. Then a communication module based on LSTM is implemented for effective global coordination. Further, random origin-destination demands (ODs) about different maximum traffic flows and occurrence times are performed to adapt the actual traffic flow pattern in practice. We conduct experiments on both synthetic and complicated real Zhongguancun road networks, and the proposed CQmix demonstrates its superiority over the baseline methods.**

I. Introduction

As the consequence of urbanization, the increasing number of vehicles induces traffic congestion inevitably and terribly [1]. Improving infrastructure construction is an effective measure to enlarge traffic network capacity. However, only by combining it with the optimization of management can realize sustainable development of traffic system [2], [3]. One valid optimization strategy is networked traffic signal control (NTSC), assisting to improve traffic conditions and shorten travel time [4], [5].

When considering approaches for traffic signal control (TSC) under large-scale scenes, whether it satisfies both large-scale and coordination requirements is of great essence. On one hand, the large-scale increase of networked controlled subjects and control scope may cause the curse of dimensionality [6] due to unbelievable joint state-action space for algorithms which may fail to operate with the guarantee of control effectiveness. On the other hand, dealing selfishly with local intersections may cause global congestion. Thus, communication between multiple traffic signals is indispensable to ensure the optimization effect and stability for the overall traffic condition.

To date, the taxonomy of NTSC can be charactered by two categories. One is traditional methods, including fixed-time control [7], actuated methods [8], [9], etc.. These methods rely on traffic knowledge or specific assumptions, but cannot adjust the control strategy rapidly [10]. Another category of methods relies on AI techniques. Recently, deep reinforcement learning(DRL) algorithms are introduced into traffic signal control problems, where intelligent traffic signal agents can sense traffic states through real-time interaction with the traffic environment. The maximum cumulative reward in DRL forces agents to focus on long-term payoffs rather than immediate traffic reward [11]–[13].

For the scale requirement in NTSC, many scholars attempted to use DRL under the multi-agent system. However, existing work tends to focus on the decentralized-training and decentralized-execution mechanism, which means that each intersection is isolated whenever in offline training or online executing [14], causing the nonstationarity of the environment. MARL with the decentralized-training and centralized-execution mechanism is an improvement to maintain control effectiveness, such as Qmix algorithm [15]. Meanwhile, there is still less work following coordination in NTSC [16], [17], which is meaningful to ensure the control effect and stability for the overall traffic condition.

In view of the challenges, we propose a novel approach for NTSC called CQmix, and contributions are threefold:

- We model NTSC as a Decentralized Partial-observation MDP (Dec-POMDP) process, and utilize the Qmix algorithm to balance large-scale and effective optimization with a decentralized-training and centralized-execution mechanism.
- For coordination, we design an effective communication module based on Long Short-Term Memory (LSTM), which integrates historical observations and actions simultaneously. Meanwhile, messages are compressed into the same dimension.
- We design a dynamic generation method of the origin-destination (OD) set for simulating various traffic flows, and develop parallel-scene sampling and segmented training mechanisms. Experiment results in both the synthetic network and the real Zhongguancun road network verify the feasibility and effectiveness of the proposed CQmix compared with the baseline algorithms.

*This work was supported in part by National Natural Science Foundation of China under Grants 61773381, U1909204, U1811463&61872365; Chinese Guangdong's S&T project (2019B1515120030, 2020B0909050001); Dongguan's Innovation Talents Project (Gang Xiong).

Xiaoyu Chen, Yisheng Lv and Bing Song are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China. They are also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, 100049, China.

Gang Xiong is with The Beijing Engineering Research Center of Intelligent Systems and Technology, Institute of Automation, and also with The Guangdong Engineering Research Center of 3D Printing and Intelligent Manufacturing, The Cloud Computing Center, Chinese Academy of Sciences, China.

Yuanyuan Chen and Fei-Yue Wang are with the State Key Laboratory for Management and Control of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China.

Xiaoyu Chen and Gang Xiong are the co-first authors.

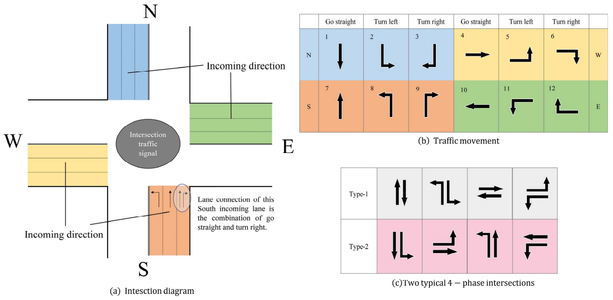
[†]Corresponding Author. E-mail: yisheng.lv@ia.ac.cn

The remainder of this paper is organized as follows. Dec-POMDP modeling for NTSC is formulated in section II. Section III introduces the proposed approach CQmix based on Qmix and LSTM communication module. In section IV, experiments under the synthetic road network and the real Zhongguancun road network are conducted. Section V concludes the paper.

II. Problem Definition and Dec-POMDP Modeling

A. Problem Definition

Consider a large-scale traffic road network $G(A, E)$, where A is the set of controlled intersections, and E is the set of roads. If there is a physical road connection directly between intersection i and intersection j, they are neighbors to each other. Note the neighborhood of i is E_i. Each traffic signal controller of an intersection is modeled as an agent, and networked traffic conditions as a global state. At each interval ΔT, every agent is response for four processes, i.e. perceiving the environment, obtaining local observation, making decision, and getting the timely feedback reward. A typical 4-leg intersection is shown in Fig. 1.



(a) Intesction diagram

(b) Traffic movement

(c)Two typical 4 − phase intersections

Fig. 1: A typical 4-leg intersection with traffic movements and traffic signal phases

B. Dec-POMDP Modeling

In this section, we formulate the NTSC as Dec-POMDP [18], which is

$$G =< A, S, \{U_i\}_{i\in A}, P, \{r_i\}_{i\in A}, \{O_i\}_{i\in A}, \gamma >, \qquad (1)$$

while the terms are detailed as follows.

1) $A = \{1, 2, ..., N\}$: The set of controlled intersections, and N is the total number.

2) O_i: Observation space of agent i. Local observation of agent i at time t taking no consideration on communication between agents is expressed as $\widetilde{o_{i,t}}$. Based on [16], we define

$$\widetilde{o_{i,t}} = \{wait(m)_t, flow(m)_t\}_{m\in L^i_{in}} \qquad (2)$$

where L^i_{in} denotes the set of incoming lanes of agent i, $wait(m)_t$ and $flow(m)_t$ refer to the maximum waiting time and the number of vehicle flow in the m_{th} incoming lane. And after adding communication message for coordination, local observation transfers to $o_{i,t}$, which will be introduced in detail in section III.

3) S: Global state space of whole agents. State $s_t \in S$ can be regarded as simple combination of $\widetilde{o_{i,t}}, i \in A$.

4) U_i: Action space of agent i. $u_{i,t}$ denotes the action agent i decides at time t, and joint action at time t is

$$u_t = \times_{i\in A} u_{i,t} \in \{U_i\}_{i\in N} \qquad (3)$$

Actions for traffic signal control are usually set as: whether or not to converse phase, setting the duration of the current phase, or selecting the phase selection at the next interval. In this paper, the last one is chosen for the action.

5) $r_{i,t}$: The local reward agent i obtained. The global reward of the whole network R_t is expressed as the sum of local rewards of all intersections at time t. Referring to settings in [17], here we define local reward $r_{i,t}$ as the weighted sum of the queue length and the maximum waiting time of all incoming lanes observed at time $t + \Delta T$.

$$r_{i,t} = \{queue(m)_{t+\Delta T}, wait(m)_{t+\Delta T}\}_{m\in L^i_{in}} \qquad (4)$$

6) γ: the discount factor. To ensure the long-term optimal traffic conditions, the discount factor prefers to be larger. We set it as 0.95.

7) P: State transition probability. Generally, the environment is opaque thus P is not clear. Consequently, the model-free approach CQmix is proposed in the subsequent section.

III. Methodology

This section presents the proposed CQmix algorithm for NTSC. Firstly, we introduce the overall architecture of the proposed method. Then, we will explain Qmix module and LSTM communication module, respectively. Further, generation method of ODs is briefly described by a parallel-scene sampling mechanism.

A. Overall Architecture of CQmix

As is shown in Fig.2, CQmix mainly has three networks: an agent network for each agent to take action independently using its local value function, a para-generator network for generating weights and biases to bridge each local value function and global value function, and a communication network for generating messages.

In centralized training, the three networks above need to make contributions to ensure global optimization. Firstly, each intersection agent i receives its own local observation $\widetilde{o_{i,t}}$, message from its neighbors generated in communication network and last action $u_{i,t-1}$ as the input of its agent network, and local value function $Q_i(\tau_i, u_{i,t})$ as output. $\tau_i = (u_{i,0}, o_{i,1}, ..., u_{i,t-1}, o_{i,t})$ denotes action-observation history of agent i, and $\tau = (\tau_1, \tau_2, ...\tau_N)$ expresses joint-history of all agents. Secondly, the para-generator network generates weights and biases to mix local value function nonlinearly to a global shared value function $Q_{total}(\tau, u_t)$ under joint-history τ and current joint-action u_t. The objective is to maximize the global value function,

$$\max \quad Q_{total} = g(\{Q_i\}_{i\in A}, \cdot) \qquad (5)$$

where $g(\cdot)$ denotes the relationship between global function and each local value functions. Loss function is the sum of TD error for b size of transitions, which is a traditional Value-based update,

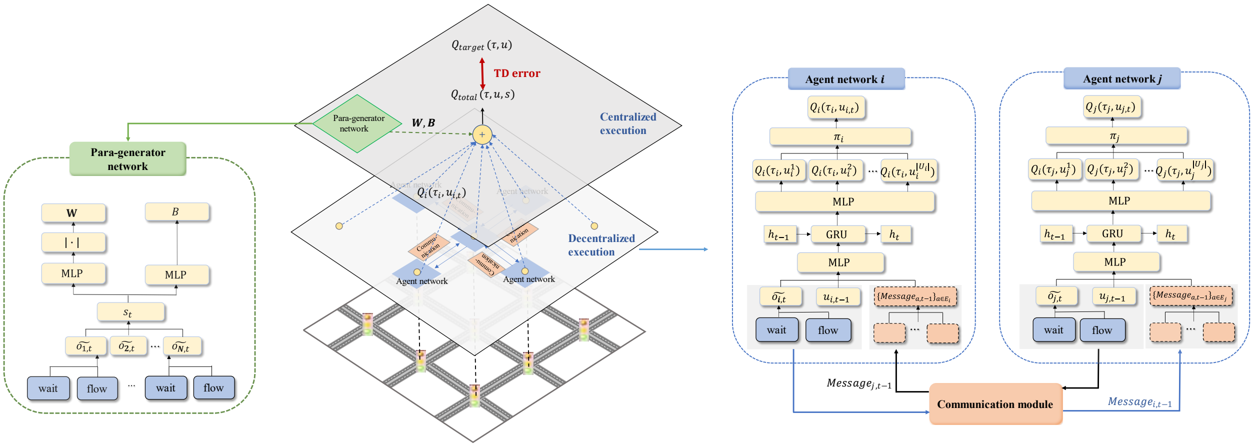


Fig. 2: The overall architecture of CQmix for NTSC

$$
\begin{cases}
L(\theta) = \sum_{k=1}^{b} \left[\left(y_{total}^{k} - Q_{total}(\tau, u, s; \theta) \right)^2 \right] \\
y_{total}^{k} = R + \gamma \max_{u'} Q_{total}(\tau', u', s'; \theta')
\end{cases}
\tag{6}
$$

While in the decentralized execution, only the first step is executed for the optimal strategy to ensure the feasibility of the large-scale network.

B. Qmix Network for Large-scale and Global Optimization

The requirements of both large scale and global optimization in NTSC are often in the clash, so we propose Qmix to balance, which satisfies the following monotonicity constraints and conclusion [15],

$$
\begin{aligned}
&if: \quad \frac{\partial Q_{total}}{\partial Q_i} = \frac{\partial(\sum_{i\in V} w_i \cdot Q_i + b_i)}{\partial Q_i} \geq 0, \forall i \in A \\
&\Rightarrow \arg\max_{u} Q_{total}(\tau, u) = \begin{pmatrix} arg\max_{u_1} Q_1(\tau_1, u_1) \\ \vdots \\ arg\max_{u_N} Q_N(\tau_N, u_N) \end{pmatrix}
\end{aligned}
\tag{7}
$$

Thus, the Qmix network contains two main components:

- Agent network: Taking observation $o_{i,t}$ and last action $u_{i,t}$ as input of DRQN (Deep Recurrent Q-Learning Network), and local value function $Q_i(\tau_i, u_{i,t})$ under action $u_{i,t}$ is generated. Theoretically, each agent owns an agent network separately supposing input or output dimensions is inconsistent.
- Para-generator network: Generate weights $\mathbf{W}$ and biases B of Mixing network based on the global network state s_t. Absolute function $|\cdot|$ in para-generator network guarantees nonnegativity of weights $\mathbf{W}$ and the monotonicity constraints.

C. LSTM Communication Module for Coordination

The unified communication module is based on LSTM to encode each agent's historical observations and actions.

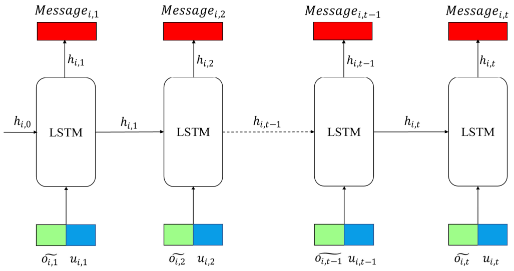


Fig. 3: LSTM Communication Module

- Message generation

As is shown in Fig.3, agent i can encode its history information and current observation through the communication module, and send message M_i to other intersections. Formally, the coding mode of the communication module is

$$
M_{i,t-1} = h_{i,t-1} = LSTM(h_{i,t-2}, \widetilde{o_{i,t-1}}, u_{i,t-1}; \phi) \tag{8}
$$

- Message receiving

Accordingly, intersection i obtains historical information from others to supplement self-observation. Simply, only messages from adjacent intersections are added based on the assumption that the impact from non-adjacent intersections can be negligible. That is,

$$
o_{i,t} = concat\left(\widetilde{o_{i,t}}, \{M_{a,t-1}\}_{a\in E_i}\right) \tag{9}
$$

The proposed communication module can not only realize coordination between other agents, but also share the following superiorities: (1) It realizes the fusion of historical information based on the LSTM network; (2) The message dimensions generated are consistent; (3)The delay of communication makes the gain of others' current information at time t impossible in reality, thus using $M_{i,t-1}$ to supplement current observation is reasonable.

D. Random Generation of ODs for Generalization

Previous studies often train and test the control methods under the same ODs, while these strategies learned under fixed ODs often fail for another. We design the double normal distribution to obey the flow pattern in reality, while randomly generate random ODs about different maximum traffic flows and occurrence times to enhance the generalization. Taking $T/2$ as the partition, X (ori, dest) demand pairs are generated individually, and the traffic flow from each OD $f_x(t)$ of demand x at time t is fitted by a normal distribution. Take it in $(0, T/2)$ as an example,

$$\begin{cases} f_x(t) = \max\left(\frac{A_x}{\sqrt{2\pi}\sigma_x} exp\left(-\frac{(t-\mu_x)^2}{2\sigma_x^2}, 0\right)\right) \\ 0 < \mu_x < T/2, A_x > 0, \sigma_x > 0 \\ x = 1, 2, .., X; t = 0, 300, ..., \lfloor T/300 \rfloor \end{cases} \tag{10}$$

where A_x, μ_x, σ_x determine the maximum traffic flow, the peak time, and the changing scope respectively. For randomness, it is reasonable to set these three parameters to meet Uniform Distribution in a particular range and randomly sampled in each training episode.

According to the Principle of large numbers, with training epochs increases, all ODs tend to be consistent. Therefore, multi-episodes parallel sampling for each training episode is recommended, and the trend of average cumulative reward in P epoch $AvgG_j(\pi_{\theta,\rho})$ indicates the gradual optimization. We present the whole process of CQmix in Algorithm 1.

Algorithm 1 Pseudo-Code of CQmix approach

```
Input: The observation of each agent
output: $\theta = [\theta_1, ..., \theta_N]$ for agent networks; $\rho$ for the
    communication network.
Initialize the parameters $\theta$ for agent networks; $\phi$ for para-
    generator network; $\rho$ for the communication network.
Initialize the replay buffer $B$ for storing $\tau$.
for each epoch do
    for each episode(multi-scenes parallel sampling) do
        Generate a random set of ODs based on (10)
        % Decentralized execution.
        Initial message $h_{i,0}, t = 0$ ($i \in A$).
        for $t = 1, 2, ..., T$ do
            for each agent $i \in A$ do
                Receive local observation $o_{i,t}$
                Obtain all optional actions' value $\mathbf{Q_i}$
                Choose action $u_{i,t} = \pi(\mathbf{Q_i})$($\epsilon$-greedy).
                Generate message $M_{i,t}$ and communicate.
            end for
            Store joint episode history $\tau$ in replay buffer.
        end for
        % Centralized training.
        Sample $b$ episode histories from replay buffer.
        for $t = 1, 2, ...T$ do
            Obtain all $Q_i(\tau_i, u_{i,t})$ and $Q_{total}$ at time $t$.
            Update the parameters $\theta, \rho, \phi$ with TD error.
        end for
    end for
end for
return The trained agent networks, and the communica-
    tion network
```

IV. Experiments

In this section, we introduce the specific experimental settings, and demonstrate results in two networks.

A. Experimental Settings

1) Experimental scenes: For convincing, we carry out experiments both on the synthetic network and the real Zhongguancun road network. The synthetic road network structure is a $5 * 5$ traffic grid, as is shown in Fig.4. The real network is extracted from the existing road network in Zhongguancun, Haidian, Beijing, China, including 27 intersections, as Fig.5 shows. The traffic simulation platform is Simulation of Urban Mobility (SUMO).

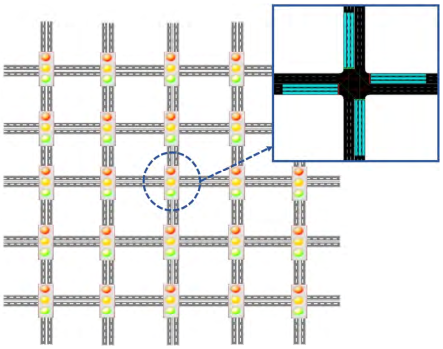


Fig. 4: $5 * 5$ traffic grid of the synthetic network

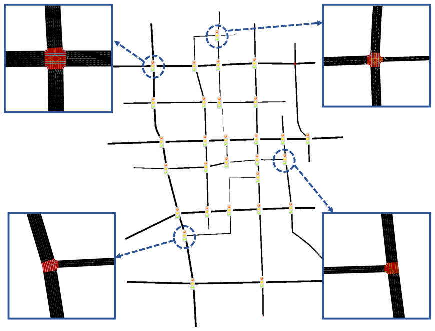


Fig. 5: Real road network in Zhongguancun

2) Baseline methods: To evaluate the effectiveness and efficiency of the proposed algorithm, we compare it with some baseline methods. Fixed-time control is a traditional control method, which is the most common method in practice yet for simple operation. IA2C and IQL are methods based on MARL with decentralized training and decentralized execution mechanism, and the former is policy-based, and the latter is value-based. MA2C is similar to IA2C except for adding some proposed stabilizing methods, like simple communication.

To intuitively reflect the Qmix network's effectiveness, the DNN network for IQL adopts the same structure as the agent network in CQmix. Meanwhile, to compare the difference in whether to use the proposed communication

TABLE I: Evaluation results in the simple synthetic network

Method	Average reward	Average queue(veh)	Average waiting time(s)	Average speed(m/s)
Fixed-time Control	-16.96	0.86	60.84	3.32
IQL	-1.76	0.11	2.10	7.02
IA2C	-15.29	0.73	28.04	2.88
MA2C	-8.81	0.57	13.52	3.71
Qmix-Base	-1.80	0.09	3.68	7.49
CQmix(ours)	**-1.54**	**0.07**	**2.47**	**7.85**

module, Qmix-Base ignores the information generation and interaction module.

Set $T = 3600s$ in each episode, and decision interval ΔT is 5s. The behavioral strategy is $\epsilon - greedy$. And the segmented training mechanism splitting the history from 0-T as multiple records is used to accelerate training.

3) Evaluation Metrics in testing: Evaluation results focus on the congestion at each intersection and the driving efficiency of vehicles, thus including four primary metrics in NTSC: the average reward, the average queue length of each intersection, and the average waiting time and average speed of each vehicle in the whole network.

B. *Experimental Results of the Synthetic Network*

Fig.6 shows training curves of all DRL algorithms. The horizontal axis represents the number of training epochs, and the vertical axis represents the average total cumulative rewards of 10 episodes in an epoch with random ODs. It is demonstrated that these algorithms are all convergent while CQmix converges to the maximum rewards.

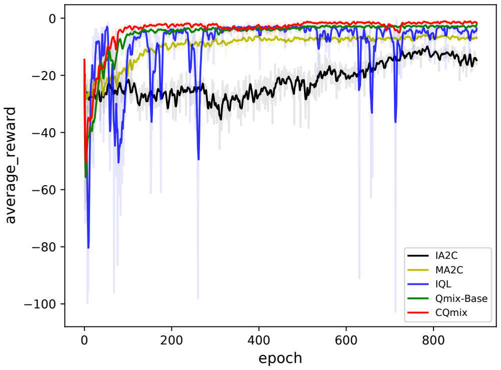


Fig. 6: Learning curves in the synthetic network

The randomly generated ODs are also applied to ensure the inconsistency between training and evaluation. Table I shows the results in evaluation episodes of the following metrics in NTSC. As can be seen from the evaluation table, CQmix has advantages over IA2C, MA2C, and the fixed-time control from all metrics' perspective. Might due to the simplicity of the synthetic scene, IQL can obtain a similar control effect as Qmix-Base. However, there is still a lag compared with CQmix. More obviously, compare Qmix-Base with CQmix, (1) from the perspective of convergence, though the final convergence rate is almost the same – achieving convergence around 120 epochs, the convergence of CQmix is more stable; (2) from the perspective of evaluation results, CQmix performs better on all metrics benefiting from its coordination between agents. Thus agent can achieve more additional information to assist decision-making.

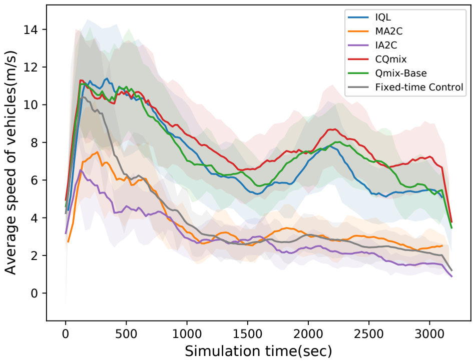


Fig. 7: Average speed in the synthetic network in evaluation

More intuitively, Fig.7. shows the average traveling speed of all vehicles in evaluation episodes, vehicles traveling in the scene controlled by CQmix have the most excellent speed in the most time of the episode, significantly shortening vehicles' traveling time and improving operation efficiency.

C. *Experimental Results of the Real Network*

Similarly, Fig.8 demonstrates that all algorithms converge after 800 epochs training except IA2C. While in terms of the convergence stability and convergence results, the proposed CQmix approach shows a significant improvement over others. For the former, CQmix tends to convergence at 300 epochs, sightly faster than others. Compared with Qmix-Base, CQmix shows a more stable convergence trend. For the convergence results, CQmix also performs best.

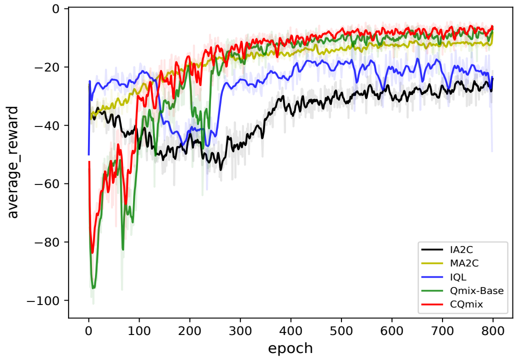


Fig. 8: Learning curves in the real Zhongguancun network

TABLE II: Evaluation results in the real Zhongguancun network

Method	Average reward	Average queue(veh)	Average waiting time(s)	Average speed(m/s)
Fixed-time Control	-18.38	1.28	40.28	6.67
IQL	-17.17	1.61	31.01	6.76
IA2C	-21.19	2.00	38.56	5.35
MA2C	-14.75	1.07	18.07	6.99
Qmix-Base	-10.57	0.56	20.71	9.30
CQmix(ours)	**-8.95**	**0.33**	**12.01**	**10.13**

Table II shows evaluation results in test episodes. CQmix demonstrates the biggest power than others at whatever metric in the more complicated scene, and Qmix-Base follows. Comparison between IQL and Qmix-Base shows the great power of Qmix network to balance large-scale requirements and global effectiveness, and comparison between CQmix and Qmix-Base shows the significance of communication module for coordination.

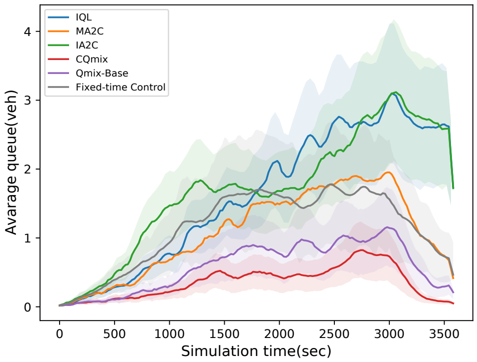


Fig. 9: Average queue length in the real Zhongguancun network in evaluation

Fig.9. shows the average queue length changing curves of all approaches in evaluation episodes. In all, the average queue length levels off to zero in CQmix and Qmix, showing the powerful effects in improving congestion at intersections and traveling efficiency of vehicles.

V. Conclusion

In this paper, we propose a Communication-Qmix approach for NTSC problem. The proposed method contributes to address two issues: the Qmix network of the concentrated-training and decentralized-execution mechanism for large-scale; the communication module for coordination. We evaluate the performance of the proposed method on both the simple synthetic scenario and real Zhongguancun road network, and compare it with the traditional method and other DRL methods. For generalization, we use a random generation strategy of ODs in experiments with the parallel-scene sampling mechanism. Experimental results show that the proposed method outperforms the baseline algorithms.

In the future, we will infuse prior knowledge, such as combining it with traditional transportation theory, to further accelerate the convergence speed and control effect of the proposed method.

References

[1] F. Y. Wang, "Parallel Control and Management for Intelligent Transportation Systems: Concepts, Architectures, and Applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.

[2] F. Z. C. C. X. A. G. Xiong, K. Wang and Z. Xie, "Parallel traffic management for the 2010 asian games," *IEEE Intelligent Systems*, vol. 25, no. 03, pp. 81–85, may 2010.

[3] G. Xiong, F. Zhu, X. Liu, X. Dong, and et al., "Cyber-physical-social system in intelligent transportation," *IEEE/CAA Journal of Automatica Sinica*, vol. 2, no. 3, pp. 320–333, 2015.

[4] L. Li, Y. Lv, and F. Y. Wang, "Traffic signal timing via deep reinforcement learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 3, no. 3, pp. 247–254, 2016.

[5] F. Zhu, Y. Lv, Y. Chen, X. Wang, G. Xiong, and F. Y. Wang, "Parallel transportation systems: Toward iot-enabled smart urban traffic control and management," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 10, pp. 4063–4071, 2020.

[6] P. Hernandez-Leal, B. Kartal, and M. E. Taylor, "A survey and critique of multiagent deep reinforcement learning," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 750–797, 2019.

[7] A. Muralidharan, R. Pedarsani, and P. Varaiya, "Analysis of fixed-time control," *Transportation Research Part B: Methodological*, vol. 73, pp. 81–90, 2015.

[8] X. Zheng, W. Recker, and L. Chu, "Optimization of control parameters for adaptive traffic-actuated signal control," *Journal of Intelligent Transportation Systems*, vol. 14, no. 2, pp. 95–108, 2010.

[9] I. Yun, "Evaluation of stochastic optimization methods of traffic signal control settings for coordinated actuated signal systems," Ph.D. dissertation, Ph. D. Thesis, University of Virginia, Charlottesville, VA, USA, 2005.

[10] K.-L. A. Yau, J. Qadir, H. L. Khoo, and M. H. Ling, "A survey on reinforcement learning models and algorithms for traffic signal control," *ACM Computing Surveys*, vol. 50, no. 3, pp. 1–38, 2017.

[11] H. Wei, G. Zheng, and et al., "Intellilight: A reinforcement learning approach for intelligent traffic light control," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, pp. 2496–2505.

[12] T. Nishi, K. Otaki, K. Hayakawa, and T. Yoshimura, "Traffic signal control based on reinforcement learning with graph convolutional neural nets," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 877–883.

[13] H. Wei, C. Chen, G. Zheng, K. Wu, and et al, "Presslight: Learning max pressure control to coordinate traffic signals in arterial network," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 1290–1298.

[14] 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 transactions on cybernetics*, vol. 50, no. 6, pp. 2687–2700, 2019.

[15] 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 *International Conference on Machine Learning*, 2018, pp. 4295–4304.

[16] T. Chu and J. Wang, "Traffic signal control by distributed reinforcement learning with min-sum communication," in *2017 American Control Conference (ACC)*, 2017, pp. 5095–5100.

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

[18] O. Aşık and H. L. Akın, "Solving multi-agent decision problems modeled as dec-pomdp: A robot soccer case study," in *Robot Soccer World Cup*. Springer, 2012, pp. 130–140.