OAM: An Option-Action Reinforcement Learning Framework for Universal Multi-Intersection Control

Enming Liang[1], Zicheng Su[2], Chilin Fang[3], Renxin Zhong[3]*

[1]School of Data Science, City University of Hong Kong
[2]Department of Advanced Design and Systems Engineering, City University of Hong Kong
[3]School of Intelligent Systems Engineering, Sun Yat-sen University
eliang4-c@my.cityu.edu.hk, zichengsu2-c@my.cityu.edu.hk, fangchlin3@mail2.sysu.edu.cn, zhrenxin@mail.sysu.edu.cn

Abstract

Efficient traffic signal control is an important means to alleviate urban traffic congestion. Reinforcement learning (RL) has shown great potentials in devising optimal signal plans that can adapt to dynamic traffic congestion. However, several challenges still need to be overcome. Firstly, a paradigm of state, action, and reward design is needed, especially for an optimality-guaranteed reward function. Secondly, the generalization of the RL algorithms is hindered by the varied topologies and physical properties of intersections. Lastly, enhancing the cooperation between intersections is needed for large network applications. To address these issues, the Option-Action RL framework for universal Multi-intersection control (OAM) is proposed. Based on the well-known cell transmission model, we first define a lane-cell-level state to better model the traffic flow propagation. Based on this physical queuing dynamics, we propose a regularized delay as the reward to facilitate temporal credit assignment while maintaining the equivalence with minimizing the average travel time. We then recapitulate the phase actions as the constrained combinations of lane options and design a universal neural network structure to realize model generalization to any intersection with any phase definition. The multiple-intersection cooperation is then rigorously discussed using the potential game theory.

We test the OAM algorithm under four networks with different settings, including a city-level scenario with 2,048 intersections using synthetic and real-world datasets. The results show that the OAM can outperform the state-of-the-art controllers in reducing the average travel time.

Introduction

With rapid urbanization and increased car ownership, traffic congestion has become one of the major issues for urban areas. Traffic control systems have shown effectiveness in improving the efficiency and robustness of road networks. Classical examples include SCATS, SCOOT, and the more recent max-pressure controller (Varaiya 2013). With the development of AI technology and the growing size of traffic data, learning-based control approaches have shown great potential in solving traffic signal control problems. In particular, reinforcement learning (RL) seems a promising solution to that in real-world scenarios (Wei et al. 2018, 2019b; Zheng et al. 2019; Chen et al. 2020; Oroojlooy et al. 2020). Although the RL methods have achieved significant improvements in intersection control, several critical issues still need to be addressed:

(1) A paradigmatic design of state, action, and reward function is missing. In the literature, these three components are generally designed manually based on experience, which would result in difficulties in generalization. Specifically, an optimality-guaranteed reward function that minimizes the average travel time is still missing.

(2) A universal framework for generalization is needed. The intersections in the real world vary in different physical capacities, topologies, and traffic flows. Besides, the parameter-sharing approach is more realistic in application owing to its data-efficient nature, comparing with training the agents for each intersection separately. To this end, the design of the RL algorithm is required to generalize to different real-life scenarios with one universal structure (Zheng et al. 2019; Oroojlooy et al. 2020).

(3) Collaboration among multiple intersections should be analyzed. Coordination between neighboring intersections is essential for efficient traffic management. The centralized approaches are computationally intractable for real-time decision making (Kuyer et al. 2008). Oppositely, the decentralized methods are more efficient by aggregating neighboring information and by executing individually (Chen et al. 2020; Wei et al. 2019b; Zhu et al. 2021). However, the mechanism of coordination in different conditions under the decentralized regime is still unclear.

To address the above issues, we propose the Option-Action RL framework for universal Multiple-intersection control (OAM). Firstly, we redesign the lane-cell-level state representation based on cell transmission model (CTM) (Daganzo 1994), which simulates the physical traffic-flow propagation. The state design can better balance the learning complexity and representation ability. We then reformulate the phase actions as the constrained combinations of lane options, where lane options are defined under different current phase as shown in Fig. 1. By disentangling the phase action into lane-level options, the action structure can generalize to any intersection topology with any phase definition. We also propose a decomposed delay with regularizer as the reward to facilitate temporal credit assignment and

*Correspondence to Renxin Zhong.

maintain the equivalence with optimizing the average travel time. Based on the decomposition scheme of delay, we rigorously discuss all possible conditions of coordination using the potential game. We then derive a decentralized Q function based on local information which considers the downstream traffic flow to promote coordination. A novel neural network structure is provided to represent the phase values and lane option values. It should be highlighted that the structure is invariant to different properties of intersections, and thus it can generalize to any intersection. Finally, we conduct extensive experiments to demonstrate the efficiency of our methods comparing to other state-of-the-art RL methods.

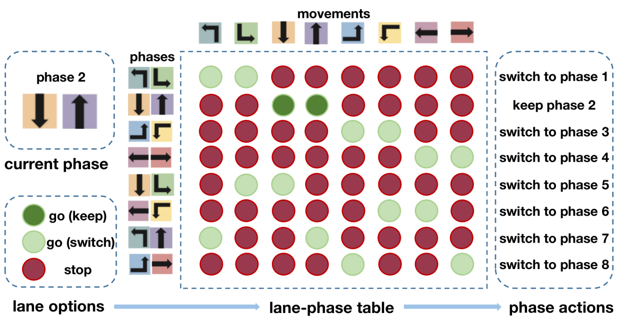


Figure 1: Option-action illustration

In summary, the contributions of this paper are as follows:

- By dividing lane features into cell level, the designed physic-informed state can represent the traffic flow propagations and facilitate generalization ability.
- We propose regularized delay as the reward to promote temporal credit assignment and prove its equivalence with minimizing the average travel time.
- The coordination mechanisms are explicitly discussed under potential game theory. In this way, the coordination is enhanced through state augmentation.
- The phase action is disentangled into lane options. By doing so, a universal structure is proposed to evaluate different phase values. To the best of our knowledge, it is the first universal structure generalizing to any intersection topology with any phase definition.

Related Work

There are different RL-based approaches dealing with intersection control and some of them try to tackle with those issues mentioned above.

Firstly, **the design of state, action, and reward**. The state design includes vehicle-specific features and lane-specific features. The aerial image of intersections or the occupation matrix captures all details of an intersection (Wei et al. 2018; Van der Pol and Oliehoek 2016). Other works select lane-level features (e.g., number of vehicles) in different lanes of an intersection as the state representation (Wei et al. 2019a; Chen et al. 2020). However, vehicle-level features are not data-efficient, while lane-level features omit the flow propagation of traffic and cannot model lanes with different lengths and speed limits.

The action of an intersection is to choose a certain phase, which is different combinations of non-conflict lane movements (Zheng et al. 2019). Intersections with different topologies have different available phases. To design a universal policy on different intersection topologies, FRAP (Zheng et al. 2019) uses pair-wise embedding of different phases based on the principle of phase competition. AttendLight (Oroojlooy et al. 2020) aggregates participating lane movements through the attention mechanism. However, both works only consider the participating lane movements for phase embedding, while other stopped lane movements are ignored. Besides, to model the next phase action under different current phases, they directly use the embedding of current active phase (e.g., one-hot encoding), which cannot generalize to intersections with different phase definitions.

As for the reward design, a wide range of reward functions (e.g., average speed, queue length, occupancy, etc.) are tested in (Egea et al. 2020), and they find that the average speed adjusted by demand performs best in the empirical study. Pressure is another widely-used reward function that evaluates the imbalance of traffic flow (Varaiya 2013). PressLight (Wei et al. 2019a) proves that the max-pressure agent can stabilize the queue length in the system. However, the equivalence of designed reward function with minimizing average travel time is not strictly proved.

Secondly, **the generalization of RL policies** deals with different intersection topologies and traffic flows. To design a universal structure on different intersection topologies, FRAP (Zheng et al. 2019) and AttendLight (Oroojlooy et al. 2020) use aggregation of lane movements to model different intersection topologies. To improve the robustness of RL policies under different traffic flows, the meta-learning approach is applied. MetaLight (Zang et al. 2020) trains the meta-learner on different traffic flows based on the gradient-based meta-learning approach. GeneraLight (Zhang et al. 2020) clusters different traffic flows and trains RL agents on flows within the same cluster, respectively. MetaVIM (Zhu et al. 2021) uses a latent variable that represents different traffic flows and takes it as input of the policy. Although the meta-learning approaches help accommodate different traffic flows, the lower-level representation structure of intersections needs further improvements.

Thirdly, **the collaboration of multiple intersections**. To obtain cooperative decisions of multiple intersections, centralized optimization approach (e.g., max-sum algorithm) is adopted in (Kuyer et al. 2008; Van der Pol and Oliehoek 2016). However, they require the maximization over a combinatorial joint action space and lack scalability in the large-scale road network. Decentralized approaches focus on aggregating adjacent intersections' information to learn cooperative policy. MPLight (Chen et al. 2020) adopts pressure-based state and reward to coordinate adjacent intersections. CoLight (Wei et al. 2019b) applies a graph attention network to aggregate neighboring information. HiLight (Xu et al. 2021) jointly optimizes the objective of neighboring intersections based on a hierarchical structure. However, to avoid the cost of frequent communication among agents, the circumstances under which the coordination between neighboring junctions is needed should be specified.

Preliminary

Single Intersection Modeling

We first consider a single four-legged intersection $i \in \mathbb{I}$ to illustrate basic definitions of the traffic control problem. An intersection is defined as a junction of several roads, where each road may have one or two directions and each direction includes several lanes.

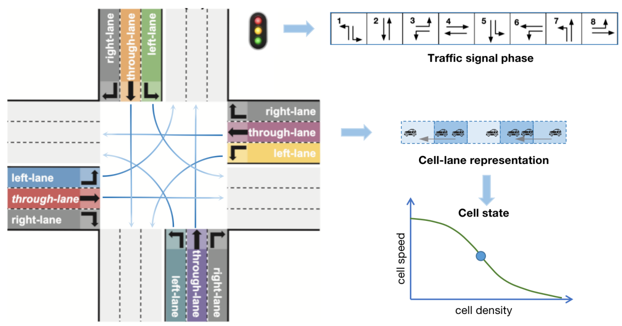


Figure 2: 4-legged intersection illustration (Zang et al. 2020)

- Lanes: we define the set of all lanes of the intersection i as $\mathbb{L}_i = \mathbb{L}_i^{\text{in}} \cup \mathbb{L}_i^{\text{out}}$, which includes incoming lanes $\mathbb{L}_i^{\text{in}}$ and outgoing lanes $\mathbb{L}_i^{\text{out}}$. Each lane $\{l_{ij}\}_{j\in\mathbb{L}_i}$ has different traffic characteristics (e.g., number of vehicles, average speed, and queue length) and physical characteristics (e.g., road length and speed limit). Those characteristics define the state of a lane.
- Cells: for each lane l_{ij}, cells C_{ij} are divided according to the speed limit as $\|l_{ijc}\| = v_{ij}^* \times \Delta t$ shown in Fig. 2, where the cell length $\|l_{ijc}\|$ denotes the maximum distance a vehicle can traverse within a single time step Δt with the maximum allowable speed v_{ij}^*. In other words, vehicles can at most travel across one cell within Δt. For each cell, according to the cell transmission model (Daganzo 1994; Su, Chow, and Zhong 2021), the cell state can be represented by the number of vehicles, the cell density[1], and the average speed.
- Movements: a traffic movement is defined as the traffic flow moving from one incoming lane $\{l_{ij}\}_{j\in\mathbb{L}_i^{\text{in}}}$ to another outgoing lane $\{l_{ik}\}_{k\in\mathbb{L}_i^{\text{out}}}$ (i.e., left turn, through, and right turn). The movements in each intersection are constrained by the traffic rules.
- Phases: different combinations of non-conflict traffic movements l_{ij} form the set of available phases $\{l_{ij}\}_{j\in\mathbb{L}_i^p}$, where $p \in \mathbb{P}_i$ indicates the phase of intersection i. At each time step t, the intersection will choose a phase and keep it for Δt (e.g., 10s). When switching to a new phase, there exists an all-red phase (e.g., 5s) to clear all vehicles within the intersection junction. Therefore, the phase duration for the switched phase is shorter (e.g., 5s) than the kept phase (e.g., 10s).

The main goal of intersection control is to minimize the average travel time of all vehicles when finishing their planning routes.

[1]the density is defined as the number of vehicles per unit length of the roadway

Method

In this section, we present our end-to-end RL framework for the intersection control problem. To formulate it under the RL context, the state, action, and reward of the intersection need to be designed.

Reinforcement Learning Design

State: considering the physical propagation of traffic flow in the incoming lanes, we divide the incoming lane into cells according to its speed limit. The state for each cell $s_{ijc} = \{m_{ijc}, k_{ijc}, v_{ijc}\}_{j\in\mathbb{L}_i^{\text{in}}, c\in C_{ij}}$ includes the number of vehicles m_{ijc}, the density k_{ijc} and the average speed v_{ijc} of vehicles within the cell. Different from using average features of the whole lane or using specific features of each vehicle, the cell-based modeling approach balances the representation ability and learning complexity. At time step t, the state of the intersection $s_{ti} = (\{s_{tijc}\}_{j\in\mathbb{L}^{\text{in}}, c\in C_{ij}}, a_{i,t-1})$ includes cell features and current phase information $a_{i,t-1}$.
Action: from the perspective of each incoming lane, its action has two options: $a_{ij} \in \{\mathbf{1}, \mathbf{0}\}$, where 1 denotes the lane movement is available and 0 otherwise. From the perspective of an intersection, the action a_i is an available phase $p \in \mathbb{P}_i$, which consists of options of all incoming lanes movements:

$$a_i = \{a_{ij,1}\}_{j\in\mathbb{L}_i^p} \cup \{a_{ij,0}\}_{j\in\mathbb{L}_i^{\text{in}}\setminus\mathbb{L}_i^p} \tag{1}$$

Pre-defined phases constrain the available combinations of lane options. For example, there are eight pre-defined phases in Figure 1, where each phase consists of two available movements and the other six stopped movements. Further, given different current phases, the effective green time is different for switching to a new phase or keeping the same phase. Existing studies (Wei et al. 2019a) deal with this issue by embedding the current phase information, which cannot generalize to intersections with different phase definitions. To solve it, we extend the definition of lane options as $\{\mathbf{1}_k, \mathbf{1}_s, \mathbf{0}\}$. As shown in Fig. 1, option $\mathbf{1}_k$ refers to that the lane movement keeps available as the current phase is kept and option $\mathbf{1}_s$ refers to that the lane movement will become available after switching to the new phase. The phase actions considering the current phase information are also extended as $\{a_{i,k}, a_{i,s}\}$, where the action of keeping the phase is $a_{i,k} = \{a_{ij,1_k}\}_{j\in\mathbb{L}_i^p} \cup \{a_{ij,0}\}_{j\in\mathbb{L}_i^{\text{in}}\setminus\mathbb{L}_i^p}$ and the action of switching to a new phase is similarly defined. With the extended definitions, the option-action modeling approach can represent any intersection topology with any phase definition.
Reward: the objective of intersection control is to minimize the average travel time of vehicles, which is equal to minimize the total delay: $\sum_n T_n - T_n^*$, where T_n is the actual travel time when vehicle n finishes its trip. T_n^* is the ideal

travel time, where vehicle n arrives its destination with the speed limits of each lane along its route without delay.

Proposition 1 *The total delay $\sum_n T_n - T_n^*$ of all vehicles is equal to the total delay of all incoming lanes. Considering the divided cells of each lane, the total delay is also equal to the total cell delay.*

$$\min \sum_n T_n - T_n^* \tag{2}$$

$$= \sum_t \sum_i \sum_j \sum_c \underbrace{d_{t,i,j,c}}_{\text{cell delay}} \tag{3}$$

Proof 1 *See Appendix A.1.*

Based on the decomposition form of total network delay in Eq. 3, we naturally derive the reward definition: $r_{ti} = \sum_j \sum_c -d_{t,i,j,c}$, which represents the total negative delay of intersection i within time interval $[t, t+\Delta t)$. However, from the perspective of credit assignment (CA) in RL algorithm (Sutton 1984), a good reward signal has to reflect the contributions of different actions. The CA problem exists in intersection control scenarios. For example, when vehicles travel with free-flow speed, their delay is zero. However, there are two cases for the zero-delay situation. The first case is that when one vehicle travels in the upstream of the incoming lane with free-flow speed, the phase action will not affect the reward instantaneously. The second case is that one vehicle crosses the junction of the intersection with free-flow speed. The zero delay of the second case is directly correlated with the chosen phase action.

Therefore, to realize better credit assignment for agents and accelerate learning process, we extend the definition of rewards in each lane as $r^*_{t,i,j} = \sum_c -d_{t,i,j,c} + \lambda m^{\text{out}}_{t,i,j}$, where $m^{\text{out}}_{t,i,j}$ is the number of outflowing vehicles of lane l_{ij} during time $[t, t+\Delta t)$ of intersection i and λ is the coefficient of regularizer $m^{\text{out}}_{t,i,j}$. The outflow term can directly reflect the instantaneous contribution of an action.

Proposition 2 *The regularizer $m^{out}_{t,i,j}$ does not affect the optimality of original objective function $\sum_n T_n - T_n^*$*

Proof 2 *See Appendix A.2.*

Multiple Intersections Cooperation

There are two directions for promoting multiple intersections coordination: explicit planning (e.g., max-sum algorithm (Kuyer et al. 2008; Van der Pol and Oliehoek 2016)) and learning-based algorithm. In comparison, explicit planning computes the optimal joint actions for all intersections directly, while the learning-based algorithm aggregates the neighboring information (Wei et al. 2019b) and optimizes the weighted neighboring rewards (Xu et al. 2021; Zhu et al. 2021). However, the explicit planning approach suffers from the computational burden, thus intractable in large-scale networks. The decentralized method with aggregation seems promising, but it still requires identifying the conditions that intersections need cooperation. To answer issues mentioned above, we introduce the state-based potential game (PG) (Marden 2012). Firstly, we model the multi-intersection control problem as a game $G = \{\mathbb{I}, \mathcal{S}, \mathcal{A}, \mathcal{R}\}$, where $\mathbb{I}$ is the set of agents (intersections), $\mathcal{S} = \{S_i\}_{i\in\mathbb{I}}$ is the joint state space of all agents and $s_i \in S_i$ is the state space for each agent. $\mathcal{A}$ and $\mathcal{R}$ are similarly defined as the joint action space and reward space.

Definition 1 *(State-based Potential Games):* A game $G = \{\mathbb{I}, \mathcal{S}, \mathcal{A}, \mathcal{R}\}$ is called an (exact) state-based potential game if there exists a measurable function $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ such that the following holds: $\forall (a_i, a_{-i}),\ (a'_i, a_{-i}) \in \mathcal{A}, \forall s \in \mathcal{S}$, and $\forall i \in \mathbb{I}$:

$$r_i\left(s, a_i, a_{-i}\right) - r_i\left(s, a'_i, a_{-i}\right) = \phi\left(s, a_i, a_{-i}\right) - \phi\left(s, a'_i, a_{-i}\right) \tag{4}$$

Condition (4) states that the change of the reward function r_i of an individual agent i equals the change in the global potential function ϕ over the joint actions. In other words, maximizing each reward function separately can achieve the objective of maximizing the global potential function ϕ in the potential game.

We set the total reward of all intersections within a single time step as $r = \sum_i r_i(s, a)$, where $s = \{s_i\}_{i\in\mathbb{I}}$ and $a = \{a_i\}_{i\in\mathbb{I}}$ are the joint states and actions, respectively. Following that, we define the potential function of all intersections as $\phi(s, a) = \sum_i r_i(s, a)$ and denote the individual reward function as $r_i(s_i, a_i)$. The necessary conditions of when the state-based potential game is formed are derived as follows:

Proposition 3 *The single step multiple intersection control forms a state-based potential game if $r_i(s, a_i, a_{-i}) = r_i(s_i, a_i)$.*

Proof 3 *See Appendix A.3.*

Proposition 4 *The condition $r_i(s, a_i, a_{-i}) = r_i(s_i, a_i)$ holds if there is no Queue spillback or Critical short road in the road network. The two terms are defined as:*

- *Queue spillback happens when a lane l_{ij} is occupied with vehicles. Vehicles from other lanes cannot cross the junction of an intersection and drive into lane l_{ij} in green light.*
- *Critical short roads refer to those roads with a length shorter than the cell length $v^*_{ij} \times \Delta t$.*

Proof 4 *See Appendix A.4.*

The proposition above clarifies under which circumstances coordination is needed between neighboring intersections. Firstly, suppose the queueing vehicles spread to the last cell of the outgoing lane. In this case, the control policy should not allow any vehicle from the incoming lane to drive into the congested outgoing lane. To guide the controller to learn such decisions, we augment the individual state with the outgoing lane state as $s_i = \{s^{\text{in}}_{ij}\}_{j\in\mathbb{L}^{\text{in}}_i} \cup \{s^{\text{out}}_{ik}\}_{k\in\mathbb{L}^{\text{out}}_i}$. In this way, the single-step delay can be completely represented given the state with both incoming and outgoing lanes. Therefore, considering the queue spillback case, the condition becomes $r_i(s, a) = r_i(\{s^{\text{in}}_{ij}\}_{j\in\mathbb{L}^{\text{in}}_i}, \{s^{\text{out}}_{ik}\}_{k\in\mathbb{L}^{\text{out}}_i}, a_i)$.

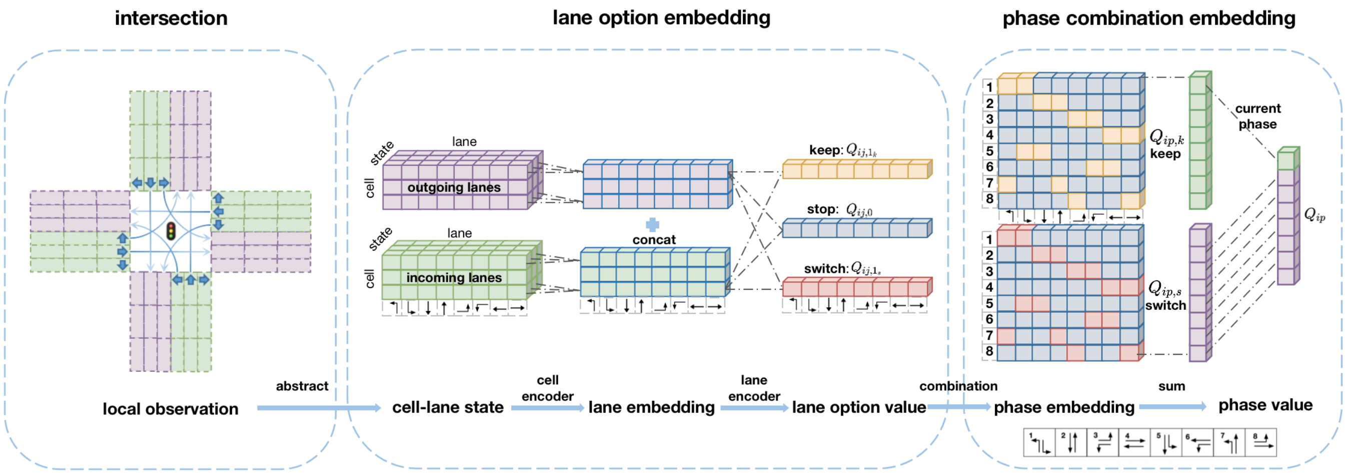


Figure 3: OAM Architecture. Local observation includes cell features for both incoming lanes and outgoing lanes. They are arranged following movement sequence and form the input cell-lane tensors. The lane option embedding component outputs three option values for each lane movement. The combinations of lane option values form the phase embeddings, including the embedding of switching phase and the embedding of keeping phase. Finally, according to the index of the current phase, we output all phase values.

Second, when there are critical short roads in the network, the reward of intersection i is directly correlated with the decision of connected intersections. Therefore, the optimization approach (e.g., max-sum algorithm) should be adopted to jointly optimize the decisions of intersections with critical roads. However, we select the decision interval as 10 seconds, where it is rare for a vehicle to travel across two intersections within 10 seconds in an urban road network. Therefore, in this paper, we do not consider the critical short roads situation.

Q Function Decomposition. To optimize the long-term reward of intersections, we introduce the Q function as $Q(s,a) = \mathbb{E}[\sum_t \gamma^t r(s_t,a_t)|s,a]$, which represents the long-term discounted reward (Sutton and Barto 2018). Following the decomposition form of single-step total delay, we give the decentralized Q function as:

Proposition 5 *The global Q function can be decomposed as the sum of local Q function as:* $Q_{tot}(s,a) = \sum_i Q_i(s_i,a_i)$

Proof 5 *See Appendix A.5.*

Further, for each intersection, the phase actions under different current phase are defined as $a_i \in \{a_{i,k}, a_{i,s}\}$, where each phase action consists of a set of lane options. Therefore, we can further decompose the intersection Q value into the sum of lane option values as:

$$Q_i(s_i,a_i) = \sum_j Q_{ij}(s_{ij},a_{ij}) \tag{5}$$

where $Q_{ij}(s_{ij},a_{ij}) = \mathbb{E}[\sum_t \gamma^t r_{tij}(s_{tij},a_{tij})|(s_{ij},a_{ij})]$ represents the lane option value. Since the lane options consist of three choices: $a_{ij} \in \{\mathbf{1}_k, \mathbf{1}_s, \mathbf{0}\}$, the corresponding lane option values represent the value of movement under keeping phase $Q_{ij,1_k}$, movement under switching phase $Q_{ij,1_s}$ and stopping $Q_{ij,0}$, respectively.

To solve the Q values for phase actions, we follow the deep Q learning algorithm (Mnih et al. 2015) to minimize the Bellman residues as:

$$L(\theta) = \mathbb{E}[(r_i + \gamma \max_{a'_i} Q_i(s'_i,a'_i,\theta) - Q_i(s_i,a_i,\theta))^2] \tag{6}$$

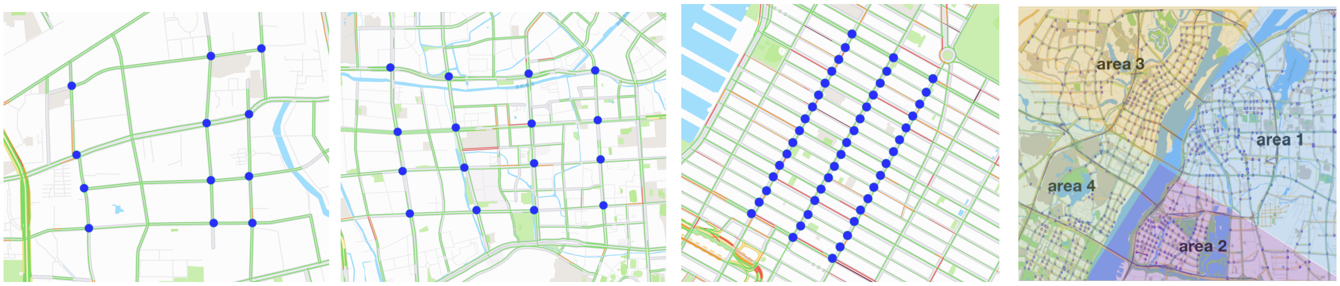
where θ denotes the parameters of neural network for Q function approximation.

The decomposition scheme above motivates us to design a universal neural network architecture to represent the lane option values and phase Q values in the next section.

Neural Network Design

Different intersections consist of different roads with different capacities and speed limits, resulting in different available movements and phases. A fixed-input-output neural network cannot generalize to different intersections (Wei et al. 2019a,b; Chen et al. 2020). Existing universal structures (Zheng et al. 2019; Oroojlooy et al. 2020) only model participating lane movements of a phase. Besides, they directly use the embedding of current phase information to model the action of switching phase. However, the phase embedding is not universal to all intersections since phase definitions are varied for different intersections. To deal with the issues mentioned above, we present the neural network architecture in Fig. 3.

- Local observation: the input state of intersection i consists of the cell state in each incoming lane and outgoing lane, and is formulated as 3-dimension tensors. For features of each incoming lane, they follow the sequence of each lane movement.
- Cell and lane embedding: we adopt a sharing-parameter multilayer perceptron (MLP) to encode the cell state as e_{ijc}. Then we concatenate all cell embedding of a lane and gain the embedding of each lane movement as e_{ij}.



(a) Jinan, 12 traffic signals (b) Hangzhou, 16 traffic signals (c) Manhattan, 48 traffic signals (d) Nanchang, 2,048 traffic signals

Figure 4: Experimental road networks

- Lane option value: each lane movement l_{ij} has three options: $a_{ij} \in \{\mathbf{1}_k, \mathbf{1}_s, \mathbf{0}\}$. We can use three different MLP encoders to output three option values as $\{Q_{ij,1_k}, Q_{ij,1_s}, Q_{ij,0}\}$, respectively. To reduce the learning complexity, motivated by the dueling structure (Wang et al. 2016), we propose a local dueling structure for option value representation as $Q_{ij,1_k} = V_{ij,0} + A_{ij,1_k}$ and $Q_{ij,1_s} = V_{ij,0} + A_{ij,1_s}$, where $Q_{ij,0} = V_{ij,0}$ is the value for default stopping option, $A_{ij,1_k}$ is the advantage option value for keeping phase compared with stopping, and $A_{ij,1_s}$ is the advantage option value for switching phase compared with stopping. Therefore, instead of outputting three option values separately, the lane encoder outputs $\{V_{ij,0}, A_{ij,1_k}, A_{ij,1_s}\}$ and add them following the dueling structure.
- Phase value: based on the specific lane-phase combinations, we concatenate different option values into phase embedding, including embedding of keeping phase and switching phase. For example, the embedding of keeping phase p is as $e_{ip,k} = \{Q_{ij,1_k}\}_{j\in\mathbb{L}_i^p} \cup \{Q_{ij,0}\}_{j\in\mathbb{L}_i^{\text{in}}\setminus\mathbb{L}_i^p}$. We then aggregate each phase embedding to corresponding phase value through summation. For instance, the value of keeping phase is as:

$$Q_{ip,k} = \sum_{j\in\mathbb{L}_i^p} Q_{ij,1_k} + \sum_{j\in\mathbb{L}_i^{\text{in}}\setminus\mathbb{L}_i^p} Q_{ij,0} \tag{7}$$

$$= \sum_{j\in\mathbb{L}_i^{\text{in}}} V_{ij,0} + \sum_{j\in\mathbb{L}_i^p} A_{ij,1_k} \tag{8}$$

  The Qmix structure (Rashid et al. 2018) and attention mechanism (Yang et al. 2020) can also be adopted for the phase embedding aggregation. Based on the index of the current phase, we output corresponding phase values at this time step.

A detailed discussion about model generalization and parameter complexity is presented in Appendix B.2.

Experiments

Experiment Setting

We conduct a series of empirical experiments on Cityflow (Zhang et al. 2019), an open-source platform for traffic simulation. Given the configurations of the road network and traffic flow, the simulator can provide the traffic information accordingly and execute the chosen phases derived by the control policy. Firstly, we conduct single-environment and multi-environment training on three different road networks to evaluate the proposed OAM controller. The public datasets[2] provide the networks of Jinan (4×3 grids), Hangzhou (4×4 grids) and Manhattan (16×3 grids). To test the generalization ability of the proposed method, we further conduct the experiments on the network of Nanchang city[3], which consists of 2,048 intersections with various junction topologies and properties (e.g., lane length, lane speed limits, etc.). A detailed training scheme is presented in Appendix B.3. Lastly, an ablation study is presented to illustrate the effectiveness of different components of the OAM method.

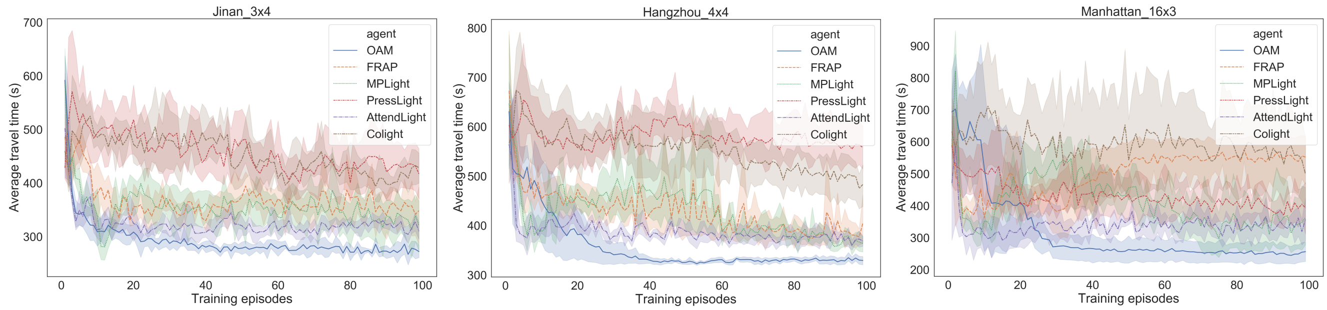
Compared Methods

Here are the brief introductions of the benchmarks. The proposed OAM method is compared with the following SOTA RL-based approaches. We do not include the conventional methods such as SOTL and Max-pressure, as they cannot outperform the RL-based controllers as presented in the past works (Zheng et al. 2019; Wei et al. 2019b; Chen et al. 2020).

- PressLight (Wei et al. 2019a): A RL controller with the use of DQN, whose reward is defined as the pressure of each intersection inspired by (Varaiya 2013).
- FRAP (Zheng et al. 2019): By modeling the phase competition mechanisms, the green signal is more likely given to the movements with higher demand. Besides, it can achieve invariance to symmetries in signal controls, thus reducing the state dimensions.
- MPLight (Chen et al. 2020): Combined FRAP structure with pressure-based reward, MPLight conducts large-scale experiments on Manhattan road network.
- AttendLight (Oroojlooy et al. 2020): This method adopts the attention network to handle the different topologies of intersections. A universal model is built up for any network configuration, and the reward is also set as the pressure of the intersections.

[2]https://traffic-signal-control.github.io/

[3]https://kddcup2021-citybrainchallenge.readthedocs.io/



(a) Jinan, 12 traffic signals (b) Hangzhou, 16 traffic signals (c) Manhattan, 48 traffic signals

Figure 5: Training performance

- CoLight (Wei et al. 2019b): This method employs the graph attention network to incorporate neighbor's information, thus enhancing the cooperation between neighboring intersections. The reward is to minimize the queue length of the intersections.

Hyperparameters. For the fairness of comparison, all the RL-based methods employ the parameter-sharing scheme and are trained by DQN (Mnih et al. 2015) with the following parameters: the discount factor, batch size and learning rate are set as 0.9, 256 and 1e-3, respectively. The buffer size is limited to four episodes, the optimizer is Adam and the exploration strategy uses $\epsilon-$greedy. The input state includes the number of vehicles, lane density, average lane speed, and phase information encoded as one-hot. For each experiment, a parameter-sharing agent is trained with transitions collected from all intersections of the road network. Each controller is trained under different random seeds three times, and the average travel time is chosen as the evaluation metric.

Single-environment Training

We start with experiments on single-environment training to compare the OAM controller with the benchmarks. Specifically, each controller is trained and tested on the same networks (Jinan, Hangzhou, and Manhattan), respectively. The progressions of average travel time along training are shown in subfigures (a), (b), and (c) in Fig. 5. Firstly, it depicts that the convergence rates of the controllers with topology-invariant structures (FRAP, AttendLight, and OAM) are much faster than those with fully connected neural network structures (PressLight and CoLight). This is due to the topology-invariant framework, which efficiently utilizes the transitions from other intersections and hence accelerates the learning process. We then find that the proposed OAM structure presents a lower variance of reward curve than other structures since OAM considers all lane movements instead of only participating lanes. In comparison, FRAP and AttendLight fit each intersection's Q function based on only part of the local information, which results in higher variance and lower accuracy. Moreover, although the AttendLight with attention-based aggregation can outperform the FRAP structure with summation-based aggregation, it cannot match the performance of OAM. This again emphasizes that the observations of the non-participating lanes play a vital role in Q function approximation.

Multiple-environment Training

We then conduct experiments on multiple environments to demonstrate the generalization of our proposed method. FRAP, MPLight, and AttendLight are used for comparison as they explicitly considered generalization in controller design. As shown in Appendix B.2, those controllers interact with the three networks in parallel for training. The result is presented in Fig. 6. Trained by the sharing transitions from different road networks, OAM can still deliver a learning curve with lower variance. On the contrary, the FRAP-based approaches present significant variance and oscillations during training. Although the variance given by the AttendLight decreases along the training process, the OAM structure can always maintain a lower variance and higher reward even without the attention mechanism. Therefore, the multi-environment experiments clearly demonstrate the superior generalization ability of the proposed OAM structure.

City-level Training and Evaluation

The three gird networks above (Jinan, Hangzhou, and Manhattan) mainly consist of regular 4-legged intersections with similar speed limits and road lengths. However, for a real-world city, the intersections are more complex and irregular. For example, there are 3-, 4-, and 5-legged intersections with different road lengths in the Nanchang network. Therefore, we conduct city-level experiments to demonstrate the scalability and generalization ability of the proposed OAM method. Specifically, as shown in subfigure (d) in Fig. 4, we select area 4 of Nanchang to train a parameter-sharing agent with randomly generated traffic flows of different volumes. We then test the trained agents in other areas with different scales of traffic flows. As shown in Table 1, OAM outperforms MPLight and AttendLight in eight scenarios out of nine, as the OAM delivers the least average travel time in all areas under different traffic demands. Besides, MPLight can handle the mild demand better as it outperforms AttendLight in scenarios with fewer vehicles (area 2 and area 3 with traffic volumes of 2,000 and 4,000), while AttendLight performs better under congested situations (area 1 and area 2 with traffic volume of 6,000). To sum up, the generaliza-

	Area 1	Area 2	Area 3	Area 1	Area 2	Area 3	Area 1	Area 2	Area 3
	traffic volume of 2,000			traffic volume of 4,000			traffic volume of 6,000		
MPLight	410	323	436	463	347	554	454	381	587
AttendLight	469	422	551	407	445	546	387	415	538
OAM	**371**	**268**	**403**	**405**	**327**	**489**	392	**364**	**513**

Table 1: Average travel time (s) in testing experiments of Nanchang

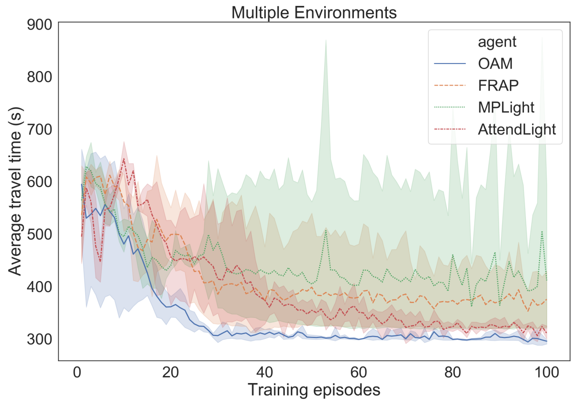


Figure 6: Multiple environments training

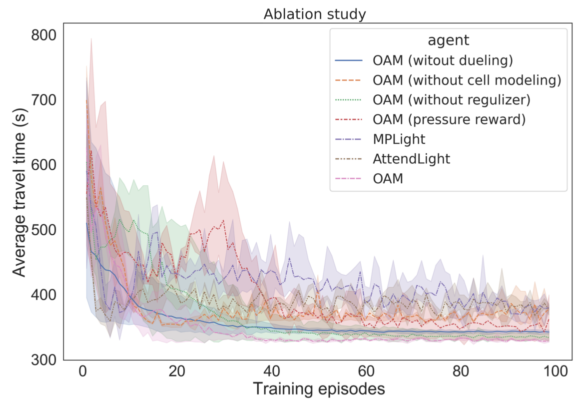


Figure 7: Ablation study of OAM

tion ability of OAM enables it to handle various intersection topologies and demand patterns on a city-wide network.

Ablation Study

Finally, to demonstrate the efficiency of each component of OAM, we conduct an ablation study on the Hangzhou network, and the performance is shown in Fig. 7. Firstly, leaving out the cell-based state representation increases the variance of the learning curve and degrades performance to the level of MPLight and AttendLight. It reveals that the lane-based features fail to represent the traffic flow propagation under different traffic volumes, especially for long road segments. Furthermore, the pressure-based OAM shows higher variance and instability in reducing the travel time, as the pressure is a rough representation of traffic flow and cannot provide a precise reward signal for intersections with varied road lengths. Without the outflow-based regularizer, the convergence rate of the learning process slows down significantly. This demonstrates the efficiency of the regularizer in alleviating the credit assignment problem. However, it still gives a similar final performance with the complete OAM, which empirically proves that the regularizer does not affect the optimality of total network delay. Lastly, the lane option dueling structure accelerates the learning process while reaching a similar performance to OAM.

Conclusion

In this paper, we propose an option-action reinforcement learning framework for universal multi-intersection control, which leverages a parameter-sharing training scheme and reaches generalization to any intersection with any phase definition. Based on three benchmarks tests and city-level experiments, the proposed method has strong generalization ability and outperforms existing structures.

We also acknowledge the limitations of our current approach. We assume that no vehicle can cross two intersections within a 10-seconds decision interval. Admittedly, this assumption can not be held for all situations. The next step is to relax the assumption and give a more generalized and theoretically guaranteed algorithm.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 72071214. Enming Liang was partially supported by a General Research Fund from Research Grants Council, Hong Kong Special Administrative Region of the People's Republic of China (Project No. 11206821). We thank the reviewers for their constructive feedback, which has helped us improve the paper.

References

Chen, C.; Wei, H.; Xu, N.; Zheng, G.; Yang, M.; Xiong, Y.; Xu, K.; and Li, Z. 2020. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. In *AAAI'20*.

Daganzo, C. F. 1994. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4): 269–287.

Egea, A. C.; Howell, S.; Knutins, M.; and Connaughton, C. 2020. Assessment of Reward Functions for Reinforcement Learning Traffic Signal Control under Real-World Limitations. In *IEEE SMC'20*.

Kuyer, L.; Whiteson, S.; Bakker, B.; and Vlassis, N. 2008. Multiagent reinforcement learning for urban traffic control using coordination graphs. In *ECML PKDD'08*.

Marden, J. R. 2012. State based potential games. *Automatica*, 48(12): 3075–3088.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A. A.; Veness, J.; Bellemare, M. G.; Graves, A.; Riedmiller, M.; Fidjeland, A. K.; Ostrovski, G.; et al. 2015. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533.

Oroojlooy, A.; Nazari, M.; Hajinezhad, D.; and Silva, J. 2020. AttendLight: Universal Attention-Based Reinforcement Learning Model for Traffic Signal Control. *arXiv:2010.05772*.

Rashid, T.; Samvelyan, M.; De Witt, C. S.; Farquhar, G.; Foerster, J.; and Whiteson, S. 2018. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. *arXiv:1803.11485*.

Su, Z.; Chow, A. H.; and Zhong, R. 2021. Adaptive network traffic control with an integrated model-based and data-driven approach and a decentralised solution method. *Transportation Research Part C: Emerging Technologies*, 128: 103154.

Sutton, R. S. 1984. *Temporal credit assignment in reinforcement learning*. Ph.D. thesis, University of Massachusetts Amherst.

Sutton, R. S.; and Barto, A. G. 2018. *Reinforcement learning: An introduction*. MIT press.

Van der Pol, E.; and Oliehoek, F. A. 2016. Coordinated deep reinforcement learners for traffic light control. *NIPS'16*.

Varaiya, P. 2013. Max pressure control of a network of signalized intersections. *Transportation Research Part C: Emerging Technologies*, 36: 177–195.

Wang, Z.; Schaul, T.; Hessel, M.; Hasselt, H.; Lanctot, M.; and Freitas, N. 2016. Dueling network architectures for deep reinforcement learning. In *ICML'16*.

Wei, H.; Chen, C.; Zheng, G.; Wu, K.; Gayah, V.; Xu, K.; and Li, Z. 2019a. Presslight: Learning max pressure control to coordinate traffic signals in arterial network. In *KDD'19*.

Wei, H.; Xu, N.; Zhang, H.; Zheng, G.; Zang, X.; Chen, C.; Zhang, W.; Zhu, Y.; Xu, K.; and Li, Z. 2019b. Colight: Learning network-level cooperation for traffic signal control. In *KDD'19*.

Wei, H.; Zheng, G.; Yao, H.; and Li, Z. 2018. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *KDD'18*.

Xu, B.; Wang, Y.; Wang, Z.; Jia, H.; and Lu, Z. 2021. Hierarchically and Cooperatively Learning Traffic Signal Control. In *AAAI'21*.

Yang, Y.; Hao, J.; Liao, B.; Shao, K.; Chen, G.; Liu, W.; and Tang, H. 2020. Qatten: A General Framework for Cooperative Multiagent Reinforcement Learning. *arXiv:2002.03939*.

Zang, X.; Yao, H.; Zheng, G.; Xu, N.; Xu, K.; and Li, Z. 2020. Metalight: Value-based meta-reinforcement learning for traffic signal control. In *AAAI'20*.

Zhang, H.; Feng, S.; Liu, C.; Ding, Y.; Zhu, Y.; Zhou, Z.; Zhang, W.; Yu, Y.; Jin, H.; and Li, Z. 2019. Cityflow: A multi-agent reinforcement learning environment for large scale city traffic scenario. In *WWW'19*.

Zhang, H.; Liu, C.; Zhang, W.; Zheng, G.; and Yu, Y. 2020. GeneraLight: Improving Environment Generalization of Traffic Signal Control via Meta Reinforcement Learning. In *CIKM'20*.

Zheng, G.; Xiong, Y.; Zang, X.; Feng, J.; Wei, H.; Zhang, H.; Li, Y.; Xu, K.; and Li, Z. 2019. Learning phase competition for traffic signal control. In *KDD'19*.

Zhu, L.; Peng, P.; Lu, Z.; Wang, X.; and Tian, Y. 2021. Meta Variationally Intrinsic Motivated Reinforcement Learning for Decentralized Traffic Signal Control. arXiv:2101.00746.