DenseLight: Efficient Control for Large-scale Traffic Signals with Dense Feedback

Junfan Lin[1] , **Yuying Zhu**[1] , **Lingbo Liu**[2,3] , **Yang Liu**[1] , **Guanbin Li**[1] and **Liang Lin**[1]

[1]The School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China.
[2]Department Land Surveying and Geo-Informatic, The Hong Kong Polytechnic University, Hongkong.
[3]Smart Cities Research Institute, The Hong Kong Polytechnic University, Hong Kong
{linjf8, zhuyy76}@mail2.sysu.edu.cn, lingbo.liu@polyu.edu.hk, {liuy856,
liguanbin}@mail.sysu.edu.cn, linliang@ieee.org

Abstract

Traffic Signal Control (TSC) aims to reduce the average travel time of vehicles in a road network, which in turn enhances fuel utilization efficiency, air quality, and road safety, benefiting society as a whole. Due to the complexity of long-horizon control and coordination, most prior TSC methods leverage deep reinforcement learning (RL) to search for a control policy and have witnessed great success. However, TSC still faces two significant challenges. 1) The travel time of a vehicle is delayed feedback on the effectiveness of TSC policy at each traffic intersection since it is obtained after the vehicle has left the road network. Although several heuristic reward functions have been proposed as substitutes for travel time, they are usually biased and not leading the policy to improve in the correct direction. 2) The traffic condition of each intersection is influenced by the non-local intersections since vehicles traverse multiple intersections over time. Therefore, the TSC agent is required to leverage both the local observation and the non-local traffic conditions to predict the long-horizontal traffic conditions of each intersection comprehensively. To address these challenges, we propose DenseLight, a novel RL-based TSC method that employs an unbiased reward function to provide dense feedback on policy effectiveness and a non-local enhanced TSC agent to better predict future traffic conditions for more precise traffic control. Extensive experiments and ablation studies demonstrate that DenseLight can consistently outperform advanced baselines on various road networks with diverse traffic flows. The code is available at https://github.com/junfanlin/DenseLight.

1 Introduction

Reducing traffic congestion is an essential task for efficient modern urban systems. As the number of vehicles in the cities increases year by year, backward traffic coordination not only does harm to the driving experience but also aggravates air contamination with more harmful fuel emissions [Zhang and Batterman, 2013]. Alleviating traffic congestion by efficient

traffic signal control (TSC) [Mirchandani and Head, 2001] is one of the most practical and economical approaches [Taylor, 2002]. Specifically, TSC aims at coordinating the traffic lights of a road network to regulate the traffic flows to minimize the average travel time of vehicles. However, the traffic data collected from a road network is usually massive yet incomprehensible [Liu et al., 2020]. Therefore, the widely-adopted TSC strategies either fix signal routine [Roess et al., 2004] or adapt the traffic signal plans in real-time according to traffic flow patterns [Cools et al., 2013].

To automatically mine useful information from the massive traffic data, more and more studies leverage the powerful representation capability of deep neural networks [LeCun et al., 2015] to learn TSC agents [Van der Pol and Oliehoek, 2016; Zhang et al., 2021; Zheng et al., 2019; Oroojlooy et al., 2020; Wei et al., 2018; Wei et al., 2019a; D'Almeida et al., 2021] through the advanced reinforcement learning (RL) methods [Mnih et al., 2015; Schulman et al., 2017; Liang et al., 2021]. A proper reward function is required for applying RL to the TSC problem and improving the policy. Since the travel time is only feasible after a vehicle leaves the road network and fails to provide instant/dense feedback to the TSC policy in time during its journey, many previous works [Varaiya, 2013; Wei et al., 2019a; Wei et al., 2019b; Xu et al., 2021] draw the traditional characteristics of traffic intersections for the reward design, such as traffic pressure and queue length at the intersection. However, most of these heuristic rewards may be biased from the ultimate goal of TSC, i.e., the average travel time minimization. The introduced biases could lead the RL methods to adjust the TSC policy in an incorrect direction.

Apart from the reward design, it is also critical to endow the RL agents with the capability of precisely predicting the future dynamics of the environment to make well-founded decisions [Van Hasselt et al., 2016; Fujimoto et al., 2018; Yang et al., 2016; Lou et al., 2020]. However, it is non-trivial for an RL agent to capture the future traffic dynamics of the intersections in the context of TSC [Chen et al., 2022]. The future arriving vehicles of one intersection may be running in a **distant** intersection at present. To this end, the **local** observations of either the intersection itself (i.e., a snapshot of vehicles at the intersection) [Varaiya, 2013; Wei et al., 2019a; D'Almeida et al., 2021] or neighboring intersections [Wei et al., 2019b; Xu et al., 2021] might be insufficient to predict

the long-horizontal traffic dynamics of the intersection. Additionally, the location and traffic flow trend of an intersection also play a role in predicting the traffic dynamics of the intersection. For example, downtown intersections tend to be more crowded than suburban intersections. For example, a large number of vehicles entering an empty intersection may indicate the beginning of the rush hour. Therefore, the non-local observations and the location of the intersections, and historical traffic conditions are all important for a TSC agent to estimate future traffic situations more accurately.

To address the issues discussed above, we propose a novel RL-based TSC method named *DenseLight*, which improves traffic light coordination by exploiting dense information from both an unbiased reward and non-local intersection information fusion. Specifically, to provide dense and unbiased feedback for the policy improvement, we propose an equivalent substitute for travel time, i.e., the gap between the ideal traveling distance (i.e., the ideal distance a vehicle could have traveled at the full speed during its journey) and the actual traveling distance during the whole journey, namely Ideal-Factual Distance Gap (IFDG). Since the length of the factual journey of a vehicle is fixed according to its traveling lanes of the road network, minimizing IFDG is equivalent to minimizing travel time. Most importantly, IFDG can be calculated at each intersection and at each timestep. Therefore, IFDG can also provide instant feedback on the effectiveness of the control policy at each intersection.

Besides an unbiased and dense reward, DenseLight also features a *Non-local enhanced Traffic Signal Control* (NL-TSC) agent to benefit TSC from the spatial-temporal augmented observation and the non-local fusion network architecture. Specifically, the NL-TSC agent supplements the original observation of an intersection with its location information and the previous observation so that each intersection can customize its own signal plans w.r.t. historical and spatial information. As for facilitating the TSC agents with a better awareness of the future traffic dynamics affected by other intersections, a non-local branch is proposed to enhance the local features of each intersection with the features of the non-local intersections. By learning to communicate the non-local information across non-local intersections, the NL-TSC agent can better predict the long horizontal traffic condition of each intersection and can thus make better coordination at present.

Overall, our contributions are three-fold: **1)** we propose a novel RL-based TSC method, i.e., DenseLight, which is optimized by an unbiased and dense reward termed IFDG; **2)** to better model the future accumulated IFDG of each intersection, we develop the NL-TSC agent, effectively gathering spatial-temporal features of each intersection and propagating the non-local intersection information to improve the multi-agent RL policy; **3)** comprehensive experiments conducted on different real-world road networks and various traffic flows show that DenseLight can consistently outperform traditional and advanced RL-based baselines.

2 Related Works

Conventional traffic signal control methods. Traditional traffic signal control methods [Little *et al.*, 1981; Roess *et* *al.*, 2004; Koonce and Rodegerdts, 2008] typically set traffic signal plans with fixed cycle lengths, phase sequences, and phase splits. They heavily relied on expert knowledge and hand-crafted rules. Adaptive traffic signal control methods formulated the task as an optimization problem and made decisions according to the pre-defined signal plans and real-time data [Hunt *et al.*, 1981; Luk *et al.*, 1982; Mirchandani and Head, 2001; Cools *et al.*, 2013; Hong *et al.*, 2022]. Recently, researchers have made progress by comparing the requests from each phase through different traffic representations [Varaiya, 2013; Zhang *et al.*, 2021].

RL-based traffic signal control methods. For individual intersection signal control, some studies investigated RL environmental settings [Van der Pol and Oliehoek, 2016; Wei *et al.*, 2018; Wei *et al.*, 2019a; Zhang *et al.*, 2021], agent network design [Zheng *et al.*, 2019], and policy transfer learning [Zang *et al.*, 2020; Oroojlooy *et al.*, 2020] to optimize the average travel time of all vehicles. Since the travel time can not be obtained until a vehicle leaves the road network, these methods presumed some objectives to be equivalent to maximizing the traffic throughput, thus minimizing travel time. For example, [Van der Pol and Oliehoek, 2016] utilized a combination of vehicle delay and waiting time as the reward, and PressLight [Wei *et al.*, 2019a] introduced the pressure of intersection to be a new objective. However, these optimization objectives are not always aligned with the minimization of the average travel time. To resolve this conflict, [D'Almeida *et al.*, 2021] defined an instant time loss for vehicles to guide the next action selection. This reward is the average loss of time at the moment when selecting traffic signals. Therefore, time loss is not able to reflect the later change in traffic congestion caused by the newly selected signals. Differently, our IFDG integrates speed over time after selecting a signal, and thus IFDG can reflect the effect of the signal.

Traffic signal coordination. To coordinate multiple intersections, a centralized controller could select actions for all intersections [Prashanth and Bhatnagar, 2010; Tan *et al.*, 2019; Van der Pol and Oliehoek, 2016] but may suffer from the curse of dimension in combinatorially large action space. Therefore, some studies modeled single agents in a decentralized way and scaled up to multi-intersection settings by parameter sharing [Chen *et al.*, 2020]. However, ignoring the influence of other intersections may affect the objective optimization above the whole network. Some studies used centralized training and decentralized execution [Rashid *et al.*, 2018; Son *et al.*, 2019; Wang *et al.*, 2020; Tan *et al.*, 2019]; some gathered neighboring information by concatenating neighborhood observations [Chu *et al.*, 2019; Liu *et al.*, 2022] or communicating through graph neural network [Wei *et al.*, 2019b; Guo *et al.*, 2021; Zeng, 2021; Huang *et al.*, 2021]. However, these methods either treated other intersections equally or only considered the neighborhood, ignoring useful information from distant intersections.

3 Preliminaries

3.1 Traffic Signal Control

Conventionally, a TSC task $\mathcal{D}$ includes a road network and a set of vehicle flows. A road network is usually composed

of N multiple intersections $I = [1, ..., N]$ which can regulate the vehicles in their entering lanes via selecting different traffic signals. Specifically, an **entering lane** of an intersection is a lane where the vehicles enter the intersection from the north, south, west, or east direction. We use L_i^{in} to represent all the entering lanes of the i-th intersection. And the **exiting lanes** L_i^{out} are the downstream lanes of the i-th intersection where the vehicles leave the intersection. A **traffic movement** is the traffic moving from an entering lane $l_{i,j}^{\text{in}} \in L_i^{\text{in}}$ to an exiting lane $l_{i,k}^{\text{out}} \in L_i^{\text{out}}$, which is generally categorized as a left turn, through and right turn. We define a combination of non-conflicting traffic movements as a **phase** a. An intersection with phase a gives the corresponding traffic movements the priority to pass through under a green light while prohibiting the others. According to the rules in most countries, right-turn traffic is allowed to pass unrestricted by signals. By convention [Zhang *et al.*, 2021], we only consider a 4-way intersection, resulting in 12 movements and 4 candidate phases.

The goal of TSC is to learn a control policy π_θ parameterized by θ for all intersections in the road network to minimize the average travel time of all the vehicles X in a road network, within a finite period T_{TSC}, i.e. $\min_\theta \frac{1}{|X|} \sum_{x \in X} x_l - x_e$, where x_e and x_l are the moments the vehicle x enters and leaves the road network, respectively. Following the conventional setting in [Zheng *et al.*, 2019; Zhang *et al.*, 2021], each phase persists for a fixed phase duration T_{phase}. To avoid confusion, we use d to denote the d-th decision, so that $t_{d+1} - t_d = T_{\text{phase}}$. We use D to denote the total number of decision-making steps in each TSC episode, where $T_{\text{TSC}} = D \times T_{\text{phase}}$. We use $v_t(x)$ to stand for the speed of vehicle x at time t, and v_{max} to represent maximum speed.

3.2 Reinforcement Learning for TSC

In this paper, we consider a standard RL framework where an agent selects the phase for each intersection after every T_{phase} seconds. At t_d, a TSC agent observes the intersection state s_d from an intersection. Then, the agent chooses a phase a_d according to the policy π. The environment returns the reward r_d and the next observation $s_{d+1} \sim \mathcal{T}(s_d, a_d)$, where $\mathcal{T}$ is the environment transition mapping. The return of a trajectory after the d-th phase is $\eta_d = \sum_{k=0}^{\infty} \gamma^k r_{d+k}$, where $\gamma \in [0, 1)$ is the discount factor. RL aims to optimize the policy to maximize the expected returns of future trajectories after observing s_d.

Elements in observation and reward of TSC. In most previous TSC methods, observations and rewards originated from the traffic statistic characteristics of intersections, including the following examples. **1) Pressure**: the difference between the number of upstream and downstream waiting vehicles (with a speed less than 0.1m/s) of the intersection in a time step; **2) Queue length**: the average number of the waiting vehicles in the incoming lanes in a time step; **3) Time loss**: sum of the time delay of each vehicle, i.e., $1 - v_{t_d}/v_{\text{max}}$, at the moment of phase selection t_d; **4) Step-wise travel time**: the local travel time at each intersection within the duration T_{phase}. As proved in Appendix, step-wise travel time is an unbiased reward for TSC, however, provides sparse feedback for each intersection at each time step.
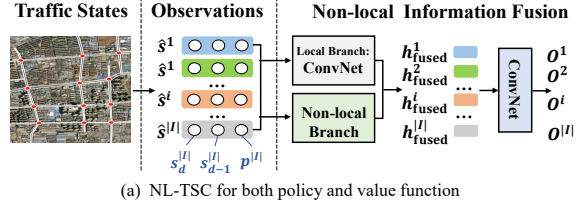
(a) NL-TSC for both policy and value function
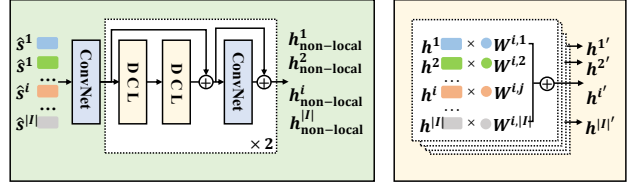
(b) Non-local Branch

(c) Dense Communication Layer (DCL)

Figure 1: The policy/value estimator architecture of NL-TSC agent. Based on the observations from the road network, an NL-TSC agent predicts action distribution/value through a local branch and a non-local branch to realize non-local information fusion. The local branch adopts a ConvNet that extracts features of each intersection using its own observation alone; the non-local branch consists of Dense Communication Layers (DCL) and ConvNets, integrating non-local features into the representation of each intersection.

4 DenseLight

In our paper, we propose a novel RL-based TSC method, namely, DenseLight, which contains two key ingredients, i.e., *Ideal-Factual Distance Gap* reward and *Non-local enhanced Traffic Signal Control* agent, as shown in Fig. 1. In this section, we elaborate on each of them in detail.

4.1 Ideal-Factual Distance Gap

As mentioned in Sect. 3, the goal of TSC is to search for a traffic signal control policy to minimize the average travel time of vehicles. Before RL takes place in TSC, traditional methods, such as MaxPressure [Varaiya, 2013], have been developed and widely adopted in real-world road networks. The elements of traditional TSC methods like pressure and queue length have inspired the later development of RL-based methods [Wei *et al.*, 2019a; Wei *et al.*, 2019b; Chen *et al.*, 2020]. By trivially adopting traditional elements as a negative reward, prior RL-based methods have witnessed a great improvement upon the traditional methods by a significant margin. However, these heuristic rewards are usually designed empirically and not derived from the target. Optimizing with these rewards might hamper the RL agent from improving in the correct direction.

To develop an unbiased and dense reward function for TSC, we propose the *Ideal-Factual Distance Gap* (IFDG) reward function. As the name suggests, IFDG characterizes the gap between the ideal distance that the vehicles could have traveled at full speed and the factual distance affected by the phase decision of the RL agent. Ideally, if vehicles are running at full speed, IFDG will be zero. We use the negative IFDG as the reward $r_{\text{IFDG},d}^i$ for i-th intersection received after d-th decision, and $x^i \in X^i$ to represent each vehicle at i-th

intersection. Formally,

$$r_{\text{IFDG},d}^i = - \sum_{\forall x^i : t_d < x_l^i \le t_{d+1}} \int_{\max\{t_d, x_e^i\}}^{x_l^i} v_{\max} - v_t(x^i)\mathrm{d}t \tag{1}$$

$$- \sum_{\forall x^i : (t_{d+1} < x_l^i) \wedge (x_e^i \le t_{d+1})} \int_{\max\{t_d, x_e^i\}}^{t_{d+1}} v_{\max} - v_t(x^i)\mathrm{d}t.$$

According to the Eq. (1), the vehicles that either stay or leave the intersection will contribute their distance gaps to the reward. And since the distance gap is sensitive to traffic congestion, $r_{\text{IFDG},d}^i$ can effectively reflect the difference in different degrees of traffic congestion caused by different phases. More importantly, IFDG turns out to be an equivalent substitute for travel time:

$$\sum_{d=0}^{D-1} \sum_{i \in I} r_{\text{IFDG},d}^i = - \sum_{x \in X} \int_{x_e}^{x_l} (v_{\max} - v_t(x))\mathrm{d}t$$

$$= \sum_{x \in X} \int_{x_e}^{x_l} v_t(x)\mathrm{d}t - \sum_{x \in X} v_{\max} \cdot (x_l - x_e)$$

$$= \text{constant} - v_{\max} \sum_{x \in X} (x_l - x_e), \tag{2}$$

where the first equation is the sum of the IFDG of all vehicles. The $\sum_{x \in X} \int_{t=x_e}^{x_l} v_t(x)\mathrm{d}t$ in Eq. (2) is the sum of the traveling distance of all vehicles, which is a constant since each vehicle has a fixed traveled path in our task. And the right-hand side of Eq.(2) is linear correlative with the travel time. Thus, our IFDG provides dense feedback for RL improvement and guides the improvement in an unbiased direction.

4.2 Non-local Enhanced TSC Agent

Considering the Markov property in RL, the decision-making and the value estimation of an agent require the observation of RL policy to be self-contained. Due to the complex relations between intersections of TSC, it is not sufficient to take the local observation of the intersection to estimate the policy's future value. In this paper, we propose an *Non-local enhanced Traffic Signal Control* (NL-TSC) agent, which takes advantage of spatial-temporal augmented observation and non-local traffic information fusion to make a better decision and estimate a more accurate value for each intersection.

Spatial-temporal augmented observation. In a road network, the dynamics of traffic congestion at an intersection generally vary across different areas and different periods. For example, the traffic congestion in downtown intersections is usually heavier than that in other areas. And the traffic tends to be more crowded at the beginning of the rush hour and more smooth at the end of it. Therefore, the TSC agent requires not only the current traffic conditions observed in each intersection but also the location and the tendency of the traffic congestion of an intersection to make more comprehensive decisions. To this end, we propose a spatial-temporal augmented observation for i-th intersection by concatenating the original observation s_d^i at time t with its position encoding

p^i and its previous observation s_{d-1}^i:

$$\hat{s}_d^i = \text{concatenate}\{s_d^i, s_{d-1}^i, p^i\}, \tag{3}$$

where concatenate$\{...\}$ puts all vectors in the bracket into one vector and p^i is a 2-D position encoding implemented as [Dosovitskiy *et al.*, 2020]. Including the position information allows NL-TSC agents of each intersection to make decisions according to their locations. Moreover, augmented with the previous observation, the changes in traffic conditions can be captured to better predict future traffic dynamics.

Non-local information fusion. On the TSC task, vehicles arriving at one intersection may come from either nearby or distant intersections. In this sense, the relations between intersections are complex and entangled. To provide sufficient information to estimate long-horizontal values, we propose a non-local branch that can automatically determine what information to communicate among non-local intersections. Specifically, the NL-TSC agent first extracts features for each intersection i by **a stack of multiple** 1×1 **convolutional layers** (abbr. ConvNet) (Eq. (4)). After that, the non-local branch propagates the features of non-local-intersections by a *Dense Communication Layers* (DCL) which is parameterized by $W \in \mathbb{R}^{|I| \times |I|}$, as shown in Fig. 1(c). The features of non-local intersections are fused and integrated into the original representations of each intersection according to Eq. (5). Then, the final non-local features $h_{\text{non-local}}^i$ for each intersection are processed by another ConvNet, as formulated in Eq. (6).

$$h^i = \text{ConvNet}_{\text{embed}}(\hat{s}^i). \tag{4}$$

$$h^{i'} = h^i + W^i \times [h^1, h^2, ..., h^{|I|}]. \tag{5}$$

$$h_{\text{non-local}}^i = h^{i'} + \text{ConvNet}_{\text{process}}(h^{i'}). \tag{6}$$

However, as the size of the road network increases, W tends to be over-parameterized and degrades the expression ability of the extracted non-local features. To address this problem, we use two consecutive DCLs with two smaller parameters $W_a \in \mathbb{R}^{|I| \times M}$ and $W_b \in \mathbb{R}^{M \times |I|}$ to replace the original DCL with W, so that $W = W_a \times W_b$, as demonstrated in Fig. 1(b). The above operations (Eq. (5) and (6)) are repeated twice to better fuse the non-local information. To allow the model to automatically fuse local and non-local information, the NL-TSC agent also extracts the local features focusing on the intersection itself:

$$h_{\text{local}}^i = \text{ConvNet}_{\text{local}}(\hat{s}^i). \tag{7}$$

Finally, both local and non-local features are forwarded to the final ConvNet to predict the categorical action distribution or the future policy value for each intersection:

$$h_{\text{fused}}^i = \text{concatenate}\{h_{\text{non-local}}^i, h_{\text{local}}^i\}, \tag{8}$$

$$o^i = \text{ConvNet}_{\text{final}}(h_{\text{fused}}^i). \tag{9}$$

o^i can be either the action logits $\log \pi_\theta(a|\hat{s})$ or the value $V_\phi(\hat{s})$ where θ and ϕ are the networks parameters of the policy and value estimator, respectively. The structure of either policy or value estimator of the NL-TSC agent is sketched in Fig. 1(a). As for RL-optimization, we adopt the well-known and stable policy gradient algorithm, proximal policy optimization (PPO-Clip) [Schulman *et al.*, 2017].

Road Network	Traffic Flow	Arrival Rate (vehicles/300s)			
		Mean	Std.	Max	Min
Jinan12	$\mathcal{D}_{\text{JN12}}$	526.64	86.70	676	256
	$\mathcal{D}_{\text{JN12(2)}}$	363.61	64.10	493	233
	$\mathcal{D}_{\text{JN12(f)}}$*	639.11	177.63	1036	281
Hangzhou16	$\mathcal{D}_{\text{HZ16}}$	250.71	38.20	335	208
	$\mathcal{D}_{\text{HZ16(f)}}$	572.48	293.39	1145	202
NewYork48	$\mathcal{D}_{\text{NY48}}$	236.42	8.13	257	216
NewYork196	$\mathcal{D}_{\text{NY196}}$	909.90	71.22	1013	522

Table 1: Data statistics of the traffic flow datasets. Dataset with the symbol * is generated based on the real data with an increasing volume of vehicles, and the others are open real-world datasets.

5 Experiments

5.1 Experiment Setting

The experimental environment is simulated in CityFlow [Zhang *et al.*, 2019], which simulates vehicles behaviors every second, provides massive observations of the road network for the agents, and executes the TSC decisions from the agents. Following the existing studies [Wei *et al.*, 2019a; Zhang *et al.*, 2021], each episode is a 3600-second simulation (i.e., $T_{\text{TSC}} = 3600$), and the action interval T_{phase} is 15 seconds, then the number of decision-making steps, i.e., D, is $3600/15 = 240$. By convention, a three-second yellow signal is set when switching from a green signal to a red signal.

Datasets. Experiments use four real-world road networks. **i) Jinan12**: with 4×3 (4 rows, 3 columns) intersections in Jinan, China; **ii) Hangzhou16**: with 4×4 intersections in Hangzhou, China; **iii) NewYork48**: with 16×3 intersections in New York, USA; **iv) NewYork196**: with 28×7 intersections in New York, USA. For each road network, the corresponding traffic flow data are collected and processed from multi-sources with detailed statistics recorded in Tab. 1. To evaluate the performance of the TSC methods on various traffic conditions, we synthesize a novel dataset $\mathcal{D}_{\text{JN12(f)}}$ with high average traffic volumes and high variances (which indicates the fluctuating traffic flow) by randomly re-sampling vehicles from the real traffic flow data and adding them back to the original data with re-assigned enter time.

Baselines. To evaluate the effectiveness of our method, we compare our DenseLight with the traditional and RL-based approaches. The results of the baselines are obtained by re-running their corrected open codes. *The traditional methods* are described in the following. **Fixed-Time [Koonce and Rodegerdts, 2008]:** Fixed-Time control consists of pre-defined phase plans that are fixed in duration; **MaxPressure [Varaiya, 2013]:** an adaptive policy that greedily selects the phase with the maximum pressure; **Efficient-MP [Wu *et al.*, 2021]:** an adaptive policy that selects the phase with the maximum efficient pressure. Efficient pressure is the difference of average queue length between the upstream and downstream of each traffic movement; **Advanced-MP [Zhang *et al.*, 2021]:** an adaptive method that defines the number of running vehicles within an effective

range near the intersection as the request of the current phase, and defines the pressure as the requests of other phases. The policy selects the phase with the maximum request. And the *advanced RL-based methods* are as follows. **FRAP [Zheng *et al.*, 2019]:** FRAP designs a network for traffic signal control that models the competition relations between different phases based on the demand prediction for each phase with queue length as the reward; **MPLight [Chen *et al.*, 2020]:** MPLight integrates pressure into the observation and uses the reward and agent as FRAP [Zheng *et al.*, 2019]; **CoLight [Wei *et al.*, 2019b]:** CoLight uses a graph attention network to learn the influences from neighboring intersections and adopts the length of vehicles in entering lanes as the reward; **Advanced-MPLight [Zhang *et al.*, 2021]:** Based on MPLight, Advanced-MPLight uses the current phase and the advanced traffic states (including efficient pressure and the number of vehicles within an effective range) as an intersection observation; **Advanced-CoLight [Zhang *et al.*, 2021]:** Advanced-CoLight adds the advanced traffic states to the observation of CoLight; **TimeLoss-FRAP [D'Almeida *et al.*, 2021]:** TimeLoss-FRAP uses the FRAP [Zheng *et al.*, 2019] agent with the instant time loss of vehicles in the entering lanes as the observation and reward; **DenseLight (Ours):** The component of observation s_d^i at the intersection i and step d is the same as Advanced-MPLight. DenseLight is optimized with IFDG reward and NL-TSC agent.

Training details. As for DenseLight, the training hyperparameters follow the defaults in PPO-Clip [Schulman *et al.*, 2017] implemented in the open source platform [Weng *et al.*, 2021]. In brief, the number of episodes in $\mathcal{B}_k$ for $k \in [0, 500)$ is 2. The size of all hidden layers of neural networks is 64. Both the local branch of NL-TSC and ConvNet$_{\text{process}}$ are ConvNets with 2 1×1 convolutional layers. Other ConvNets in NL-TSC are single 1×1 convolutional layers. The size of the training batch is 64. The learning rate is 3e-4 at the beginning and linearly decays to 0. M of W_a and W_b is set as $|I|$. For $\mathcal{D}_{\text{NY196}}$, the batch size is set as 16 to reduce the memory occupancy, M is set as $0.1 \times |I|$ to prevent over-parameterization, and the 28×7 road network is divided into four 7×7 road networks to reduce the dimension of the state. These measures are adopted to ease the optimization without causing conflicts with our contributions. The dimension of position encoding is 16. Following the conventional settings in [Chen *et al.*, 2020; Zhang *et al.*, 2021], all results in the tables of experiments are the average of the final 10 evaluation episodes.

5.2 Results

We test each method in 60-minute simulations separately and report the average performance of the last ten episodes. We have following findings from the comparison results in Tab. 2.

Consistent improvement. As shown in Tab. 2, our DenseLight achieves consistent improvement by a clear margin across different road networks and traffic flows. Particularly, DenseLight hits a travel time of 800 seconds on $\mathcal{D}_{\text{NY196}}$ while the previous state-of-the-art result is about 1000 seconds, saving about 20% of the time. Another significant improvement can be observed under $\mathcal{D}_{\text{HZ16(f)}}$ where DenseLight achieves nearly 10% improvement over the previous best result.

Method	$\mathcal{D}_{JN12}$	$\mathcal{D}_{JN12(2)}$	$\mathcal{D}_{JN12(f)}$	$\mathcal{D}_{HZ16}$	$\mathcal{D}_{HZ16(f)}$	$\mathcal{D}_{NY48}$	$\mathcal{D}_{NY196}$
Fixed-Time	415.63	351.84	413.24	475.44	393.92	1070.45	1506.85
MaxPressure	259.87	229.74	279.88	268.37	336.78	775.63	1178.71
Efficient-MP	256.00	224.45	277.20	264.25	315.38	293.27	1122.32
Advanced-MP	241.43	225.71	262.32	263.49	309.29	197.84	1060.36
FRAP	273.50	249.35	298.99	287.07	352.20	180.04	1241.54
MPLight	278.27	251.39	303.36	297.80	349.29	1841.86	1909.92
CoLight	252.87	235.09	278.59	276.08	329.56	168.12	988.78
Advanced-MPLight	239.10	219.21	250.36	255.06	306.25	1617.80	1339.09
Advanced-CoLight	232.16	217.20	251.21	251.44	303.15	160.25	1004.52
TimeLoss-FRAP	234.34	221.90	249.62	258.47	321.06	896.25	1143.90
DenseLight (Ours)	**226.97**	**215.82**	**239.58**	**248.43**	**272.27**	**156.30**	**803.42**

Table 2: The average travel time results of baselines across different traffic flow datasets in different road networks.
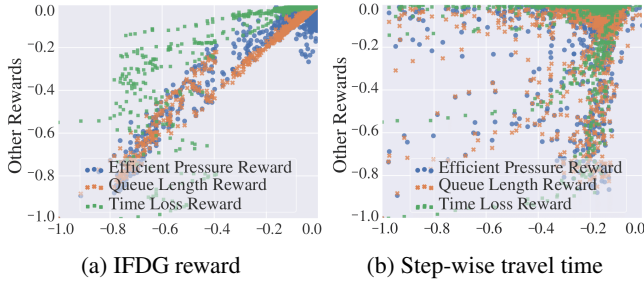

(a) IFDG reward (b) Step-wise travel time

Figure 2: Relations between the normalized IFDG reward/step-wise travel time reward and other normalized rewards.

(a) $\mathcal{D}_{HZ16(f)}$ (b) $\mathcal{D}_{NY195}$

Figure 3: Errors of the value prediction with different components.

Generalization to various traffic flows. We observe that under two existing flow data $\mathcal{D}_{JN12}$ and $\mathcal{D}_{JN12(2)}$ with the Jinan12 road network, different methods have close performances. However, on another similar road network, i.e., Hangzhou16, DenseLight can obtain a more significant improvement under $\mathcal{D}_{HZ16(f)}$. The potential reason is that the traffic flow in $\mathcal{D}_{HZ16(f)}$ shows a more diverse pattern than that of the Jinan12 road network. As shown in Tab. 1, the numbers of arriving vehicles in every 300 seconds in $\mathcal{D}_{HZ16(f)}$ vary considerably. To further verify this point, we have designed a fluctuating flow $\mathcal{D}_{JN12(f)}$ with its detailed statistics presented in Tab. 1. From the results of $\mathcal{D}_{JN12(f)}$, we observe that the improvement of DenseLight becomes more evident.

5.3 Ablations and Analyses

In this part, we sequentially add modules for ablation.

Rewards

To evaluate the effect of reward, we compare our IFDG reward with other rewards. In the experiments, all the elements except for the reward are strictly controlled to be the same. The upper part of Tab. 3 shows that our IFDG obtains the best performance. As the size of the road network and the variance of flows increase, the advantage of IFDG becomes more significant. Especially on $\mathcal{D}_{HZ16(f)}$ and $\mathcal{D}_{NY196}$, our IFDG can obtain the most clear improvements. The potential reason could be that IFDG not only aligns with the ultimate goal as deduced in Eq. (1), but also carries dense information about
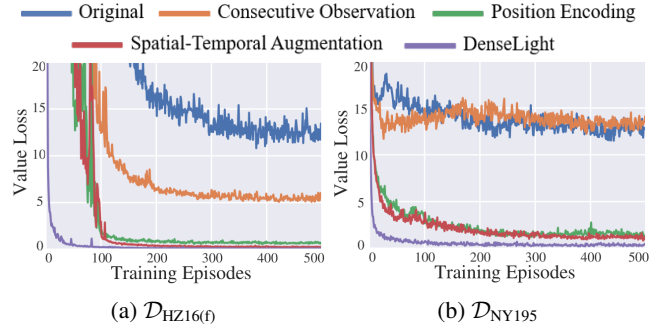
traffic congestion of different degrees. To intuitively demonstrate that our IFDG carries more dense information about the intersection than step-wise travel time, we show their associations with other dense rewards, respectively in Fig. 2a and Fig. 2b. We observe that IFDG is positively associated with the other dense rewards while step-wise travel time is not.

Spatial-Temporal Augmentation

To investigate the individual effect of position encoding and consecutive observation separately, we conduct experiments that concatenate the original observation with either position encoding (P) or consecutive observation (C). As shown in the middle part of Tab. 3, in comparison to the results of the "Ideal-Factual Distance Gap" which only uses the original observation, we find that TSC agents can obtain further improvements with either spatial or temporal information. Especially with position encoding (P), TSC agents can achieve consistent gains. However, we also discover that in the case of $\mathcal{D}_{NY196}$, the performance degrades with consecutive observation alone while using both information does not witness the same degradation. To this end, we suppose the potential reason is that in a large road network, the location of the intersection plays a critical role in predicting future traffic congestion. And the consecutive observations without the context of the location of the intersection, i.e., Consecutive Observation (C) in Tab. 3, might be a piece of misleading information. We visualize the curves of future congestion estimation losses with different components in Fig. 3. In Fig. 3b, the future traffic

Component	Method	$\mathcal{D}_{JN12}$	$\mathcal{D}_{JN12(2)}$	$\mathcal{D}_{JN12(f)}$	$\mathcal{D}_{HZ16}$	$\mathcal{D}_{HZ16(f)}$	$\mathcal{D}_{NY48}$	$\mathcal{D}_{NY196}$
Reward	EfficientPressure [Zhang *et al.*, 2021]	230.31	217.30	244.50	251.03	334.55	1556.77	1106.46
	Queue Length [Wei *et al.*, 2019b]	229.76	217.30	247.25	250.53	293.42	162.43	1026.24
	TimeLoss [D'Almeida *et al.*, 2021]	233.88	217.07	257.78	250.49	313.18	161.67	1049.19
	Step-wise Travel Time	230.10	217.78	244.47	253.16	288.41	162.91	1056.36
	Ideal-Factual Distance Gap	229.60	216.77	243.50	250.38	276.75	161.02	963.21
S.T. Aug.	Consecutive Observation (C)	229.04	217.08	242.71	250.30	274.53	160.90	1052.93
	Position Encoding (P)	228.73	216.70	242.29	249.61	276.48	159.59	954.99
	Spatial-temporal Augmentation (C+P)	228.20	216.31	240.03	249.38	274.16	159.50	953.48
Non-local	1-hop	228.17	216.12	240.21	249.17	288.81	157.78	926.85
	2-hop	228.23	216.31	240.36	249.10	284.52	158.18	931.62
	Transformer [Vaswani *et al.*, 2017]	227.79	216.12	240.44	249.14	274.04	158.66	917.25
	DenseLight	**226.97**	**215.82**	**239.58**	**248.43**	**272.27**	**156.30**	**803.42**

Table 3: The average travel time results of ablation studies, ranging from rewards to agent components. For briefness, we use S.T. Aug. to stand for spatial-temporal augmentation.

congestion prediction is worse with only consecutive observations. Moreover, from Fig. 3, we can observe that the patterns of estimation losses are in accordance with the results in Tab. 3, which implies that the average travel time has an association with the accuracy of value estimations. This is because, by using IFDG as a reward, the value is the expectation of the discounted accumulated ideal-factual distance gap, positively associated with the average travel time. This observation also justifies the benefit of our IFDG reward as an unbiased measurement of the average travel time.

Non-Local Fusion

As shown in Tab. 3, adding the non-local branch achieves consistent improvement on all tasks. Especially under $\mathcal{D}_{NY196}$, non-local information enhancement results in a breakthrough in that the average travel time approaches 800 seconds. To comprehensively analyze the advantages of our non-local branch, we conduct analyses from three aspects including *the accuracy of future traffic conditions*, *communication mechanisms* and *dense communication layers*.

The accuracy of future traffic conditions. From Fig. 3, we observe that, by adding a non-local branch, the value losses of DenseLight obtain further improvements in comparison with others. Especially on tasks $\mathcal{D}_{HZ16(f)}$ and $\mathcal{D}_{NY196}$, the value losses when using non-local branch drop significantly, which is consistent with the results in Tab. 3 (the last row of "S.T. Aug." part v.s. "DenseLight"). Given the positive association between value using IFDG and future average travel time, this observation highlights the importance of aggregating non-local traffic conditions in predicting the long-horizon future traffic condition of each intersection.

Non-local communication. To highlight the advantage of learning to propagate and integrate information from non-local intersections against manually specified communication among neighboring intersections, we include two sets of experiments that use fixed attention weights, i.e., $W_{1\text{-hop}}$ and $W_{2\text{-hop}}$, in the dense communication layer. $W_{1\text{-hop}}$, for example, assigns non-zero $\frac{1}{|I_{1\text{-hop}}^i|}$ to the $W_{1\text{-hop}}^{i,j}, \forall j \in I_{1\text{-hop}}^i$, where

$I_{n\text{-hop}}^i$ includes intersections that can be reached at most n steps from the intersection i. As shown in Tab. 3 ("1-hop", "2-hop" and "DenseLight"), with the learnable W, Dense-Light can outperform both $W_{1\text{-hop}}$ and $W_{2\text{-hop}}$ consistently.

The effectiveness of dense communication layer. To investigate the effectiveness, we modify our DenseLight by replacing the non-local branch with a transformer encoder [Vaswani *et al.*, 2017; Wang *et al.*, 2022]. The transformer is a well-known architecture to capture the global relationships among language tokens/image patches. From the results in Tab. 3, DenseLight+Transformer can outperform DenseLight without a non-local branch consistently and has better results against both 1-hop and 2-hop in most cases. This observation again justifies the benefit of aggregating non-local information in solving TSC problems. However, efficient as the transformer is, it is usually overparameterized (the total size of parameters is about 2.8 MB) and computation-heavy, making RL difficult to improve. Different from the transformer, our non-local branch only uses a linear weighted sum to aggregate non-local information, resulting in a more slim model (the total size of parameters is about 0.2 MB) and an easier RL optimization process.

6 Conclusion

In this paper, we propose a novel method DenseLight for a multi-intersection traffic signal control problem (TSC) based on deep reinforcement learning (RL). Specifically, Dense-Light optimizes the average travel time of vehicles in the road network under the guidance of an unbiased and dense reward named Ideal-Factual Distance Gap (IFDG) reward and further benefits the future accumulated IFDG modeling by a Non-local enhanced TSC (NL-TSC) agent through spatial-temporal augmented observation and non-local information fusion. We conduct comprehensive experiments on several real-world road networks and various traffic flows, and the results demonstrate consistent performance improvement. In the future, we will extend our method to learn policies with generalization ability over different road networks.

Acknowledgments

Contribution Statement

Junfan Lin and Yuying Zhu have equal contributions. Yang Liu is the corresponding author. All of the authors have contributed to the experiments and writing of this work.

References

[Chen *et al.*, 2020] Chacha Chen, Hua Wei, Nan Xu, Guanjie Zheng, Ming Yang, Yuanhao Xiong, Kai Xu, and Zhenhui Li. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. In *AAAI*, volume 34, pages 3414–3421, 2020.

[Chen *et al.*, 2022] Changlu Chen, Yanbin Liu, Ling Chen, and Chengqi Zhang. Bidirectional spatial-temporal adaptive transformer for urban traffic flow forecasting. *TNNLS*, 2022.

[Chu *et al.*, 2019] Tianshu Chu, Jie Wang, Lara Codecà, and Zhaojian Li. Multi-agent deep reinforcement learning for large-scale traffic signal control. *TITS*, 21(3):1086–1095, 2019.

[Cools *et al.*, 2013] Seung-Bae Cools, Carlos Gershenson, and Bart D'Hooghe. Self-organizing traffic lights: A realistic simulation. In *Advances in applied self-organizing systems*, pages 45–55. Springer, 2013.

[D'Almeida *et al.*, 2021] Marcelo D'Almeida, Aline Paes, and Daniel Mossé. Designing reinforcement learning agents for traffic signal control with the right goals: a time-loss based approach. In *ITSC*, pages 1412–1418. IEEE, 2021.

[Dosovitskiy *et al.*, 2020] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[Fujimoto *et al.*, 2018] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *ICML*, pages 1587–1596. PMLR, 2018.

[Guo *et al.*, 2021] Xin Guo, Zhengxu Yu, Pengfei Wang, Zhongming Jin, Jianqiang Huang, Deng Cai, Xiaofei He, and Xiansheng Hua. Urban traffic light control via active multi-agent communication and supply-demand modeling. *TKDE*, pages 1–1, 2021.

[Hong *et al.*, 2022] Wanshi Hong, Gang Tao, Hong Wang, and Chieh Wang. Traffic signal control with adaptive online-learning scheme using multiple-model neural networks. *TNNLS*, 2022.

[Huang *et al.*, 2021] Hao Huang, Zhiqun Hu, Zhaoming Lu, and Xiangming Wen. Network-scale traffic signal control via multiagent reinforcement learning with deep spatiotemporal attentive network. *IEEE trans cybern*, 2021.

[Hunt *et al.*, 1981] PB Hunt, DI Robertson, RD Bretherton, and RI Winton. Scoot-a traffic responsive method of coordinating signals. *Publication of: Transport and Road Research Laboratory*, LR 1014 Monograph, Jan 1981.

[Koonce and Rodegerdts, 2008] Peter Koonce and Lee Rodegerdts. *Traffic signal timing manual.* United States. Federal Highway Administration, 2008.

[LeCun *et al.*, 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[Liang *et al.*, 2021] Enming Liang, Kexin Wen, William HK Lam, Agachai Sumalee, and Renxin Zhong. An integrated reinforcement learning and centralized programming approach for online taxi dispatching. *TNNLS*, 2021.

[Little *et al.*, 1981] John DC Little, Mark D Kelson, and Nathan H Gartner. Maxband: A versatile program for setting signals on arteries and triangular networks. *Transportation Research Record 795*, pages 40–46, 1981.

[Liu *et al.*, 2020] Lingbo Liu, Jiajie Zhen, Guanbin Li, Geng Zhan, Zhaocheng He, Bowen Du, and Liang Lin. Dynamic spatial-temporal representation learning for traffic flow prediction. *TITS*, 22(11):7169–7183, 2020.

[Liu *et al.*, 2022] Yang Liu, Yu-Shen Wei, Hong Yan, Guan-Bin Li, and Liang Lin. Causal reasoning meets visual representation learning: A prospective study. *Machine Intelligence Research*, pages 1–27, 2022.

[Lou *et al.*, 2020] Jungang Lou, Yunliang Jiang, Qing Shen, Ruiqin Wang, and Zechao Li. Probabilistic regularized extreme learning for robust modeling of traffic flow forecasting. *TNNLS*, 2020.

[Luk *et al.*, 1982] JY Luk, AG Sims, and PR Lowrie. Scatsapplication and field comparison with a transyt optimised fixed time system. In *International Conference on Road Traffic Signalling*, number 207, 1982.

[Mirchandani and Head, 2001] Pitu Mirchandani and Larry Head. A real-time traffic signal control system: architecture, algorithms, and analysis. *Transportation Research Part C: Emerging Technologies*, 9(6):415–432, 2001.

[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.

[Oroojlooy *et al.*, 2020] Afshin Oroojlooy, Mohammadreza Nazari, Davood Hajinezhad, and Jorge Silva. Attendlight:

Universal attention-based reinforcement learning model for traffic signal control. *NeurIPS*, 33:4079–4090, 2020.

[Prashanth and Bhatnagar, 2010] LA Prashanth and Shalabh Bhatnagar. Reinforcement learning with function approximation for traffic signal control. *TITS*, 12(2):412–421, 2010.

[Rashid *et al.*, 2018] Tabish Rashid, Mikayel Samvelyan, Christian Schroeder, Gregory Farquhar, Jakob Foerster, and Shimon Whiteson. QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *ICML*, volume 80, pages 4295–4304. PMLR, 2018.

[Roess *et al.*, 2004] Roger P Roess, Elena S Prassas, and William R McShane. *Traffic engineering*. Pearson/Prentice Hall, 2004.

[Schulman *et al.*, 2017] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[Son *et al.*, 2019] Kyunghwan Son, Daewoo Kim, Wan Ju Kang, David Earl Hostallero, and Yung Yi. QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *ICML*, volume 97 of *PMLR*, pages 5887–5896. PMLR, 09–15 Jun 2019.

[Tan *et al.*, 2019] Tian Tan, Feng Bao, Yue Deng, Alex Jin, Qionghai Dai, and Jie Wang. Cooperative deep reinforcement learning for large-scale traffic grid signal control. *IEEE trans cybern*, 50(6):2687–2700, 2019.

[Taylor, 2002] Brian D Taylor. Rethinking traffic congestion. *Access Magazine*, 1(21):8–16, 2002.

[Van der Pol and Oliehoek, 2016] Elise Van der Pol and Frans A Oliehoek. Coordinated deep reinforcement learners for traffic light control. *NeurIPS*, 2016.

[Van Hasselt *et al.*, 2016] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI*, volume 30, 2016.

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

[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. *NeurIPS*, 30, 2017.

[Wang *et al.*, 2020] Xiaoqiang Wang, Liangjun Ke, Zhimin Qiao, and Xinghua Chai. Large-scale traffic signal control using a novel multiagent reinforcement learning. *IEEE transactions on cybernetics*, 51(1):174–187, 2020.

[Wang *et al.*, 2022] Tian Wang, Jiahui Chen, Jinhu Lü, Kexin Liu, Aichun Zhu, Hichem Snoussi, and Baochang Zhang. Synchronous spatiotemporal graph transformer: A new framework for traffic data prediction. *TNNLS*, 2022.

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

[Weng *et al.*, 2021] Jiayi Weng, Huayu Chen, Dong Yan, Kaichao You, Alexis Duburcq, Minghao Zhang, Hang Su, and Jun Zhu. Tianshou: A highly modularized deep reinforcement learning library. *arXiv preprint arXiv:2107.14171*, 2021.

[Wu *et al.*, 2021] Qiang Wu, Liang Zhang, Jun Shen, Linyuan Lü, Bo Du, and Jianqing Wu. Efficient pressure: Improving efficiency for signalized intersections. *arXiv preprint arXiv:2112.02336*, 2021.

[Xu *et al.*, 2021] Bingyu Xu, Yaowei Wang, Zhaozhi Wang, Huizhu Jia, and Zongqing Lu. Hierarchically and cooperatively learning traffic signal control. In *AAAI*, volume 35, pages 669–677, 2021.

[Yang *et al.*, 2016] Hao-Fan Yang, Tharam S Dillon, and Yi-Ping Phoebe Chen. Optimized structure of the traffic flow forecasting model with a deep learning approach. *TNNLS*, 28(10):2371–2381, 2016.

[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.

[Zeng, 2021] Zheng Zeng. Graphlight: Graph-based reinforcement learning for traffic signal control. In *ICCCS*, pages 645–650. IEEE, 2021.

[Zhang and Batterman, 2013] Kai Zhang and Stuart Batterman. Air pollution and health risks due to vehicle traffic. *Science of the total Environment*, 450:307–316, 2013.

[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 *W3C*, pages 3620–3624, 2019.

[Zhang *et al.*, 2021] Liang Zhang, Qiang Wu, Jun Shen, Linyuan Lü, Jianqing Wu, and Bo Du. Expression is enough: Improving traffic signal control with advanced traffic state representation. *arXiv preprint arXiv:2112.10107*, 2021.

[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.