

PV-TSC: Learning to Control Traffic Signals for Pedestrian and Vehicle Traffic in 6G Era

Kangjie Xu¹, Student Member, IEEE, Junqin Huang², Student Member, IEEE,
Linghe Kong¹, Senior Member, IEEE, Jiadi Yu¹, Senior Member, IEEE, and Guihai Chen

Abstract—Recent advances in traffic signal control have witnessed the success of reinforcement learning. However, most of these approaches have focused on vehicle traffic and lack consideration for pedestrians. This can be attributed in part to the fact that the existing underlying technologies are not yet practical to deploy in real-world environments. Vision technologies, for example, can easily be obscured from view in reality. The direction of movement and position of pedestrians is difficult to estimate accurately. The emergence of 6G localization and tracking services offer new opportunities. With this base service, we intend to improve the efficiency, safety, and scalability of multi-intersection traffic signal control with mixed traffic flows. This problem is challenging for its coordination, scalability, and access of new traffic. To solve these challenges, we propose PV-TSC, a distributed reinforcement learning motivated traffic signal control with pedestrian access. We analyze different behaviors of pedestrian traffic, and integrate pedestrian traffic with the proven traffic signal control scheme for vehicle traffic. Finally, we conduct simulation experiments to illustrate the superiority of PV-TSC against classic methods, and further analyze the effectiveness of PV-TSC design by exploring its variants.

Index Terms—Traffic signal control, reinforcement learning, pedestrian traffic.

I. INTRODUCTION

NOWADAYS, traffic congestion is one of the biggest urban governance concerns, which seriously affects people's travel efficiency, causes more traffic accidents, and contributes to more environmental pollution. According to Forbes News [1], the total cost of traffic congestion is estimated as high as 74.1\$ billion annually with 66.1\$ billion occurring in urban areas. From this point of view, it is necessary to improve travel efficiency. Meanwhile, when other indicators such as pollution and accidents are taken into account, traffic congestion can cost cities billions of dollars each year. INRIX [2] reported that drivers lost more than 88\$ billion due to rear-end collisions in 2019 with the average cost for each of them coming to 1,377\$. Furthermore, not only is

vehicle safety important but also pedestrian road safety should not be ignored. Centers for Disease Control and Prevention (CDC) [3] estimated that 137,000 pedestrians were treated in emergency departments for nonfatal crash-related injuries in Feb. 2017 per trip. Pedestrians are 1.5 times more likely than vehicle passengers to be killed in a car crash.

All of these issues regarding efficiency and safety can be optimized by controlling traffic signals. Signal timing optimization has been proved to be an NP-complete problem, while the traffic signal control scenarios require real-time and scalability. In recent years, many studies have applied reinforcement learning (RL) to the field of traffic signal control. Due to the development of technologies, more information can be obtained from the environment, and the computing power of the devices is more powerful, paving the ground for adopting reinforcement learning in the field of traffic signal control. The performance of traditional traffic signal control schemes relies heavily on the parameter settings, and cannot dynamically adapt to changes in traffic flows. In contrast, reinforcement learning can learn from the data and dynamically adjust the control strategy based on real-time traffic changes. Either in the single intersection or in the multi-intersection scenarios, the reinforcement learning method has achieved a prominent performance and has been tested in large-scale real-world scenarios [4].

Most of the traffic control algorithms combined with reinforcement learning focus only on the vehicle traffic flow, and ignore the pedestrian traffic flow, which is also an important component of intersection traffic. A few studies [5] that considered pedestrian flow failed to analyze in depth the impact that pedestrians bring to intersections. Thus, there is still room for improvement for reinforcement learning motivated traffic signal control with pedestrian access. The reason for the lack of research in this field is the special nature of sidewalks. The traffic lane is one-way, and the direction of car movement can be determined by judging the lane where the vehicle is located, but the sidewalk is two-way, and in reality, it is difficult to determine the direction of pedestrian movement. Current related works rely on computer vision techniques [6] to estimate pedestrian movement direction. However, the performance of computer vision technology is still impaired by visual obstacles and insufficient computational resources. The reasons above make traffic signal control with pedestrian access difficult to deploy and use in practice.

The recent development of 6G and MIMO (Multi-input Multi-output) [7], [8] offers new opportunities for traffic

Manuscript received 22 December 2021; revised 13 February 2022; accepted 28 February 2022. Date of publication 14 March 2022; date of current version 7 July 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2020YFB1710900; in part by NSFC under Grant 62141220, Grant 61972253, Grant U1908212, Grant 72061127001, Grant 62172276, and Grant 61972254; and in part by the Program for Professor of Special Appointment (Eastern Scholar) at the Shanghai Institutions of Higher Learning. The Associate Editor for this article was S. Mumtaz. (Corresponding author: Linghe Kong.)

The authors are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: xkjac04452017@sjtu.edu.cn; junqin.huang@sjtu.edu.cn; linghe.kong@sjtu.edu.cn; jiadiyu@sjtu.edu.cn; chen-gh@sjtu.edu.cn).

Digital Object Identifier 10.1109/TITS.2022.3156816

signal control, and the stability of these wireless technologies for location tracking is better than vision technologies inherently. In addition, hundreds of meters of localization range, decimeter-level localization accuracy, and high-accuracy movement direction recognition are perfectly adequate for obtaining pedestrian-related data at intersections.

Moreover, pedestrian traffic involvement introduces a lot of uncertainty and complexity. The speed, physical volume, and movement pattern of pedestrians are quite different from those of vehicles. Pedestrians and vehicles can impede each other's movement, reducing the efficiency of traffic flow and even causing some traffic accidents. How to strike a balance between the two traffic flows, between safety and efficiency remains to be settled in this context.

Another challenge comes from the multi-intersection scenario. An intuitive way to apply reinforcement learning to traffic signal control is to rely on one agent to control the traffic lights at all intersections [9]. The advantage of this is that one agent can view global information and can optimize the problem from a global perspective. However, such a framework is not well scalable. The state space and action space are exponentially large, which is not applicable in practice due to the real-time requirements of control. Although existing single-intersection optimization schemes are scalable, and perform well in the single-intersection scenario, it is not guaranteed to be equally good in a multi-intersection environment. Therefore, researchers have tried to extend single-intersection schemes by introducing some collaborative mechanisms, such as adding queue length in the neighboring intersections [5], modeling relationship between neighboring intersections [10], for achieving both scalability and efficiency at the same time in multi-intersection scenarios.

In this paper, we combine the localization services of 6G and learning-based traffic signal control to achieve joint control of pedestrian and vehicle traffic, seeking to reduce the waiting time of pedestrians and vehicles, and improve the traffic safety of multi-intersections. The key challenges to solve this problem are: (1) What is special about pedestrian traffic as opposed to vehicle traffic? (2) How can we exploit these specialties to optimize the efficiency, safety, scalability of traffic signal control in the multi-intersection scenario? To answer these questions, we propose our traffic signal control scheme: Pedestrian-Vehicle Traffic Signal Control (PV-TSC), the contributions of our PV-TSC includes:

- 1) PV-TSC considers both pedestrian and vehicle traffic, and designs a reinforcement learning scheme based on different queuing behaviors of the traffic flow to effectively schedule the traffic signal for safe and efficient travel.
- 2) The proposed method is decentralized, and has good scalability to multi-intersections. Meanwhile, we extend the "pressure" reward to pedestrian traffic flows, which integrates information of neighboring intersections to the local model efficiently.
- 3) We evaluate PV-TSC in the multi-intersections scenario in SUMO, compared with some traditional methods, and shows an edge in the multiple dimensions, such as waiting time/travel time for vehicle and pedestrian traffic

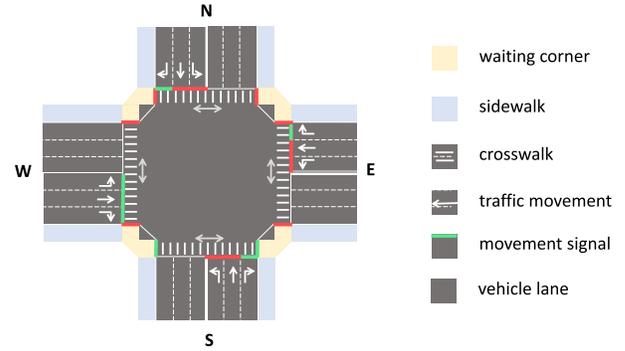


Fig. 1. Example of traffic intersection topology.

flows, safety score. Furthermore, we validate some of our design choices in evaluations by comparing several variants of our method.

II. PROBLEM DESCRIPTION

In this paper, we focus on traffic signal control in the case of multiple intersections. In this section, we present some definitions related to this research problem. Most of these definitions are consistent with research in the field [11], but the involvement of pedestrian flow introduces some differences. We will use the intersection example in Fig. 1 to illustrate these definitions and describe the problem.

A. Preliminaries

1) *Vehicle Lane*: A vehicle lane at an intersection is where the vehicle traffic flow enters or leaves the intersection. A vehicle lane can be either an incoming lane or an outgoing lane. It's unidirectional. At the same moment, the width of a lane can accommodate one vehicle.

2) *Pedestrian Lane*: Pedestrian lanes are places where pedestrians walk. They can be divided into several types: sidewalks, waiting corners, and crosswalks. Sidewalks are parallel to the adjacent lanes. Waiting corners are where pedestrians wait for green traffic signals. Crosswalks are where the pedestrians cross the intersection, in that vehicles are obliged to stop when someone has indicated their intent to cross by waiting by the crossing. For pedestrian traffic flow, pedestrian lanes are both incoming and outgoing, which is bidirectional. Another difference between a vehicle lane and a pedestrian lane is that the width of a pedestrian lane can accommodate several pedestrians.

3) *Lane Segment*: A lane can be divided into several segments, different segments have different distances from the intersection center and they don't overlap with each other.

4) *Traffic Movement*: A travel movement (l, m) means a travel of pedestrians or vehicles going from incoming lane l to outgoing lane m . $l \in LI_i$, $m \in LO_i$, LI_i is the set of incoming lanes in intersection i and LO_i is the set of outgoing lanes in intersection i . In Fig. 1, directions of traffic movements are shown as the white arrows in each incoming lane. Particularly, the traffic movements of pedestrian lane is bidirectional.

5) *Movement Signal*: A movement signal $ms(l, m)$ controls the state of a traffic movement (l, m) . As we experienced

Phase id	Permissible traffic movements
0 (all-red phase)	None
1	 (E)
2	 (S)
3	 (W)
4	 (N)

Fig. 2. Phase definition of the intersection shown in Fig. 1.

in our daily lives, a green movement signal means the corresponding movement is allowed and we can denote it as $ms(l, m) = 1$. While a red movement signal prohibits the corresponding movement, we denote it as $ms(l, m) = 0$. In this paper, we only discuss the cases of driving on the right side of the road. For the case of driving on the left, an analogous conversion is possible. In addition, the traffic signal for the right turn vehicle lane is always green, as it is in our real lives.

6) *Phase*: A phase is a set of movement signals of traffic movements as demonstrated in Fig. 2, it can be denoted as $p_i = \{(l, m) | ms(l, m) = 1\}$, where $l \in LI_i$ and $m \in LO_i$. In Fig. 2, the bidirectional arrows means the direction of crosswalks. ‘‘E’’, ‘‘S’’, ‘‘W’’, ‘‘N’’ indicate the locations of crosswalks. Usually, each phase is combined of non-conflicted permissible traffic movements, otherwise conflicted traffic movements can easily block each other and cause congestion. What’s more, traffic lights are controlled by changing the sequence or duration of each phase, rather than flipping one movement signal particularly.

7) *All-Red Phase*: An all-red phase means, all movement signals for vehicle lanes and pedestrian lanes are red signals in this phase.

8) *Phase Sequence and Signal Plan*: A phase sequence defines the order of phase changes. A signal plan sets duration for each phase in the phase sequence.

9) *Cycle-Based Signal Plan*: A cycle-based signal plan means the phase will iterate in cyclic order, while the time for each phase is not constant.

10) *Pressure*: The concept ‘‘pressure’’ was proposed by Varaiya *et al.* [12]. It can be defined over traffic movements, which indicates the difference between the traffic density of incoming lanes and outgoing lanes. For a traffic movement (l, m) . We denote pressure as $w(l, m)$, $x(l)$ is the number of vehicles on the vehicle lane l (or the number of pedestrians for the pedestrian lane). $x_{max}(l)$ is the maximum number of vehicles (pedestrians) a lane can accommodate. $d(l)$ is the density of the lane l , which is defined as

$$d(l) = \frac{x(l)}{x_{max}(l)}. \quad (1)$$

The pressure $w(l, m)$ is defined as

$$w(l, m) = d(l) - d(m). \quad (2)$$

The pressure of vehicle lanes is the sum of pressures of all vehicle traffic movements (l_v, m_v) , going from l_v to m_v , which is defined as

$$P_i^{veh} = \sum_{(l_v, m_v)} w(l_v, m_v), \quad (3)$$

where $l_v \in LI_i^{veh}$ and $m_v \in LO_i^{veh}$, LI_i^{veh} is the set of incoming vehicle lanes in intersection i and LO_i^{veh} is the set of outgoing vehicle lanes in intersection i .

In the literature [13], it interprets pressure as the degree of imbalance in the density of vehicles in the incoming and outgoing lanes in that intersection. Here we extend the pressure to pedestrian lanes. With pedestrian access, the sidewalks and waiting corners are also inputs of the intersection, and we should include it in the intersection pressure. The pressure of the pedestrian lanes for all the pedestrian traffic movements (l_p, m_p) is defined as

$$P_i^{ped} = \sum_{(l_p, m_p)} w(l_p, m_p), \quad (4)$$

where $l_p \in LI_i^{ped}$ and LO_i^{ped} , LI_i^{ped} is the set of incoming pedestrian lanes in intersection i and LO_i^{ped} is the set of outgoing pedestrian lanes in intersection i . Actually, $LI_i^{ped} = LO_i^{ped}$ since a pedestrian lane is bidirectional. Thus, the whole pressure for the intersection is

$$P_i = \alpha_1 * P_i^{veh} + \alpha_2 * P_i^{ped}, \quad (5)$$

where α_1 and α_2 are weighting factors.

Take the intersection in Fig. 3 as an example. Assume the intersection identifier is $i = 0$, the maximum number of pedestrians in the waiting corners l_{wc} is $x_{max}(l_{wc})$, the maximum number of pedestrians on the sidewalks l_{sw} is $x_{max}(l_{sw})$, and the maximum number of vehicles in the vehicle lane l_v is $x_{max}(l_v)$. There are 3 vehicles and 2 pedestrians in the incoming lanes, 1 vehicle and 3 pedestrians in the outgoing lanes. Thus, $P_0^{veh} = \frac{3}{x_{max}(l_v)} - \frac{1}{x_{max}(l_v)} = \frac{2}{x_{max}(l_v)}$ and $P_0^{ped} = \frac{2}{x_{max}(l_{wc})} - \frac{3}{x_{max}(l_{sw})}$. For crosswalks, their contributions to the overall pressure is 0, since their incoming and outgoing lanes are at the same intersection. Thus, the whole pressure for the intersection in Fig. 3 is $P_0 = \alpha_1 * \frac{2}{x_{max}(l_v)} + \alpha_2 * (\frac{2}{x_{max}(l_{wc})} - \frac{3}{x_{max}(l_{sw})})$.

11) *Safety Score*: The safety score is to quantify the degree of an intersection’s safety. Considering non-compliance of pedestrians, we define the safety score S_i as the number of jaywalking pedestrians (those who do not obey the traffic signal schedule) in the intersection i .

$$S_i = \#(\text{jaywalking pedestrians}) \quad (6)$$

A pedestrian signal permits a pedestrian to begin crossing a street during the green movement signal. Pedestrians are usually considered to be ‘‘jaywalking’’ only if they enter the crosswalk some other time. Therefore, the higher the safety score, the higher the probability pedestrian-vehicle accidents happen in the intersection.

All the definitions and notations used in the paper can be referred to in Table I.

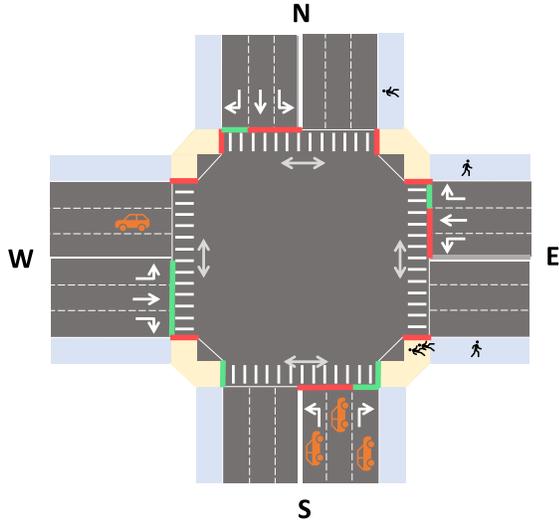


Fig. 3. The description of “pressure” concept.

TABLE I
NOTATIONS AND DEFINITIONS

Notation	Definition
t	Time step
i	Intersection index
l, m	Lane
N	Number of intersections
T	Number of time steps in an episode
M	Number of episodes
LI_i	Set of incoming lanes of intersection i
LO_i	Set of outgoing lanes of intersection i
p_r	All-red phase
P_i	Pressure of intersection i
S_i	Safety score in intersection i
s_i^t	State of agent i at time step t
a_i^t	Action of agent i at time step t
o_i^t	Observation of agent i at time step t
\mathcal{S}	State space
\mathcal{O}	Observation space
\mathcal{A}	Action space
γ	Discount factor
TF	Transition function
R	Reward function
G_t	Total reward until time step t
T_r	Phase duration for all-red phase
T_g	Phase duration for regular phase
α_1	The weighting factor for vehicle pressure
α_2	The weighting factor for pedestrian pressure
α_3	The weighting factor for safety score
$p_i(k)$	The k -th phase of intersection i
$x(l)$	Number of pedestrians (vehicles) on lane l
$x(l)(t)$	Number of pedestrians (vehicles) on lane l at time step t
$x_{max}(l)$	Maximum number of pedestrians (vehicles) on lane l
$x(l, m)$	Number of vehicles leaving lane l and entering m
$x(l, m)(t)$	Number of vehicles leaving lane l and entering m at time step t
$length(l)$	Length of lane l
(l, m)	Traffic movement going from l to m
$w(l, m)$	Pressure of traffic movement (l, m)
$ms(l, m)$	State of movement signal over traffic movement (l, m)

B. Problem Definition

The problem we focus on is the multi-intersection traffic signal control problem. By controlling the traffic lights in the whole road network, we aim to reduce the travel time of pedestrians and vehicles, as well as lowering the risks (safety score). In this paper, we resort to the multi-agent reinforcement framework to solve the problem.

Usually a single-agent reinforcement problem is modeled as a Markov Decision Process (MDP). The generalization of MDP to the multi-agent case is the stochastic game (SG). A stochastic game is defined by a tuple $\Gamma = \langle \mathcal{S}, TF, \mathcal{A}, R, \mathcal{O}, N, \gamma \rangle$, where $\mathcal{S}, TF, \mathcal{A}, R, \mathcal{O}, N, \gamma$ are the sets of states, transition probability functions, joint actions, reward functions, private observations, number of agents and a discount factor respectively. The definitions are given as follows:

- 1) N : N agents identified by $i \in I = \{1, \dots, N\}$.
- 2) \mathcal{S}, \mathcal{O} : At each time step t , agent i draws observation $o_i^t \in \mathcal{O}$ correlated with the true environment state $s^t \in \mathcal{S}$ according to the observation function $\mathcal{S} \times I \rightarrow \mathcal{S}$.
- 3) TF, \mathcal{A} : Agent i 's action set \mathcal{A}_i is defined as a group of phases. At time step t , each agent takes an action $a_i^t \in \mathcal{A}_i$, forming a joint action $a^t = a_1^t, \dots, a_N^t$, which induces a transition in the environment according to the state transition function

$$TF(s^{t+1} | s^t, a^t) : \mathcal{S} \times \mathcal{A}_1 \times \dots \times \mathcal{A}_N \rightarrow \Omega(\mathcal{S})$$

where $\Omega(\mathcal{S})$ denotes the space of state distributions.

- 4) R : In a stochastic game setting, the reward an agent obtains is also influenced by the actions of other agents. Therefore, at time t , each agent i obtains rewards r_i^t by a reward function

$$R_i(s_i^t, a_i^t) : \mathcal{S} \times \mathcal{A}_1 \times \dots \times \mathcal{A}_N \rightarrow \mathbb{R}$$

- 5) γ : Intuitively, the joint actions have long-term effects on the environment. Each agent i chooses an action following a certain policy π_i , aiming to maximize its total reward, $G^t := \sum_{j=0}^{\infty} \gamma^j r_i^{t+j}$, where the discount factor $\gamma \in [0, 1]$ controls the importance of immediate rewards versus future rewards.

1) *Multi-Intersection Traffic Signal Control Problem*: For a network with multiple intersections, agents are defined as signal controllers of N intersections in the environment. The goal of the traffic signal agents controlled with reinforcement learning is to learn the optimal policy for each agent, as well as to optimize the traffic conditions of the global traffic network. At each time point t , each agent i observes part of the environment as an observation o_i^t . The agent will predict the next action a_i^t to be taken. In the real world, \mathcal{A}_i is mostly predetermined, i.e., traffic signals can only change in a few phases. These actions will be executed in the environment and generate a reward r_i^t , where the reward can be defined at the level of a single intersection or a set of intersections in the environment.

III. PV-TSC DESIGN

In this section, we firstly introduce the design of PV-TSC. And then, we discuss the feasibility and justification of the design from several aspects, including the difference between the pedestrian and vehicular traffic, and how we optimize the safety and efficiency of our method in the state, action, reward design in the distributed agent premise. Furthermore, the learning process of the agent will be detailed.

A. Agent Design

The reinforcement learning method we adopted in PV-TSC is DQN [14]. Previously it was designed to play the Atari games. Many recent studies in this field [4], [15], [16] have applied it to the traffic signal control problem, and some of them have proved the practicality of DQN. Each DQN agent in our method controls one intersection. Next, we will elaborate on the definition for the state, action, and reward definition for our DQN agent.

1) *State*: At each time step t , the agent will observe the state in the road network, quantify them with an observation function and get observation o_i^t , then combine them as the state. The state of our DQN agent is composed of several parts, including the pedestrian state, vehicle state, and the current phase p_i .

For the vehicle state, it includes the density of vehicles $d(l_v(j))$ on each road segment of incoming lanes, $l_v \in LI_i^{veh}$, $l_v(j)$ is the j -th segment of lane l_v . By default, $j \in \{1, 2, 3\}$ and each vehicle lane is divided into 3 segments of the same length.

For the pedestrian state, it includes the number of queuing pedestrians on each waiting corners $d(l_{wc}), l_{wc} \in LI_i^{wc}$, the number of pedestrians on the crosswalk $d(l_{cw}), l_{cw} \in LI_i^{cw}$, the density of 3 sidewalk $d(l_{sw}(j)) \in LI_i^{sw}$, $j \in \{1, 2, 3\}$. Assuming the length of crosswalk is $length_{cw}$ and the length of waiting corners is $length_{wc}$, walking speed of pedestrians is v_{ped} , minimum green time is g_{min} (minimum time for pedestrian to pass through the crosswalk). The length of the first sidewalk segment is defined as:

$$length_{l_{sw}(1)} = g_{min} * v_{ped} - length_{cw} - length_{wc}, \quad (7)$$

while the length of the other two segments equally divide the rest of the length.

Finally the state definition of DQN is $\langle \{d(l_v(j))\}, \{d(l_{wc})\}, \{d(l_{cw})\}, \{d(l_{sw}(j))\}, p_i \rangle$.

2) *Action*: Previous vehicle traffic signal control related researches have not dealt with the crosswalk lane. Commonly, there are several action types for traffic signal control: setting the phase duration, setting the phase duration ratio (fixed cycle time), keeping or changing the current phase, or directly choosing the next phase.

Firstly, we define each phase for the intersection. Take the intersection in Fig. 1 for example, the four regular phases are shown in Fig. 4. They will iterate in a cyclic order. Besides 4 non-conflicted phases, we add an all-red phase considering much larger pedestrian flows and intersection emergencies. Specifically, we insert an all-red phase between each two regular phases. Assume the $p_i(k)$ is the k -th phase in regular phases and p_r is the all-red phase. Thus the phase sequence for intersection in Fig. 1 is $p_i(1) \rightarrow p_r \rightarrow p_i(2) \rightarrow p_r \rightarrow p_i(3) \rightarrow p_r \rightarrow p_i(4) \rightarrow p_r \rightarrow p_i(1) \rightarrow \dots$

Furthermore, we adopt the keep and change action type. When phases operate in cyclic order, each phase except the all-red phase, will last at least for a duration of T_g seconds (all-red phase duration is T_r). When the time for the current phase runs out, the agent will decide on whether to keep the current phase and operate for another T_r . If $a_i^t = 0$, the traffic light

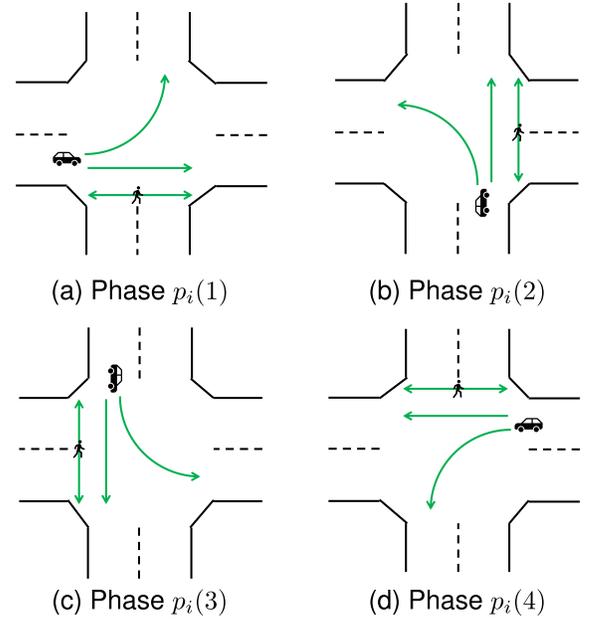


Fig. 4. Definition for four regular phases of intersection i .

will keep the traffic light in the last phase, else $a_i^t = 1$ enter the next phase in the phase sequence mentioned above. The action space $\mathcal{A}_i = \{0, 1\}$. The agent is in charge of the decision of keeping or changing. The detailed workflow is shown in Algorithm 1.

3) *Reward*: The reward for one intersection agent is defined as weighted sum of the vehicle pressure P_i^{veh} , pedestrian pressure P_i^{ped} and safety score S_i . $\alpha_1, \alpha_2, \alpha_3$ are weighting factors.

$$r_i^t = -|\alpha_1 * P_i^{veh} + \alpha_2 * P_i^{ped}| - \alpha_3 * S_i \quad (8)$$

It is noted that in the all-red phase, the pedestrians on the crosswalks are not counted in the safety score. Intuitively, the less pressure in the system, the larger throughput of the intersection. The fewer jay-crossing pedestrians on the crosswalk less likely will pedestrian-vehicle accidents happen.

B. Design Philosophy

To prove the effectiveness of PV-TSC design, we explain our design philosophy, and justify some designs for the safety and efficiency concerns of traffic signal control.

1) *State Comprehensively Modeling the Traffic*: The state definition actually means what information we should provide for the agent. Lack of information may prevent the agent from estimating the value and performing the best action. Hence, the chosen state features should describe the environment comprehensively. But more does not mean better, too complicated features may confuse and prolong the training process. Enhanced deep neural networks can be powerful enough to extract these useful features, but the unstable training process may not lead to performance improvement, and the longer computation time violates the real-time requirement of the traffic signal control system. What's more, a larger computation overhead puts more burden on the hardware resources. Thus, a reasonable design of state is necessary.

Algorithm 1 Keep and Change Action Workflow

$agent_i$ is the DQN agent for intersection i .
 T_r is the all-red phase duration.
 T_g is the regular phase duration.
 p_r is the all-red phase, $p_i(k)$ is the k -th regular phase of intersection i .
 $phases_i \leftarrow [p_i(1), p_r, p_i(2), p_r, p_i(3), p_r, p_i(4), p_r]$
 $t \leftarrow 0$
 $t_l \leftarrow T_g$
 $p_{cur} = 0$
while *true* **do**
 if $t_l == 0$ **then**
 Observe state s_i^t
 Query the action a_i^t of the current state to the $agent_i$
 if $a_i^t == 0$ **then**
 $p_{next} \leftarrow p_{cur}$
 else
 $p_{next} \leftarrow (p_{cur} + 1) \bmod size(phases_i)$
 end if
 $p_{cur} \leftarrow p_{next}$
 if $phases_i[p_{cur}]$ is all-red phase **then**
 $t_l \leftarrow T_r$
 else
 $t_l \leftarrow T_g$
 end if
 end if
 $t_l \leftarrow t_l - 1$
 $t = t + 1$
end while

The choice of state features is motivated by several reasons. Firstly, previous studies [17] have proved that the number of vehicles on each lane and the current phase have fully described the system dynamics, which are simple and comprehensive. While some information such as waiting time without bound may cause a large state space and tend to interfere with the training process. We follow this idea [17] in our design, and thus the state of PV-TSC is also composed of the number of vehicles on each lane segment. Secondly, the different definitions for pedestrians attribute to our two observations on different queuing behaviors of pedestrians. 1) Due to a larger physical length of vehicles, the vehicle queue is usually much longer than pedestrian queue. 2) Besides, the acceleration time for pedestrians is much smaller than that of vehicles. A vehicle may have to wait for the vehicle ahead of it to speed up, which can take up several seconds. While for the pedestrian queue, the time for pedestrians to accelerate nearly costs no more than a second. These phenomena lead to the result that the queue in pedestrian lane can drain out much faster than the queue in vehicle lane. Based on the above observations, we take into consideration the number of pedestrians in the first segment of the sidewalk, because these pedestrians are most likely to cross the crosswalk within the duration of a phase.

To further theoretically support our design, we can justify it by proving that the state has the Markov property and the state transition can be formulated as a Markov chain. That is to say, s^t only depends on the s^{t-1} and the control

policy. Furthermore, we can justify that the reward can actually minimize the travel time.

Wei *et al.* [13] has demonstrated that a evolution equation of the $x(l)(t)$, which is the number of vehicles on lane l in time step t . We can treat one bidirectional pedestrian lane as two overlapped unidirectional pedestrian lanes, where the same evolution equation applies as well. In original equation, $x(l, m)(t)$ means the number of vehicles leaving lane l and entering lane m at time step t . With pedestrian access, $x(l, m)(t)$ changes its meaning, which stands for the number of vehicles or pedestrians that leave lane l and enter m at time step t . The modified evolution equation 9 is shown below.

$$\begin{aligned}
 x(l, m)(t + 1) &= x(l, m)(t) \\
 &+ \underbrace{\sum_{k \in In_l} \min[c(k, l) \cdot a(k, l)(t), x(k, l)(t)] \cdot r(l, m)}_{\text{receiving traffic}} \\
 &- \underbrace{\min\{c(l, m) \cdot a(l, m)(t), x(l, m)(t)\} \cdot 1}_{\text{discharging traffic}} \cdot 1(x(m) \leq x_{\max}(m))
 \end{aligned} \tag{9}$$

In this equation, $c(l, m)$ means the discharging rate, which is a non-negative and bounded value. $r(l, m)$ is the turning ratio, meaning the proportion of vehicles changing from lane l to lane m , In_l means the set of lanes which are the incoming lanes of lane l . $a(l, m)(t) = a_i^t$, the action for traffic movement (l, m) is consistent with the action a_i^t of the entire intersection. Because $a(l, m)(t)$ is a function of $x(l, m)(t)$, from the equation, we can further infer that $x(l, m)(t + 1)$ depends on the only random variable $x(k, l)(t)$, which indicates the iterative process is Markov chain. The probabilities of transition depend on the control policy $a(l, m)(t)$. In this way, we ensure the Markov property of the state of PV-TSC.

2) *Action Improving Intersection Safety and RL Training*: The special consideration for pedestrian participation is that we add an all-red phase when traffic signals switch between regular phases. The reason is that during switches, the pedestrians may still be walking on the crosswalk. Some pedestrians may be non-compliant, who take a chance to cross the crosswalk in the last seconds during the green signal. In order to ensure the safe crossing of these pedestrians, the agent can extend one or more all-red phases.

Another consideration is the keep and change action type. One crucial benefit for keep and change action type is that the action space is relatively small, which greatly simplifies and stabilizes the learning process. On the other hand, the cyclic signal plan makes pedestrians and vehicle drivers ready and clear about the signal plan, and prepare for their next steps in advance. Since reinforcement learning relies on hyper-parameters to train, the cycle-based signal plan reduces the risk that pedestrians can keep waiting infinitely. With the two considerations mentioned above, PV-TSC can adapt the all-red phase duration dynamically based on the intersection state.

3) *Reward Achieving Scalability, Balancing Safety and Efficiency*: Since we adopt the distributed design, i.e., each

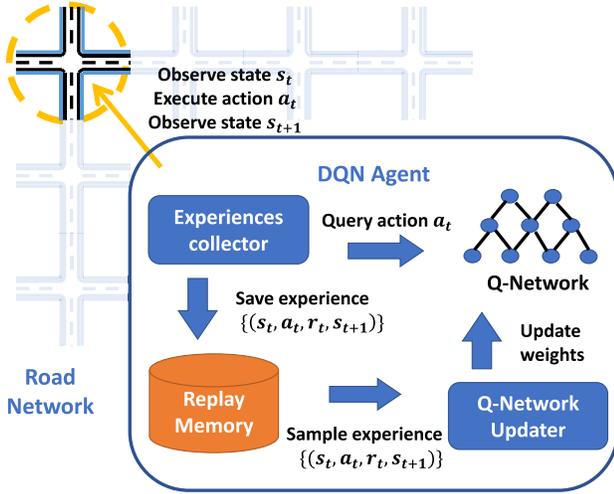


Fig. 5. DQN learning process.

agent is responsible for one intersection, the scalability of our method should be guaranteed. Thus, we optimize the reward design to realize both the efficiency and safety of our method.

Though the objective of traffic signal control is to optimize travel time, we cannot directly adopt it. Because the travel time is a delayed signal for the reinforcement learning agent, it cannot be measured immediately when the whole trip has not been finished. Thus, the reward function usually comprises other factors, such as queue length, waiting time, and so on. Presslight [13] theoretically proves that “pressure” can lead reinforcement learning agents to achieve good performance, we follow this idea and extend it to the pedestrian traffic scenario.

We can show the advantages of our methods from the perspectives of the safety and efficiency concerns: For the justification of pressure, Chen *et al.* [18] have proved that the max-pressure control policy with the weighted pressure (for pedestrian and vehicle traffic flow) is stabilizing, which means the queue lengths of each lane will remain bounded in expectation. Equivalently, the control policy is throughput optimal. DQN agent always chooses the action that maximizes the reward (including pressure), so it tries to optimize the throughput for the intersection as well. For the safety score part, when traffic signals enter a regular phase from the all-red phase, the pedestrians walking on the crosswalk will be regarded as jaywalking pedestrians, thus less reward the agent may gain. In this way, the safety score part encourages the agent extending the all-red phase when pedestrians are still walking on crosswalks. However, the agent should balance between the safety issue and efficiency issue in this situation.

C. Learning Process

The design of the agent above describes the definitions of basic elements for DQN agent. Based on these definitions, DQN agent will try to learn from the collected experiences and estimate the Q value for each <state, action> pairs. The framework of DQN agent is shown in Fig. 5. The learning process consists of two parts: the experience collector and the Q-network updater.

Algorithm 2 Experience Collector

\mathcal{D} is the replay memory
 Q is the action-value function approximated by an neural network with weights θ
for episode = 1 ... M **do**
 observe the state s_i^1
 for $t = 1 \dots T$ **do**
 With probability ϵ select a random action a_i^t
 otherwise select $a_i^t = \max_a Q^*(s_t, a; \theta)$
 Execute action a_i^t
 Observe reward r_i^t and state s_i^{t+1}
 Store transition $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$ to \mathcal{D}
 end for
end for

The experience collector will observe states, execute actions, and compute reward. The collector records the transition $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$, and stores it in the experience replay memory \mathcal{D} . Meanwhile, when executing each action, the agent will adopt ϵ -greedy algorithm [19]. The agent will follow the greedy strategy with probability $1 - \epsilon$ and selects a random action with probability ϵ . The detailed execution steps can be referred in Algorithm 2.

The Q-network updater will randomly sample the experience batches from the replay memory \mathcal{D} , then it computes the target y_i for iteration i . Finally, θ will be updated based on the equation 10.

$$\nabla_{\theta_i} L_i(\theta_i) = \mathbb{E}_{s, a \sim \rho(\cdot); s' \sim \mathcal{E}} \left[\left(r + \gamma \max_{a'} Q(s', a'; \theta_{i-1}) - Q(s, a; \theta_i) \right) \nabla_{\theta_i} Q(s, a; \theta_i) \right] \quad (10)$$

The reason for proposing the experience collector and the Q-network updater is that we can use multi-processes to achieve distributed learning. In this way, we can accelerate collecting the experiences and the learning process.

The detailed update algorithm can be referred to in Algorithm 3. It is noted that the terminal state means the intersection is empty (no vehicle and no pedestrians on each lane) in our traffic signal control problem.

IV. EXPERIMENTS

In this section, we conduct some experiments to evaluate our proposed method, PV-TSC. We will introduce the settings of experiments, evaluation metrics, and some compared methods. Performance comparison between different methods will be demonstrated in this section as well.

A. Experiment Setup

1) *Simulation Platform Settings*: The simulation platform is SUMO (Simulation of Urban MObility) [20], which is a widely-used and open-source traffic simulation package. Leveraging SUMO, we can define the road network topology, generate traffic flows and control the traffic signals.

About road network topology, we investigated several cases: 3×3 intersection, 1×4 intersection, 4×4 intersection,

TABLE II
 CONFIGURATION OF EVALUATIONS

Configuration	Road network topology type	Vehicle Arrival rate (vehicles/h)	Pedestrian arrival rate (pedestrians/h)	Volume ratio	Simulation start time (s)	Simulation end time (s)	Simulation duration (s)
1	a	224 * 3 * 3	224 * 3 * 3	1	3600	25200	21600
2	b	224 * 1 * 4	224 * 1 * 4	1	3600	25200	21600
3	c	224 * 4 * 4	224 * 4 * 4	1	3600	25200	21600
4	a	224 * 3 * 3 * 1	224 * 3 * 3 * 2	0.5	3600	25200	21600
5	a	224 * 3 * 3 * 2	224 * 3 * 3 * 1	2	3600	25200	21600
6	a	224 * 3 * 3 * 2 * 1/3	224 * 3 * 3 * 2 * 2/3	0.5	3600	25200	21600
7	a	224 * 3 * 3 * 2 * 2/3	224 * 3 * 3 * 2 * 1/3	2	3600	25200	21600

Algorithm 3 Q-Network Updater

\mathcal{D} is the replay memory

Q is the action-value function approximated by a neural network with weights θ

for episode = 1 ... M **do**

for $t = 1 \dots T$ **do**

 Sample random batch of transitions $(s_i^t, a_i^t, r_i^t, s_i^{t+1})$
 from \mathcal{D}

 Set $y_t = \begin{cases} r_i^t, & (\text{for terminal } s_i^{t+1}) \\ r_i^t + \gamma \max_{a'} Q(s_i^{t+1}, a'; \theta) & (\text{for non-terminal } s_i^{t+1}) \end{cases}$

 Perform a gradient descent step on $(y_t - Q(s_i^t, a_i^t; \theta))^2$ according to equation 10

end for

end for

as shown in Fig. 6. They are marked as road type a , b , c , respectively. The road speed limit for vehicle lanes is 45km/h and the default speed for pedestrians is 1m/s. The length of each vehicle lane and sidewalk is 500 meters. The length of crosswalks is 20 meters, which means the minimum time for pedestrians to cross the intersection is 20 seconds. To ensure that pedestrians can cross the intersection safely, phase time for each regular phase should be larger than 20 seconds.

About the traffic flow generation settings, we use the randomTrips tool in SUMO, which can generate pedestrian and vehicle traffic routes in binomial distribution. We can control the arrival rates of the traffic. For each route, its start point and end point are also random in the road network.

Based on the environmental settings and traffic routes settings, we define several different configurations, including arrival rates for traffic flows, simulation duration, volume ratio ($volume\ ratio = \frac{traffic\ volume_{veh}}{traffic\ volume_{ped}}$, the ratio of traffic volume of vehicles to the traffic volume of pedestrians), simulation start time and end time, as shown in the Table II. We will let this method run for 3600 time steps and then start to compute some relevant metrics afterwards, since we cannot start from an empty intersection in reality. In addition, a green light signal is always followed by a 3-seconds yellow light signal. This applies to all methods in experiments.

2) *Compared Methods*: We compare our method with some traditional methods and our proposed method's variants. All methods are carefully tuned in our settings, to optimize the travel time for all traffic flows.

- Fixed Time Control: Fixed Time Control means the phase duration for each regular phase is set to a fixed time.

In our experiments, the phase duration for each regular phase is set as 20 seconds, the same as the minimum green time.

- Max Pressure [12]: Max Pressure is a distributed traffic signal control method without a priori knowledge for vehicle traffic. It proposes the ‘‘pressure’’ definition and chooses the phase with maximum pressure. The method is analytically proven to maximize the network throughput.
- Webster’s method [21]: Webster’s method is one of the most widely used and classic traffic signal control methods. It has a closed-form solution to a single intersection scenario given some prior knowledge, including saturation rate, traffic volume, and etc. It generates an optimal cycle length and signal plan that minimizes the travel time for vehicles.
- SOTL (Self-Organizing Traffic Light Control) [22]: SOTL is also a single-intersection solution. It takes traffic waiting time and queue length into consideration. The traffic signal change when some statistics exceed a certain threshold, which is a hyper-parameter.
- Max W Pressure: Max W Pressure is a modified version of Max Pressure method mentioned above. In this modified version, the pressure definition is changed from the original vehicle pressure to the whole pressure.

Furthermore, we design two variants of our proposed method as listed below:

- PV-TSC Variant 1: In this variant, the information about pedestrian lane is removed in state definition, the all-red phase will not appear in the phase sequence, safety score part and pedestrian pressure do not exist in reward. This variant is similar to PressLight [13] for vehicle traffic signal control.
- PV-TSC Variant 2: In this variant, the all-red phase and safety score part in reward are removed. This variant is aimed to optimize the traveling efficiency only.

3) *Evaluation Metrics*: We evaluate the performance of those methods from the following three metrics.

- Vehicle/Pedestrian travel time: The travel time for vehicles/pedestrians is the average time that vehicles/pedestrians spend when finishing the whole trip.
- Vehicle/Pedestrian queue length: The queue length for vehicles/pedestrians is the average queue length in vehicle/pedestrian lanes.
- Safety score: As defined in the preliminary section, it reflects the degree of emergency. We average the safety score of all time steps and all intersections in our evaluations.

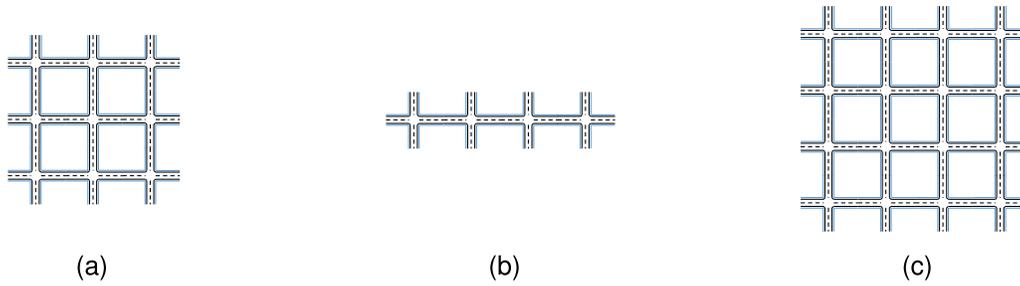


Fig. 6. Road network topology for evaluations, including (a) 3×3 network, (b) 1×4 network, (c) 4×4 network.

Model parameter	Value
Reward coefficient α_1	1
Reward coefficient α_2	1
Reward coefficient α_3	1
Exploration rate ϵ	0.03
Batch size	32
M(#episode)	1000
γ for future reward	0.8
Replay memory size	1500

4) *PV-TSC Parameter Settings*: We adopt a full-connected layer network with 3 hidden layers in our experiments, which is not computationally intensive and has enough representational power in this scenario. Other parameters settings of PV-TSC is shown in Table III.

B. Performance Analysis

1) *Comparison With Traditional Methods*: We evaluate the traditional methods and our proposed PV-TSC in configuration 1. The evaluation results are shown in the Table IV.

In Table IV, among all the evaluated methods, our proposed method PV-TSC has an advantage over other methods in all the evaluated metrics. The reduction for travel time is approximately about 40% (vehicle) and 13% (pedestrian) versus Fixed Time control. Closer inspection of the table can lead us to find that actually, the reduction of travel time is close to the reduction of waiting time. The reduction for waiting time is more prominent, about 60% for the vehicle traffic and 45% for pedestrian traffic. Though pedestrian and vehicle routes are generated using the same algorithm, the average waiting time for pedestrians is larger than that of vehicles. The reason is that the pedestrians sometimes cross the intersection to the diagonal and cross two crosswalks. Therefore the waiting time is larger. Table IV also reveals that Max Pressure has the longest pedestrian waiting time, which can be attributed to its unfixed phase sequence. Compared with Max Pressure, other cycle-based signal plans can ensure there is no infinite waiting and thus has better performance. The performance of the modified version Max W Pressure has been greatly improved because the definition of whole pressure takes pedestrians into account. As for the safety score, because the Max Pressure has a larger pedestrian travel time, the traffic signal control efficiency of pedestrian traffic is at disadvantage. In other words, the pedestrian traffic is larger than that in other cases, which may contribute more to safety score.

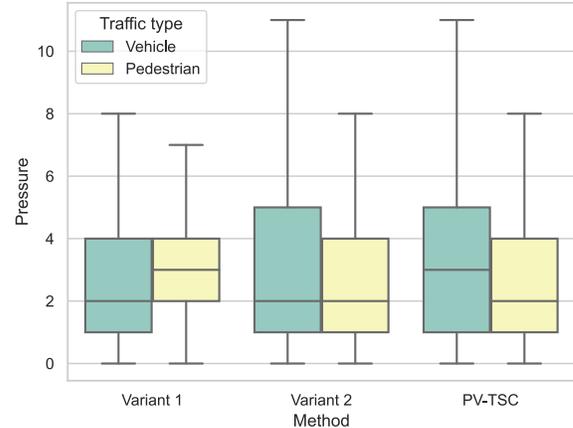


Fig. 7. Pressure for PV-TSC and its variants.

2) *Comparison With Variants of PV-TSC*: To further validate the effectiveness of our PV-TSC design, we compare the PV-TSC with the two variants mentioned above. The corresponding evaluations are PV-TSC and two variants (Variant 1 and Variant 2) in configuration 1. Variant 1 is aimed to optimize the vehicle travel time while ignoring pedestrian travel time and safety issues, thus the performances of pedestrian-related travel time, waiting time, safety score are in inferior positions as shown in Table IV. Variant 2 removes the safety consideration and show an edge over Variant 1 in pedestrian travel time and waiting time, but the safety score of Variant 2 is still large. In Fig. 7, we show the average intersection pressure of two types of traffic. Lower pressure indicates better performance. Variant 1 shows an advantage in vehicle pressure and falls short in pedestrian pressure. Variant 2 shows good performance in both two types of pressure as it is aimed to optimize both. PV-TSC adds safety concerns, thus the performance is degraded compared with Variant 2. To conclude, the complete PV-TSC shows good performance in overall, and demonstrates the efficiency and safety of our design from the side.

3) *Comparison in Different Configuration Settings*: We evaluate the performance of PV-TSC in terms of different traffic networks and volume ratios, for extensively demonstrating the superiority of our method.

Different network topologies: We compare the simulation results for different road network types. The corresponding experiments are evaluated in the 1, 2, 3 configuration in Table II. The traffic volume increases accordingly with the number of intersections. First, we plot the queue lengths over time in Fig. 8 for different road network types. We intercepted

TABLE IV
EVALUATION RESULTS OF DIFFERENT METHODS UNDER DIFFERENT CONFIGURATIONS

Method	Configuration	Vehicle travel time(s)	Pedestrian travel time(s)	Vehicle waiting time(s)	Pedestrian waiting time(s)	Safety score
Max Pressure	1	96.12	336.84	45.92	119.82	0.1254
Webster	1	106.04	301.63	54.09	84.18	0.0565
Fixed Time	1	150.93	313.54	98.78	95.78	0.0442
SOTL	1	119.72	296.09	62.08	79.45	0.0435
Max W Pressure	1	104.73	280.33	53.46	65.69	0.0332
PV-TSC	1	90.24	270.42	39.82	54.00	0.0137
Variant 1	1	87.03	288.35	36.13	71.93	0.0612
Variant 2	1	89.71	269.60	39.17	52.63	0.0602
PV-TSC	4	95.74	278.55	45.19	63.57	0.0204
PV-TSC	5	102.88	271.98	51.92	55.30	0.0179
PV-TSC	6	88.62	272.41	37.93	57.62	0.0094
PV-TSC	7	92.74	267.02	41.44	51.57	0.0083

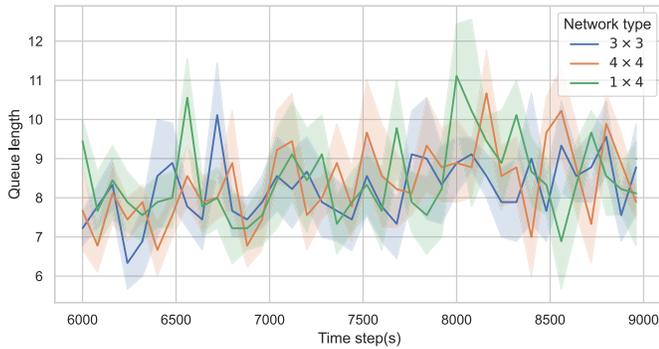


Fig. 8. Queue length versus time in different road network topology.

a time period [6800, 6900] in the simulation. The queue length at an intersection is the sum of waiting vehicles and waiting pedestrians. To make a fair comparison, the results are further averaged by the number of intersections. The solid line in the graph indicates the mean queue length, and the light-colored area around it indicates the variance. From the curves and areas in the figure, we can tell that the queue lengths are relatively stable. More importantly, there is no significant separation of the queue length curves and areas for the three types of road networks, which demonstrates the scalability of our distributed approach when applied to road networks of different sizes.

Different volume ratios: To compare the results in different volume ratios, we set two extra environmental configurations 4 and 5 in Table II: one with larger vehicle traffic, the other with larger pedestrian traffic. They correspond to the downtown area (larger pedestrian traffic) and suburbs (larger vehicle traffic). From the results, we can conclude with some insights: 1) With a constant total traffic volume, the more the share of a single type of traffic, the more its waiting time and travel time. As shown in Table IV, the waiting time and travel time in configuration 6 (PV-TSC with larger pedestrian traffic share) increases compared to configuration 1. 2) With a larger traffic volume, the waiting time and travel time of both two types of traffic increase. We doubled the pedestrian traffic volume and keep the vehicle traffic volume constant in configuration 4 compared to configuration 1, resulting in an increase in travel time for both waiting times and travel times. The correlation between the two is also illustrated from the side. 3) What’s more, the performance gap in different traffic ratios is within acceptable limits, which shows the stability and robustness of our PV-TSC method.

V. RELATED WORKS

A. Traffic Signal Control With Reinforcement Learning

Classical algorithms in the past usually modeled the traffic control problem as an optimization problem with multiple prior assumptions on the model that one needs to specify manually. However, when the assumptions do not match with the reality, the performance of the model degrades.

Researchers need methods that do not rely on realistic priori assumptions, so many studies resort to reinforcement learning. What these approaches have in common is that they quantify some information about the road, pass it to the model, and the model makes judgments based on the quantified information to manipulate the changes in the traffic signals. With the development of deep learning, deep reinforcement learning came into being, and deep learning models have given reinforcement learning a stronger ability to fit state-action pair value.

Many studies in traffic signal control using reinforcement learning exist, you can refer to the survey [23]. To describe the intersection environment and approximate objective (optimization objective), researchers have used queue length, waiting time, travel speed as state or reward. For the action, the literature has practiced several schemes: pick a phase, control the phase time or ratio, keep or change. The learning methods can be divided into value-based and policy-based. Value-based methods are to approximate the state-value function or state-action pair value. Policy-based method directly update policy parameters to maximize the objective return. Some techniques in other related areas are also adopted. With the success of image feature extraction in computer vision, some studies [15], [24], [25] draw on the this idea and use 2D density image to represent state. However, some researchers [13], [17] argue that the complicated state features and reward do not necessarily bring good performance. Reference [10] adopted graph neural network in reinforcement learning to facilitate communication between neighboring intersections. Reference [26] used the demonstrations in traditional traffic signal control to accelerate learning, which is similar to behavior cloning and mastering.

B. Traffic Signal Control With Pedestrian Access

Some previous studies [27], [28] examine the trade-offs and relationship between pedestrian and vehicle traffic flows.

Other research related to pedestrians mainly focus on improving the intersection efficiency, reducing the travel time for the two traffic flows. Most of research use mathematical programming approaches [29], [30], meta-heuristic algorithms [31]–[34] solve the traffic signal control problem. Most of these works pay attention to the single-intersection optimization, which may not apply to the multi-intersection scenario. On the other hand, For multiple intersection scenarios, most solutions adopt centralized algorithms, which may not scale well and lack timely feedback on intersection situations.

Some work consider the safety issue of pedestrians. References [35] and [36] indicate that the coordination of traffic signals can improve intersection safety. Reference [37] consider to reduce the accident rate by adding an dynamic all-red phase, the duration of which depends on the number of non-compliant pedestrians. References [31] and [32] add an exclusive pedestrian phase (EPP) based on the original signal plan to accommodate the pedestrian traffic. All the mentioned approaches to solve the safety problem use heuristic algorithms (e.g., genetic algorithms) that may not dynamically adjust to real-time traffic conditions.

Reinforcement learning is also practiced in pedestrian-vehicle mixed flows in [5]. In their solution, distributed multi-agent Q learning is adopted, with neighboring intersections states and information exchange taken into account.

C. 6G Localization and Tracking

Previously, vehicle location detection have been investigated thoroughly even in complicated urban environments [38]. The latency and bandwidth of 6G [39] also fully meets transmission requirements in traffic signal control. For the pedestrian localization and detection, new technologies of 6G create new opportunities. In 6G white paper on sensing and localization [40], 6G aim to develop towards even higher frequency ranges, wider bandwidths, and massive antenna arrays. In turn, this will enable sensing solutions with very fine range, Doppler and angular resolutions, as well as localization to cm-level degree of accuracy. Recent study [8] proposed KEF, which can track position and velocity of the UE (User Equipment), with measurements of the angle of arrival and time of flight information obtained along an outdoor track, to provide a mean accuracy of 24.8 cm at 142 GHz, over 34 UE locations, using a single base station in line-of-sight and non-line-of-sight. Therefore, combing the traffic signal control and 6G localization and tracking is promising.

Most of the previous studies have considered only the vehicle traffic flow, because the information of vehicles can be easily obtained by some sensor data. Since the travel lane is one-way, the direction of car movement can be known by observing which road the car is on. But these are difficult to measure and obtain for pedestrian traffic. Pedestrian lanes are two-way and people are smaller in size, which is difficult to be accurately detected and recognized by ordinary sensors. Many previous traffic signal control studies with pedestrian traffic are mostly based on visual techniques, such as using camera for pedestrian recognition, pose estimation, etc. But the drawback

of visual techniques is that the recognition accuracy decreases when encountering visual obstruction or blurred lens. The 6G wireless localization and tracking technology compensates for this shortcoming by providing stable and accurate pedestrian tracking and localization, thus providing a reliable base service for pedestrian-related traffic signal control.

VI. CONCLUSION

In this paper, we propose the PV-TSC, which controls the traffic signals to coordinate the pedestrian traffic and vehicle traffic with the assistance of reinforcement learning and 6G positioning services. Our further evaluations demonstrate that PV-TSC ensures scalability and improves the safety and efficiency of intersection transportation.

We still admit some limitations of our works. PV-TSC asks for parameter tuning due to the shortcomings of reinforcement learning tuning work. What's more, It s difficult to rank the importance of safety score, vehicle travel time, pedestrian travel time. From this point, our work can be further improved using methods such as the multi-objective deep reinforcement learning method [41].

REFERENCES

- [1] N. McCarthy. *Traffic Congestion Costs U.S. Cities Billions of Dollars Every Year*. Accessed: Mar. 10, 2020. [Online]. Available: <https://www.forbes.com/sites/miallmcCarthy/2020/03/10/traffic-congestion-costs-us-cities-billions-of-dollars-every-year-infographic/>
- [2] INRIX. (2021). *Inrix 2021 Traffic Scorecard Report*. [Online]. Available: <https://inrix.com/scorecard/>
- [3] C. for Disease Control and Prevention. *Pedestrian Safety*. Accessed: Mar. 6, 2020. [Online]. Available: https://www.cdc.gov/transportationsafety/pedestrian_safety/index.html
- [4] C. Chen *et al.*, "Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 4, 2020, pp. 3414–3421.
- [5] Y. Liu, L. Liu, and W.-P. Chen, "Intelligent traffic light control using distributed multi-agent Q learning," in *Proc. IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2017, pp. 1–8.
- [6] A. Dominguez-Sanchez, M. Cazorla, and S. Orts-Escolano, "Pedestrian movement direction recognition using convolutional neural networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3540–3548, Dec. 2017.
- [7] P. Khomchuk, I. Stainvas, and I. Bilik, "Pedestrian motion direction estimation using simulated automotive MIMO radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 52, no. 3, pp. 1132–1145, Jun. 2016.
- [8] O. Kanhere and T. S. Rappaport, "Outdoor sub-THz position location and tracking using field measurements at 142 GHz," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.
- [9] L. A. Prashanth and S. Bhatnagar, "Reinforcement learning with average cost for adaptive control of traffic lights at intersections," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 1640–1645.
- [10] H. Wei *et al.*, "CoLight: Learning network-level cooperation for traffic signal control," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1913–1922.
- [11] H. Wei, G. Zheng, V. Gayah, and Z. Li, "A survey on traffic signal control methods," 2020, *arXiv:1904.08117*, doi: [10.48550/arXiv.1904.08117](https://doi.org/10.48550/arXiv.1904.08117).
- [12] P. Varaiya, "Max pressure control of a network of signalized intersections," *Transp. Res. C, Emerg. Technol.*, vol. 36, pp. 177–195, Nov. 2013.
- [13] H. Wei *et al.*, "PressLight: Learning max pressure control to coordinate traffic signals in arterial network," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1290–1298.
- [14] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.
- [15] H. Wei, G. Zheng, H. Yao, and Z. Li, "IntelliLight: A reinforcement learning approach for intelligent traffic light control," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2496–2505.

[16] E. Van der Pol and F. A. Oliehoek, "Coordinated deep reinforcement learners for traffic light control," *Proc. Learn., Inference Control Multi-Agent Syst. (NIPS)*, 2016, pp. 1–8.

[17] G. Zheng *et al.*, "Diagnosing reinforcement learning for traffic signal control," 2019, *arXiv:1905.04716*.

[18] R. Chen, J. Hu, M. W. Levin, and D. Rey, "Stability-based analysis of autonomous intersection management with pedestrians," *Transp. Res. C, Emerg. Technol.*, vol. 114, pp. 463–483, May 2020.

[19] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.

[20] P. A. Lopez *et al.*, "Microscopic traffic simulation using SUMO," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2575–2582.

[21] F. V. Webster, "Traffic signal settings," Dept. Sci. Ind. Res., London, U.K., Tech. Rep. 39, 1958.

[22] S.-B. Cools, C. Gershenson, and B. D'Hooghe, "Self-organizing traffic lights: A realistic simulation," in *Advances in Applied Self-Organizing Systems*. Cham, Switzerland: Springer, 2013, pp. 45–55.

[23] H. Wei, G. Zheng, V. Gayah, and Z. Li, "Recent advances in reinforcement learning for traffic signal control: A survey of models and evaluation," *ACM SIGKDD Explor. Newslett.*, vol. 22, no. 2, pp. 12–18, Jan. 2021.

[24] J. A. Calvo and I. Dusparic, "Heterogeneous multi-agent deep reinforcement learning for traffic lights control," in *Proc. AICS*, 2018, pp. 2–13.

[25] M. Coskun, A. Baggag, and S. Chawla, "Deep reinforcement learning for traffic light optimization," in *Proc. IEEE Int. Conf. Data Mining Workshops (ICDMW)*, Nov. 2018, pp. 564–571.

[26] G. Zheng *et al.*, "Learning phase competition for traffic signal control," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, Nov. 2019, pp. 1963–1972.

[27] M. Ishaque and R. Noland, "Multimodal microsimulation of vehicle and pedestrian signal timings," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1939, pp. 107–114, Dec. 2005.

[28] M. M. Ishaque and R. B. Noland, "Trade-offs between vehicular and pedestrian traffic using micro-simulation methods," *Transp. Policy*, vol. 14, no. 2, pp. 124–138, 2007.

[29] C. Yu, W. Ma, K. Han, and X. Yang, "Optimization of vehicle and pedestrian signals at isolated intersections," *Transp. Res. B, Methodol.*, vol. 98, pp. 135–153, Apr. 2017.

[30] Y. Zhang, R. Su, K. Gao, and Y. Zhang, "Traffic light scheduling for pedestrians and vehicles," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, Aug. 2017, pp. 1593–1598.

[31] W. Ma, Y. Liu, and K. L. Head, "Optimization of pedestrian phase patterns at signalized intersections: A multi-objective approach," *J. Adv. Transp.*, vol. 48, no. 8, pp. 1138–1152, 2014.

[32] W. Ma, D. Liao, Y. Liu, and H. K. Lo, "Optimization of pedestrian phase patterns and signal timings for isolated intersection," *Transp. Res. C, Emerg. Technol.*, vol. 58, pp. 502–514, Sep. 2015.

[33] M. Li, W. Alhajyaseen, and H. Nakamura, "A traffic signal optimization strategy considering both vehicular and pedestrian flows," in *Proc. Compendium Papers CD-ROM, 89th Annu. Meeting Transp. Res. Board*, 2010, pp. 10–14.

[34] Y. Zhang, K. Gao, Y. Zhang, and R. Su, "Traffic light scheduling for pedestrian-vehicle mixed-flow networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1468–1483, Apr. 2019.

[35] F. Guo, X. Wang, and M. A. Abdel-Aty, "Modeling signalized intersection safety with corridor-level spatial correlations," *Accident Anal. Prevention*, vol. 42, no. 1, pp. 84–92, Jan. 2010.

[36] S. Midenet, N. Saunier, and F. Boillot, "Exposure to lateral collision in signalized intersections with protected left turn under different traffic control strategies," *Accident Anal. Prevention*, vol. 43, no. 6, pp. 1968–1978, Nov. 2011.

[37] Y. Zhang, Y. Zhang, and R. Su, "Pedestrian-safety-aware traffic light control strategy for urban traffic congestion alleviation," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 178–193, Jan. 2021.

[38] X. Wang *et al.*, "VLD: Smartphone-assisted vertical location detection for vehicles in urban environments," in *Proc. 19th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw. (IPSN)*, Apr. 2020, pp. 25–36.

[39] K. Z. Ghafoor *et al.*, "Millimeter-wave communication for internet of vehicles: Status, challenges, and perspectives," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 8525–8546, Sep. 2020.

[40] A. Bourdoux *et al.*, "6G white paper on localization and sensing," 2020, *arXiv:2006.01779*.

[41] T. T. Nguyen, N. D. Nguyen, P. Vamplew, S. Nahavandi, R. Dazeley, and C. P. Lim, "A multi-objective deep reinforcement learning framework," *Eng. Appl. Artif. Intell.*, vol. 96, Nov. 2020, Art. no. 103915.



Kangjie Xu (Student Member, IEEE) received the B.Eng. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2020, where he is currently pursuing the master's degree with the Department of Computer Science and Engineering. His research interests include computer networking and mobile computing.



Junqin Huang (Student Member, IEEE) received the B.Eng. degree in computer science and technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include crowdsensing, the Internet of Things, blockchain, and mobile computing.



Linghe Kong (Senior Member, IEEE) received the B.Eng. degree in automation from Xidian University in 2005, the master's degree in telecommunication from Telecom SudParis in 2007, and the Ph.D. degree in computer science from Shanghai Jiao Tong University in 2013. He is currently a Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. Before that, he was a Post-Doctoral Researcher with Columbia University, McGill University, and the Singapore University of Technology and Design. His

research interests include the Internet of Things, 5G, blockchain, and mobile computing.



Jiadi Yu (Senior Member, IEEE) received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2007. He is currently an Associate Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University. Prior to join Shanghai Jiao Tong University, he was at the Stevens Institute of Technology, USA, as a Post-Doctoral Researcher. He has published more than 100 refereed papers in international leading journals and key conferences in the areas of wireless communications and networking, mobile computing, and security and privacy. His current research interests include mobile computing and sensing, cyber security and privacy, the Internet of Things (IoT), and smart healthcare. He is a member of the IEEE Communication Society.



Guihai Chen received the B.S. degree from Nanjing University in 1984, the M.E. degree from Southeast University in 1987, and the Ph.D. degree from The University of Hong Kong in 1997. He is currently a Distinguished Professor with Shanghai Jiao Tong University, China. He was a Visiting Professor with many universities, including the Kyushu Institute of Technology, Japan, in 1998; The University of Queensland, Australia, in 2000; and Wayne State University, USA, from 2001 to 2003. His research interests include sensor networks, peer-to-peer computing, and high-performance computer architecture and combinatorics.