8-2021

Dynamic lane traffic signal control with group attention and multi-timescale reinforcement learning

Qize JIANG
Fudan University

Jingze LI
Fudan University

Weiwei SUN
Fudan University

Baihua ZHENG
Singapore Management University, bhzheng@smu.edu.sg

Citation

Dynamic Lane Traffic Signal Control with Group Attention and Multi-Timescale Reinforcement Learning

Qize Jiang[1,2,3] , **Jingze Li**[1,2,3] , **Weiwei Sun**[1,2,3] and **Baihua Zheng**[4]

[1]School of Computer Science, Fudan University
[2]Shanghai Key Laboratory of Data Science, Fudan University
[3]Shanghai Institute of Intelligent Electronics & Systems
[4]School of Computing and Information Systems, Singapore Management University
{qzjiang18,jingzeli20,wwsun}@fudan.edu.cn, bhzheng@smu.edu.sg

Abstract

Traffic signal control has achieved significant success with the development of reinforcement learning. However, existing works mainly focus on intersections with normal lanes with fixed outgoing directions. It is noticed that some intersections actually implement dynamic lanes, in addition to normal lanes, to adjust the outgoing directions dynamically. Existing methods fail to coordinate the control of traffic signal and that of dynamic lanes effectively. In addition, they lack proper structures and learning algorithms to make full use of traffic flow prediction, which is essential to set the proper directions for dynamic lanes. Motivated by the ineffectiveness of existing approaches when controlling the traffic signal and dynamic lanes simultaneously, we propose a new method, namely *MT-GAD*, in this paper. It uses a group attention structure to reduce the number of required parameters and to achieve a better generalizability, and uses multi-timescale model training to learn proper strategy that could best control both the traffic signal and the dynamic lanes. The experiments on real datasets demonstrate that MT-GAD outperforms existing approaches significantly.

1 Introduction

Traffic congestion has become a hot button issue in major cities all round the world. Traffic signals and traffic lanes are two common approaches adopted to help control traffic flows and improve traffic safety. The former refers to signalling devices positioned mainly at road intersections to regulate the traffic flow; and the latter partitions a road into lanes with each used by a single line of vehicles to control and guide the movements of vehicles and to reduce traffic conflicts.

Most traffic signals are controlled by simple timers, which are simple yet effective for busy areas with *consistent* or *predictable* volume of traffic. However, they may force drivers to stop unnecessarily when no traffic is present in areas where traffic is unpredictable or sporadic. Recently, *intelligent* traffic lights have been widely exploited, which are expected to

be able to detect cars and change the traffic lights accordingly to better manage the traffic flows.

Among many efforts to equip the traffic signals with intelligence, control algorithms using reinforcement learning have achieved good results, such as IntelliLight [Wei *et al.*, 2018] and FRAP [Zheng *et al.*, 2019]. They use the current traffic status of an intersection as the state, select a practicable combination of the traffic signals as the action, and consider data such as queue length and vehicle travel time as the value. With deep reinforcement learning, these methods have achieved better results than traditional methods. However, they do not capture the characteristics of traffic data well. They either simply use state observation, without considering the symmetric structure of intersection; or consider only part of data but are not able to capture global information, which makes them not generalizable to different situations. Although there are algorithms on improving the generalizability, such as [Zang *et al.*, 2020; Zheng *et al.*, 2019], their structures are not well-designed and suffer from slow convergence speed. Consequently, low model learning efficiency or poor generalizability stop them from learning good traffic signal control strategies or being employed at intersections with different traffic flows.

In terms of traffic lanes, we focus on *dynamic lanes* (a.k.a. variable lanes), a special type of traffic lanes that could change the direction of the lanes dynamically, mainly during peak hours, to improve traffic flow. The lane labelled 2 in Figure 1 is a typical dynamic lane. We want to highlight that the dynamic lanes focused in this work refer to the lanes whose directions could be changed between *go straight* and *turn left* (or *turn right*), but not the reversible lanes whose directions could be reversed.

Dynamic lanes have achieved big success in many cities. For example, Shanghai set 7 dynamic lanes in 2005, with all making positive impact at various locations. Take the one on Tianyaoqiao Road as an example. It helps to reduce the average number of traffic light cycles that left-turning vehicles have to wait during rush hours by 50%.

Controlling of traffic signal has been well-studied, but not that of dynamic lanes. We would like to highlight that these two problems have very different characteristics. First, traffic signal controlling algorithms typically only consider the current volume of traffic flow, as the change of traffic signal is

Figure 1: A dynamic lane with direction

expected to have an immediate impact on the traffic flow. On the other hand, the change of the directions of dynamic lanes does affect the traffic, but the immediate results might not be positive. However, it will generate positive results which actually take some time. In other words, an algorithm controlling traffic signal cares more about immediate impacts while that of dynamic lanes focuses on non-immediate results. Second, the change of traffic signal has a much higher frequency, as compared with that of dynamic lanes. We actually celebrate their differences in this paper as they provide opportunities to develop a controlling system that could coordinate the change of traffic signal and that of dynamic lanes for a better controlling of traffic flow.

In order to enable the coordination of the above two controlling tasks, we propose a novel algorithm named *Multi-Timescale Group Attention Dynamic lane traffic signal control* (MT-GAD). To the best of our knowledge, this is the *first* attempt to study the traffic signal control with dynamic lanes. Our main contributions are listed below.

- We propose MT-GAD as the first solution to coordinate the control of traffic signal and that of dynamic lanes to better control the overall traffic flows.

- We adopt group attention to achieve better traffic signal control results and enhance the model's generalizability.

- We design a multi-timescale learning method to enable the learning of signal control and that of lane control simultaneously.

- We perform an experimental study based on real data to demonstrate the superior performance of our model.

2 Related Work

In this section, we review existing works on traffic control with dynamic lanes and traffic signal control.

Traffic control with dynamic lanes. Controlling traffic lanes to ease traffic congestion is common. Dynamic lanes, as an example, are used in many major cities, especially in China, to improve traffic efficiency in rush hour. [Li *et al.*, 2009] proposes a pre-signal control method based on a fictitious stop line to control dynamic lanes. [Zhou *et al.*, 2019] proposes a traffic priority method as a solution. [Yao *et al.*, 2018; Li *et al.*, 2013] propose an optimization model which minimizes the total delay.

Traffic signal control. For traffic signal control, the fixed-timer which switches traffic signal phase with pre-defined time interval is simple and easy, but can't make use of real time traffic flows. SOTL [Cools *et al.*, 2013] switches traffic signals based on vehicle numbers of approaching lanes.

Recently, many traffic signal control methods with deep reinforcement learning have been proposed. [Li *et al.*, 2016] uses stacked auto-encoders to approximate rewards based on queue lengths. [Gao *et al.*, 2017] uses CNN to approximate value function. [Wei *et al.*, 2018] uses deep reinforcement learning to solve the problem. [Zheng *et al.*, 2019] considers the signal phase competition. [Zang *et al.*, 2020] uses MAML to learn a model with better training speed. [Kim and Jeong, 2020] learns a policy with predicted future traffic flow. [Zhang *et al.*, 2020] tries to control traffic signals when only part of vehicles can be detected. [Rizzo *et al.*, 2019] uses time critic policy gradient methods to handle congestion scenarios. [Oroojlooy *et al.*, 2020] uses two attention model to handle different road number intersections and traffic signal phases. Although all the above works and our study presented in this paper focus on *single* intersection, there are also many works analyzing traffic signal control with multiple intersections. [Wei *et al.*, 2019a] proposes to use pressures as the reward to perform better coordination. [Wei *et al.*, 2019b] uses attention to learn different weights between neighbors. [Zhu *et al.*, 2021] learns polices which only based on local observations by meta-learning. [Yu *et al.*, 2020] uses communication and action rectification to learn multi-agent policies.

Discussion. To our best knowledge, the control of traffic signal and that of dynamic lanes are considered as two *independent* tasks in the literature. However, they are interrelated. An effective learning model that could perform the two tasks simultaneously is expected to offer a better control of the traffic flow and that's the focus of this paper.

3 Problem Definition and Preliminaries

Following existing problem definitions on traffic signal control [Zheng *et al.*, 2019; Wei *et al.*, 2019b], we adjust the definitions to introduce the traffic signal control problem with dynamic lanes.

Traffic flow refers to the number of vehicles passing through a road within a time unit. Given one intersection, without loss of generality, we assume that there are four approaching roads with each having three outgoing directions, excluding rarely appeared U-turns. We differentiate approaching roads by their positions relative to the intersection, i.e., *North*, *South*, *East*, and *West*, and use $\{S, L, R\}$ to name outgoing directions corresponding to *go Straight*, *turn Left*, and *turn Right* respectively. Consequently, there are in total $4 \times 3 = 12$ directional traffic flows, as defined below.

Definition 1. *A* directional traffic flow (DTF) *at an intersection is denoted as* $F_{X,Y}$, *where* $X \in \{N, S, E, W\}$ *and* $Y \in \{S, L, R\}$.

Traffic signals are employed to regulate the traffic flow at busy intersections. A signal will be set for every DTF to control vehicles' movements, e.g., a green light signals that it's safe to pass through the intersection and a red light stops all the vehicles. For road safety, the routes of DTFs receiving the green signal shouldn't cross each other. Note that DTF $F_{X,Y}$ with $Y = R$ (i.e., the outgoing direction is to turn right) will never cross any other DTFs and it usually receives green signal all the time. For other eight DTFs, there are eight non-conflict phases, with each enabling two DTFs. We plot non-
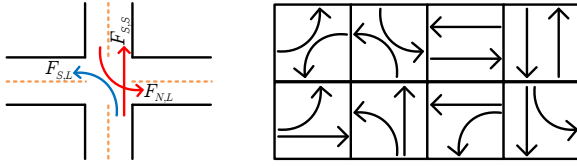
Figure 2: DTFs example and all non-conflict phases following right-handed driving

conflict phases and three DTFs in Figure 2 for illustration, $F_{S,S}$ and $F_{N,L}$ conflict with each other, while $F_{S,L}$ in blue conflicts with neither one.

Definition 2. *A traffic signal phase at time step t, denoted as P_t, sets every DTF $F_{X,Y}$ ($Y \neq R$) one signal $sig_{X,Y} \in \{red, green\}$ such that i) $F_{X,Y}$s with corresponding signal being green do not cross each; and ii) the number of DTFs with signal being green is maximized. In addition, when a different phase is selected in time step $t+1$, i.e. $P_t \neq P_{t+1}$, a short all-red phase P_{red} is used to clear vehicles going through the intersection.*

For approaching roads, each normal lane has one fixed outgoing direction, usually painted on the road; dynamic lanes could change their outgoing directions dynamically. In this paper, we set $\{S, L\}$ as the only available directions for the dynamic lane, which are most common outgoing directions in countries using right-handed traffic (RHT). We also assume a dynamic lane has only one outgoing direction at any time, which is consistent with the real setting used in China.

Definition 3. *The outgoing direction of a dynamic lane located on road X at time step t is denoted as $V_t^X \in \{S, L\}$. For safety purpose, when a dynamic lane changes its direction, V_{red} is added to stop the entering of dynamic lane for several time steps.*

According to the above definitions, we formalize the problem to be studied in this paper as follows. We assume that there is only one dynamic lane at an intersection, which is most common in the real world for intersections with four-lane roads.

Problem 1. *Dynamic lane traffic signal control. At each time step t, given observation o_t of traffic conditions at the intersection, an agent should find an optimal policy to decide P_{t+1} and V_{t+1} such that the reward r (to be formalized next) is maximized.*

4   Method Framework

In this section, we will first explain the reinforcement learning environment setting, and then detail the proposed algorithm MT-GAD.

4.1   Reinforcement Learning Environment Setting

Traffic signal control can be defined as a Markov Decision Process (MDP), which could be solved using reinforcement learning methods. MDP is defined by 5-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$, corresponding to state space $\mathcal{S}$, action space $\mathcal{A}$, state transition probability $\mathcal{P}$, reward function $\mathcal{R}$, and discount factor γ.

We adopt Deep Q-Learning (DQN) [Mnih *et al.*, 2015] as the reinforcement learning structure. With given state observation s_t and action $a = \{P_t, V_t\}$ at time step t, DQN approximates the state-action function $Q(s_t, a; \theta)$. DQN updates the network θ using the loss function defined in Eq. (1).

$$L = Q(s_t, a; \theta) - (r_a + \gamma \max_{a'} Q(s_{a,t+1}, a'; \theta')) \quad (1)$$

The state, the action, and the reward considered in this work are presented in the following.

State. The state for every DTF $F_{X,Y}$ is described by following five pieces of information. 1) $w_{X,Y}$, the number of vehicles currently waiting in $F_{X,Y}$; 2) $q_{X,Y}$, the number of vehicles in DTF trying to pass the intersection, i.e., considering the vehicles in $F_{X,Y}$ approaching the intersection and vehicles in $w_{X,Y}$; 3) $sig_{X,Y}$, the current signal received by $F_{X,Y}$; 4) $t_{X,Y}$, the time passed since the current signal has been set; and 5) $p_{X,Y}$, the predicted number of approaching vehicles in a predefined t_p time steps. In addition to the above observations related to DTFs, there are two additional pieces of information, the traffic signal phase P and dynamic lane direction V. Here, we use a simple LSTM to estimate $p_{X,Y}$, which could be replaced with any modern method to get a higher accuracy. As to be reported in the experiments, our method is *not* sensitive to the accuracy, once it is above certain threshold.

Action. The action here is to set two signals, including A^P and A^V. A^P selects a traffic signal phase for the next time step; A^V selects the direction for the dynamic lane. As there are 8 legal traffic signal phases and 2 directions for dynamic lanes, there are in total 16 different actions.

Reward. There are many available reward metrics that could be used as the performance metrics. However, many of them are based on vehicle's movement, which might not be directly available. Consequently, we use *average queue length* of every approaching lane between two time steps as the reward.

We now propose our method MT-GAD, whose structure is illustrated in Figure 3.

4.2   Traffic Flow Information Extraction

State information such as $q_{X,Y}$ and $w_{X,Y}$ is based on DTFs, so the first step of MT-GAD is to extract flow information to describe a state accurately. As stated previously, traffic signal control focuses more on the current flow and its impact is expected to be immediate and last only a few minutes; while dynamic lane control aims at improving the traffic condition in the near future and one setup could last multiple hours. Consequently, they take in different information as inputs.

To better consider corresponding features and avoid learning from unimportant information, we input $q_{X,Y}$, $w_{X,Y}$, $sig_{X,Y}$, and $t_{X,Y}$ to traffic signal controller and input $q_{X,Y}$ and $p_{X,Y}$ to dynamic lane controller. Thereafter, we use two fully-connected layers to generate traffic signal control embedding $h_{F_{X,Y}}^P$ and dynamic lane control embedding $h_{F_{X,Y}}^V$. We want to highlight that we purposely design our embeddings $h_{F_{X,Y}}^P$ and $h_{F_{X,Y}}^V$ to be based on every DTF, in order to remove the impact of different input order of DTFs. As there are eight DTFs, this step is trained eight times using one input, one per DTF.
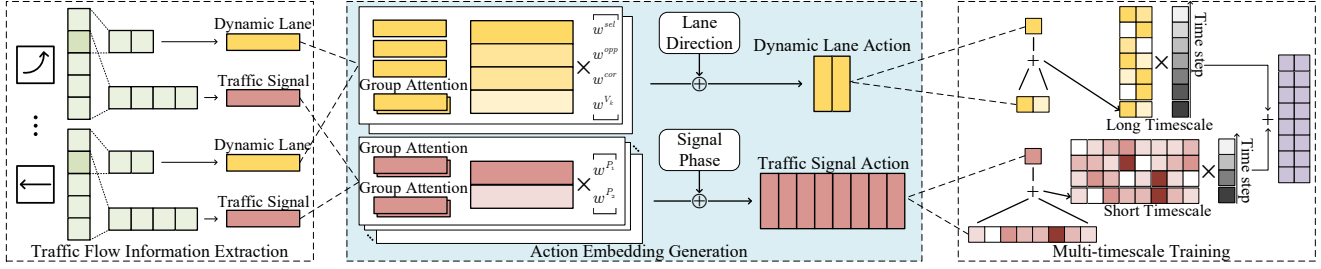
Figure 3: MT-GAD Structure. It consists of three main steps, *traffic flow information extraction* extracts corresponding information for traffic signal control and dynamic lane control; *action embedding generation with group attention* combines different information by attention and prior knowledge weighting; and *multi-timescale training* trains the two controllers with different timescales simultaneously.

4.3 Action Embedding Generation

After generating $h^P_{F_{X,Y}}$ and $h^V_{F_{X,Y}}$, we extract embeddings for every action based on $h^P_{F_{X,Y}}$ and $h^V_{F_{X,Y}}$. Attention mechanism [Vaswani *et al.*, 2017] provides a good solution to make input order of DTF states irrelevant. Attention takes query Q to combine key-value pairs $\langle K, V \rangle$ to an output based on Q.

Attention allows us to combine DTFs together based on their importance. However, attention also introduces a large number of parameters which slow down the training process significantly and increase the risk of overfitting the model. Because action embedding contains all input information, it's common to set hidden size of action embedding larger than h_{F_i}, and the biggest parameter matrix is W^Q. Consequently, we propose to calculate the attention without using W^Q.

We notice that attention has many parameters as it doesn't fully utilize the characteristics of traffic flows and control actions. Take the top left phase in Figure 2 as an example. Information related to DTFs $F_{W,L}$ and $F_{E,L}$ is significantly more important than the information of other DTFs. Accordingly, we group DTFs based on their impacts on the current control phase. When calculating the k-th phase P^k, we split DTFs into two groups based on their signals received in P^k, where group G_g contains DTFs receiving green signal, and group G_r for DTFs receiving red signal. For example, the top-left phase in Figure 2 has $F_{W,L}$ and $F_{E,L}$ in G_g, and other DTFs in G_r. Because the DTFs in the same group share the same impact on P^k, we use attention to generate one embedding for every group, with all-one vector I as Q and $h^P_{F_{X,Y}}$ as K and V. Then, we use predefined weights w^g and w^r to calculate the weighted average H^{P_k} of embeddings, i.e., $H^{P_k} = w^g A^{P_k}_g + w^r A^{P_k}_r$, where $A^{P_k}_i = \text{Attention}(I, G^{P_k}_i, G^{P_k}_i), (i \in g, r)$.

For a dynamic lane, it controls two different directions with opposite impacts on its controlling. Considering one outgoing direction $Y \in \{S, L\}$ as the selected direction, we use $F_{X,Y}$ to represent the selected DTF F_{sel}. Accordingly, $F_{X,Y'}$ with $Y' = \{S, L\} \backslash Y$ indicates the opposite DTF F_{opp}. We further introduce the coordinate DTF F_{cor} to represent the DTF that has the chance to receive green signal together with F_{sel}. For example, given a dynamic lane $V_E = L$, we have $F_{sel} = F_{E,L}$, $F_{opp} = F_{E,S}$, and $F_{cor} = F_{W,L}$. We strongly believe that F_{cor}, as compared with other DTFs, is more important for the controlling of the dynamic lanes.

Consequently, we extract these three DTFs (i.e., F_{sel}, F_{opp}, and F_{cor}) individually but consider the other five DTFs as a group, represented by an attention. We apply the same idea to traffic signal control to calculate the weighted average H^{V_k}.

$$G^{V_k} = \{h^V_{F_j} | F_j \notin \{F^{V_k}_{sel}, F^{V_k}_{opp}, F^{V_k}_{cor}\}\}$$
$$A^{V_k} = \text{Attention}(I, G^{V_k}, G^{V_k})$$
$$H^{V_k} = w^{sel} h^{V_k}_{F_{sel}} + w^{opp} h^{V_k}_{F_{opp}} + w^{cor} h^{V_k}_{F_{cor}} + w^{V_k} A^{V_k}$$

Finally, we concatenate H^{P_k} and H^{V_k} with current phase embedding that goes through one fully-connected layer, , i.e., $E^{P_k} = \sigma(W(H^{P_K} \oplus \text{Embedding}(P)) + b)$, and the same as E^{V_k}. Then, we use the Dueling DQN structure to calculate Q_P and Q_V based on E^P and E^V separately.

4.4 Multi-Timescale Training

The differences between traffic signal control and dynamic lane control affect not only the extraction of relevant information, but also the learning process. A multi-timescale reinforcement learning method that is able to capture both close time information and future information is preferred.

To overcome the challenge of multi-timescale learning, we propose to focus on the discount factor γ for the MDP environment. When γ is small (e.g., $\gamma = 0.8$), the environment considers short-term reward, which fits traffic signal control. Meanwhile, a small γ allows the learning algorithm to ignore the reward achieved by successful direction changes of dynamic lanes, as $\gamma^{V_{red}}$ is expected to be so small that is almost negligible (e.g., $0.8^{30} \approx 0.001$ based on our parameter setups). On the other hand, when γ is large (e.g., $\gamma = 0.95$), the reward achieved by dynamic lane control becomes dominant. In other words, the control of traffic signal and that of dynamic lane use γ of different scales. When calculating $Q_P(s_t, a_t; \theta)$ for traffic signal control, we use small γ (e.g., 0.8); when calculating dynamic lane control $Q_V(s_t, a_t; \theta)$, we use bigger γ (e.g., 0.95). We then calculate the Q value for combined action $a_{ij} = \langle P_i, V_j \rangle$ as $Q(s_t, a_{ij}; \theta) = Q_P(s_t, P_i; \theta)/2 + Q_V(s_t, V_j; \theta)/2$.

Instead of using the standard loss function defined in Eq. (1), we consider the one defined in Eq. (2) to take into account the difference between γ_P and γ_V.

$$L = Q(s_t, a_{ij}; \theta) - \big(0.5 \cdot \gamma_P \cdot \max_{P'_i} Q(s_{a_{ij}, t+1}, P'_i; \theta')$$
$$+ 0.5 \cdot \gamma_V \cdot \max_{V'_j} Q(s_{a_{ij}, t+1}, V'_j; \theta') + r_{a_{ij}, t}\big) \quad (2)$$

5 Experiments

5.1 Experiment Settings

Our experimental study is based on a modified version of CityFlow [Zhang *et al.*, 2019] to support dynamic lanes. In addition to traffic signal control available in original CityFlow, we further implement dynamic lane controller.

In our experimental study, we focus on a single intersection with one dynamic lane, which is a common setup used in many existing works [Li *et al.*, 2009; Zhou *et al.*, 2019]. Every road has 3 lanes, with 1 turn-left lane, and 2 go-straight lanes. We want to highlight that our model considers input with respective to the directions of the DTFs, so the number of lanes is actually *irrelevant*. As stated before, we assume there is only one road with a dynamic lane, and the road with a dynamic lane $V_X \in \{L, S\}$ replaces one of the two go-straight lanes it has with the dynamic lane. We set $P_{red} = 5s$, $V_{red} = 300s$, and the time interval for agent to make action is $10s$. The predict length of traffic flows $t_p = 600s$.

We run the experimental study using a personal computer with AMD ThreadRipper 2950X CPU, 2080 Ti GPU, and 128GB memory. All methods are trained with 240,000 frames.

Datasets. Existing datasets available for traffic signal control problem (e.g., [Zheng *et al.*, 2019]) only contain traffics in 3600 seconds, which is too short to evaluate the impact of dynamic lane changes. Consequently, we develop two datasets based on real taxi trajectories in Porto and Shanghai to simulate the traffic within one whole day. Porto (PT) dataset [Moreira-Matias *et al.*, 2015] contains close to 1 million trajectories generated by 438 taxis from 1 July 2013 to 20 March 2014. Shanghai (SH) dataset contains 11,861,593 trajectories generated by 13,767 taxis in the whole month of April, 2015. We use map-matching [Song *et al.*, 2012] to convert GPS points to trajectories. To overcome the high variance caused by small amount of taxis compared with real traffic, we ignore the day of trajectories to make data dense, and balance the average vehicle number to simulate *high-volume* and *low-volume* traffic flows. To benchmark the average volume, we refer to the data collected by the Los Angeles government [Alcazar, 2020] and we find that the most busy intersections, such as Angeles Vista Boulevard and Slauson Avenue intersection, have over $44,000$ vehicles passed in a day, i.e. 0.51 vehicle per second on average. Consequently, we use this in our setting to simulate high-demand situation, and reduce the flow by 50% to simulate low-volume traffic.

A dynamic lane should be implemented at intersections satisfying following two rules, rule 1) they have large numbers of samples to guarantee they are busy; and rule 2) the directions of their flows do change. We select two intersections randomly from each dataset based on these two rules, denoted as SH1, SH2, PT1, and PT2, and another two intersections SH3 and PT3 that not fully satisfy two rules to showcase that our algorithm is also able to learn effective signal control methods when there is no need for dynamic lane. We simulate two different flow volumes for every intersection, denoted by subscriptions low and high, e.g., $SH1_{low}$ and $SH1_{high}$ refer to SH1 with low volume and high volume respectively.

Compared methods. To evaluate the efficiency of MT-GAD, we implement the following methods as the main competitors, including **FixedTime** and **SOTL** [Cools *et al.*, 2013], two representatives of traditional methods, and **PressLight** [Wei *et al.*, 2019a], **CoLight** [Wei *et al.*, 2019b] and **FRAP** [Zheng *et al.*, 2019], three representatives of reinforcement learning (RL) approaches. Due to space limitation, we skip their details that have been presented in related work. We also extend these methods to control dynamic lanes, except FixedTime, which is unable to control dynamic lanes. For SOTL, it changes the direction of a dynamic lane when the vehicle number on corresponding DTFs is unbalanced; for RL based algorithms, we extend their action space from 8 to 16. Last but not the least, we include a variant of MT-GAD, labelled as **MT-GAD-w/o-DL**, that controls only traffic signal by fixing the dynamic lane to one direction (e.g., $A_t^V = S$ all the time).

Evaluation metrics. We use *vehicles' average delay* $\mathcal{D}$ as the main performance metric. $\mathcal{D}$ refers to the difference between the real travel time T_{real} spent by a vehicle from entering the intersection until passing through the intersection and the expected travel time T_{exp}, which is the ratio of the route length to the speed limit [Wei *et al.*, 2019c], i.e., $\mathcal{D} = T_{real} - T_{exp}$. As controlling algorithms do not affect T_{exp}, this metric is the same as *vehicle's average travel time*.

5.2 Evaluate Results

Overall performance. The overall performance is reported in Table 1. In general, reinforcement learning algorithms perform better than traditional algorithms, because they can learn detailed strategies. When traffic flow is low, the differences among RL algorithms are rather minor, because it's easier to learn a good strategy in low traffic and the room for further improvement is very limited. When traffic volume becomes higher, our model is able to consistently outperform all the competitors by a significant margin. When we compare a method with its variance that implements dynamic lane controller, it is observed that dynamic lane is able to improve the traffic only in some cases. This is because they all only combine traffic signal control and dynamic lane control via a simple strategy, without considering the unique features of the two controlling tasks. On the other hand, MT-GAD outperforms MT-GAD w/o DL consistently when the traffic volume is high.

Generalizability. Due to space limitation, we only report the result under SH dataset. We train the models based on $SH1_{high}$, and then deploy them directly to other datasets, including $SH2_{high}$, $SH3_{high}$ and $SH1_{rotate}$ that rotates all traffic flows and the dynamic lane of $SH1_{high}$ in clockwise by 90 degree, e.g., $V_W \rightarrow V_N$, $F_{W,S} \rightarrow F_{N,S}$, etc. Table 2 reports the results, with % indicating the ratio of $\mathcal{D}$ to $\mathcal{D}'$. Here, $\mathcal{D}$ refers to the performance a model could achieve if the training and testing are for the same intersection, and $\mathcal{D}'$ refers to the performance achieved based on the model trained on top of $SH1_{high}$. MT-GAT* is a variant of MT-GAD which replaces the group attention with standard attention. Obviously, MT-GAD demonstrates a superior generalizability, significantly better than other models. MT-GAT* is observed to perform well in some situations but perform badly in others. This is

Datasets		$PT1_{high}$		$PT1_{low}$		$PT2_{high}$		$PT2_{low}$		$PT3_{high}$		$PT3_{low}$	
Dynamic Lane		w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/
Trad. Algo.	FixedTime	1753.8	-	98.8	-	3024.9	-	638.3	-	1810.7	-	104.3	-
	SOTL	879.1	477.7	66.2	59.6	2018.5	2118.2	203.2	218.1	780.3	294.7	63.1	58.0
RL Algo.	PressLight	208.9	2433.8	32.6	40.0	769.1	2787.5	61.0	97.8	456.5	865.0	43.6	47.9
	CoLight	127.4	992.1	24.5	28.3	948.5	2306.6	59.6	73.0	635.2	247.5	33.4	33.1
	FRAP	165.1	209.1	20.4	23.5	814.3	1032.6	55.3	61.5	526.7	179.8	30.2	23.5
	MT-GAD	105.0	104.9	20.8	23.4	715.6	706.6	48.6	49.9	87.2	95.2	22.6	24.3
Datasets		$SH1_{high}$		$SH1_{low}$		$SH2_{high}$		$SH2_{low}$		$SH3_{high}$		$SH3_{low}$	
Dynamic Lane		w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/	w/o	w/
Trad. Algo.	FixedTime	3499.6	-	1031.7	-	2340.8	-	722.1	-	4551.9	-	1058.8	-
	SOTL	2220.6	2182.2	225.5	195.4	1393.8	1221.8	86.0	84.6	3238.2	3237.2	210.8	209.8
RL Algo.	PressLight	114.9	1544.8	33.4	43.5	252.8	402.2	45.1	50.1	45.7	120.3	28.3	34.3
	CoLight	194.5	488.2	25.1	36.8	217.4	69.4	31.8	32.4	42.3	42.5	16.8	19.3
	FRAP	196.1	133.6	21.8	23.3	62.1	65.1	27.2	30.5	37.0	39.6	14.4	16.6
	MT-GAD	89.8	70.1	27.5	20.4	64.1	56.1	24.5	26.2	37.6	32.8	16.2	16.1

Table 1: The overall vehicle's average delay $\mathcal{D}$ in seconds of MT-GAD and its competitors.

Dataset	$SH2_{high}$	$SH3_{high}$	$SH1_{rotate}$
PressLight	3335.6 (8%)	899.0 (5%)	4269.5 (3%)
CoLight	2881.2 (8%)	373.2 (11%)	2510.2 (8%)
FRAP	510.4 (12%)	108.8 (54%)	1814.6 (71%)
MT-GAD *	1388.4 (4%)	85.2 (39%)	3312.2 (3%)
MT-GAD	**53.2 (106%)**	**40.3 (81%)**	**70.0 (100%)**

Table 2: Models' $\mathcal{D}$ in seconds with training dataset $SH1_{high}$ and different test datasets.

Dataset	$SH1_{high}$		$PT1_{high}$	
Metric	$\mathcal{D}_V$	# D.C.	$\mathcal{D}_V$	# D.C.
PressLight-DL	34.160	58.2	397.457	26.4
CoLight-DL	108.727	26.8	266.474	18.8
FRAP-DL	15.149	0.0	47.787	12.8
MT-GAD-0.8	15.807	17.8	35.746	33.6
MT-GAD-0.95	19.503	22.0	42.618	35.2
MT-GAD	**14.372**	20.2	**33.830**	18.7

Table 3: Delay of the dynamic lane ($\mathcal{D}_V$) in seconds and the number of direction changes (# D.C.).

because the large number of parameters could easily overfit the model. With group attention, our model achieves the best generalizability consistently.

Multi-timescale training evaluation. To demonstrate the effectiveness of our Multi-timescale training, we report the values of two parameters, $\mathcal{D}_V$ and #D.C. in Table 3. $\mathcal{D}_V$ is the delay of vehicles driving on L or S directions of a dynamic lane, for example, for $V_W \in \{L, S\}$, only vehicles in DTFs $F_{W,L}$ and $F_{W,S}$ are considered. #D.C. is the average number of direction changes the dynamic lane experiences during the experiment. With the environment setting, the maximum #D.C. is 80. Because of the space limitation, we only report their values under $SH1_{high}$ and $PT1_{high}$, while we do make similar observations in other intersections. Note that $\mathcal{D}_V$ is also relevant to the traffic signal control efficiency. Here MT-GAD-0.8 and MT-GAD-0.95 refer to two variants of MT-GAD that fix γ to 0.8 and 0.95 respectively, that means, we consider the traffic signal control and dynamic lane control using same timescale, either short (0.8) or long (0.95). We can find that existing RL methods fail to learn an effective strategy to control dynamic lanes, as they either never change the lane direction (i.e., #D.C. = 0) or keep changing the direction in some datasets. MT-GAD performs much better, but not MT-GAD-0.8 or MT-GAD-0.95. The latter two fail to learn direction change strategy in $PT1_{high}$. Although they do change dynamic lane directions, their $\mathcal{D}_V$ is much longer than that of MT-GAD, because with wrong timescale input, an agent can't learn strategies effectively.

The impact of traffic flow prediction. We adopt LSTM to predict the traffic flow. However, our model is not sensitive to the prediction accuracy, as long as it reaches a minimum expectation. We conduct a simulated traffic predictor $\mathcal{S}$, with the accuracy $a \in (0, 1]$ as an input. $\mathcal{S}$ takes real traffic prediction p_t from dataset, and calculates the output prediction p'_t as $p'_t = rp_t$, with $r \sim \mathcal{N}\left(0, (1-a)^{-2}\right)$. We observe that once a reaches 0.4, its impact on the model performance is not significant.

6 Conclusion

In this paper, we explore the traffic signal control problem with dynamic lanes. To the best of our knowledge, it's the first attempt to control dynamic lane and traffic signal together. We propose MT-GAD as a solution, that uses group attention to improve learning speed and generalizability, and adopts multi-timescale learning to find optimal strategies for both traffic signal control and dynamic lane control. Extensive experiments demonstrate that our method outperforms the state-of-the-art methods, with better generalizability. In the near future, we plan to explore other related directions, such as control of traffic signal and reversible lanes.

Acknowledgements

References

[Alcazar, 2020] Christian Alcazar. Traffic count data. https://data.lacounty.gov/Transportation/Traffic-Count-Data/uvew-g569, 2020. Accessed: 2021-05-12.

[Cools *et al.*, 2013] Seung-Bae Cools, Carlos Gershenson, and Bart D'Hooghe. *Self-Organizing Traffic Lights: A Realistic Simulation*, pages 45–55. Springer London, London, 2013.

[Gao *et al.*, 2017] Juntao Gao, Yulong Shen, Jia Liu, Minoru Ito, and Norio Shiratori. Adaptive traffic signal control: Deep reinforcement learning algorithm with experience replay and target network. *arXiv preprint arXiv:1705.02755*, 2017.

[Kim and Jeong, 2020] Daeho Kim and Okran Jeong. Cooperative traffic signal control with traffic flow prediction in multi-intersection. *Sensors*, 20(1):137, 2020.

[Li *et al.*, 2009] L. Li, Z. Qu, X. Song, and D. Wang. Research on variable lane signalized control method. In *ICMTMA*, volume 3, pages 575–578, 2009.

[Li *et al.*, 2013] Xu Li, Jun Chen, and Hao Wang. Study on flow direction changing method of reversible lanes on urban arterial roadways in china. *Intelligent and Integrated Sustainable Multimodal Transportation Systems Proceedings from COTA*, 96:807 – 816, 2013.

[Li *et al.*, 2016] Li Li, Yisheng Lv, and Fei-Yue Wang. Traffic signal timing via deep reinforcement learning. *IEEE/CAA Journal of Automatica Sinica*, 3(3):247–254, 2016.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[Moreira-Matias *et al.*, 2015] Luis Moreira-Matias, Michel Ferreira, and Joao Mendes-Moreira. Ecml/pkdd 15: Taxi trajectory prediction. https://www.kaggle.com/c/pkdd-15-predict-taxi-service-trajectory-i/data, 2015. Accessed: 2021-05-12.

[Oroojlooy *et al.*, 2020] Afshin Oroojlooy, MohammadReza Nazari, Davood Hajinezhad, and Jorge Silva. Attendlight: Universal attention-based reinforcement learning model for traffic signal control. In *NIPS*, 2020.

[Rizzo *et al.*, 2019] Stefano Giovanni Rizzo, Giovanna Vantini, and Sanjay Chawla. Time critic policy gradient methods for traffic signal control in complex and congested scenarios. In *SIGKDD*, pages 1654–1664, 2019.

[Song *et al.*, 2012] Renchu Song, Wei Lu, Weiwei Sun, Yan Huang, and Chunan Chen. Quick map matching using multi-core cpus. In *SIGSPATIAL*, pages 605–608, 2012.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017.

[Wei *et al.*, 2018] Hua Wei, Guanjie Zheng, Huaxiu Yao, and Zhenhui Li. Intellilight: A reinforcement learning approach for intelligent traffic light control. In *SIGKDD*, pages 2496–2505, 2018.

[Wei *et al.*, 2019a] Hua Wei, Chacha Chen, Guanjie Zheng, Kan Wu, Vikash Gayah, Kai Xu, and Zhenhui Li. Presslight: Learning max pressure control to coordinate traffic signals in arterial network. In *SIGKDD*, pages 1290–1298, 2019.

[Wei *et al.*, 2019b] Hua Wei, Nan Xu, Huichu Zhang, Guanjie Zheng, Xinshi Zang, Chacha Chen, Weinan Zhang, Yanmin Zhu, Kai Xu, and Zhenhui Li. Colight: Learning network-level cooperation for traffic signal control. In *CIKM*, pages 1913–1922, 2019.

[Wei *et al.*, 2019c] Hua Wei, Guanjie Zheng, Vikash Gayah, and Zhenhui Li. A survey on traffic signal control methods. *arXiv preprint arXiv:1904.08117*, 2019.

[Yao *et al.*, 2018] Ronghan Yao, Xiaotong Zhang, Na Wu, and Xianmin Song. Modeling and control of variable approach lanes on an arterial road: a case study of dalian. *Canadian Journal of Civil Engineering*, 45(11):986–1003, 2018.

[Yu *et al.*, 2020] Zhengxu Yu, Shuxian Liang, Long Wei, Zhongming Jin, Jianqiang Huang, Deng Cai, Xiaofei He, and Xian-Sheng Hua. Macar: Urban traffic light control via active multi-agent communication and action rectification. In Christian Bessiere, editor, *IJCAI*, pages 2491–2497. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Main track.

[Zang *et al.*, 2020] Xinshi Zang, Huaxiu Yao, Guanjie Zheng, Nan Xu, Kai Xu, and Zhenhui Li. Metalight: Value-based meta-reinforcement learning for traffic signal control. In *AAAI*, volume 34, pages 1153–1160, 2020.

[Zhang *et al.*, 2019] Huichu Zhang, Siyuan Feng, Chang Liu, Yaoyao Ding, Yichen Zhu, Zihan Zhou, Weinan Zhang, Yong Yu, Haiming Jin, and Zhenhui Li. Cityflow: A multi-agent reinforcement learning environment for large scale city traffic scenario. In *WWW*, pages 3620–3624, 2019.

[Zhang *et al.*, 2020] Rusheng Zhang, Akihiro Ishikawa, Wenli Wang, Benjamin Striner, and Ozan K Tonguz. Using reinforcement learning with partial vehicle detection for intelligent traffic signal control. *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[Zheng *et al.*, 2019] Guanjie Zheng, Yuanhao Xiong, Xinshi Zang, Jie Feng, Hua Wei, Huichu Zhang, Yong Li, Kai Xu, and Zhenhui Li. Learning phase competition for traffic signal control. In *CIKM*, pages 1963–1972, 2019.

[Zhou *et al.*, 2019] Lihua Zhou, Juanjuan Li, and Kangkang Ding. Research on variable lane control method based on traffic priority. In *AIIPCC*, New York, NY, USA, 2019.

[Zhu *et al.*, 2021] Liwen Zhu, Peixi Peng, Zongqing Lu, Xiangqian Wang, and Yonghong Tian. Variationally and intrinsically motivated reinforcement learning for decentralized traffic signal control. *arXiv preprint arXiv:2101.00746*, 2021.