



# EMVLight: A multi-agent reinforcement learning framework for an emergency vehicle decentralized routing and traffic signal control system

Haoran Su <sup>a,1</sup>, Yaofeng D. Zhong <sup>b,1</sup>, Joseph Y.J. Chow <sup>a</sup>, Biswadip Dey <sup>b</sup>, Li Jin <sup>c,\*</sup>

<sup>a</sup> New York University, USA

<sup>b</sup> Siemens Corporation, Technology, USA

<sup>c</sup> UM Joint Institute and Department of Automation, Shanghai Jiao Tong University, China



## ARTICLE INFO

### Keywords:

Emergency vehicle management  
Traffic signal control  
Deep reinforcement learning  
Multi-agent system

## ABSTRACT

Emergency vehicles (EMVs) play a crucial role in responding to time-critical calls such as medical emergencies and fire outbreaks in urban areas. Existing methods for EMV dispatch typically optimize routes based on historical traffic-flow data and design traffic signal pre-emption accordingly; however, we still lack a systematic methodology to address the coupling between EMV routing and traffic signal control. In this paper, we propose EMVLight, a decentralized reinforcement learning (RL) framework for joint dynamic EMV routing and traffic signal pre-emption. We adopt the multi-agent advantage actor–critic method with policy sharing and spatial discounted factor. This framework addresses the coupling between EMV navigation and traffic signal control via an innovative design of multi-class RL agents and a novel pressure-based reward function. The proposed methodology enables EMVLight to learn network-level cooperative traffic signal phasing strategies that not only reduce EMV travel time but also shortens the travel time of non-EMVs. Simulation-based experiments indicate that EMVLight enables up to a 42.6% reduction in EMV travel time as well as an 23.5% shorter average travel time compared with existing approaches.

## 1. Introduction

Emergency vehicles (EMVs) include ambulances, fire trucks, and police cars, which respond to critical events such as medical emergencies, fire disasters, and public security crisis. Emergency response time is the key indicator of a city's incidents management ability and resiliency. Reducing response time saves lives and prevents property losses. For instance, Berdowski et al. (2010) indicates that the survivor rate from a sudden cardiac arrest without treatment drops 7%–10% for every minute elapsed, and there is barely any chance to survive after 8 min. EMV travel time, the time interval for an EMV to travel from a rescue station to an incident site, accounts for a major portion of the emergency response time. However, overpopulation and urbanization have exacerbated road congestion, making it more challenging to reduce the average EMV travel time. Records (Analytics, 2021) have shown that even with a decline in average emergency response time, the average EMV travel time increased from 7.2 min in 2015 to 10.1 min in 2021 in New York City, an approximately 40% increase over six years even accounting for post-Covid traffic conditions. Therefore, there is a severe urgency and significant benefit for shortening the average EMV travel time on increasingly crowded roads.

\* Corresponding author.

E-mail address: [li.jin@sjtu.edu.cn](mailto:li.jin@sjtu.edu.cn) (L. Jin).

<sup>1</sup> Equal contribution.

Existing work has studied strategies to reduce the travel time of EMVs by route optimization and traffic signal pre-emption (Lu and Wang, 2019; Humagain et al., 2020). Route optimization refers to the search for a time-based shortest path. The traffic network (e.g., city road map) is modeled as a graph with intersections as nodes and road segments between intersections as edges. Based on the time a vehicle needs to travel through each edge (road segment), route optimization calculates an optimal route with the minimal EMV travel time (Humagain et al., 2020). In addition, as the EMV needs to be as fast as possible, the law in most places requires non-EMVs to yield to emergency vehicles sounding sirens, regardless of the traffic signals at intersections (De Lorenzo and Eilers, 1991). Even though this practice gives the right-of-way to EMVs, it poses safety risks for vehicles and pedestrians at intersections (Grant, 2017). To address this safety concern, existing methods (Nelson and Bullock, 2000; Qin and Khan, 2012; Humagain et al., 2020; Huang et al., 2015) have also studied traffic signal pre-emption which refers to the process of deliberately altering the signal phases at each intersection to prioritize EMV passage.

However, a major challenge for adaptive EMV operation is the coupling between EMV route optimization and traffic light pre-emption (Humagain et al., 2020). As the traffic condition constantly changes, static route optimization can potentially become suboptimal as an EMV travels through the network; i.e. the traffic is highly time-dependent and exhibits transient properties during a dispatch (Coogan and Arcak, 2015). Moreover, traffic signal pre-emption has a significant impact on the traffic flow, which would change the fastest route as well. Thus, the optimal route should be updated with real-time traffic flow information, i.e., the route optimization should be solved in a dynamic (time-dependent) way. As an optimal route can change as an EMV travels through the traffic network, the traffic signal pre-emption would need to adapt accordingly. In other words, the subproblems of dynamic route optimization and traffic signal pre-emption are coupled and should be solved ideally simultaneously in real-time. Existing approaches have limited consideration to this coupling.

In addition, most of the existing models on emergency vehicle service have a single objective of reducing the EMV travel time (Haghani et al., 2004, 2003; Panahi and Delavar, 2008; Shaaban et al., 2019). As a result, their traffic signal control strategies have an undesirable effect of increasing the travel time of non-EMVs. Non-EMVs on the path of approaching EMVs would pull over or stop, and they usually do not receive clear guidance from traffic signals on resuming their trips (Hsiao et al., 2018), causing unnecessary delay. Non-EMVs elsewhere are also likely indirectly and negatively impacted if adjacent intersections lack coordination to address the incurred delay (Humagain et al., 2020). Therefore, traffic signal control strategies accommodating both EMVs and non-EMVs need to be recommended.

In this paper, we aim to perform route optimization and traffic signal pre-emption to not only reduce EMV travel time but also to reduce the average travel time of non-EMVs. In particular, we address the following two key challenges:

1. **How to dynamically route an EMV to a destination under time-dependent traffic conditions in a computationally efficient way?** As the congestion level of each road segment changes over time, the routing algorithm should be able to update the remaining route as the EMV passes each intersection. Running the shortest-path algorithm each time the EMV passes through an intersection is not efficient. A computationally efficient decentralized routing algorithm is desired.
2. **How to coordinate traffic signals to not only reduce EMV travel time but reduce the average travel time of non-EMVs as well?** To reduce EMV travel time, only the traffic signals along the route of the EMV need to be altered. However, to further reduce average non-EMV travel time, traffic signals in the whole traffic network need to cooperate.

Reinforcement learning (RL), which gained significant traction in assorted domains of traffic signal control recently, has been extensively studied and proven effective for learning stochastic traffic conditions and dealing with randomness. Thus, to tackle the above challenges, we propose **EMVLight**, a decentralized multi-agent reinforcement learning framework with a dynamic routing algorithm to control traffic signal phases for efficient EMV passage. In the proposed RL design, each intersection is an agent and each agent is responsible for deciding local traffic signal phases. Multiple agents coordinate to facilitate EMV passage as well as alleviate congestion. Our experimental results demonstrate that EMVLight outperforms traditional traffic engineering methods and existing RL methods under two metrics – EMV travel time and the average travel time of all vehicles – on different traffic configurations. We extend the preliminary work (Su et al., 2022) by taking into account extra capacity of each road segments and the possibility of forming “emergency lanes” for full speed EMV passage. In addition, we demonstrate EMVLight’s performance on synthetic and real-world maps with extra capacities. We also present the difference in EMV routing between EMVLight and benchmark methods with an emphasize on the number of successfully formed emergency lanes.

Our contributions are threefold. First, we capture the emergency capacity in road segments for emergencies and incidents. We also propose a mathematical model to decide whether an emergency lane can be formed for full speed EMV passage based on emergency capacity and number of vehicles of a road segment. Second, we incorporate a decentralized path-finding scheme for EMVs based on real time traffic information. Third, we propose to solve EMV routing and traffic signal control problems simultaneously in a multi-agent reinforcement learning framework. In particular, we set up different types of reinforcement learning agents based on the location of the EMV and design different rewards for each type. This leads to up to a 42.6% reduction in EMV travel time as well as an 23.5% shorter average travel time of all trips completed in the network compared with existing benchmark methods.

The rest of the paper is organized as follows. Section 2 reviews relevant literature. Section 3 introduces our definition of pressure and emergency capacity. Section 4 presents our EMVLight methodology, i.e., dynamic routing and reinforcement learning. Benchmark methods and experimental setup are presented in Section 5. Section 6 discussed and compared the performance of EMVLight and benchmark methods in terms of EMV travel time, average travel time of all vehicles as well as EMV route choices. We conclude in Section 7 and share inspirations on future directions.

## 2. Literature review

### 2.1. Conventional routing optimization and traffic signal pre-emption for emvs.

Although, routing and pre-emption are coupled in reality, existing methods usually solve them separately. Many of the existing approaches leverage Dijkstra's shortest path algorithm to get the optimal route (Wang et al., 2013a; Mu et al., 2018; Kwon et al., 2003; Jotshi et al., 2009). An A\* algorithm for ambulance routing was proposed by Nordin et al. (2012). However, this line of work assumed that the routes and traffic conditions are fixed and static and fails to address the dynamic nature of real-world traffic flows. Another line of work considered the change of traffic flows over time. Ziliaskopoulos and Mahmassani (1993) proposed a shortest-path algorithm for time-dependent traffic networks, but the travel time associated with each edge at each time step is assumed to be known prior. Musolino et al. (2013) proposed different routing strategies for different times in a day (e.g., peak/non-peak hours) based on traffic history data at those times. However, in the problem of our consideration, routing and pre-emption strategies can significantly affect the travel time associated with each edge during the EMV passage, and the existing methods cannot deal with this kind of real-time changes. Haghani et al. (2003) formulated the dynamic shortest path problem as a mixed integer programming problem. Koh et al. (2020) have used RL for real-time vehicle navigation and routing. Related lines of work (Miller-Hooks and Mahmassani, 2000; Gao and Chabini, 2006; Kim et al., 2005; Fan et al., 2005; Yang and Zhou, 2014; Huang and Gao, 2012; Gao and Huang, 2012; Samaranayake et al., 2012; Nie et al., 2012; Thomas and White III, 2007) tackled the optimal adaptive routing problem in a variety of stochastic and time-dependent settings. However, none of these studies have considered the traffic signal pre-emption coupling, nor did they took the context of EMV passage into account solving shortest path problems.

Once an optimal route for the EMV has been determined, traffic signal pre-emption is deployed to further reduce the EMV travel time. A common pre-emption strategy is to extend the green phases of green lights to let the EMV pass each intersection along a fixed optimal route (Wang et al., 2013a; Bieker-Walz and Behrisch, 2019). Pre-emption strategies for multiple EMV requests were introduced by Asaduzzaman and Vidyasankar (2017). Wu et al. (2020) approached the emergency vehicle lane clearing from an microscopic motion planning perspective and Hosseinzadeh et al. (2022) investigated an EMV-centered traffic control scheme through multiple intersections to alleviate traffic congestion. These work scrutinized strategies on pre-emption, but they had not considered EMV's dynamic routing which leads to the optimal path.

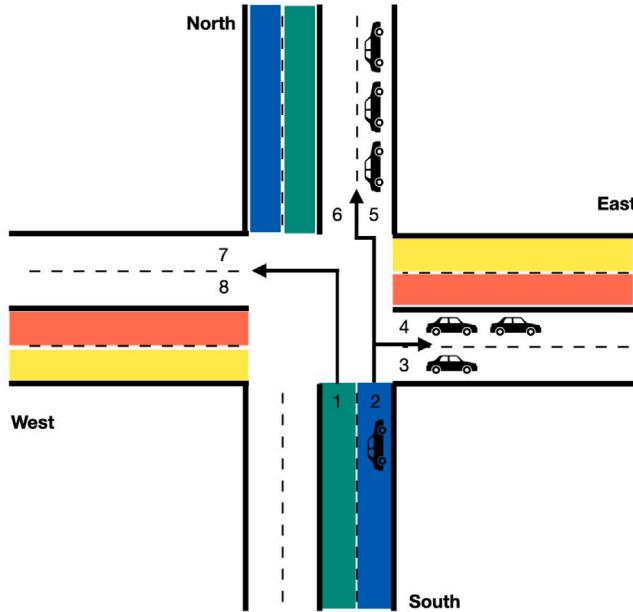
Please refer to Lu and Wang (2019) and Humagain et al. (2020) for a thorough survey of conventional routing optimization and traffic signal pre-emption methods. We would also like to point out that the conventional methods prioritize EMV passage and have significant disturbances on the traffic flow which increases the average non-EMV travel time.

### 2.2. Rl-based traffic signal control

Traffic signal pre-emption only alters the traffic phases at the intersections where an EMV travels through. However, to reduce congestion, traffic phases at nearby intersections also need to be changed cooperatively. The coordination of traffic signals to mitigate traffic congestion is referred to as traffic signal control which has been addressed by leveraging deep RL in a growing body of work. Abdulhai et al. (2003) is among the first to explore Q-learning, which is a model-free reinforcement learning method to learn the optimal action in a particular state, for adaptive traffic signal control. The idea was outlined for an isolated traffic signal controller as well as a network of traffic signal controllers. However, the size of the state representation grows exponentially as more traffic signals are considered and the learning process becomes extremely expensive. Only the isolated scenario was numerically shown to have an advantage over traditional methods. To address this curse of dimensionality, Prashanth and Bhatnagar (2010) introduced feature-based state representations, which grows linearly in the number of lanes considered. The improved state representation reduces computational complexity and improves performance. El-Tantawy et al. (2013) avoided the curse of dimensionality by introducing game-theoretic approaches into Q-learning where each agent converges to the best response policy to all neighbors' policies.

The increasing compute power in recent years enables the training of deep neural networks, in particular, the deep Q-learning RL method. Van der Pol and Oliehoek (2016) are the first to incorporate deep Q network (DQN) into traffic jam assessment and extend it to coordinated adjacent intersections. Based on that inspiration, a line of work proposes to leverage the Q-learning framework for traffic signal control. CoLight (Wei et al., 2019b) facilitated communication and cooperation between nearby traffic signals by graph attentional networks. FRAP (Zheng et al., 2019) improved RL-based methods by adding a phase competition model. PressLight (Wei et al., 2019a) incorporated the method of max pressure, which aggressively selects the traffic signal phase with maximum pressure to smooth congestion (Varaiya, 2013; Li and Jabari, 2019; Levin et al., 2020; Wang et al., 2022; Lazar et al., 2021), into the reward design and shows improvement over traditional reward design. Chen et al. (2020) combined the idea from FRAP (Zheng et al., 2019) and PressLight (Wei et al., 2019a) to achieve city-level traffic signal control. Zang et al. (2020) leveraged meta-learning algorithms to speed up Q-learning for traffic signal control. This line of Q-learning methods assumed the traffic signal control agents to be homogeneous and the neural network parameters are shared among all the agents, which enables the methods to scale up. However, this assumption makes it hard to adapt these methods to non-regular traffic grids where different intersections might have different numbers of incoming and outgoing links.

The actor-critic method is another category of reinforcement learning methods which does not suffer from the curse of dimensionality challenge in multi-agent settings. Aslani et al. (2017) studied actor-critic methods for adaptive traffic signal control under different traffic disruption scenarios and show that actor-critic controllers outperforms Q-learning-based controllers. Xu et al. (2021) proposed a hierarchical actor-critic method to encourage cooperation between intersections. Chu et al. (2019) leveraged



**Fig. 1.** Traffic movements of a four-way two-lane bidirectional intersection.

a multi-agent independent advantage actor–critic algorithm for adaptive traffic signal control. Ma and Wu (2020) introduced a manager-worker hierarchy and split the entire traffic networks into unique regions, so that it can manage traffic conditions locally based on the actor–critic framework. Existing RL-based traffic control methods focus on reducing the congestion in the traffic network and are not designed specifically for EMV pre-emption. In contrast, our RL framework is built upon state-of-the-art ideas such as max pressure and is designed to reduce both the EMV travel time and the overall congestion. The centralized training with decentralized execution framework empowers the scheme to be compatible with stochastic traffic settings and realize negligible communication and synchronization cost among vehicles. Under the actor–critic framework, Mo et al. (2022) investigated the traffic signal control strategy based on connected vehicle communications. We recommend (Noaeen et al., 2022; Wei et al., 2019c) for a comprehensive and systematic review on RL-based traffic signal control methods. Although RL methods have been extensively studied on the particular problem of traffic signal control and proven beneficial and effective, no existing study has considered EMV passages as well as their negative impacts on non-EMVs traffic flow.

### 2.3. Intra-link emv traversal strategies.

Recent advances of vehicle-to-vehicle communication technology (LeBrun et al., 2005) have drawn attention to designing optimal link level traversal strategies for EMVs in congested roads. Agarwal and Paruchuri (2016) proposed Fixed Lane Strategy (FLS) and Best Lane Strategy (BLS) for EMVs to travel through a congested urban roadway. Based on the findings of Agarwal and Paruchuri (2016), Ismath et al. (2019) studied the performance of both strategies from a communication perspective and conclude that FLS outperforms BLS when speed variance is small. Hannoun et al. (2019, 2021) designed and improved a semi-automated warning system to instruct downstream traffic to yield using mixed integer linear programming. The proposed strategy adapts to varying market penetration rate of connected vehicles and is validated to be computationally efficient for the real-time deployment. Su et al. (2021) suggested the dynamic queue-jump lane (DQJL) strategy using multi-agent reinforcement learning.

## 3. Preliminaries

In this section, we introduce relevant preliminary terms and definitions which facilitate the problem formulation.

### 3.1. Traffic map, movement and signal phase

A traffic map can be represented by a graph  $G(\mathcal{V}, \mathcal{E})$ , with intersections as nodes and road segments between intersections as edges. We refer to a one-directional road segment between two intersections as a link. A link has a fixed number of lanes, denoted as  $h(l)$  for lane  $l$ . Vehicles are allowed to switch lane between two intersections. Fig. 1 shows 8 links and each link has 2 lanes. A traffic movement  $(l, m)$  is defined as the traffic traveling across an intersection from an incoming lane  $l$  to an outgoing lane  $m$ . The intersection shown in Fig. 1 has 24 permissible traffic movements. As an example, vehicles on lane 1 are turning left, and vehicles on lane 2 may go straight or turn right. After turning into the link, vehicles will enter either lane. Thus, the incoming South link has

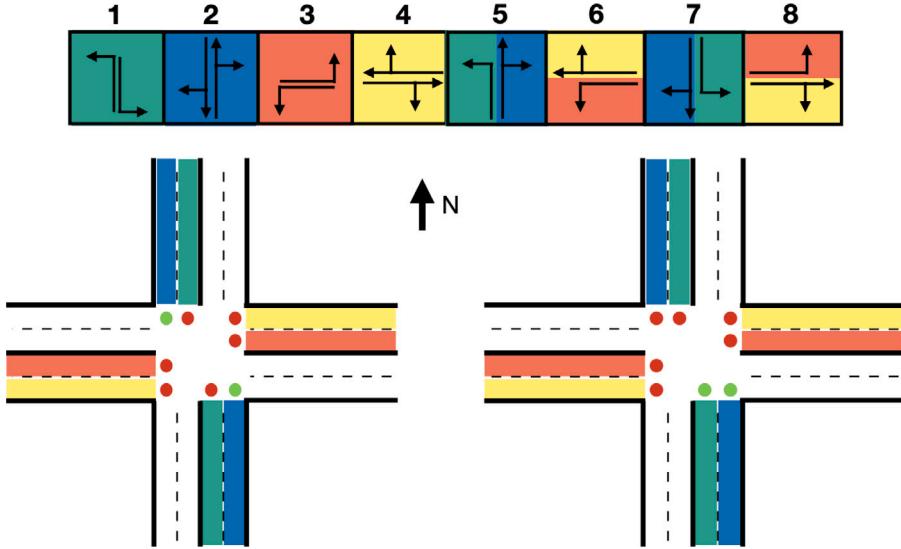


Fig. 2. Top: 8 signal phases; Left: phase #2 illustration; Right: phase #5 illustration.

the potential traffic movements of  $\{(1, 7), (1, 8), (2, 5), (2, 6), (2, 3), (2, 4)\}$ . The set of all permissible traffic movements of an intersection is denoted as  $\mathcal{M}$ .

A traffic signal phase is defined as the set of permissible traffic movements. As shown in Fig. 2, an intersection with 4 links has 8 phases.

### 3.2. Pressure

The pressure of an incoming lane  $l$  measures the unevenness of vehicle density between lane  $l$  and corresponding outgoing lanes in permissible traffic movements. The vehicle density of a lane is  $x(l)/x_{max}(l)$ , where  $x(l)$  is the number of vehicles on lane  $l$  and  $x_{max}(l)$  is the vehicle capacity on lane  $l$ , which is related to the length of a lane. Then the pressure of an incoming lane  $l$  is

$$w(l) = \left| \frac{x(l)}{x_{max}(l)} - \sum_{\{m|(l,m) \in \mathcal{M}\}} \frac{1}{h(m)} \frac{x(m)}{x_{max}(m)} \right|, \quad (1)$$

where  $h(m)$  is the number of lanes of the outgoing link which contains  $m$ . In Fig. 1,  $h(m) = 2$  for all the outgoing lanes. An example for Eq. (1) is shown in Fig. 1.

Taking the intersection's traffic conditions shown in Fig. 1 as an example, assuming the maximum capacity for each lane is 5 vehicles, we can calculate the lane pressure for lane 2 to be

$$w(2) = \left| \underbrace{\frac{1}{5}}_{\text{lane 2}} - \frac{1}{2} \left( \underbrace{\frac{1}{5} + \frac{2}{5}}_{\text{lane 3 and 4}} \right) - \frac{1}{2} \left( \underbrace{\frac{3}{5} + \frac{0}{5}}_{\text{lane 5 and 6}} \right) \right| = \frac{2}{5}$$

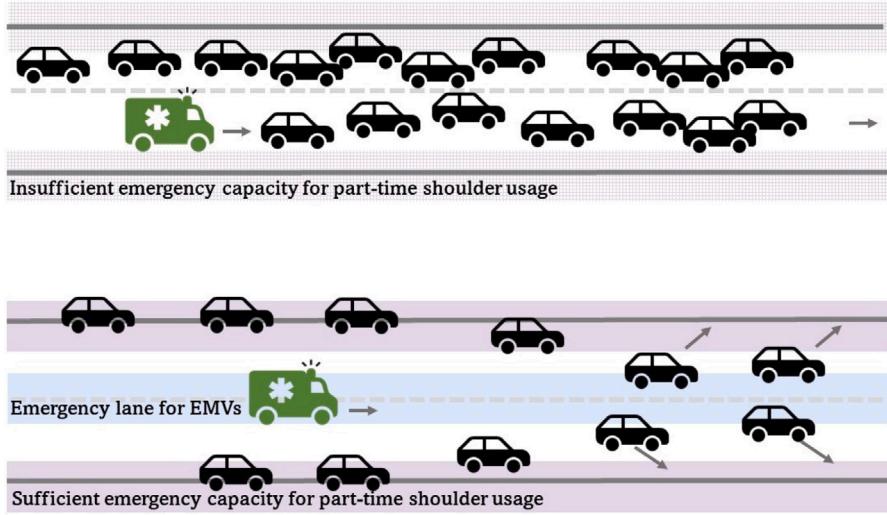
The pressure of an intersection indicates the unevenness of vehicle density between incoming and outgoing lanes in an intersection. Intuitively, reducing the pressure leads to more evenly distributed traffic, which indirectly reduces congestion and average travel time of vehicles. EMVLight defines pressure of an intersection in EMVLight as the average of the pressure of all incoming lanes,

$$P_i = \frac{1}{|\mathcal{I}_i|} \sum_{l \in \mathcal{I}_i} w(l),$$

where  $\mathcal{I}_i$  represents the set of all incoming lanes of intersection  $i$ . According to such definition, the intersection pressure shown in Fig. 1 is computed to be  $\frac{25}{80}$ .

PressLight (Wei et al., 2019a) assumes that traffic movements are lane-to-lane, i.e., vehicles in one lane can only move into a particular lane in a link. Because of the lane-to-lane assumption, in PressLight, the pressure is defined per movement. PressLight defines the pressure of a movement as the difference of the vehicle density between an incoming lane  $l$  and the outgoing lane  $m$ , i.e.,

$$w^*(l, m) = \frac{x(l)}{x_{max}(l)} - \frac{x(m)}{x_{max}(m)},$$



**Fig. 3.** A demonstration of an emergency lane for EMV passage. *Top:* the EMV has to follow the congested queue due to insufficient emergency capacity; *Bottom:* the EMV is traversing on the yielded emergency lane due to sufficient emergency capacity.

For instance, lane 2, shown in Fig. 1, carries  $w^*(2, 3)$ ,  $w^*(2, 4)$ ,  $w^*(2, 5)$ ,  $w^*(2, 6)$ ,  $w^*(2, 7)$  and  $w^*(2, 8)$ . Taking the permissible traffic movement from lane 2 to lane 4 as an example, we can get the pressure for this movement as

$$w^*(2, 4) = \frac{1}{5} - \frac{2}{5} = -\frac{1}{5}$$

PressLight then defines the pressure of an intersection  $i$  as the absolute value of the sum of pressure of movements of intersection  $i$ , i.e., When calculating the pressure intersection, PressLight has

$$P_i^* = \left| \sum_{(l, m) \in \mathcal{M}_i} w^*(l, m) \right|,$$

where  $\mathcal{M}_i$  is the set of permissible traffic movements of intersection  $i$ . According to PressLight's definition of intersection pressure, Fig. 1 shows an intersection with pressure of 6.

*Pressure in our work.* EMVLight assumes a lane-to-link style traffic movement as vehicles can enter either lane on the target link, see Fig. 1. Following lane pressure defined in (1),

*Comparison.* The first difference between the two definitions is that  $w^*(l, m)$  can be both positive or negative, but  $w(l)$  can only take positive values that measures the unevenness of the vehicle density in the incoming lane and that of the corresponding outgoing lanes. We take the absolute value since the direction of pressure is irrelevant here, and the goal of each agent is to minimize this unevenness. The second difference is that at the intersection level,  $P_i^*$  takes a sum but  $P_i$  takes an average. The average is more suitable for our purpose since it scales the pressure down and the unit penalty for normal agents would be relatively large as compared to rewards for pre-emption agents (Eq. (3)). This design puts the efficient passage of EMV vehicles at the top priority. Our experimentation results, presented in Section 6 indicate the proposed pressure design produces a more robust reward signal during training and outperforms PressLight in congestion reduction.

### 3.3. Emergency capacity

A roadway segment may have additional capacity, e.g. shoulder, parking lanes, bike lanes, dedicated to providing extra space that can be used under emergencies and incidents. In the presence of EMVs, existing vehicles are allowed to pull-over or park on the shoulders temporarily, forming an *emergency lane* for the emergency vehicle to pass. An emergency lane is a lane formed between original lanes dedicated for EMV passage, see Fig. 3 bottom. EMVs are assumed to travel freely on the emergency lane through the dense traffic. This is referred as an *emergency yielding* and non-EMVs are experiencing an *part-time shoulder use* (Administration, 2020).

Adaptive traffic management strategies based on part-time shoulder use, such as dynamic hard shoulder running (D-HSR) (Ma et al., 2016), have proven beneficial and cost-effective for such scenarios.

We use an intuitive mathematical model to determine whether such an emergency lane for EMV passage can be established. First, we define *emergency capacity*  $C_i^{EC}$  of a link  $i$  to be the additional capacity in the link for emergencies and incidents. The emergency capacity depends on the segment's shoulder width, lane width, geometric clearance and other factors. We say that a link

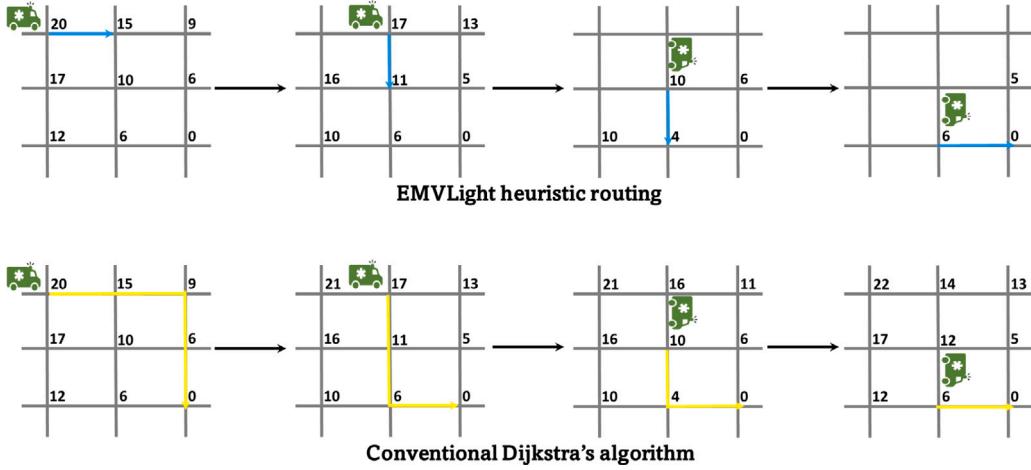


Fig. 4. EMVLight routing (top) vs conventional Dijkstra's routing (bottom).

is *emergency-capacitated* if it has a nonzero emergency capacity. In order to express the maximum number of vehicles allowed in a link, for forming an emergency lane, we assume there are  $n_i$  non-EMVs in the link  $i$  with an average speed of  $s_i$ . We also assume that the normal capacity  $k_i$  is evenly distributed among  $l_i$  lanes so that each lane has a capacity of  $k_i/l_i$ . The overall capacity of the link is then  $k_i + C_i^{\text{EC}}$ . To form an emergency lane for EMV passage with maximum speed, all the non-EMVs need to move out of the emergency lane. Thus the maximum number of vehicles allowed is  $k_i + C_i^{\text{EC}} - k_i/l_i$ . As a result, the travel speed of the EMV is

$$s_i^{\text{EMV}} = \begin{cases} s_f & n_i \leq k_i + C_i^{\text{EC}} - \frac{k_i}{l_i}, \\ s_i & \text{else,} \end{cases} \quad (2)$$

where  $s_f$  represents the maximum speed allowed for EMVs.

#### 4. Methodology

In this section, we elaborate the methodology of EMVLight. We begin with implementing a decentralized shortest path onto EMV navigation in traffic networks, and then incorporate it into the proposed multi-class RL agent design. Subsequently, we introduce the multi-agent advantage actor–critic framework as well as the RL training workflow in details.

##### 4.1. Decentralized routing for EMVLight

Dijkstra's algorithm is an algorithm that finds the shortest path between a given node and every other nodes in a graph. The time-based Dijkstra's algorithm finds the fastest path and has been widely used for EMV routing. In order to find such a path, the EMV travel time along each link need to be estimated first and we refer to it as the *intra-link travel time*. Dijkstra's algorithm takes as input the traffic graph, the intra-link travel time and a destination, and can return the time-based shortest path as well as estimated travel time from each intersection to the destination. The latter is usually referred to as the *estimated time of arrival* (ETA) of each intersection.

In a traffic network, the intra-link travel time usually depends on the link's emergency capacity and number of vehicles in that link. In our model, this dependency is captured by EMV speed Eq. (2). The intra-link travel time is then calculated as the link length divided by the EMV speed. However, traffic conditions are constantly changing and so does EMV travel time along each link. Moreover, EMV pre-emption techniques alter traffic signal phases, which will significantly change the traffic condition as the EMV travels. The pre-determined shortest path might become congested due to stochasticity and pre-emption. Thus, updating the optimal route dynamically can facilitate EMV passage. One option is to run Dijkstra's algorithm repeatedly as the EMV travels through the network in order to take into account the updated EMV intra-link travel time. However, this requires global traffic information of the entire traffic network throughout EMVs' trips. Even when a centralized controller is established to navigate the EMV, the synchronization and communication cost grows exponentially when the network size increases and the nonscalability disallows the centralized scheme to be real-time for the navigation.

Decentralized routing approaches (Adacher et al., 2014; Chen et al., 2006; He et al., 2015) were introduced to find the shortest path with partial observability of the system. However, these models require network decomposition or partitioning in advance, and solve optimal paths with polynomial-bounded time complexity at best (Johnson et al., 2016). Some other approach heavily relies on V2I communication (Mostafizi et al., 2021). Considering the massive amount of iterations of trial-and-error in reinforcement learning, none of these decentralized routing methods provide a suitable design for a learning-based framework.

**Table 1**

Time complexities of the proposed routing heuristic and the dynamic shortest-path approach.  $M$  can be arbitrarily set to determine the updating frequency. The larger  $M$  is, the shorter selected route can be.

	EMVLight heuristic	Dynamic shortest-path based on Dijkstra's
Initialization	$\mathcal{O}( \mathcal{V}  \log  \mathcal{V}  +  \mathcal{E} )$	–
updating	$\mathcal{O}( \mathcal{V} )$	$\mathcal{O}( \mathcal{V}  \log  \mathcal{V}  +  \mathcal{E} )$
updating frequency	$ \mathcal{V} $	$M$

---

**Algorithm 1:** Decentralized routing for EMVs
 

---

```

Input :
   $G = (\mathcal{V}, \mathcal{E})$  traffic map as a graph
   $T^t = [T_{ij}^t]$  intra-link travel time at time  $t$ 
   $i_d$  index of the destination

Output:
   $\text{ETA}^t = [\text{ETA}_i^t]$  ETA of each intersection
   $\text{Next}^t = [\text{Next}_i^t]$  next intersection to go from each intersection
  /* pre-populating */ */
1  $\text{ETA}^0, \text{Next}^0 = \text{Dijkstra}(G, T^0, i_d)$  */
  /* dynamic routing */
2 for  $t = 0 \rightarrow T$  do
3   foreach  $i \in \mathcal{V}$  do (in parallel)
4      $\text{ETA}_i^{t+1} \leftarrow \min_{(i,j) \in \mathcal{E}} (\text{ETA}_j^t + T_{ji}^t)$ 
5      $\text{Next}_i^{t+1} \leftarrow \arg \min_{\{j | (i,j) \in \mathcal{E}\}} (\text{ETA}_j^t + T_{ji}^t)$ 
  
```

---

To achieve efficient decentralized dynamic routing, we extend Dijkstra's algorithm to update the optimal route based on the updated intra-link travel times. As shown in Algorithm 1, first a pre-populating process is carried out where a standard Dijkstra's algorithm is run to get the  $\text{ETA}^0$  from each intersection to the destination. For each intersection, the next intersection  $\text{Next}^0$  along the shortest path is also calculated. For an intersection  $i$ , the result  $\text{ETA}_i^0$  and  $\text{Next}_i^0$  are stored locally in the intersection. We assume this process can be done before the EMV leaves the dispatching hub. This is reasonable since a sequence of processes, including call-taker processing, are performed before the EMVs are dispatched. Once the pre-populating process is finished, we can update ETA and Next for each intersection efficiently in parallel in a decentralized way, since the update only depends on information of neighboring intersections. Fig. 4 provides an example comparing between Algorithm 1 and conventional Dijkstra's algorithm on an 3-by-3 traffic network. The numerical value represents the ETA of each intersection, and it gets updated as described above. EMVLight, rather than solving the full shortest path like the conventional Dijkstra's algorithm, only determines the next link to travel each iteration. A comparison of the worst-case time complexity between the EMVLight routing heuristic and a Dijkstra-based dynamic shortest path method is provided in Table 1. Notice that the Dijkstra's algorithm can be implemented with a Fibonacci heap min-priority queue and solve the shortest path with a time complexity of  $\mathcal{O}(|\mathcal{V}| \log |\mathcal{V}| + |\mathcal{E}|)$  (Fredman and Tarjan, 1984).

By adopting the proposed heuristic routing algorithm, we facilitate the RL agent design, which is introduced in Section 4.2.

**Remark 1.** In static Dijkstra's algorithm, the shortest path is obtained by repeatedly querying the Next attribute of each node from the origin until we reach the destination. In our dynamic Dijkstra's algorithm, since the shortest path changes, at a intersection  $i$ , we only care about the immediate next intersection to go to, which is exactly  $\text{Next}_i$ .

#### 4.2. Reinforcement learning agent design

While dynamic routing directs the EMV to the destination, it does not take into account the possible waiting times for red lights at the intersections. Thus, traffic signal pre-emption is also required for the EMV to arrive at the destination in the least amount of time. However, since traditional pre-emption only focuses on reducing the EMV travel time, the average travel time of non-EMVs can increase significantly. Thus, we set up traffic signal control for efficient EMV passage as a decentralized RL problem. In our problem, each intersection is an RL agent. Each agent makes decisions on the traffic signal phases of this intersection based on local information. Aside from communicating with the approaching EMV, agents are also communicating with each other. Multiple agents coordinate the control signal phases of intersections cooperatively to (1) reduce EMV travel time and (2) reduce the average travel time of non-EMVs. First we design 3 agent types. Then we present agent design and multi-agent interactions.

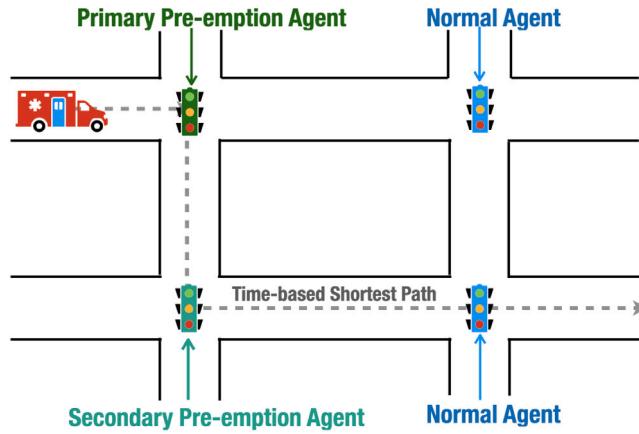


Fig. 5. Types of agents through the EMV's passage to the destination.

#### 4.2.1. Types of agents for EMV passage

When an EMV is on duty, we distinguish 3 types of traffic control agents based on EMV location and routing (Fig. 5). An agent is a *primary pre-emption agent*  $i_p$  if an EMV is on one of its incoming links. The agent of the next intersection  $i_s = \text{Next}_{i_p}$  is referred to as a *secondary pre-emption agent*. The rest of the agents are *normal agents*. We design these types since different agents have different local goals, which is reflected in their reward designs.

#### 4.2.2. Agent design

**State:** The state of an agent  $i$  at time  $t$  is denoted as  $s_i^t$  and it includes the number of vehicles on each outgoing lanes and incoming lanes, the distance of the EMV to the intersection, the estimated time of arrival (ETA), and which link the EMV will be routed to (Next), i.e.,

$$s_i^t = \{x^t(l), x^t(m), d_{\text{EMV}}^t[L_{ji}], \text{ETA}_i^t, \text{Next}_i^t\}, \quad (3)$$

where  $L_{ji}$  represents the links incoming to intersection  $i$  from its adjacent intersections  $j \in \mathcal{N}_i$ . With a slight abuse of notation,  $l$  and  $m$  denote the set of incoming and outgoing lanes, respectively. The vector  $d_{\text{EMV}}^t$  contains the information about the distance of an EMV to the intersection it is present. For the intersection shown in Fig. 1,  $d_{\text{EMV}}^t$  is a vector of four elements. In particular, for primary pre-emption agents, one of the elements represents the distance of EMV to the intersection in the corresponding link and the rest of the elements are set to -1. For all other agents,  $d_{\text{EMV}}^t$  are padded with -1.

**Action:** Prior work has focused on using phase switch, phase duration and phase itself as actions. In this work, we define the action of an agent as one of the 8 phases in Fig. 2; this enables more flexible signal patterns as compared to the traditional cyclical patterns. Due to safety concerns, once a phase has been initiated, it should remain unchanged for a minimum amount of time, e.g. 5 s. Because of this, we set our MDP time step length to be 5 s to avoid rapid switch of phases.

**Reward:** PressLight has shown that minimizing the pressure is an effective way to encourage efficient vehicle passage. For normal agents, we adopt a similar idea, as shown in Eq. (4). For secondary pre-emption agents, we additionally encourage less vehicles on the link where the EMV is about to enter in order to encourage efficient EMV passage. Thus, the reward is a weighted sum of the pressure and this additional term, with a weight  $\beta$ , as shown in Eq. (4). For a default setting of balancing EMV navigation and traffic congestion alleviation, we choose  $\beta = 0.5$ . For primary pre-emption agents, we simply assign a unit penalty at each time step to encourage fast EMV passage, as shown in Eq. (4). Thus, depending on the agent type, the local reward for agent  $i$  at time  $t$  is as follows

$$r_i^t = \begin{cases} -P_i^t & i \notin \{i_p, i_s\}, \\ -\beta P_{i_s}^t - \frac{1-\beta}{|L_{i_p i_s}|} \sum_{l \in L_{i_p i_s}} \frac{x(l)}{x_{\max}(l)} & i = i_s, \\ -1 & i = i_p. \end{cases} \quad (4)$$

**Justification of agent design.** The quantities in local agent state can be obtained at each intersection using various technologies. Numbers of vehicles on each lane ( $x^t(l), x^t(m)$ ) can be obtained by vehicle detection technologies, such as inductive loop (Gajda et al., 2001) based on the hardware installed underground. The distance of the EMV to the intersection  $d_{\text{EMV}}^t[L_{ji}]$  can be obtained by *vehicle-to-infrastructure* technologies such as VANET (Buchenscheit et al., 2009), which broadcasts the real-time position of a vehicle to an intersection. Prior work by Wang et al. (2013b) and Noori et al. (2016) have explored these technologies for traffic signal pre-emption.

The dynamic routing algorithm (Algorithm 1) can provide (ETA, Next) for each agent at every time step. However, due to the stochastic nature of traffic flows, updating the route too frequently might confuse the EMV driver, since the driver might be instructed

a new route, say, every 5 s. There are many ways to ensure reasonable frequency. One option is to inform the driver only once while the EMV travels on a single link. We implement it by updating the state of a RL agent  $(\text{ETA}_i^{t'}, \text{Next}_i^{t'})$  at the time step when the EMV travels through half of a link. For example, if the EMV travels through a link to agent  $i$  from time step 11 to 20 in constant speed, then dynamic routing information in  $s_i^{16}$  to  $s_i^{20}$  are the same, which is  $(\text{ETA}_i^{15}, \text{Next}_i^{15})$ , i.e.,  $t' = 15$ .

As for the reward design, one might wonder how an agent can know its type. As we assume an agent can observe the state of its neighbors, agent type can be inferred from the observation. This will become clearer in Section 4.3.

#### 4.3. Multi-agent advantage actor–critic

We adopt a multi-agent advantage actor–critic (MA2C) framework similar to Chu et al. (2019) to address the coupling of EMV navigating and traffic signal control simultaneously in a decentralized manner. The difference is that our local state includes dynamic routing information and our local reward encourages efficient passage of EMV. Here we briefly introduce the MA2C framework.

In a multi-agent network  $G(\mathcal{V}, \mathcal{E})$ , the neighborhood of agent  $i$  is denoted as  $\mathcal{N}_i = \{j | ji \in \mathcal{E} \text{ or } ij \in \mathcal{E}\}$ . The local region of agent  $i$  is  $\mathcal{V}_i = \mathcal{N}_i \cup i$ . We define the distance between two agents  $d(i, j)$  as the minimum number of edges that connect them. For example,  $d(i, i) = 0$  and  $d(i, j) = 1, \forall j \in \mathcal{N}_i$ . In MA2C, each agent learns a policy  $\pi_{\theta_i}$  (actor) and the corresponding value function  $V_{\phi_i}$  (critic), where  $\theta_i$  and  $\phi_i$  are learnable neural network parameters of agent  $i$ .

**Local observation.** In an ideal setting, agents can observe the states of every other agent and leverage this global information to make a decision. However, this is not practical in our problem due to communication latency and will cause scalability issues. We assume an agent can observe its own state and the states of its neighbors, i.e.,  $s_{\mathcal{V}_i}^t = \{s_j^t | j \in \mathcal{V}_i\}$ . The agents feed this observation to its policy network  $\pi_{\theta_i}$  and value network  $V_{\phi_i}$ .

**Fingerprint.** In multi-agent training, each agent treats other agents as part of the environment, but the policy of other agents are changing over time. Foerster et al. (2017) introduce *fingerprints* to inform agents about the changing policies of neighboring agents in multi-agent Q-learning. Chu et al. (2019) bring fingerprints into MA2C. Here we use the probability simplex of neighboring policies  $\pi_{\mathcal{N}_i}^{t-1} = \{\pi_j^{t-1} | j \in \mathcal{N}_i\}$  as fingerprints, and include it into the input of policy network and value network. Thus, our policy network can be written as  $\pi_{\theta_i}(a_i^t | s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1})$  and value network as  $V_{\phi_i}(s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1})$ , where  $s_{\mathcal{V}_i}^t$  is the local observation with spatial discount factor introduced below.

**Spatial discount factor and adjusted reward.** MA2C agents cooperatively optimize a global cumulative reward. We assume the global reward is decomposable as  $r_t = \sum_{i \in \mathcal{V}} r_i^t$ , where  $r_i^t$  is defined in Eq. (4). Instead of optimizing the same global reward for every agent, here we employ the spatial discount factor  $\alpha$ , introduced by Chu et al. (2019), to let each agent pay less attention to rewards of agents farther away. The adjusted reward for agent  $i$  is

$$\tilde{r}_i^t = \sum_{d=0}^{D_i} \left( \sum_{j \in \mathcal{V} | d(i,j)=d} (\alpha)^d r_j^t \right), \quad (5)$$

where  $D_i$  is the maximum distance of agents in the graph from agent  $i$ . When  $\alpha > 0$ , the adjusted reward include global information, it seems this is in contradiction to the local communication assumption. However, since reward is only used for offline training, global reward information is allowed. Once trained, the RL agents can control a traffic signal without relying on global information.

**Temporal discount factor and return.** The local return  $\tilde{R}_i^t$  is defined as the cumulative adjusted reward  $\tilde{R}_i^t := \sum_{\tau=t}^T \gamma^{\tau-t} \tilde{r}_i^\tau$ , where  $\gamma$  is the temporal discount factor and  $T$  is the length of an episode. We can estimate the local return using value function,

$$\tilde{R}_i^t = \tilde{r}_i^t + \gamma V_{\phi_i^-}(s_{\mathcal{V}_i}^{t+1}, \pi_{\mathcal{N}_i}^t | \pi_{\theta_{-i}}^-), \quad (6)$$

where  $\phi_i^-$  means parameters  $\phi_i$  are frozen and  $\theta_{-i}$  means the parameters of policy networks of all other agents are frozen.

**Network architecture and training.** As traffic flow data are spatial temporal, we leverage a long–short term memory (LSTM) layer along with fully connected (FC) layers for policy network (actor) and value network (critic). Fig. 6 provides an overview for the MA2C frameworks for EMVLight.

**Value loss function** With a batch of data  $B = \{(s_i^t, \pi_i^t, a_i^t, s_i^{t+1}, r_i^t)_{i \in \mathcal{V}}^T\}$ , each agent's value network is trained by minimizing the difference between bootstrapped estimated value and neural network approximated value

$$\mathcal{L}_v(\phi_i) = \frac{1}{2|B|} \sum_B \left( \tilde{R}_i^t - V_{\phi_i}(s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1}) \right)^2. \quad (7)$$

**Policy loss function** Each agent's policy network is trained by minimizing its policy loss

$$\mathcal{L}_p(\theta_i) = -\frac{1}{|B|} \sum_B \left( \ln \pi_{\theta_i}(a_i^t | s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1}) \tilde{A}_i^t \right. \quad (8)$$

$$\left. - \lambda \sum_{a_i \in \mathcal{A}_i} \pi_{\theta_i} \ln \pi_{\theta_i}(a_i | s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1}) \right), \quad (9)$$

where  $\tilde{A}_i^t = \tilde{R}_i^t - V_{\phi_i^-}(s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1})$  is the estimated advantage which measures how much better the action  $a_i^t$  is as compared to the average performance of the policy  $\pi_{\theta_i}$  in the state  $s_i^t$ . The second term is a regularization term that encourage initial exploration, where  $\mathcal{A}_i$  is the action set of agent  $i$ . For an intersection as shown in Fig. 1,  $\mathcal{A}_i$  contains 8 traffic signal phases.

**Training algorithm** Algorithm S1 shows the multi-agent A2C training process.

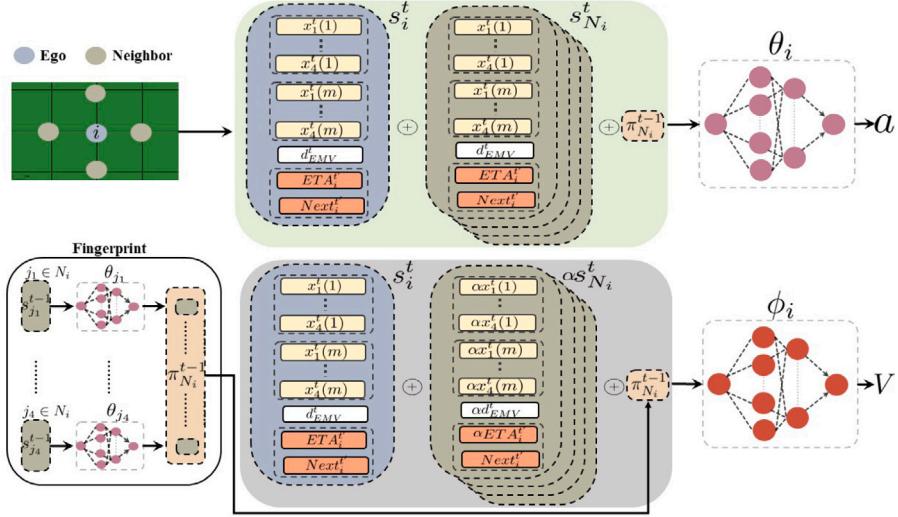


Fig. 6. Overview of MA2C framework for EMVLight's navigation and traffic signal control.

**Algorithm S1:** Multi-agent A2C Training

---

**Input :**

- $T$  maximum time step of an episode
- $N_{bs}$  batch size
- $\eta_\theta$  learning rate for policy networks
- $\eta_\phi$  learning rate for value networks
- $\alpha$  spatial discount factor
- $\gamma$  (temporal) discount factor
- $\lambda$  regularizer coefficient

**Output:**

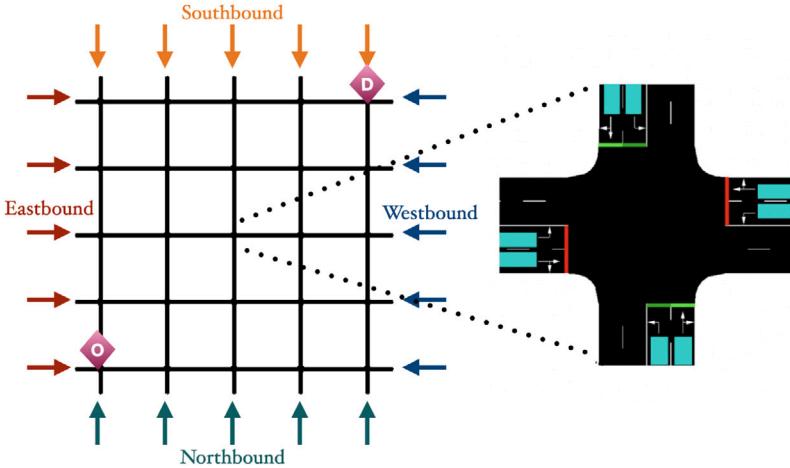
- $\{\phi_i\}_{i \in \mathcal{V}}$  learned parameters in value networks
- $\{\theta_i\}_{i \in \mathcal{V}}$  learned parameters in policy networks

```

1 initialize  $\{\phi_i\}_{i \in \mathcal{V}}, \{\theta_i\}_{i \in \mathcal{V}}, k \leftarrow 0, B \leftarrow \emptyset$ ; initialize SUMO,  $t \leftarrow 0$ , get  $\{s_i^0\}_{i \in \mathcal{V}}$ 
2 repeat
    /* generate trajectories */ */
    foreach  $i \in \mathcal{V}$  do (in parallel)
        | sample  $a_i^t$  from  $\pi_i^t$ 
        | receive  $\tilde{r}_i^t$  and  $s_i^{t+1}$ 
    6  $B \leftarrow B \cup \{(s_i^t, \pi_i^t, a_i^t, s_i^{t+1}, r_i^t)_{i \in \mathcal{V}}\}$ 
    7  $t \leftarrow t + 1, k \leftarrow k + 1$ 
    8 if  $t == T$  then
        | initialize SUMO,  $t \leftarrow 0$ , get  $\{s_i^0\}_{i \in \mathcal{V}}$  */
        /* update actors and critics
    10 if  $k == N_{bs}$  then
        | foreach  $i \in \mathcal{V}$  do (in parallel)
            | | calculate  $\tilde{r}_i^t$  (Eqn. (4)),  $\tilde{R}_i^t$  (Eqn. (5))
            | |  $\phi_i \leftarrow \phi_i - \eta_\phi \nabla \mathcal{L}_v(\phi_i)$ 
            | |  $\theta_i \leftarrow \theta_i - \eta_\theta \nabla \mathcal{L}_p(\theta_i)$ 
    15  $k \leftarrow 0, B \leftarrow \emptyset$ 
16 until Convergence

```

---



**Fig. 7.** *Left:* the synthetic grid<sub>5×5</sub>. Origin and destination for EMV are labeled. *Right:* an intersection illustration in SUMO, the teal area are inductive loop detected area.

**Table 2**

Configuration for Synthetic Grid<sub>5×5</sub>. Peak flow is assigned from 400s to 800s and non-peak flow is assigned out of this period. For Config. 1 and 2, the vehicles enter the grid from North and South, and exit towards East and West.

Config	Traffic flow (veh/lane/hr)		Origin	Destination
	Non-peak	Peak		
1	200	240	N,S	E,W
2	160	320		
3	200	240	Randomly	
4	160	320	generated	

## 5. Experimentation

In this section, we demonstrate our RL framework using Simulation of Urban MObility (SUMO) (Lopez et al., 2018). SUMO is an open-source traffic simulator capable of simulating both microscopic and macroscopic traffic dynamics, suitable for capturing the EMV's impact on the regional traffic as well as monitoring the overall traffic flow. An RL-simulator training pipeline is established between the proposed RL framework and SUMO, i.e., the agents collect observations from SUMO and preferred signal phases are fed back into SUMO. Notice that in order to reflect the proposed emergency lane in Fig. 3, we enable the built-in *Sublane* model (German Aerospace Center, 2022b) and *Blue light device* (German Aerospace Center, 2022a) to establish the emergency lane for EMV passage. Under this scenarios, non-EMVs pull over to the sides when the EMV is driving between, and they resume normal driving when the EMV leaves the segment, see Appendix C for more details.

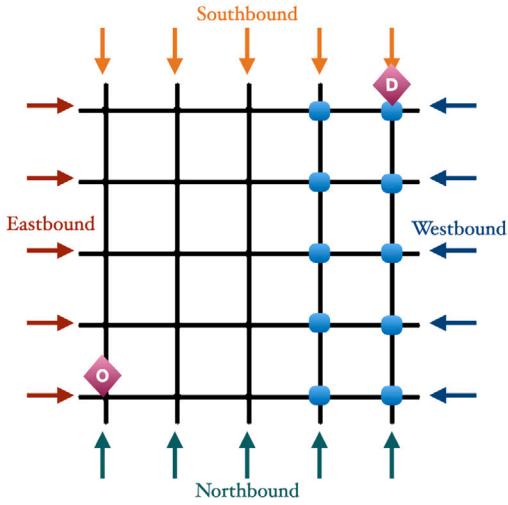
### 5.1. Datasets and maps descriptions

We conduct the following experiments based on both synthetic and real-world map.

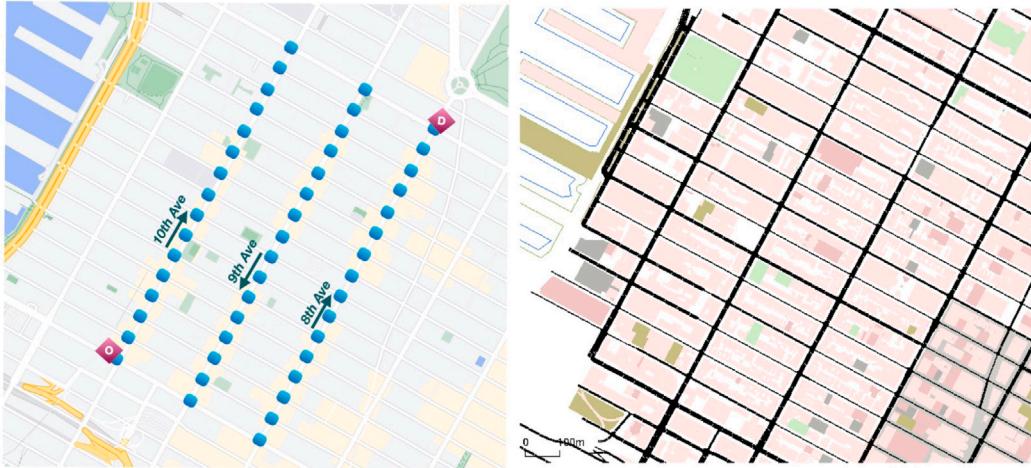
*Synthetic Grid<sub>5×5</sub>.* We synthesize a 5 × 5 traffic grid, where intersections are connected with bi-directional links. Each link contains two lanes. We assume all the links have zero emergency capacity. We design 4 configurations of time-varying traffic flows, listed in Table 2. The origin (O) and destination (D) of the EMV are labeled in Fig. 7. The traffic for this map has a time span of 1200s. We dispatch the EMV at  $t = 600$  s to ensure the roads are compacted when it starts travel.

*Emergency-capacitated (EC) synthetic Grid<sub>5×5</sub>.* This map adopts the same network layout as the Synthetic Grid<sub>5×5</sub> but with emergency-capacitated segments. As shown in Fig. 8, segments approaching intersections highlighted by blue are emergency-capacitated with  $C^{EC} = 0.2k$ , with  $k$  represents the normal vehicle capacity of this segment. All other segments are not emergency-capacitated.

*Manhattan<sub>16×3</sub>.* This is a 16 × 3 traffic network extracted from Manhattan Hell's Kitchen area (Fig. 9) and customized for demonstrating EMV passage. In this traffic network, intersections are connected by 16 one-directional streets and 3 one-directional avenues. We assume each avenue contains four lanes and each street contains two lanes so that the right-of-way of EMVs and pre-emption can be demonstrated. We assume the emergency capacity for avenues and streets are  $C_{\text{avenue}}^{EC} = 0.2k_{\text{avenue}}$  and



**Fig. 8.** Emergency-capacitated Synthetic Grid<sub>5x5</sub>. Segments towards intersections highlighted by blue have emergency capacities. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 9.** Manhattan<sub>16x3</sub>: a 16-by-3 traffic network in Hell's Kitchen area. Origin and destination for the EMV dispatching are labeled. *Left:* on Google Map; *Right:* in SUMO simulator.

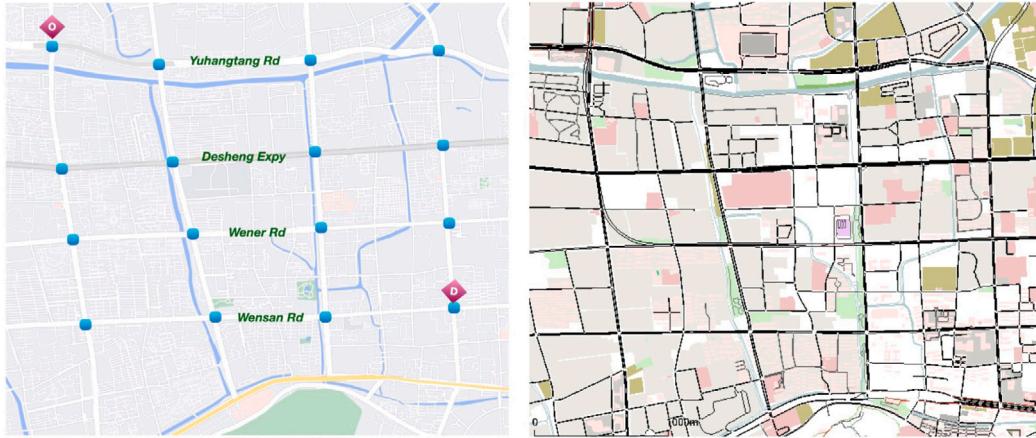
$C_{\text{street}}^{\text{EC}} = 0.15k_{\text{street}}$ , respectively. The traffic flow for this map is generated from open-source NYC taxi data. Both the map and traffic flow data are publicly available.<sup>2</sup> The origin and destination of EMV are set to be far away as shown in Fig. 9.

*Hangzhou*<sub>4x4</sub>. An irregular 4 × 4 road network represents major avenues in Gudang sub-district in Hangzhou, China. All the road segments are bi-directional with two lanes in each direction. Both the map and traffic flow data are publicly available. We set the origin and destination for EMV routing as shown in Fig. 10. The emergency capacity for each segment is set as  $C^{\text{EC}} = 0.2k$ . Notice that this map only includes primary arterials and intersections because secondary arterials and interchanges usually lack ITS infrastructure to assist EMVs' passage. They are also hard to monitor and control, considering Hangzhou Gudang district is a densely populated area. It is common to see double parking or merchants occupying road space on these secondary roads. The associated traffic flow dataset (Chen, 2019) also excludes vehicle trajectories information on these secondary arterials for simplification.

## 5.2. Benchmark methods

Due to the lack of existing RL methods for efficient EMV passage, we select traditional methods and RL methods for each subproblem and combine them to set up benchmarks.

<sup>2</sup> <https://traffic-signal-control.github.io/>



**Fig. 10.** Hangzhou<sub>4x4</sub>: a 4-by-4 irregular and asymmetric network. Origin and destination for the EMV dispatching are labeled. *Left:* on Google Map; *Right:* in SUMO simulator.

For traffic signal pre-emption, the most intuitive and widely-used approach is the idea of extending green light period for EMV passage at each intersection which results in a *Green Wave* (Corman et al., 2009). **Walabi (W)** (Bieker-Walz and Behrisch, 2019) is an effective rule-based method that implemented Green Wave for EMVs in SUMO environment. We integrate Walabi with combinations of routing and traffic signal control strategies introduced below as benchmarks. We first present two routing benchmarks.

- **Static routing:** static routing is performed only when EMV starts to travel and the route remains fixed. We adopt A\* search as the implementation of static routing since it is a powerful extension to the Dijkstra's shortest path algorithm and is used in many real-time applications.<sup>3</sup>
- **Dynamic routing:** dynamic routing updates the route by taking into account real-time information of traffic conditions. The route is then updated by repeatedly running static routing algorithms. To set up the dynamic routing benchmark, we run A\* every 50 s as EMV travels. The update interval is set to 50 s since running the full A\* to update the route is not as efficient as our proposed dynamic Dijkstra's algorithm.

#### Traffic signal control benchmarks:

- **Fixed Time (FT):** Cyclical fixed time traffic phases with random offset (Roess et al., 2004) is a policy that split all phases with an predefined green ratio. The coordination between traffic signals are predefined so it is not updated based on real-time traffic. Because of its simplicity, it is the default strategy in real traffic signal control for steady traffic flow.
- **Max Pressure (MP):** Varaiya (2013) studies max pressure control and use it as the main criterion for selecting traffic signal phases. It defines pressure for each signal phases and aggressively selects the traffic signal phase with maximum pressure to smooth congestion. Hence the name Max Pressure. It is the state-of-the-art network-level signal control strategy that is not based on learning.
- **Coordinated Deep Reinforcement Learners (CDRL):** CDRL (Van der Pol and Oliehoek, 2016) is a Q-learning based coordinator which directly learns joint local value functions for adjacent intersections. It extends Q-learning from single-agent scenarios to multi-agent scenarios. It also employs transfer planning and max-plus coordination strategies for joint intersection coordination.
- **PressLight (PL):** PL (Wei et al., 2019a) is also a Q-learning based method for traffic signal coordination. It aims at optimizing the pressure at each intersection. However, it defines pressure for each intersection, which is slightly different from the definition in Max Pressure. Our definition of pressure Eq. (4). is also different from that in PL.
- **CoLight (CL):** CoLight (Wei et al., 2019b) uses a graph-attentional-network-based reinforcement learning method for large scale traffic signal control. It adjusts queue length with information from neighbor intersections.

#### 5.3. Metrics

We evaluate performance of all strategies under two metrics: *EMV travel time*, which reflects the routing and pre-emption ability, and *average travel time*, which indicates the ability of traffic signal control for efficient vehicle passage. Vehicles which have completed their trips during the simulation interval are counted when calculating the average travel time.

<sup>3</sup> Our implementation of A\* search employs a Manhattan distance as the heuristic function.

**Table 3**

EMV travel time in the four configurations of Synthetic Grid<sub>5x5</sub>. Lower value indicates better performance and the lowest values are highlighted in bold.

Method	EMV travel time $T_{\text{EMV}}$ [s]			
	Config 1	Config 2	Config 3	Config 4
FT w/o EMV	N/A	N/A	N/A	N/A
W + static + FT	<b>258.18 ± 5.32</b>	273.32 ± 9.74	256.40 ± 6.20	240.84 ± 4.43
W + static + MP	260.22 ± 10.87	272.40 ± 10.92	265.74 ± 11.98	242.32 ± 9.48
W + static + CDRL	269.42 ± 7.32	282.20 ± 5.28	276.14 ± 2.58	280.32 ± 4.82
W + static + PL	270.68 ± 9.13	279.14 ± 9.22	281.42 ± 5.62	266.10 ± 8.32
W + static + CL	255.72 ± 4.23	272.06 ± 8.13	270.22 ± 2.81	277.12 ± 6.10
W + dynamic + FT	229.38 ± 8.28	212.87 ± 3.17	218.46 ± 4.28	220.69 ± 7s.96
W + dynamic + MP	220.48 ± 9.26	208.08 ± 12.90	212.46 ± 9.82	220.98 ± 10.62
W + dynamic + CDRL	239.84 ± 5.24	219.15 ± 8.26	211.86 ± 7.13	232.46 ± 10.16
W + dynamic + PL	243.32 ± 13.86	244.82 ± 10.52	250.12 ± 8.13	255.02 ± 12.76
W + dynamic + CL	220.12 ± 4.19	209.12 ± 4.76	224.00 ± 5.31	226.32 ± 4.13
EMVLIGHT	<b>195.46 ± 7.48</b>	<b>190.66 ± 8.28</b>	<b>183.12 ± 6.43</b>	<b>189.44 ± 8.32</b>

**Table 4**

Average travel time for all vehicles which have completed trips in the four configurations of Synthetic Grid<sub>5x5</sub>.

Method	Average travel time $T_{\text{avg}}$ [s]			
	Config 1	Config 2	Config 3	Config 4
FT w/o EMV	353.43 ± 4.65	371.13 ± 4.58	314.25 ± 2.90	334.10 ± 3.73
W + static + FT	380.42 ± 13.35	395.17 ± 15.37	350.16 ± 13.66	363.90 ± 15.39
W + static + MP	355.10 ± 15.36	362.09 ± 16.15	318.76 ± 14.90	330.69 ± 15.52
W + static + CDRL	559.19 ± 3.60	540.81 ± 12.04	568.13 ± 13.25	568.13 ± 6.67
W + static + PL	369.52 ± 8.72	372.32 ± 16.05	339.18 ± 7.17	339.12 ± 5.33
W + static + CL	365.64 ± 14.08	380.13 ± 8.20	328.42 ± 17.52	333.74 ± 5.76
W + dynamic + FT	380.76 ± 10.70	404.81 ± 18.76	345.09 ± 11.60	358.90 ± 15.27
W + dynamic + MP	360.38 ± 10.31	365.10 ± 8.33	327.98 ± 18.90	351.62 ± 3.79
W + dynamic + CDRL	565.38 ± 16.10	544.29 ± 19.23	598.73 ± 11.01	572.22 ± 13.94
W + dynamic + PL	373.17 ± 17.98	387.25 ± 13.98	349.12 ± 16.25	330.21 ± 17.23
W + dynamic + CL	359.14 ± 9.52	370.45 ± 4.02	320.64 ± 4.10	335.27 ± 7.62
EMVLIGHT	<b>335.09 ± 4.13</b>	<b>333.28 ± 8.81</b>	<b>307.90 ± 3.89</b>	<b>321.02 ± 5.87</b>

## 6. Results and discussion

In this section, we demonstrate the performance of EMVLIGHT and compare it against that of all benchmark methods on four experimentation maps. The results show a clear advantage of EMVLIGHT under the two metrics. In addition, we illustrate the difference of underlying route selection by EMVLIGHT and benchmark methods. We further conduct ablation studies to investigate the contribution of different components to EMVLIGHT's performance.

### 6.1. Performance comparison

To evaluate the performance of the proposed EMVLIGHT and all benchmark methods, we conduct SUMO simulation with five independent runs for each setting. Randomly generated seeds are used in learning-based methods. Means as well as standard deviations of the simulation results are reported for a full numerical assessment. The differences in simulation results for the same setting under independent SUMO runs are coming from configuration noise during generation, such as vehicles' lengths/accelerations/lane-changing eagerness, and, for RL-based methods, random seeds for initialization.

We provide implementation details of EMVLIGHT on different experimentation settings in [Appendix A](#). Hyper-parameters choices for EMVLIGHT and RL-based benchmarks are provided in [Appendix B](#).

#### 6.1.1. Synthetic Grid<sub>5x5</sub> results

[Tables 3](#) and [4](#) present the experimental results on average EMV travel time and average travel time on Synthetic Grid<sub>5x5</sub>. In terms of EMV travel time  $T_{\text{EMV}}$ , the dynamic routing benchmark performs better than static routing benchmarks. This is expected since dynamic routing considers the time-dependent nature of traffic conditions and update optimal route accordingly. The best learning and non-learning benchmark methods are dynamics routing with CoLight and Max Pressure, respectively. EMVLIGHT further reduces EMV travel time by 16% on average as compared to dynamic routing benchmarks. This advantage in performance can be attributed to the design of secondary pre-emption agents. This type of agents learns to "reserve a link" by choosing signal phases that help clear the vehicles in the link to encourage high speed EMV passage (Eq. (4)).

**Table 5**  
EMV travel time in the four configurations of Synthetic Grid<sub>5x5</sub> with an emergency capacity co-efficient of 0.25.

Method	EMV travel time $T_{\text{EMV}}$ [s]			
	Config 1	Config 2	Config 3	Config 4
FT w/o EMV	N/A	N/A	N/A	N/A
W + static + FT	254.04 ± 7.42	260.18 ± 12.03	252.12 ± 11.03	232.47 ± 12.23
W + static + MP	233.76 ± 8.05	258.60 ± 9.06	252.74 ± 13.05	233.20 ± 8.96
W + static + CDRL	240.10 ± 8.65	266.28 ± 8.54	258.10 ± 9.27	270.43 ± 6.18
W + static + PressLight	265.28 ± 7.28	269.10 ± 6.65	270.18 ± 8.83	259.20 ± 7.13
W + static + CoLight	250.82 ± 6.73	267.08 ± 10.21	266.12 ± 4.13	270.18 ± 8.12
W + dynamic + FT	210.28 ± 8.90	206.18 ± 7.19	210.28 ± 8.81	207.64 ± 10.02
W + dynamic + MP	202.28 ± 8.54	203.20 ± 9.07	206.64 ± 7.98	210.86 ± 8.59
W + dynamic + CDRL	218.36 ± 8.12	209.28 ± 7.19	208.180 ± 10.54	230.22 ± 9.22
W + dynamic + PressLight	270.08 ± 10.20	238.10 ± 9.22	242.10 ± 6.98	248.24 ± 10.24
W + dynamic + CoLight	216.04 ± 4.91	206.12 ± 6.27	219.26 ± 6.87	223.78 ± 5.10
EMVLight	<b>150.28 ± 7.48</b>	<b>158.20 ± 6.28</b>	<b>154.28 ± 4.19</b>	<b>159.28 ± 6.03</b>

**Table 6**  
Average travel time for all vehicles which have completed trips in the four configurations of Synthetic Grid<sub>5x5</sub> with an emergency capacity coefficient of 0.25.

Method	Average travel time $T_{\text{avg}}$ [s]			
	Config 1	Config 2	Config 3	Config 4
FT w/o EMV	353.43 ± 4.65	371.13 ± 4.58	314.25 ± 2.90	334.10 ± 3.73
W + static + FT	395.28 ± 5.17	410.94 ± 9.16	365.82 ± 6.14	379.64 ± 6.21
W + static + MP	370.52 ± 6.18	375.44 ± 6.48	331.62 ± 5.92	345.13 ± 8.63
W + static + CDRL	575.28 ± 7.76	555.62 ± 10.04	574.91 ± 19.86	585.20 ± 7.53
W + static + PL	385.28 ± 12.09	380.83 ± 10.07	360.09 ± 11.62	369.72 ± 18.02
W + static + CL	382.17 ± 6.02	380.13 ± 8.21	344.19 ± 16.02	352.07 ± 4.10
W + dynamic + FT	389.12 ± 18.21	411.98 ± 15.31	353.72 ± 9.09	367.74 ± 16.82
W + dynamic + MP	370.28 ± 12.51	362.82 ± 9.05	335.10 ± 9.16	360.02 ± 17.18
W + dynamic + CDRL	575.05 ± 9.67	550.92 ± 14.06	609.26 ± 11.12	578.10 ± 12.09
W + dynamic + PL	380.29 ± 6.10	395.28 ± 5.62	359.16 ± 14.07	337.26 ± 4.96
W + dynamic + CL	366.14 ± 8.21	380.74 ± 15.84	330.44 ± 17.29	343.58 ± 14.27
EMVLight	<b>334.96 ± 5.52</b>	<b>336.18 ± 17.09</b>	<b>309.10 ± 15.56</b>	<b>323.76 ± 17.24</b>

As for average travel time  $T_{\text{avg}}$ , we first notice that the traditional pre-emption technique (W + static + FT) indeed increases the average travel time by around 10% as compared to a traditional Fix Time strategy without EMV (denoted as “FT w/o EMV” in [Table 4](#)), thus decreasing the efficiency of vehicle passage. Different traffic signal control strategies have a direct impact on overall efficiency. Fixed Time is designed to handle steady traffic flow. Max Pressure, as a SOTA traditional method, outperforms Fix Time and, surprisingly, nearly outperforms all RL benchmarks in terms of overall efficiency. This shows that pressure is an effective indicator for reducing congestion and this is why we incorporate pressure in our reward design. Coordinate Learner performs the worst probably because its reward is not based on pressure. PressLight does not beat Max Pressure because it has a reward design that focuses on smoothing vehicle densities along a major direction, e.g. an arterial. Grid networks with the presence of EMV make PressLight less effective. CoLight achieves similar performance as Max Pressure and is the best learning benchmark method. Our EMVLight improves its pressure-based reward design to encourage smoothing vehicle densities of all directions for each intersection. This enable us to achieve an advantage of 7.5% over our best benchmarks (Max Pressure).

#### 6.1.2. Emergency-capacitated synthetic Grid<sub>5x5</sub> results

[Table 5](#) shows  $T_{\text{EMV}}$  of all the methods implemented on the emergency-capacitated synthetic Grid<sub>5x5</sub> map. By comparing [Tables 3](#) and [5](#), we conclude that the emergency-capacitated map exhibits overall shorter  $T_{\text{EMV}}$  in all configurations. For benchmark methods, the nonzero emergency capacity shorten  $T_{\text{EMV}}$  by an average of approximately 12 s. In particular, dynamic routing-based methods benefit more from the additional emergency capacity, resulting in an average reduction of in 16.28 s  $T_{\text{EMV}}$ . This is due to the adaptive nature of dynamic navigation. By comparing EMVLight and benchmark methods in [Table 5](#), we observe that EMVLight reduce  $T_{\text{EMV}}$  by up to 50 s (25%) as compared to the best benchmark method in all configurations. The substantial difference in  $T_{\text{EMV}}$  reduction between benchmark methods and EMVLight is due to high success rate of emergency lane forming under coordination, which is investigated further in [Section 6.2](#).

As the EMV travels faster and more emergency lanes are formed, the average travel time of non-emergency vehicles increases. [Table 6](#) shows  $T_{\text{avg}}$  with emergency capacity added. By comparing [Tables 4](#) and [6](#), we observe that with added emergency capacity, the average increase in  $T_{\text{avg}}$  for non-learning-based and learning-based benchmarks are 12.04 s and 7.78 s, respectively. Learning-based methods lead to a smaller average increase since agents gradually learn to direct non-EMVs, which are interrupted by EMV passages, to resume their trips as soon as possible. As a result, potential congested queues would not be formed on these segments, effectively reducing the overall  $T_{\text{avg}}$ .

**Table 7**

$T_{\text{EMV}}$  and  $T_{\text{avg}}$  for Manhattan<sub>16x3</sub>. The average travel time without the presence of EMVs (1649.64) is retrieved from data.

Method	Manhattan <sub>16x3</sub>	
	$T_{\text{EMV}}$	$T_{\text{avg}}$
FT w/o EMV	N/A	1649.64
W + static + FT	817.37 ± 17.40	1816.43 ± 68.96
W + static + MP	686.72 ± 19.23	917.52 ± 52.16
W + static + CDRL	702.62 ± 24.29	1247.67 ± 83.47
W + static + PL	626.88 ± 24.82	992.06 ± 47.67
W + static + CL	545.26 ± 30.21	855.28 ± 41.29
W + dynamic + FT	820.54 ± 28.86	1808.25 ± 68.04
W + dynamic + MP	632.68 ± 13.29	921.18 ± 49.29
W + dynamic + CDRL	680.62 ± 20.17	1262.39 ± 60.09
W + dynamic + PL	521.42 ± 27.62	977.62 ± 53.45
W + dynamic + CL	501.26 ± 28.71	862.94 ± 45.19
EMVLight	292.82 ± 16.23	782.13 ± 39.31

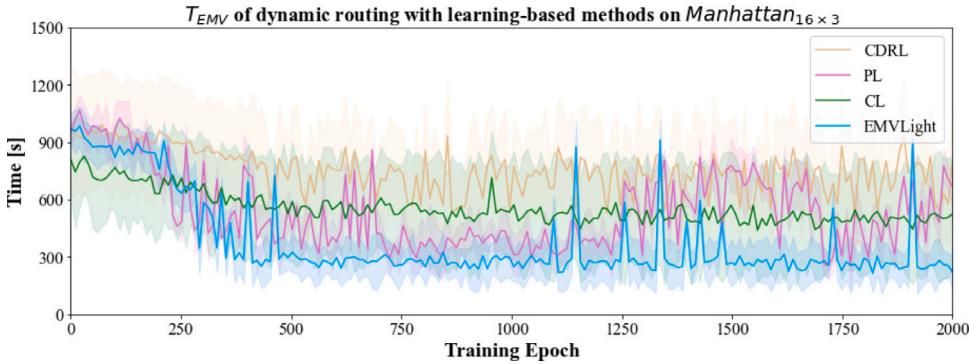


Fig. 11.  $T_{\text{EMV}}$  convergence by learning-based dynamic routing strategies on Manhattan<sub>16x3</sub>.

EMVLight, surprisingly, manages to achieve nearly no increase in  $T_{\text{avg}}$  with added emergency capacity, even though more emergency lanes are formed as indicated by smaller  $T_{\text{EMV}}$ . This result shows that EMVLight is able to learn a strong traffic signal coordination strategy while navigating EMVs simultaneously. The proposed multi-class agent design demonstrates EMVLight's capability of addressing the coupled problems of EMV routing and traffic signal control simultaneously. EMVLight manages to prepare segments for incoming EMVs by reducing the number of vehicles on those segments and restores the impacted traffic in a timely manner after the EMV passage.

#### 6.1.3. Manhattan<sub>16x3</sub> results

Table 7 presents EMV travel time and average travel time of all the methods on the Manhattan<sub>16x3</sub> map. In terms of  $T_{\text{EMV}}$ , dynamic routing benchmarks in general result in faster EMV passage, as expected. Compared with benchmark methods, EMVLight produces a considerably low average  $T_{\text{EMV}}$  of 292.82 s, which is 38% faster than the best benchmark (W+static+CL). As for  $T_{\text{avg}}$ , We have similar observation as in the synthetic maps that Max Pressure achieves a similar level of performance on reducing congestion as PressLight and beats CDRL by a solid margin of 25%. CoLight stands out among benchmarks regarding both metrics. Particularly, CoLight shortens  $T_{\text{avg}}$  by approximately one minute than Max Pressure strategies on this map.

Figs. 11 and 12 shows the learning curves of  $T_{\text{EMV}}$  and  $T_{\text{avg}}$ , respectively, in the four RL methods. From both figures, we observe that EMVLight has the fastest convergence – in 500 epochs for  $T_{\text{EMV}}$  and in 1000 epoches for  $T_{\text{avg}}$  – among all four methods. In Fig. 11, both PressLight and CDRL struggle with converging to a stable  $T_{\text{EMV}}$ . In Fig. 12, CDRL converges very slowly as compared to the other three methods. Both figures shows that CDRL behaves the worst since its DQN design hardly scales with an increasing number of intersections. These learning curves demonstrate the fast and stable learning of EMVLight.

#### 6.1.4. Hangzhou<sub>4x4</sub> results

Table 8 presents  $T_{\text{EMV}}$  and  $T_{\text{avg}}$  of EMVLight and benchmark methods on Hangzhou<sub>4x4</sub>. EMVLight achieves the lowest  $T_{\text{EMV}}$  of 194.52 s, beating the best benchmark (W+dynamic+CL) by 115 s (37%). As for  $T_{\text{avg}}$ , EMVLight also has excellent performance, exhibiting a 10% advantage over W+dynamic+CL, and a 20% advantage over W+dynamic+MP. Once again, Max Pressure outperforms PressLight in terms of  $T_{\text{avg}}$ . This is consistent with our observations with other maps above, suggesting Max Pressure result in great traffic coordination strategies to reduce overall congestion.

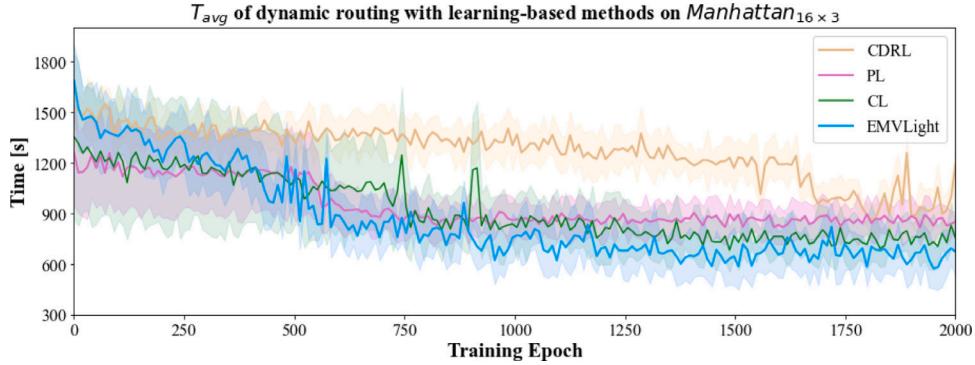


Fig. 12.  $T_{\text{avg}}$  convergence by learning-based dynamic routing strategies on Manhattan<sub>16x3</sub>.

**Table 8**

$T_{\text{EMV}}$  and  $T_{\text{avg}}$  for Gudang<sub>4x4</sub>. The average travel time without the presence of EMVs (764.08) is retrieved from data.

Method	Hangzhou <sub>4x4</sub>	
	$T_{\text{EMV}}$	$T_{\text{avg}}$
FT w/o EMV	N/A	764.08
W + static + FT	466.19 ± 10.25	779.13 ± 12.90
W + static + MP	377.20 ± 14.42	404.37 ± 8.12
W + static + CDRL	409.56 ± 12.06	749.10 ± 10.02
W + static + PL	380.82 ± 6.72	425.46 ± 9.74
W + static + CL	368.20 ± 14.66	366.14 ± 8.25
W + dynamic + FT	415.63 ± 9.03	783.89 ± 10.03
W + dynamic + MP	328.42 ± 12.28	410.25 ± 6.23
W + dynamic + CDRL	401.08 ± 15.25	755.28 ± 12.82
W + dynamic + PL	321.52 ± 14.58	431.27 ± 8.24
W + dynamic + CL	319.84 ± 11.09	370.20 ± 7.13
EMVLight	194.52 ± 9.65	331.42 ± 6.18

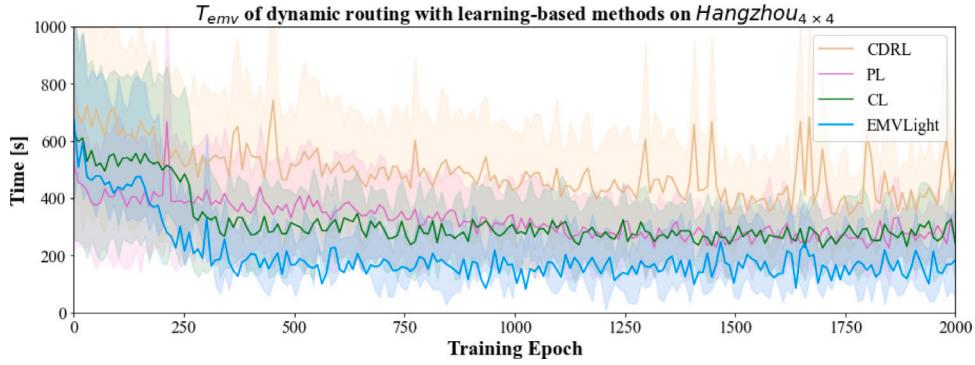


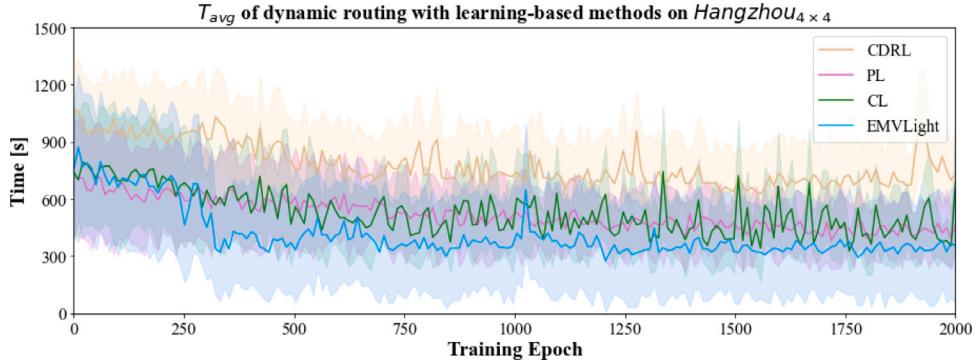
Fig. 13.  $T_{\text{EMV}}$  convergence by learning-based dynamic routing strategies on Hangzhou<sub>4x4</sub>.

Based on results of all introduced experiments, Max Pressure has evinced the consistency to restrict congestion, particularly when restoring stagnant traffic after EMV passages. Consequently, Max Pressure accomplishes the comparable congestion reduction acquirement with, if not better than, RL-based strategies.

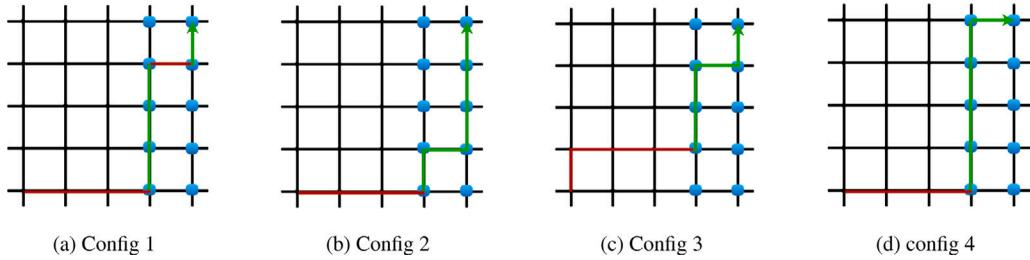
CDRL fails to learn an effective coordination strategy to manage congestion, where its  $T_{\text{avg}}$  trivially differs from Fix Time strategy's  $T_{\text{avg}}$ . CoLight surpasses other methods by shortening  $T_{\text{avg}}$  by at least 40 s but it does not illustrate significant improvement in terms of  $T_{\text{EMV}}$  from alternative benchmarks. Learning curves in Figs. 13 and 14 show that EMVLight converges in 500 epochs.

## 6.2. Routing choices

In this section, we investigate the reason behind EMVLight's best performance in  $T_{\text{EMV}}$  from a routing perspective. In particular, we show how EMVLight is able to leverage emergency capacity to achieve maximum speed passage. To understand different route



**Fig. 14.**  $T_{\text{avg}}$  convergence by learning-based dynamic routing strategies on  $\text{Hangzhou}_{4 \times 4}$ .



**Fig. 15.** EMV's routing choice on Emergency-capacitated Synthetic Grid<sub>5x5</sub> based on EMVLIGHT. Emergency lane established for EMV passage on segments highlighted in green, and not established on segments in red. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



**Fig. 16.** EMV's routing choice on config 1 of Emergency-capacitated Synthetic Grid<sub>5x5</sub> under MP-based benchmarks.

choices between benchmark methods and EMVLIGHT, we analyze EMV routes on emergency-capacitated Synthetic Grid<sub>5x5</sub> as well as Hangzhou<sub>4x4</sub> to gain insight into the advantage of EMVLIGHT.

*Emergency-capacitated synthetic Grid<sub>5x5</sub> routes.* Routes selected by EMVLIGHT are demonstrated in Fig. 15 for all four configurations. As this is a regular grid, first we notice that the length of all EMV routes are the Manhattan distance between the origin and destination. This is the shortest length possible to achieve successful EMV dispatch. The four routes confirm that EMVLIGHT directs EMVs to enter the region in the east as soon as possible to leverage the extra emergency capacity for full speed passage. This results in 4 full speed links in Config 1 and 3 as well as 5 full speed links in Config 2 and 4.

We also present the route choices of W+static+MP and W+dynamic+MP in Config 1, of this map, as shown in Fig. 16. By comparing these routes with EMVLIGHT (Fig. 15(a)), we can clearly see the benchmark method cannot actively leverage the emergency capacity, generating routes with only 1 full speed segment (Fig. 16(a)) and 2 full speed segment (Fig. 16(b)).

*Hangzhou<sub>4x4</sub> routes.* Since Hangzhou<sub>4x4</sub> is an irregular grid where different links have different lengths, the routes optimized by different algorithms have different total lengths. This provides another perspective on evaluating routing performance of different models. Fig. 17 shows EMV routes given by W+static+MP, W+dynamic+MP and EMVLIGHT in this grid. By comparing the total distances of the three routes, we find that the route chosen by EMVLIGHT is the longest among the three models. However, EMVLIGHT

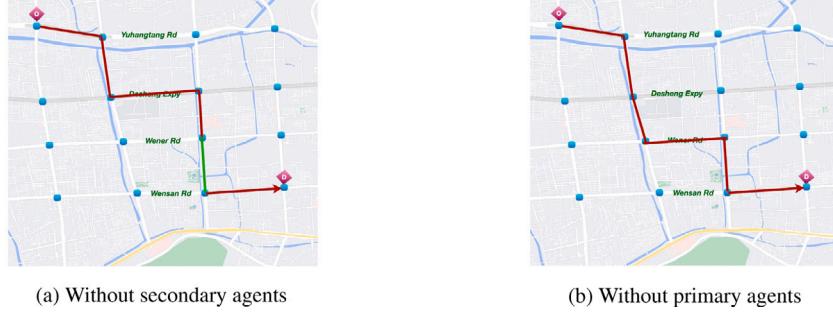


**Fig. 17.** The corresponding route selected by (a) W+static+MP, (b) W+dynamic+MP, (c) EMVLight on Hangzhou<sub>4x4</sub>. Distance and  $T_{EMV}$  for the selected path are indicated. The lowest values are highlighted in bold.

**Table 9**

Ablation studies on pressure-based reward design and agent types. Experiments are conducted on Hangzhou<sub>5x5</sub>. The lowest value are highlighted.

Ablations	Ablation 1	Ablation 2	Ablation 3	Ablation 4	EMVLight
$T_{EMV}$ [s]	$205.20 \pm 6.92$	$237.40 \pm 10.08$	$311.52 \pm 5.18$	$384.71 \pm 8.52$	<b>194.52</b> $\pm 9.65$
$T_{avg}$ [s]	$389.14 \pm 8.40$	$382.07 \pm 6.92$	$442.73 \pm 6.65$	$444.15 \pm 7.02$	<b>331.42</b> $\pm 6.18$



**Fig. 18.** EMV's route choice on Hangzhou<sub>4x4</sub> with replaced primary (a) and secondary (b) agents.

achieves the smallest EMV travel time on this route. This is because EMVLight is able to coordinate traffic signals to leverage the emergency capacity to let the EMV travel at its maximum speed on more than half of the route, indicated by the green segments.

As for static routing (W+static+MP), it chooses a fixed route given the traffic conditions upon dispatching, favoring the shortest-in-distance route. The dynamic routing method (W+dynamic+MP) recalculate the time-based shortest path every 30 s as EMV travels. Most of the time, however, Max Pressure fails to reduce the number of vehicles on the upcoming links to enable an emergency lane. **Fig. 17(b)** shows that the emergency lane is only formed in one of the six links. EMVLight is able to further reduce EMV travel time partly because the time-based shortest path is updated in real time. More importantly, EMVLight is able to reduce the number of vehicles in upcoming links so that emergency lanes can be formed. This can be attributed to the design of primary and secondary preemption agents in EMVLight, which will be further examined in Section 6.3.1.

### 6.3. Ablation studies

#### 6.3.1. Ablation studies on reward

We propose three types of agents and design their rewards (Eq. (4)) based on our improved pressure definition and heuristics. In order to see how our improved pressure definition and proposed agent types influence the results, we propose three ablation studies:

1. replacing our pressure definition by that defined in PressLight
2. replacing our pressure definition by  $P_i = \sum_{l \in I_i} w(l)$
3. replacing secondary pre-emption agents with normal agents
4. replacing primary pre-emption agents with normal agents

**Table 9** shows the results of these ablations on the Hangzhou<sub>4x4</sub> map. We observe that PressLight-style pressure yields a slightly larger  $T_{EMV}$  but significantly increases the  $T_{avg}$ . Without secondary pre-emption agents,  $T_{EMV}$  increases by 60% since almost no “link

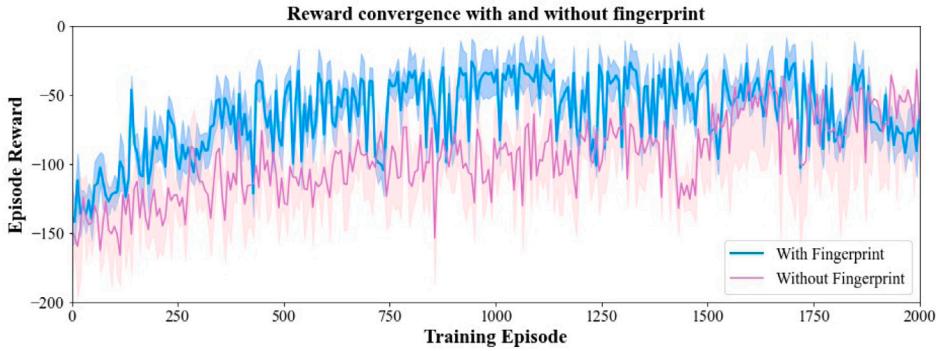


Fig. 19. Reward convergence with and without fingerprint. Experiments are conducted on Config 1 synthetic grid<sub>5x5</sub>.

**Table 10**

EMVLight's performance on Emergency-capacitated Synthetic Grid<sub>5x5</sub> with Configuration 1 given different values of  $\beta$ .

	0	0.25	0.5	0.75	1
$T_{\text{EMV}}$	$119.10 \pm 5.14$	$134.40 \pm 5.83$	$158.20 \pm 6.28$	$172.60 \pm 5.71$	$185.20 \pm 4.19$
$T_{\text{avg}}$	$340.23 \pm 4.75$	$338.17 \pm 7.25$	$334.92 \pm 5.52$	$331.81 \pm 9.28$	$327.10 \pm 8.18$

reservation" happened. In Ablation 2, if we do not take the average while keep the sum of absolute value of lane pressures as the intersection pressure, we notice a slightly smaller  $T_{\text{avg}}$  but a 15% increase in  $T_{\text{EMV}}$ . EMVLight's pressure design outperforms the ones in Ablation 1 and 2 and has been proven as the most suitable pressure design for this particular task. Moreover, without primary pre-emption agents,  $T_{\text{EMV}}$  increases considerably, which again proves the importance of pre-emption. We can further confirm the importance of agent designs by inspecting the selected routes in the last two ablation studies. Figs. 18(a) and 18(b) shows routes selected by EMVLights after replacing secondary agents and primary agents with normal agents, respectively. Even though the routes are similar as that in Fig. 17(c), much fewer emergency lanes are successfully formed. This failure of utilizing emergency capacity lead to the significant increase in EMV travel time as shown in Table 9. As we can see from the routes, EMVs barely take advantage of emergency yielding during their trips.

### 6.3.2. Ablation study on policy exchanging

In multi-agent RL, fingerprint has been shown to stabilize training and enable faster convergence. In order to see how fingerprint affects training in EMVLight, we remove the fingerprint design, i.e., policy and value networks are changed from  $\pi_{\theta_i}(a_i^t | s_i^t, \pi_{\mathcal{N}_i}^{t-1})$  and  $V_{\phi_i}(\tilde{s}_i^t, \pi_{\mathcal{N}_i}^{t-1})$  to  $\pi_{\theta_i}(a_i^t | s_i^t)$  and  $V_{\phi_i}(\tilde{s}_i^t)$ , respectively. Fig. 19 shows the influence of fingerprint on training. With fingerprint, the reward converges faster and suffers from less fluctuation, confirming the effectiveness of fingerprints, i.e. policy exchanging.

### 6.4. Sensitivity analysis on $\beta$

The value of  $\beta$ 's in Eq. (4) represents the weights assigned to the two tasks. When  $\beta = 1$ , the secondary pre-emption agents will act like normal agents and only focus on congestion alleviation. When  $\beta = 0$ , the secondary pre-emption agents concentrate on emptying non-EMVs and reserve the whole segment for the approaching EMV. In order to illustrate the tradeoff, we conduct experimental runs on the emergency-capacitated synthetic Grid<sub>5x5</sub> with configuration 1 as the sensitivity analysis on  $\beta$ , see Table 10. We perform 5 independent runs for each  $\beta$ 's value.

As we can see from Table 10,  $T_{\text{EMV}}$  obtains the minimum value when  $\beta$  is 0 since the EMV's path has been cleared and reserved in advance. Comparing to  $T_{\text{EMV}}$  when  $\beta = 1$ , we witness a 35% reduction in  $T_{\text{EMV}}$  when secondary agents effortlessly reserve the segments. Likewise,  $T_{\text{avg}}$  hits the smallest value when secondary agents shift their responsibility onto congestion reduction entirely. Under this scenario, pre-emption only happens on the segment which EMV is currently traveling on and the EMV will travel with the slowest speed since more vehicles appear ahead of its road. Although every non-EMV saves about 10 s for the whole trip in this case, an ambulance might delay for more than one minute and a precious life might be lost. We believe it would be the best for the traffic system administrator to decide  $\beta$  given the emergency of the case.

## 7. Conclusion

In this paper, we introduced a decentralized reinforcement learning framework, EMVLight, to facilitate the efficient passage of EMVs and reduce traffic congestion at the same time. Leveraging the multi-agent A2C framework, agents incorporate dynamic routing and cooperatively control traffic signals to reduce EMV travel time and average travel time of non-EMVs. Our work considers the realistic settings of emergency capacitated road segments and traffic patterns before, during and after EMV passages. Extending Dijkstra's algorithm and embedding into the multi-class RL agent design, EMVLight fundamentally addresses the coupling challenge

of EMV's dynamic routing and traffic signal control, filling the research gap on this particular task. Evaluated on both synthetic and real-world map, EMVLight shortens the EMV travel time and average travel time by up to 42.6% and 23.5% respectively, comparing with existing methods from traditional and learning-based traffic signal control for EMV-related managements. Both quantitative and qualitative assessments on EMVLight, including its EMV navigation as well as pre-clear and restore traffic conditions under emergency state, have concluded that EMVLight is a promising control scheme for such task.

Non-trivial future directions for this study including, but not limited to, the followings. First, the interactions among vehicles, especially under emergency, are extremely complicated. As an effort to close the gap between simulation and reality, we are motivated to extend current ETA estimation model to capture more realistic traffic patterns with EMVs so that agents are more responsive in the field tests. Second, we are looking forward to navigating multiple EMVs simultaneously in the same traffic network. The reward design for pre-emption agents (imagine two or more EMVs appears within one intersection) is worth a technical and ethical discussion. Last but not least, as one of the RL applications in the field of transportation, our method has required a massive number of updated iterations, even in the simulated environments, to achieve the desirable result. How to learn efficiently so that our trial-and-error attempts would not bring catastrophic impacts on real traffic is a critical question for all ITS RL applications.

#### CRediT authorship contribution statement

**Haoran Su:** Conceptualization, Investigation, Methodology, Visualization, Validation, Writing – original draft. **Yaofeng D. Zhong:** Conceptualization, Investigation, Methodology, Validation, Writing – review & editing. **Joseph Y.J. Chow:** Conceptualization, Methodology, Supervision, Validation, Writing – review & editing. **Biswadip Dey:** Conceptualization, Methodology, Validation, Writing – review & editing. **Li Jin:** Conceptualization, Methodology, Supervision, Validation, Writing – review & editing.

#### Data availability

Data will be made available on request.

#### Acknowledgment

The authors are thankful to Dr. Amit Chakraborty from Siemens for discussion on the conceptualization. The authors appreciated the editor as well as anonymous reviewers for their precious comments and suggestions. This work was partially supported by Siemens, Dwight David Eisenhower Transportation Fellowship, C2SMART University Transportation Center, NSFC Project 62103260, SJTU UM Joint Institute, and J. Wu & J. Sun Endowment Fund. Joseph Y.J. Chow was partially supported by the NYU University Research Challenge Fund 2020. These supports are gratefully acknowledged, but imply no endorsement of the findings.

#### Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

#### Appendix A. Implementation details

**MDP time step.** Although MDP step length can be arbitrarily small enough for optimality, traffic signal phases should be maintained a minimum amount of time so that vehicles and pedestrians can safely cross the intersections. To avoid rapid switching between the phases, we set our MDP time step length to be 5 s.

##### Implementation details for Non-emergency-capacitated/Emergency capacitated Synthetic Grid<sub>5x5</sub>

- dimension of  $s_{\mathcal{V}_i}^t$ :  $5 \times (8 + 8 + 4 + 2) = 110$
- dimension of  $\tilde{s}_{\mathcal{V}_i}^t$ :  $5 \times (8 + 8 + 4 + 2) = 110$
- dimension of  $\pi_{\mathcal{N}_i}^{t-1}$ :  $4 \times 8 = 32$
- Policy network  $\pi_{\theta_i}(a_i^t | s_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1})$ : concat [110  $\xrightarrow{\text{FC}}$  128ReLU, 32  $\xrightarrow{\text{FC}}$  64ReLU]  $\rightarrow$  64LSTM  $\xrightarrow{\text{FC}}$  8Softmax
- Value network  $V_{\phi_i}(\tilde{s}_{\mathcal{V}_i}^t, \pi_{\mathcal{N}_i}^{t-1})$ : concat [110  $\xrightarrow{\text{FC}}$  128ReLU, 32  $\xrightarrow{\text{FC}}$  64ReLU]  $\rightarrow$  64LSTM  $\xrightarrow{\text{FC}}$  1Linear
- Each link is 200 m. The free flow speed of the EMV is 12 m/s and the free flow speed for non-EMVs is 6 m/s.
- Temporal discount factor  $\gamma$  is 0.99 and spatial discount factor  $\alpha$  is 0.90.
- Initial learning rates  $\eta_\phi$  and  $\eta_\theta$  are both 1e-3 and they decay linearly. Adam optimizer is used.
- MDP step length  $\Delta t = 5$  s and for secondary pre-emption reward weight  $\beta$  is 0.5.
- Regularization coefficient is 0.01.

**Implementation details for Manhattan<sub>16x3</sub>** The implementation is similar to the synthetic network implementation, with the following differences:

- Initial learning rates  $\eta_\phi$  and  $\eta_\theta$  are both 5e-4.

**Table B.11**  
Hyper-parameters selected for RL-based methods.

Hyper-parameters	CDRL	PL	CL	EMVLight
Temporal discount	0.99			
Batch size	32	128	128	128
Buffer size	1e5	1e4	1e4	1e4
Sample size	2048	1000	1000	1000
$\eta$ and decay	0.5&0.975	0.8&0.95	0.8&0.95	0.8&0.95
Optimizer	Adam	RMSprop	RMSprop	Adam
Learning rate	0.00025	1e-3	1e-3	1e-3
# Conv layers	1	–	3	–
# MLP layers	1	4	3	1
# MLP units	(168,168)	(32,32)	(32,32)	(192,1)
MLP Activation	ReLU			
Initialization	Random Normal			
Step length $\Delta T$			5 s	

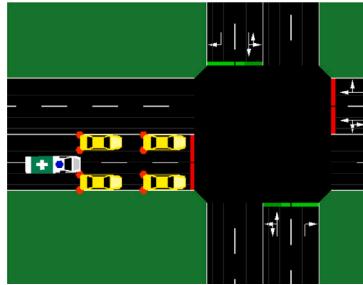


Fig. C.20. A blue light device approaching an intersection with the sublane model activated.

- Since the avenues and streets are both one-directional, the number of actions of each agent are adjusted accordingly.
- Avenues and streets length are based on real Manhattan block size with each block of 80 m  $\times$  274 m.

**Implementation details for Hangzhou<sub>4x4</sub>** The implementation is similar to the synthetic network implementation, with the following differences:

- Initial learning rates  $\eta_\phi$  and  $\eta_\theta$  are both 5e-4.
- Streets length are based on real map of Hangzhou Gudang district.

## Appendix B. Hyperparameters

We provide the choice of hyper-parameters for RL-based methods in Table B.11.

## Appendix C. Emergency lane in SUMO

With increased lateral resolution after enabling the sublane model, SUMO is able to simulate the virtual lane, i.e. emergency lane, formation for emergency traffic. Vehicles can drive between lanes under this mode. At the same time, we define the EMV as a *blue light device* to incorporate with the sublane model. Thus, we can observe that non-EMVs on the left would pull over towards the left side, i.e. latAlignment=“left”, and non-EMVs on the right would pull over towards the right side, i.e., latAlignment=“right”. When the EMV is traveling on the emergency lane, these vehicles do not perform lane changes and they do not move onto the emergency lane. After the EMV leaves this segments, these vehicles resume normal driving and move back with their previous lateral alignment. These two SUMO built-in modules together is able to achieve our proposed emergency lane. See Fig. C.20 below from Bieker-Walz et al. (2018).

## References

- Abdulhai, B., Pringle, R., Karakoulas, G.J., 2003. Reinforcement learning for true adaptive traffic signal control. *J. Transp. Eng.* 129 (3), 278–285.  
 Adacher, L., Oliva, G., Pascucci, F., 2014. Decentralized route guidance architectures with user preferences in urban transportation networks. *Proc. - Soc. Behav. Sci.* 111, 1054–1062. <http://dx.doi.org/10.1016/j.sbspro.2014.01.140>, URL <https://www.sciencedirect.com/science/article/pii/S1877042814001414>, Transportation: Can we do more with less resources? – 16th Meeting of the Euro Working Group on Transportation – Porto 2013.

- Administration, D.F.H., 2020. Use of freeway shoulders for travel — Guide for planning, evaluating, and designing part-time shoulder use as a traffic management strategy. <https://ops.fhwa.dot.gov/publications/fhwahop15023/ch1.htm>, Accessed: 2022-04-27.
- Agarwal, A., Paruchuri, P., 2016. V2V communication for analysis of lane level dynamics for better EV traversal. <http://dx.doi.org/10.1109/IVS.2016.7535412>, arXiv.org.
- Analytics, N., 2021. End-to-end response times. URL <https://www1.nyc.gov/site/fdn/y/about/resources/data-and-analytics/end-to-end-response-times.page>.
- Asaduzzaman, M., Vidyasankar, K., 2017. A priority algorithm to control the traffic signal for emergency vehicles. In: 2017 IEEE 86th Vehicular Technology Conference (VTC-Fall). pp. 1–7. <http://dx.doi.org/10.1109/VTCFall.2017.8288364>.
- Aslani, M., Mesgari, M.S., Wiering, M., 2017. Adaptive traffic signal control with actor-critic methods in a real-world traffic network with different traffic disruption events. *Transp. Res. C* 85, 732–752.
- Berdowski, J., Berg, R.A., Tijssen, J.G., Koster, R.W., 2010. Global incidences of out-of-hospital cardiac arrest and survival rates: systematic review of 67 prospective studies. *Resuscitation* 81 (11), 1479–1487.
- Bieker-Walz, L., Behrisch, M., 2019. Modelling green waves for emergency vehicles using connected traffic data. *EPiC Series Comput.* 62, 1–11.
- Bieker-Walz, L., Behrisch, M., Junghans, M., 2018. Analysis of the traffic behavior of emergency vehicles in a microscopic traffic simulation. *EPiC Series Eng.* 2, 1–13.
- Buchenscheit, A., Schaub, F., Kargl, F., Weber, M., 2009. A VANET-based emergency vehicle warning system. In: 2009 IEEE Vehicular Networking Conference. VNC, IEEE, pp. 1–8.
- Chen, C., 2019. Synthesized Hangzhou Dataset, [https://github.com/traffic-signal-control/sample-code/tree/master/data/hangzhou\\_4x4\\_gudang\\_18041610\\_1h](https://github.com/traffic-signal-control/sample-code/tree/master/data/hangzhou_4x4_gudang_18041610_1h).
- Chen, Y., Bell, M.G.H., Wang, D., Bogenberger, K., 2006. Risk-averse time-dependent route guidance by constrained dynamic a\* search in decentralized system architecture. *Transp. Res.* 1944 (1), 51–57. <http://dx.doi.org/10.1177/036119810619440010>.
- Chen, C., Wei, H., Xu, N., Zheng, G., Yang, M., Xiong, Y., Xu, K., Li, Z., 2020. Toward a thousand lights: Decentralized deep reinforcement learning for large-scale traffic signal control. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34, (04), pp. 3414–3421. <http://dx.doi.org/10.1609/aaai.v34i04.5744>, URL <https://ojs.aaai.org/index.php/AAAI/article/view/5744>.
- Chu, T., Wang, J., Codecà, L., Li, Z., 2019. Multi-agent deep reinforcement learning for large-scale traffic signal control. *IEEE Trans. Intell. Transp. Syst.*
- Coogan, S., Arcak, M., 2015. A compartmental model for traffic networks and its dynamical behavior. *IEEE Trans. Automat. Control* 60 (10), 2698–2703. <http://dx.doi.org/10.1109/TAC.2015.2411916>.
- Corman, F., D'Ariano, A., Pacciarelli, D., Pranzo, M., 2009. Evaluation of green wave policy in real-time railway traffic management. *Transp. Res. C* 17 (6), 607–616.
- De Lorenzo, R.A., Eilers, M.A., 1991. Lights and siren: A review of emergency vehicle warning systems. *Ann. Emerg. Med.* 20 (12), 1331–1335.
- El-Tantawy, S., Abdulhai, B., Abdelgawad, H., 2013. Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (MARLIN-ATSC): methodology and large-scale application on downtown toronto. *IEEE Trans. Intell. Transp. Syst.* 14 (3), 1140–1150.
- Fan, Y., Kalaba, R., Moore II, J., 2005. Shortest paths in stochastic networks with correlated link costs. *Comput. Math. Appl.* 49 (9–10), 1549–1564.
- Foerster, J., Nardelli, N., Farquhar, G., Afouras, T., Torr, P.H., Kohli, P., Whiteson, S., 2017. Stabilising experience replay for deep multi-agent reinforcement learning. In: International Conference on Machine Learning. PMLR, pp. 1146–1155.
- Fredman, M., Tarjan, R., 1984. Fibonacci heaps and their uses in improved network optimization algorithms. In: 25th Annual Symposium OnFoundations of Computer Science, 1984. pp. 338–346. <http://dx.doi.org/10.1109/SFCS.1984.715934>.
- Gajda, J., Sroka, R., Stencel, M., Wajda, A., Zeglen, T., 2001. A vehicle classification based on inductive loop detectors. In: IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics. Vol. 1, IEEE, pp. 460–464.
- Gao, S., Chabini, I., 2006. Optimal routing policy problems in stochastic time-dependent networks. *Transp. Res. B* 40 (2), 93–122.
- Gao, S., Huang, H., 2012. Real-time traveler information for optimal adaptive routing in stochastic time-dependent networks. *Transp. Res. C* 21 (1), 196–213.
- German Aerospace Center, 2022a. SUMO emergency vehicle simulation. URL <https://sumo.dlr.de/docs/Simulation/Emergency.html>.
- German Aerospace Center, 2022b. SUMO sublane model. URL <https://sumo.dlr.de/docs/Simulation/SublaneModel.html>.
- Grant, P., 2017. The human factors associated with responding to emergency vehicles. (Doctorates and Masters). Edith Cowan University Theses.
- Haghani, A., Hu, H., Tian, Q., 2003. An optimization model for real-time emergency vehicle dispatching and routing. In: 82nd Annual Meeting of the Transportation Research Board, Washington, DC. Citeseer, p. 1.
- Haghani, A., Tian, Q., Hu, H., 2004. Simulation model for real-time emergency vehicle dispatching and routing. *Transp. Res.* 1882 (1), 176–183. <http://dx.doi.org/10.3141/1882-21>.
- Hannoun, G.J., Murray-Tuite, P., Heaslip, K., Chantem, T., 2019. Facilitating emergency response vehicles' movement through a road segment in a connected vehicle environment. *IEEE Trans. Intell. Transp. Syst.* 20 (9), 3546–3557. <http://dx.doi.org/10.1109/TITS.2018.2877758>.
- Hannoun, G.J., Murray-Tuite, P., Heaslip, K., Chantem, T., 2021. Sequential optimization of an emergency response vehicle's intra-link movement in a partially connected vehicle environment. *Transp. Res.* 03611981211017914, URL: <https://doi.org/10.1177/03611981211017914>.
- He, Y., Liu, Z., Shi, J., Wang, Y., Zhang, J., Liu, J., 2015. K-shortest-path-based evacuation routing with police resource allocation in city transportation networks. *PLOS ONE* 10 (7), 1–23. <http://dx.doi.org/10.1371/journal.pone.0131962>.
- Hosseiniyadeh, M., Sinopoli, B., Kolmanovsky, I., Baruah, S., 2022. MPC-based emergency vehicle-centered multi-intersection traffic control. *IEEE Trans. Control Syst. Technol.*
- Hsiao, H., Chang, J., Simeonov, P., 2018. Preventing emergency vehicle crashes: status and challenges of human factors issues. *Human Factors* 60 (7), 1048–1072.
- Huang, H., Gao, S., 2012. Optimal paths in dynamic networks with dependent random link travel times. *Transp. Res. B* 46 (5), 579–598.
- Huang, Y.-S., Weng, Y.-S., Zhou, M., 2015. Design of traffic safety control systems for emergency vehicle preemption using timed Petri nets. *IEEE Trans. Intell. Transp. Syst.* 16 (4), 2113–2120.
- Humagain, S., Sinha, R., Lai, E., Ranjitkar, P., 2020. A systematic review of route optimisation and pre-emption methods for emergency vehicles. *Transp. Rev.* 40 (1), 35–53.
- Ismath, I., Samarasinghe, T., Dias, D., Wimalarathna, M., Rasanga, W., Jayaweera, N., Nugera, Y., 2019. Emergency vehicle traversal using DSRC/WAVE based vehicular communication. In: 2019 IEEE Intelligent Vehicles Symposium. IV, pp. 1981–1986. <http://dx.doi.org/10.1109/IVS.2019.8813981>.
- Johnson, P., Nguyen, D., Ng, M., 2016. Large-scale network partitioning for decentralized traffic management and other transportation applications. *J. Intell. Transp. Syst.* 20 (5), 461–473. <http://dx.doi.org/10.1080/15472450.2016.1151792>, URL <https://www.sciencedirect.com/science/article/pii/S1547245022000767>.
- Jotshi, A., Gong, Q., Batta, R., 2009. Dispatching and routing of emergency vehicles in disaster mitigation using data fusion. *Socio-Econ. Plan. Sci.* 43 (1), 1–24. <http://dx.doi.org/10.1016/j.seps.2008.02.005>, URL <http://www.sciencedirect.com/science/article/pii/S0038012108000165>.
- Kim, S., Lewis, M.E., White, C.C., 2005. Optimal vehicle routing with real-time traffic information. *IEEE Trans. Intell. Transp. Syst.* 6 (2), 178–188.
- Koh, S., Zhou, B., Fang, H., Yang, P., Yang, Z., Yang, Q., Guan, L., Ji, Z., 2020. Real-time deep reinforcement learning based vehicle navigation. *Appl. Soft Comput.* 96, 106694.
- Kwon, E., Kim, S., Betts, R., 2003. Route-based dynamic preemption of traffic signals for emergency vehicle operations. In: Transportation Research Board 82nd Annual Meeting Transportation Research Board. p. 1.
- Lazar, D.A., Coogan, S., Pedarsani, R., 2021. Routing for traffic networks with mixed autonomy. *IEEE Trans. Automat. Control* 66 (6), 2664–2676. <http://dx.doi.org/10.1109/TAC.2020.3020059>.

- LeBrun, J., Chuah, C.-N., Ghosal, D., Zhang, M., 2005. Knowledge-based opportunistic forwarding in vehicular wireless ad hoc networks. In: 2005 IEEE 61st Vehicular Technology Conference. Vol. 4, pp. 2289–2293. <http://dx.doi.org/10.1109/VETEC.2005.1543743>.
- Levin, M.W., Hu, J., Odell, M., 2020. Max-pressure signal control with cyclical phase structure. *Transp. Res. C* 120, 102828. <http://dx.doi.org/10.1016/j.trc.2020.102828>, URL <https://www.sciencedirect.com/science/article/pii/S0968090X20307324>.
- Li, L., Jabari, S.E., 2019. Position weighted backpressure intersection control for urban networks. *Transp. Res. B* 128, 435–461. <http://dx.doi.org/10.1016/j.trb.2019.08.005>, URL <https://www.sciencedirect.com/science/article/pii/S0191261518307896>.
- Lopez, P.A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., Lücken, L., Rummel, J., Wagner, P., Wiesner, E., 2018. Microscopic traffic simulation using sumo. In: 2018 21st International Conference on Intelligent Transportation Systems. ITSC, IEEE, pp. 2575–2582.
- Lu, L., Wang, S., 2019. Literature review of analytical models on emergency vehicle service: Location, dispatching, routing and preemption control. In: 2019 IEEE Intelligent Transportation Systems Conference. ITSC, pp. 3031–3036. <http://dx.doi.org/10.1109/ITSC.2019.8916885>.
- Ma, J., Hu, J., Hale, D.K., Bared, J., 2016. Dynamic hard shoulder running for traffic incident management. *Transp. Res. Rec.* 2554 (1), 120–128. <http://dx.doi.org/10.3141/2554-13>.
- Ma, J., Wu, F., 2020. Feudal multi-agent deep reinforcement learning for traffic signal control. In: Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems. AAMAS, Auckland, New Zealand, pp. 816–824.
- Miller-Hooks, E.D., Mahmassani, H.S., 2000. Least expected time paths in stochastic, time-varying transportation networks. *Transp. Sci.* 34 (2), 198–215.
- Mo, Z., Li, W., Fu, Y., Ruan, K., Di, X., 2022. Cvlight: Decentralized learning for adaptive traffic signal control with connected vehicles. *Transp. Res. C* 141, 103728. <http://dx.doi.org/10.1016/j.trc.2022.103728>, URL <https://www.sciencedirect.com/science/article/pii/S0968090X22001644>.
- Mostafizi, A., Koll, C., Wang, H., 2021. A decentralized and coordinated routing algorithm for connected and autonomous vehicles. *IEEE Trans. Intell. Transp. Syst.* 1–13. <http://dx.doi.org/10.1109/TITS.2021.3105057>.
- Mu, H., Song, Y., Liu, L., 2018. Route-based signal preemption control of emergency vehicle. *J. Control Sci. Eng.* 2018, 1–11. <http://dx.doi.org/10.1155/2018/1024382>.
- Musolino, G., Polimeni, A., Rindone, C., Vitetta, A., 2013. Travel time forecasting and dynamic routes design for emergency vehicles. *Proc.-Soc. Behav. Sci.* 87, 193–202.
- Nelson, E.J., Bullock, D., 2000. Impact of emergency vehicle preemption on signalized corridor operation: An evaluation. *Transp. Res. Rec.* 1727 (1), 1–11.
- Nie, Y.M., Wu, X., Homem-de Mello, T., 2012. Optimal path problems with second-order stochastic dominance constraints. *Netw. Spat. Econ.* 12 (4), 561–587.
- Noaeen, M., Naik, A., Goodman, L., Crebo, J., Abrar, T., Abad, Z.S.H., Bazzan, A.L., Far, B., 2022. Reinforcement learning in urban network traffic signal control: A systematic literature review. *Expert Syst. Appl.* 199, 116830. <http://dx.doi.org/10.1016/j.eswa.2022.116830>, URL <https://www.sciencedirect.com/science/article/pii/S0957417422002858>.
- Noori, H., Fu, L., Shiravi, S., 2016. A connected vehicle based traffic signal control strategy for emergency vehicle preemption. In: Transportation Research Board 95th Annual Meeting, pp. 16–6763.
- Nordin, N.A.M., Zaharudin, Z.A., Maasar, M.A., Nordin, N.A., 2012. Finding shortest path of the ambulance routing: Interface of A-star algorithm using c programming. In: 2012 IEEE Symposium on Humanities, Science and Engineering Research. IEEE, pp. 1569–1573.
- Panahi, S., Delavar, M., 2008. A GIS-based dynamic shortest path determination in emergency vehicles. *World Appl. Sci. J.* 3 (1), 88–94.
- Prashanth, L., Bhatnagar, S., 2010. Reinforcement learning with function approximation for traffic signal control. *IEEE Trans. Intell. Transp. Syst.* 12 (2), 412–421.
- Qin, X., Khan, A.M., 2012. Control strategies of traffic signal timing transition for emergency vehicle preemption. *Transp. Res. C* 25, 1–17.
- Roess, R.P., Prassas, E.S., McShane, W.R., 2004. Traffic Engineering. Pearson/Prentice Hall.
- Samaranayake, S., Blandin, S., Bayen, A., 2012. A tractable class of algorithms for reliable routing in stochastic networks. *Transp. Res. C* 20 (1), 199–217.
- Shaaban, K., Khan, M.A., Hamila, R., Ghanim, M., 2019. A strategy for emergency vehicle preemption and route selection. *Arab. J. Sci. Eng.* 44 (10), 8905–8913.
- Su, H., Shi, K., Chow, J.Y.J., Jin, L., 2021. Dynamic queue-jump lane for emergency vehicles under partially connected settings: A multi-agent deep reinforcement learning approach. [arXiv:2003.01025](https://arxiv.org/abs/2003.01025).
- Su, H., Zhong, Y.D., Dey, B., Chakraborty, A., 2022. Emvlight: A decentralized reinforcement learning framework for efficient passage of emergency vehicles. *AAAI Proc.* <http://dx.doi.org/10.48550/ARXIV.2109.05429>.
- Thomas, B.W., White III, C.C., 2007. The dynamic shortest path problem with anticipation. *European J. Oper. Res.* 176 (2), 836–854.
- Van der Pol, E., Oliehoek, F.A., 2016. Coordinated deep reinforcement learners for traffic light control. In: Proceedings of Learning, Inference and Control of Multi-Agent Systems (at NIPS 2016).
- Varaiya, P., 2013. Max pressure control of a network of signalized intersections. *Transp. Res. C* 36, 177–195.
- Wang, J., Ma, W., Yang, X., 2013a. Development of degree-of-priority based control strategy for emergency vehicle preemption operation. *Discrete Dyn. Nat. Soc.* 2013.
- Wang, Y., Wu, Z., Yang, X., Huang, L., 2013b. Design and implementation of an emergency vehicle signal preemption system based on cooperative vehicle-infrastructure technology. *Adv. Mech. Eng.* 5, 834976.
- Wang, X., Yin, Y., Feng, Y., Liu, H.X., 2022. Learning the max pressure control for urban traffic networks considering the phase switching loss. *Transp. Res. C* 140, 103670. <http://dx.doi.org/10.1016/j.trc.2022.103670>, URL <https://www.sciencedirect.com/science/article/pii/S0968090X22001139>.
- Wei, H., Chen, C., Zheng, G., Wu, K., Gayah, V., Xu, K., Li, Z., 2019a. Presslight: Learning max pressure control to coordinate traffic signals in arterial network. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 1290–1298.
- Wei, H., Xu, N., Zhang, H., Zheng, G., Zang, X., Chen, C., Zhang, W., Zhu, Y., Xu, K., Li, Z., 2019b. Colight: Learning network-level cooperation for traffic signal control. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1913–1922.
- Wei, H., Zheng, G., Gayah, V., Li, Z., 2019c. A survey on traffic signal control methods. [arXiv preprint arXiv:1904.08117](https://arxiv.org/abs/1904.08117).
- Wu, J., Kulcsár, B., Ahn, S., Qu, X., 2020. Emergency vehicle lane pre-clearing: from microscopic cooperation to routing decision making. *Transp. Res. B* 141, 223–239.
- Xu, B., Wang, Y., Wang, Z., Jia, H., Lu, Z., 2021. Hierarchically and cooperatively learning traffic signal control. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 35, pp. 669–677.
- Yang, L., Zhou, X., 2014. Constraint reformulation and a Lagrangian relaxation-based solution algorithm for a least expected time path problem. *Transp. Res. B* 59, 22–44.
- Zang, X., Yao, H., Zheng, G., Xu, N., Xu, K., Li, Z., 2020. MetaLight: Value-based meta-reinforcement learning for traffic signal control. In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34, (01), pp. 1153–1160.
- Zheng, G., Xiong, Y., Zang, X., Feng, J., Wei, H., Zhang, H., Li, Y., Xu, K., Li, Z., 2019. Learning phase competition for traffic signal control. In: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1963–1972.
- Ziliaskopoulos, A.K., Mahmassani, H.S., 1993. Time-dependent, shortest-path algorithm for real-time intelligent vehicle highway system applications. In: *Transportation Research Record* 1408, pp. 94–100.