AdaptLight: Toward Cross-Space-Time Collaboration for Adaptive Traffic Signal Control

Xintian Cai, Yilin Liu*, Quan Yuan, Guiyang Luo, and Jinglin Li

State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
{caixintian,liuyilin10,yuanquan,luoguiyang,jlli}@bupt.edu.cn

Abstract. Recent multi-agent deep reinforcement learning (MADRL) approaches have shown notable benefits in traffic signal control. However, the spatial-temporal coupling, hysteresis, and heterogeneity of collaborative agents are usually ignored. States and actions among multiple intersections induce complex coupling and hysteresis in both space and time dimensions, while the actions also present spatial-temporal heterogeneity due to fluctuated traffic. These characteristics impose a critical impact on the efficiency and flexibility of coordinated control. In this paper, we propose *AdaptLight*, an MADRL-based model to achieve cross-space-time collaboration. It captures the interactions among spatial-temporal traffic components and exploits action repetition to adaptively adjust decision granularity for heterogeneous traffic. For the spatial-temporal coupling and hysteresis issue, *AdaptLight* first establishes a feature extraction network based on spatial-temporal graph Transformer. To tackle the spatial-temporal action heterogeneity problem, an action-repetition-enabled MADRL module is designed, which can decide asynchronous-cooperative actions spanning multiple timesteps. Experiments present that *AdaptLight* shows competitive performance on different datasets.

Keywords: Multi-agent deep reinforcement learning · Traffic signal control · Graph transformer · Action repetition.

1 Introduction

Multi-agent deep reinforcement learning (MADRL) based multi-intersection traffic signal control (M-TSC) approaches have shown superior performance over traditional methods in improving traffic efficiency. However, there are still critical puzzles that remain unresolved. The first issue is **spatial-temporal coupling and hysteresis among states and actions**. *Coupling* is raised because the combinations of action-action, state-state, and action-state trajectories interact in different modes and result in various effects in time-space dimensions. This requires policies to identify distinct interaction modes and generate optimal coordinated behaviors accordingly. *Coupling* further prompts *hysteresis* caused by water-like traffic flows being blocked by intersections and gradually spreading

* Corresponding author.

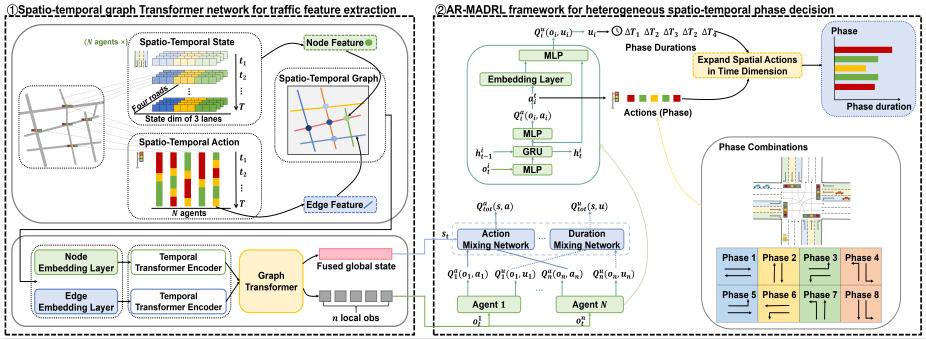


Fig. 1. The overall network structure of the proposed *AdaptLight*.

to nearby regions. The second issue is **spatial-temporal heterogeneity of actions**. Actions are *heterogeneous* in space-time dimensions because the policy decision intervals should vary with the different frequencies of traffic flow change. At the same timestep, different intersections observe traffic of different fluctuation frequencies, requiring distinct decision intervals. At different timesteps, the traffic at a single intersection also presents different fluctuation frequencies, which still requires dynamic decision intervals. Existing RL-based methods usually ignore this characteristic because Dec-POMDP induces synchronous decision intervals. Neglecting these problems can result in poor efficiency and adaptability of cooperative signal control. Therefore, we propose an MADRL-based model for cross-space-time M-TSC collaboration, *AdaptLight* as shown in Figure 1.

- We propose a spatial-temporal graph Transformer network to extract coupling and hysteresis features among multiple state-action combinations.
- We design an action repetition MADRL (AR-MADRL) framework to decide spatial-temporal heterogeneous policy decision intervals for drastic traffic.
- We evaluate *AdaptLight* on synthetic and real-world datasets. Our method outperforms both conventional and RL-based approaches, especially in dynamic and complicated environments.

2 Method

We consider M-TSC as a decentralized partially-observable SMDP problem, which is characterized by a tuple $G = \langle I, S, A, U, P, R, \Omega, O, n, \gamma \rangle$, where I is the finite set of n agents, $s \in S$ is the state, A is the finite action set. Each agent i only has access to a partial observation $o_i \in \Omega$ according to the observation function $O(s, i)$. At each step, each agent i selects an action $a_i \in A$, resulting in a joint action $\boldsymbol{a} \in A^n$. Conditioned on the observation o_i and the next action a_i, the agent also decides the corresponding duration $u_i^a \in U$ of the chosen action, where U is the finite discrete duration set. The joint action $\boldsymbol{a}$ and joint duration $\boldsymbol{u}$ transit the current state s to next state s' according to the transition function $P(s' \mid s, \boldsymbol{a}, \boldsymbol{u})$. Each agent shares a joint reward $\boldsymbol{r}$. The joint policy $\boldsymbol{\pi}$

induces a joint action-value function $\mathcal{Q}_{tot}^{\pi}(s, \boldsymbol{a})$ and a joint duration-value function $\mathcal{Q}_{tot}^{\pi}(s, \boldsymbol{u})$. The goal of the joint policy $\boldsymbol{\pi}$ is to maximize the cumulative joint rewards $\boldsymbol{r}$ of all intersections.

2.1 Spatial-Temporal Graph Transformer Network

Input and Embedding. For each time interval t, the node features $v_i^t \in \mathbb{R}^{d_n \times T}$ of node i consist of its observation history, where d_n is the dimension of concatenated observations, and T represents the window length. For edge from node i to node j, its edge features $e_{ij}^t \in \mathbb{R}^{d_e \times T}$ consist of four components: the action history of the entering intersection, the action history of the exiting intersection, the action duration transition of the entering intersection, and action duration transition of the exiting intersection. The four components are concatenated along the first dimension into a $d_e \times T$-dimensional tensor. The node features, edge features, and positional encodings are then passed through fully connected layers to get d_h-dimensional embedded node features $v_i'^t \in \mathbb{R}^{d_h}$, embedded edge features $e_{ij}'^t \in \mathbb{R}^{d_h}$, and embedded Laplacian positional encodings $\widehat{PE} \in \mathbb{R}^{d_h}$.

Temporal Transformer Layer. To handle temporal coupling and hysteresis, we use temporal Transformer to capture the time dependence of node-edge feature trajectories. Temporal Transformer learns the modes of traffic state history and intersection action history separately. For each node, the node features are permuted as a $T \times d_n$ tensor, which is then passed through a Transformer encoder layer, resulting in updated hidden features with temporal dependencies. We utilize an MLP layer as the aggregation function to output the final embedded features without the time dimension as $\hat{v}_i^t \in \mathbb{R}^{d_n}$. Similarly, the edge features are also updated with another temporal Transformer layer as $\hat{e}_{ij}^t \in \mathbb{R}^{d_e}$. Then, we add positional encodings to the input node embedding: $\widehat{h}_i^t = \widehat{v}_i^t + \widehat{PE}$. We omit the time symbol t in the following formula for the convenience of expression.

Spatial Transformer Layer. Based on the outputs of the temporal Transformer layer, we use spatial Transformer to combine temporal features with spatial features to get spatial-temporal encoded features. To solve spatial coupling and hysteresis, we specially design the attention encoding process in the spatial graph Transformer layer to extract the complex interaction information among combinations of decision intention and traffic states, which is defined as:

$$\omega_{ij}^{m,l} = Q_h^{m,l}\hat{h}_i^l K_h^{m,l}\hat{h}_j^l + Q_h^{m,l}\hat{h}_i^l K_e^{k,l}\hat{e}_{ji}^l + Q_e^{m,l}\hat{e}_{ij}^l K_h^{m,l}\hat{h}_j^l + Q_e^{m,l}\hat{e}_{ij}^l K_e^{m,l}\hat{e}_{ji}^l, \quad (1)$$

where $Q_h^{m,l}$, $Q_e^{m,l}$, $K_h^{m,l}$, $K_e^{m,l} \in \mathbb{R}^{d_m \times d_h}$ are learnable parameters, m is the number of multi-attention heads, $m = 1, \ldots, H$. This process is aimed to extract spatial interaction patterns among state-state, action-state, state-action, and action-action components in the system. The attention score $\omega_{ij}^{m,l}$ is then scaled and passed through a Softmax layer to get the final attention $\alpha_{ij}^{m,l}$. Further, node features are updated through spatial graph Transformer layers following the paradigm of vanilla graph Transformer network [1]. Edge features are also propagated to represent pairwise attention: $\widehat{e}_{ij}^{l+1} = f_{O,e}^l([\omega_{ij}^{m,l}]_{m=1}^H)$, where $f_{O,e}^l$ is a linear function that merges concatenated head.

After L sub-layers, node features obtained at the last layer $\widehat{h}_i^L$ are treated as the local observation o_i of the followed AR-MADRL network. For providing more comprehensive global state information, all node features are fused as the global state encoding s, which is also utilized in the AR-MADRL network: $s = \frac{\sum_{i \in I} \hat{h}_i^L}{n}$.

2.2 AR-MADRL Framework for Heterogeneous Decision

AR-MADRL has a hierarchical dual-objective architecture as illustrated in Figure 1 (right). The hierarchy consists of a higher-level policy to select asynchronous actions. To tackle temporal action heterogeneity, a lower-level policy is designed to choose timesteps for which the chosen actions will act.

For agent i at time t, given the observation o_i^t, action policy π_a outputs an action a_i^t based on the $\mathcal{Q}$-function: $\mathcal{Q}_i^a\left(o_i, a_i\right) := \mathbb{E}\left[r_t + \gamma \mathcal{Q}_i^a\left(o_i^{t+1}, a_i^{t+1}\right)\right]$. Conditioned on the observation o_i^t and chosen action a_i^t, duration policy π_u outputs a discrete duration u_i based on the n-step $\mathcal{Q}$-function: $\mathcal{Q}_i^u(o_i, u_i \mid a_i) := \mathbb{E}[\sum_{k=0}^{u_i-1} \gamma^k r_{t+k} + \gamma^{u_i} \mathcal{Q}_i^u(o_i^{t+u_i}, a_i^{t+u_i})]$, where $u_i \in \{1, \ldots, U\}$, U is the maximum duration steps. Then the chosen action will be executed for u_i steps.

To handle spatial action heterogeneity, we use multi-agent DQN to approximate both policies and coordinate agent behaviors. Policy π_a and π_u share a linear layer and a GRU to share observation context o between two policies, which is encoded as a vector δ_o and then mapped into action $a \in \mathbb{R}^{d_a}$ by action policy. Next, action a is encoded into action representation δ_a via a linear layer. Observation representation δ_o and action representation δ_a are concatenated as the input of the final linear layer, which is designed to approximate π_u and outputs the duration $u \in \mathbb{R}^{d_u}$, where we represent u as a d_u-dimension one-hot vector. To use global state features, the local $\mathcal{Q}$-values $\mathcal{Q}^a$ and $\mathcal{Q}^u$ are then mixed by mixing networks QMIX to estimate the global action-value $\mathcal{Q}_{\text{tot}}^a\left(s, \boldsymbol{a}\right)$ and duration-value $\mathcal{Q}_{\text{tot}}^u\left(s, \boldsymbol{u}\right)$, respectively.

The introduction of action repetition into multi-agent systems leads to asynchronous rewards since the action end time is not uniform. To solve this problem, we introduce Mac-JERTs [2] to build replay buffers. A joint reward is collected when any agent terminates an action, and agents share a joint cumulative reward $r^c = \sum_{t=t_a}^{t_{\text{end}}} r_t$, where t_a denotes the starting step of a joint action, and t_{end} refers to the timestep at which any agent ends a local action. *AdaptLight* is trained end-to-end to optimize the following objective function:

$$\begin{aligned} \mathcal{L}(\theta) = \mathbb{E}_D[(r^c + (\gamma \max_{\boldsymbol{a}'} \overline{\mathcal{Q}}_{\text{tot}}^a\left(s', \boldsymbol{a}'\right) - \mathcal{Q}_{\text{tot}}^a\left(s, \boldsymbol{a}\right)) \\ + (\gamma \max_{\boldsymbol{u}'} - \overline{\mathcal{Q}}_{\text{tot}}^u\left(s', \boldsymbol{u}'\right) - \mathcal{Q}_{\text{tot}}^a\left(s, \boldsymbol{u}\right)))], \end{aligned} \tag{2}$$

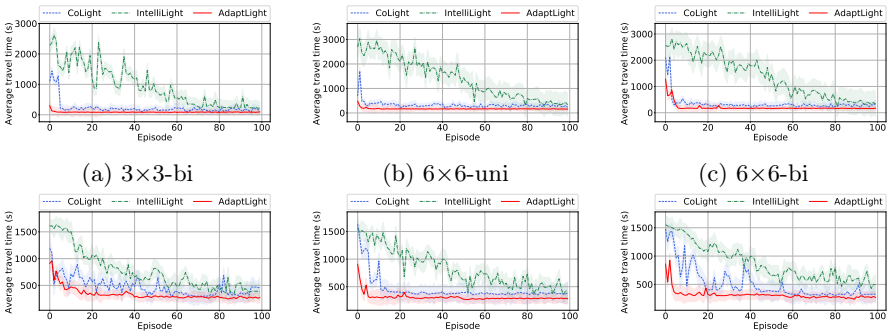
where $\overline{\mathcal{Q}}_{\text{tot}}^a$, $\overline{\mathcal{Q}}_{\text{tot}}^u$ are target networks, θ denotes the parameters of the networks.

3 Experiments

We use CityFlow as the simulation platform. At each decision time, an agent chooses an action from 8 signal phase combinations. For our method, in addition

Table 1. Performance of different methods on 6 road networks, and *action oscillations* of *AdaptLight* and CoLight (as the baseline method) during an episode.

	Model	3×3-Bi	6×6-Uni	6×6-Bi	Hangzhou	Jinan-1	Jinan-2
Travel Time	Fixedtime	105.49	210.94	210.93	718.89	882.11	814.68
	MaxPressure	101.46	186.56	195.49	416.36	337.17	356.95
	IntelliLight	154.77	395.34	316.88	414.97	501.73	560.19
	CoLight	90.42	181.92	182.69	366.53	349.19	368.16
	AdaptLight	**85.46**	**167.42**	**168.00**	**312.00**	**285.57**	**299.48**
Action Oscillations	CoLight	169	178	179	177	170	172
	AdaptLight	**105**	**155**	**158**	230	209	215



(a) 3×3-bi (b) 6×6-uni (c) 6×6-bi

(d) Hangzhou (e) Jinan-1 (f) Jinan-2

Fig. 2. Convergence curves of different methods on 6 road networks.

to selecting an action, an agent also selects a phase duration that lasts for 10, 20, 30, or 40 steps. For compared methods, the phase duration is averaged as a 20-step fixed duration. Following CoLight [3], we conduct experiments on 3 synthetic grid networks (3×3 bi-direction, 6×6 uni-direction, and 6×6 bi-direction) and 3 real-world networks (4×4 in Hangzhou, 3×4 in Jinan, and 3×4 in Jinan with more dynamic traffic flows and higher throughput). We compare our method with conventional TSC methods (Fixedtime, and Maxpressure) and state-of-the-art RL-based methods (IntelliLight and Colight).

As presented in Table 1, *AdaptLight* achieves the best performance under different road networks and traffic flows. The cross-space-time collaboration is more evident on real-world datasets since these datasets have more drastic traffic flows, more complex network structures, and thus more complicated spatial-temporal dependencies among multi-intersections. Our method presents better stability. Figure 2 illustrates the convergence curves over *AdaptLight* and other approaches. The convergence speed of *AdaptLight* outperforms all the compared models, which shows the training efficiency of our method. The improvement in convergence speed is attributed to the high parallel computing efficiency of an entire attention-based Transformer mechanism without recurrence and convo-

lutions. Meanwhile, learning to choose phase intervals as well as phase settings also improves the speed of targeting the optimal policy patterns.

In Table 1, we compare the number of action changes (the average number of switched phases for an intersection) required for one round of experiment over *AdaptLight* and CoLight. We discover that our method achieves the best performance with fewer actions in synthetic maps which have stable vehicle arrival rates, while our method requires more actions in real-world maps that have dynamic traffic. The experiments indicate that *AdaptLight* learns when it is necessary to act and decides adaptive optimal action durations for distinct flow status. This improvement can strike a balance between optimizing performance and decreasing action oscillations in TSC environments, enhancing driver experience and reducing potential safety threats.

We perform ablation experiments on the spatial-temporal graph Transformer network and AR-MADRL framework. The absence of both components leads to a loss in performance. We discover that an intersection fails to allocate proper attention scores without graph Transformer modules, which can identify spatial-temporal coupling and hysteresis features from state-action components.

4 Conclusion

In this paper, we have proposed a collaborative cross-space-time M-TSC method *AdaptLight* to handle spatial-temporal coupling, hysteresis, and heterogeneity issues. For the spatial-temporal coupling and hysteresis puzzle, we propose a spatial-temporal graph Transformer model. For the spatial-temporal action heterogeneity problem, we are the first to extend action repetition to MADRL to learn heterogeneous-asynchronous actions and decision intervals. Experiments show that our approach has strong efficiency and adaptability in various environments, especially for dynamic traffic. Experiments also give evidence that *AdaptLight* balances performance and action oscillations properly.

Acknowledgements This paper is supported in part by the National Key Research and Development Program of China under Grant 2022YFB4300402, the Natural Science Foundation of China under Grant 62272053, Grant 62102041, and in part by the Young Elite Scientists Sponsorship Program by China Association for Science and Technology (CAST) under Grant 2022QNRC001.

References

1. Dwivedi, V.P., Bresson, X.: A generalization of transformer networks to graphs. arXiv preprint arXiv:2012.09699 (2020)
2. Xiao, Y., Hoffman, J., Amato, C.: Macro-action-based deep multi-agent reinforcement learning. In: Conference on Robot Learning, pp. 1146–1161 (2020)
3. Wei, H., Xu, N., Zhang, H., Zheng, G., Li, Z.: Colight: Learning network-level cooperation for traffic signal control. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1913–1922 (2019)