Cooperative Sensing and Heterogeneous Information Fusion in VCPS: A Multi-Agent Deep Reinforcement Learning Approach

Xincao Xu, Member, IEEE, Kai Liu, Senior Member, IEEE, Penglin Dai, Member, IEEE, Ruitao Xie, Jingjing Cao, Member, IEEE, and Jiangtao Luo, Senior Member, IEEE

Abstract—Cooperative sensing and heterogeneous information fusion are critical to realize vehicular cyber-physical systems (VCPSs). This paper makes the first attempt to quantitatively measure the quality of VCPS by designing a new metric called Age of View (AoV). Specifically, we first present the system architecture where heterogeneous information can be cooperatively sensed and uploaded via vehicle-to-infrastructure (V2I) communications in vehicular edge computing (VEC). Logical views are constructed by fusing the heterogeneous information at edge nodes. Further, we formulate the problem by deriving a cooperative sensing model based on the multi-class M/G/1 priority queue, and defining the AoV by modeling the timeliness, completeness and consistency of the logical views. On this basis, a multi-agent difference reward based actor-critic with V2I bandwidth allocation (MDRAC-VBA) solution is proposed. In particular, the system state includes vehicle sensed information, edge cached information and view requirements. The vehicle action space consists of the sensing frequencies and uploading priorities of information. A difference-reward-based credit assignment is designed to divide the system reward, which is defined as the VCPS quality, into the difference reward for vehicles. Edge node allocates V2I bandwidth to vehicles based on predicted vehicle trajectories and view requirements.

I. INTRODUCTION

Recent advances in sensing technologies and vehicular communications drive the development of vehicular cyber-physical systems (VCPSs) [1], which is a key enabler of the next generation of intelligent transportation systems (ITSs) [2], [3]. In VCPS, heterogeneous information such as traffic light status, vehicle locations, point cloud data and surveillance videos can be cooperatively sensed and uploaded by vehicles [4]. The view, which is logical mapping of physical status of the elements in vehicular networks, such as the locations, velocities, and heading directions of vehicles, and the status of traffic lights can be constructed at the edge node based on the fusing of sensed information. The particular elements to be constructed in a logical view is determined by the specific ITS applications. On the other hand, vehicular edge computing (VEC) [5] becomes a promising paradigm in supporting high-density vehicular communications, massive data transmission, and adaptive computation offloading at the edge of vehicular networks. Therefore, it is of great significance to investigate quality cyber-physical fusion in VEC.

Great efforts have been devoted to data dissemination [6], [7], information caching [8], [9], [10] and task offloading [11], [12], [13] in vehicular networks. However, none of them have investigated the synergistic effects of cooperative sensing and heterogeneous information fusion. A number of studies have been studied on predicting [14], [15], scheduling [16], [17], and controlling [18], [19], [20] technologies, which facilitated the implementation of various ITS applications. Nevertheless, they are based on assumed sufficient and reliable information collected by the edge/cloud nodes. Several studies have concerned the information quality evaluation in VCPS [21], [22], [23]. However, they only evaluated the quality at the data item level, while ignoring the quality evaluation for the fusing of heterogeneous information. Some studies have focused on vehicle sensing and information fusion using deep reinforcement learning (DRL) in vehicular networks [24], [25].

1558-0016 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.
vehicular networks [27], [28]. However, none of the solutions can be directly applied in VCPS for cooperative sensing and heterogeneous information fusion. To the best of our knowledge, this is the first work on investigating the synergistic effect of cooperative sensing and heterogeneous information fusion by quantitatively measuring the quality of VCPS.

The critical issues and challenges to be addressed in this paper are summarized as follows. First, the physical information is highly dynamic. Therefore, it is critical to consider the synergistic effect of the sensing frequency, queuing delay, and transmission delay to ensure information freshness and timeliness. Second, the physical information is temporal-spatial correlated. Meanwhile, vehicles have different sensing capabilities and they are scheduled in a distributed manner. Consequently, vehicles are expected to be cooperated in information sensing and uploading to reduce resource consumption and enhance information quality. Third, vehicle-to-infrastructure (V2I) communications have limited radio coverage, and unreliable due to the nature of wireless communications. Thus, it is also critical to alleviate the impact of intermittent connection and packet loss during uploading. Fourth, the physical information is intrinsically heterogeneous in terms of distribution, updating frequency, and modality, which brings great challenges to the quality modeling of information fusion.

With above motivations, we jointly investigate the evaluation metric and the scheduling algorithm, aiming at enhancing the quality of VCPS by synergizing the cooperative sensing and heterogeneous information fusion. The main contributions are outlined as follows.

1) A novel problem is investigated to maximize the quality of VCPS by integrating the sensing, uploading, modeling and evaluation of heterogeneous information in VCPS. In particular, a cooperative sensing model is derived based on the multi-class M/G/1 priority queue and the Shannon theory. On this basis, a new metric called Age of View (AoV) is designed to quantitatively measure the quality of information fusion by evaluating the timeliness, completeness, and consistency of heterogeneous information in VCPS. To the best of our knowledge, this is the first work on quantitatively evaluating the quality of VCPS with the consideration of unique characteristics captured by the newly designed metric AoV.

2) A dedicated solution named multi-agent difference-reward based actor-critic with V2I bandwidth allocation (MDRAC-VBA) is proposed based on multi-agent actor-critic. Specifically, vehicles act as independent agents with action space of sensing frequencies and uploading priorities. The system state consists of vehicle sensed information, edge cached information, and view requirements. The system reward is defined as the achieved VCPS quality. Then, a difference reward (DR) based credit assignment scheme is designed to evaluate the contributions of individual vehicles on view construction by dividing the system reward into difference rewards, so as to enhance the evaluation accuracy in term of the action of each agent. Further, the solution manages to achieve smaller action space of each agent and speed up the convergency compared with conventional DRL algorithms. Meanwhile, a V2I bandwidth allocation (VBA) scheme is designed at the edge node based on vehicle trajectories and view requirements.

3) A comprehensive performance evaluation is conducted based on real-world vehicular trajectories. The proposed MDRAC-VBA algorithm and four competitive algorithms, including random allocation (RA), centralized deep deterministic policy gradient (C-DDPG) [26], multi-agent actor-critic (MAC) [28] and MAC with VBA scheme (MAC-VBA) are implemented. The simulation results demonstrated that MDRAC-VBA outperforms RA, C-DDPG, MAC, and MAC-VBA by around 61.8%, 23.8%, 22.0%, and 8.0%, respectively, in terms of maximizing the VCPS quality, and speeds up the convergence by around 6.8 ×, 1.4 × and 1.3 × compared with C-DDPG, MAC, and MAC-VBA, respectively.

The rest of this paper is organized as follows. Section II reviews the related work. Section III presents the system architecture. Section IV formulates the problem. Section V proposes the solution. Section VI evaluates the performance. Finally, Section VII concludes this paper and discusses future research directions.

II. RELATED WORK

There have been numerous studies on data dissemination, information caching, and task offloading in vehicular networks. Singh et al. [6] proposed an intent-based network control framework, where a neural network is used to train the flow table and enables intelligent data dissemination. Liu et al. [7] considered the cooperative data dissemination problem in a vehicular end-edge-cloud architecture, and proposed a clique searching-based scheduling scheme to enable collaborative data encoding and dissemination. Zhang et al. [8] proposed a social-aware vehicular edge caching mechanism, which dynamically orchestrates the cache capability of edge nodes and smart vehicles according to user preference similarity and service availability. Dai et al. [9] proposed a blockchain-enabled distributed information caching framework, which integrates DRL and permissioned blockchain and achieves intelligent and secure information caching. Su et al. [10] developed a dynamic information caching scheme based on the analyzed vehicular content request features. Shang et al. [11] studied energy-efficient task offloading and developed a deep-learning-based algorithm to minimize the energy consumption. Liao et al. [12] presented a task offloading strategy for air-ground integrated VEC that enables vehicles to learn long-term strategies with a multi-dimensional awareness of intent. Zhao et al. [13] proposed a mobility-aware dependent task offloading scheme to minimize the average response time and average energy consumption in urban vehicular edge computing. These studies mainly focused on scheduling algorithms for data dissemination, information caching, and task offloading in vehicular networks. However, none of them have investigated the synergistic effects of cooperative sensing and heterogeneous information fusion in VCPS.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Great efforts have been devoted to predicting, scheduling, and controlling technologies in VCPS. Zhang et al. [14] proposed a hybrid velocity-profile prediction method, which integrates the traffic flow state with individual driving behaviors. Zhang et al. [15] predicted the vehicle status based on a lane-change behavioral prediction model and an acceleration prediction model. Li et al. [16] considered vehicle mobility and developed a physical-ratio-K interference model-based broadcast scheme to ensure communication reliability. Lian et al. [17] presented a scheduling method for path planning based on an established map model to optimize the path utilization efficiency. Hu et al. [18] proposed a fuel-optimal controller to optimize the vehicle speed and continuously variable transmission gear ratio based on the leading vehicle status. Lv et al. [19] presented an adaptive algorithm to control the vehicle acceleration under three typical driving styles with different protocol selections. These studies focused on different technologies to support VCPS, such as trajectory predicting, path scheduling, and vehicle controlling, which facilitated the implementation of various ITS applications. Nevertheless, these studies are based on the assumption of the availability of quality information to model the physical elements in vehicular networks, without giving quantitative analysis on the quality of the logical views.

Several studies have evaluated the information quality in VCPS. Liu et al. [21] proposed a scheduling algorithm for temporal data dissemination in VCPS, which strikes a balance between real-time data dissemination and timely information sensing. Rager et al. [22] developed a framework to enhance the information quality by modeling random data loads to capture the stochastic nature of real networks. Yoon et al. [23] presented a unified cooperative perception framework to obtain the accurate motion states of vehicles, considering communication losses in vehicular networks and the random vehicle motions. These studies focused on information quality evaluation with respect to data timeliness, accuracy, or consistency in VCPS. Nevertheless, existing studies only considered the quality measurement at the homogeneous data item level, which may not sufficient when considering at the application level where the required logical views are constructed by fusing heterogeneous information.

Some studies have focused on vehicle sensing and information fusion by using DRL algorithms. Zhao and Liu [24] designed a social-aware incentive mechanism based on proximal policy optimization (PPO) to derive the optimal long term sensing strategy. Dong et al. [25] presented a deep Q networks (DQN) based approach to fuse information obtained on the local downstream environment for reliable lane change decisions. Mili and Cherkaoui [26] proposed a deep deterministic policy gradient (DDPG) based solution to minimize the age of information by scheduling resource block and broadcast coverage. These technologies are mainly proposed for vehicle sensing and information fusion using single agent DRL algorithms such as DQN, DDPG, and PPO. However, these algorithms cannot be directly applied for the cooperative sensing and heterogeneous information fusion in VCPS, and they are not suitable when considering multiple vehicles. A few studies applied the multi-agent DRL to allocate resources in vehicular networks. Xu et al. [27] presented a multi-agent distributed distributional deep deterministic policy gradient (MAD4PG) to maximize the service ratio by scheduling the task offloading in vehicular edge computing. He et al. [28] proposed a multi-agent actor-critic (MAC) algorithm to allocate resources for vehicles with strict delay requirements and minimum bandwidth consumption. Nevertheless, these solutions only considered one type of agents (i.e., vehicles or edge nodes) in vehicular networks.

III. System Architecture

In this section, we present a cooperative sensing and heterogeneous information fusion architecture in VCPS at the edge of vehicular networks. As shown in Fig. 1(a), the architecture can be abstracted into two layers, i.e., the physical vehicular environment and the logical views constructed by edge nodes. In particular, edge nodes such as 5G stations and roadside units (e.g., $e_1 \sim e_s$) are installed at the roadside. Vehicles are able to communicate with edge nodes within their radio coverage via V2I communications and can sense heterogeneous information via equipped onboard sensors such as LiDAR, GPS, and cameras. Clearly, the physical information in vehicular networks are highly dynamic and temporal-spatial correlated. Meanwhile, the sensing vehicles are with heterogeneous capacities and limited resources, and the vehicular communications are intermittent and unreliable. Therefore, it is critical to have a tailored metric to quantitatively evaluate the quality of the logical views constructed by the edge node, so as to measure the overall VCPS performance effectively.

As shown in Fig. 1(b), the logical view construction at edge node $e_1$ consists of three steps. Step 1 (Sensing): Each vehicle may sense different information based on their locations and sensing capacities. The sensed information is queued at each vehicle for uploading to the edge node, and each vehicle will determine the sensing frequencies and uploading priorities of these information. Step 2 (Uploading): The edge node allocates V2I bandwidth (i.e., different ranges of non-overlapped spectrums) to vehicles with uploading tasks, so that these vehicles are able to upload their sensed information simultaneously without interference. Step 3 (View Constructing): The edge node constructs the logical view by mapping the received physical information to the corresponding logical elements based on the requirement of specific ITS applications.

The system characteristics are summarized as follows. First, the heterogeneous information is sensed by vehicles at different sensing frequencies. Therefore, the arrival moments of different information could be not the same. In addition, improving the sensing frequency may enhance information freshness, but also lengthen the queuing delay. Second, it is essential to determine the uploading priorities of different information in vehicles by considering the different data sizes of information, connectivity of V2I communications, and view requirements comprehensively. Third, due to the limited bandwidth resources of edge nodes and changeable vehicular channel conditions, the allocated V2I bandwidth may not be sufficient to support the data uploading timely. It is meaningful to allocate larger bandwidth to vehicles that are prepared to upload fresher and more urgent information rather than in
worse channel conditions (e.g., leaving the V2I coverage) to maximize bandwidth efficiency. The channel conditions of different vehicles are considered by modeling the signal to noise ratio (SNR) between vehicles and the edge nodes, and the V2I transmission rate is determined by the distance between the two nodes and the allocated bandwidth.

Further, we give an example to better illustrate the idea. As shown in Fig. 1(a), a logical view is constructed in edge node $e_1$ at time $t$ to enable the speed advisory application at the intersection based on the information sensed and uploaded by vehicles $s_1$, $s_2$, and $s_3$. In general, the goal of such applications is to provide appropriate speed recommendations to vehicles approaching intersections, aiming to achieve smooth vehicle passage and maximize overall traffic efficiency. Suppose vehicles $s_2$ and $s_3$ can sense the traffic light information, but the values are not consistent at time $t$. For example, $s_2$ observes 17s remaining of the red light, whereas $s_3$ observes 16s, resulting in the information inconsistency. On the other hand, note that the status of the same physical element (e.g., the location of pedestrian $p_1$) might be sensed by multiple vehicles simultaneously (e.g., $s_1$ and $s_2$). In such case, it only needs to be uploaded by one of the vehicles (e.g., vehicle $s_1$) at certain time to save the V2I bandwidth. As long as the physical elements are modeled at the edge node with the same quality level, it can be applied to different applications without the need of repeatedly uploading by different vehicles. Moreover, the packet loss may cause a gap between the physical environment and the view. For example, suppose the packet for $s_2$’s location update is lost, which results in the significant inconsistency between its true location and modeled location at time $t$. As illustrated above, it is critical yet challenging to quantitatively measure the quality of views constructed at edge nodes, and design an effective scheduling mechanism for cooperative sensing and information fusion to maximize the overall quality of VCPS.

IV. PROBLEM FORMULATION

A. Notations

The set of discrete time slots of the system is denoted by $T = \{1, 2, \ldots, t, \ldots, T\}$, where $T$ is the number of time slots. The set of heterogeneous information is denoted by $D = \{1, 2, \ldots, d, \ldots, D\}$. Each information $d \in D$ is characterized by a two-tuple $d = (\text{type}_d, |d|)$, where $\text{type}_d$ is the type and $|d|$ is data size, measured by bit. The set of vehicles is denoted by $S = \{1, 2, \ldots, s, \ldots, S\}$. Each vehicle $s \in S$ is characterized by a three-tuple $s = (l_s, D_s, \pi_s)$, where $l_s$ is the location of vehicle $s$ at time $t$; $D_s$ is the set of information that can be sensed by vehicle $s$; and $\pi_s$ is the transmission power of vehicle $s$. The set of edge nodes is denoted by $E = \{1, 2, \ldots, e, \ldots, E\}$. Each edge node $e \in E$ is characterized by a three-tuple $e = (l_e, r_e, b_e)$, where $l_e$ is the location, $r_e$ is the communication range, and $b_e$ is the bandwidth capacity, measured by Hz. The distance between vehicle $s$ and edge node $e$ at time $t$ is denoted by $\text{dis}_{se}^t \triangleq \text{distance}(l_s, l_e)$, $\forall s \in S, e \in E, \forall t \in T$, where distance $(\cdot, \cdot)$ is the Euclidean distance.

The set of information sensed by vehicle $s$ at time $t$ is denoted by $D_s^t \subseteq D_t$. The information types are distinct for any information $d \in D_s^t$, i.e., $\text{type}_d^t \neq \text{type}_d, \forall d^* \in D_s^t \setminus \{d\}, \forall d \in D_s^t$. The sensing frequency of information $d$ in vehicle $s$ at time $t$ is denoted by $\lambda_{s,d,t}^t$. Due to the limited sensing ability, we have $\lambda_{s,d,t}^t \in \left[\lambda_{s,d,t}^{\min}, \lambda_{s,d,t}^{\max}\right]$, $\forall d \in D_s^t, \forall s \in S, \forall t \in T$, where $\lambda_{s,d,t}^{\min}$ and $\lambda_{s,d,t}^{\max}$ are the minimum and

Fig. 1. System architecture.
maximum of sensing frequency for information with type $d$ in vehicle $s$, respectively. The uploading priority of information $d$ in vehicle $s$ at time $t$ is denoted by $p_{d,s}^t$, and we have $p_{d,s}^t < p_{d',s}^t$, $\forall d,d' \in D_s, \forall s \in S, \forall t \in T$. The set of vehicles within the radio coverage of edge node $e$ at time $t$ is denoted by $S_e^t = \{s | \text{dis}_{s,e}^t \leq r_e, \forall s \in S\}$, $S_e^t \subseteq S$. The V2I bandwidth allocated by edge node $e$ for vehicle $s$ at time $t$ is denoted by $b_{e,s}^t$, and we have $b_{e,s}^t \in [0, b_e], \forall s \in S, \forall e \in E$. The sum of V2I bandwidth allocated by edge node $e$ cannot exceed its capacity $b_e$, i.e., $\sum_{s \in S, e \in E} b_{e,s}^t \leq b_e, \forall t \in T$. The primary notations are summarized in Table I.

### Table I: Summary of Primary Notations

| Notations | Descriptions | Notes |
|-----------|--------------|-------|
| $T$ | Set of discrete time slots | $T = \{t_1, t_2, \ldots, t_t, \ldots, T\}$ |
| $D$ | Set of heterogeneous information | $d \in D$ and $d = (d_e, |d|)$ |
| $S$ | Set of vehicles | $s \in S$ and $s = (l_s, D_s, p_s) |
| $E$ | Set of edge nodes | $e \in E$ and $e = (l_e, r_e, b_e)$ |
| $\text{type}_d$ | Type of information $d$ | |
| $l_s$ | Location of vehicle $s$ at time $t$ | $s \in S$ |
| $D_s$ | Set of information that can be sensed by vehicle $s$ | |
| $p_s$ | Transmission power of vehicle $s$ | |
| $l_e$ | Location of edge node $e$ | |
| $r_e$ | Communication range of edge node $e$ | |
| $b_e$ | Bandwidth capacity of edge node $e$ | |
| $\text{dis}_{s,e}$ | Distance between vehicle $s$ and edge node $e$ at time $t$ | $\text{dis}_{s,e}^t \triangleq \text{distance}(l_s, l_e)$ |
| $S_e^t$ | Set of vehicles within the radio coverage of edge node $e$ at time $t$ | $S_e^t = \{s | \text{dis}_{s,e}^t \leq r_e, \forall s \in S\}$, $S_e^t \subseteq S$ |
| $\lambda_{d,s}^t$ | Sensing frequency of information $d$ in vehicle $s$ at time $t$ | $\lambda_{d,s}^t \in [\lambda_{d,s}^{\min}, \lambda_{d,s}^{\max}], \forall d \in D_s$ |
| $p_{d,s}^t$ | Uploading priority of information $d$ in vehicle $s$ at time $t$ | $p_{d,s}^t \neq p_{d',s}^t, \forall d,d' \in D_s \setminus \{d\}, \forall d \in D_s$ |
| $b_{e,s}^t$ | V2I bandwidth allocated by edge node $e$ for vehicle $s$ at time $t$ | $b_{e,s}^t \in [0, b_e], \forall s \in S, \forall e \in E$ |
| $D_{d,s}^e$ | Set of elements with higher uploading priority than $d$ in vehicle $s$ | $D_{d,s}^e = \{d | p_{d,s}^t > p_{d',s}^t, \forall d' \in D_s\}$ |
| $c_{d,s,e}^t$ | Queuing time of information $d$ in vehicle $s$ to edge node $e$ at time $t$ | $c_{d,s,e}^t \in \{0, 1\}$ |
| $D_{d,s,e}^t$ | Set of information transmitted from vehicle $s$ to edge node $e$ at time $t$ | $D_{d,s,e}^t = \{d | c_{d,s,e}^t = 1, \forall d \in D_s\}, D_{d,s,e}^t \subseteq D_s$ |
| $O$ | Set of physical traffic elements | |
| $V$ | Set of views | |
| $\mathcal{V}_v$ | Set of views required by vehicle $v$ | $\mathcal{V}_v = \{\alpha^t | y_{v,e} = 1, \forall e \in O\}$ |
| $\mathcal{D}_{v,e}$ | Set of views required by edge node $e$ at time $t$ | $\mathcal{D}_{v,e} = \bigcup_{v \in V, e \in E} (\mathcal{D}_v \cap \mathcal{D}_e), \forall v \in \mathcal{V}_v, \forall e \in E$ |
| $\mathcal{D}_{v,e}$ | Set of views received by edge node $e$ at time $t$ and required by vehicle $v$ | |

#### B. Cooperative Sensing Model

The cooperative sensing model is shown in Fig. 2, where three vehicles denoted as $s_1$, $s_2$, and $s_3$ exhibit diverse sensing capabilities. Specifically, vehicle $s_1$ can sense heterogeneous information $d_1$, $d_4$, and $d_7$, and $d_8$ of types 1, 2, 4, and 5, whereas vehicle $s_2$ is capable of sensing information $d_2$, $d_3$, and $d_9$ of types 1, 3, and 5. Each vehicle may sense heterogeneous information at distinct frequencies and with varying uploading priorities. For instance, vehicle $s_1$ senses information, designated as $d_1$, with a sensing frequency $\lambda_{1,1}^s$ and an uploading priority $p_{1,1}^s$. Further, edge node $e$ facilitates V2I communications by assigning bandwidth to each vehicle. For example, the bandwidth allocated to vehicle $s_1$ by edge node $e$ for transmitting the information $d_8$ is $b_{1,e}^t$. Due to the unreliable nature of wireless communications, data item $d_{10}$ may be omitted from the view constructed by edge node $e$.

The inter-arrival time and queuing time of information sensed by vehicles are modeled by the multi-class M/G/1 priority queue [29]. We assume that the distribution of information transmission time with the same type $d$ in vehicle $s$ stays stable within each time slot. The transmission time $\hat{g}_{d,s,e}^t$ of information with type $d$ follows a class of General distribution with mean $\alpha_{d,s}^t$, and finite second and third moments $\beta_{d,s}^t$, $\gamma_{d,s}^t$, and the distribution set is represented by

$$
P = \{\hat{g}_{d,s,e}^t : \mathbb{E}[\hat{g}_{d,s,e}^t] = \alpha_{d,s}^t, \\
\mathbb{E}[\hat{g}_{d,s,e}^t - \alpha_{d,s}^t]^2 = \beta_{d,s}^t, \\
\mathbb{E}[\hat{g}_{d,s,e}^t - \alpha_{d,s}^t]^3 = \gamma_{d,s}^t\} 
$$

(1)

Therefore, the uploading workload $\rho_s^t$ is represented by

$$
\rho_s^t = \sum_{d \in D_s} \lambda_{d,s}^t \alpha_{d,s}^t 
$$

(2)

To guarantee the existence of the queue steady-state, it requires $\rho_s^t < 1$. The inter-arrival time $\hat{a}_{d,s}^t$ is the duration between the arrival of two adjacent information with type $d$ in vehicle $s$, which is computed by

$$
\hat{a}_{d,s}^t = \frac{1}{\lambda_{d,s}^t} 
$$

(3)

The set of elements with higher uploading priority than information $d$ in vehicle $s$ at time $t$ is denoted by

$$
D_{d,s}^t = \{d^* | p_{d,s,s}^t > p_{d,s,s}^t, \forall d^* \in D_s\} 
$$

(4)

where $p_{d,s,s}^t$ is the uploading priority of information $d^* \in D_s$. Therefore, the uploading workload ahead of information $d$
Suppose there are \( n \) types of information, and the information \( d^1 \) has the highest uploading priority, i.e., \( D_{d^1,s}^n = \emptyset \). Then, the mean queuing time of information \( d^1 \) can be computed by

\[
\bar{q}_{d^1,s}^t = \frac{\lambda_{d^1,s}^t \beta_{d^1,s}^t}{2}
\]

where \( \lambda_{d^1,s}^t \) and \( \beta_{d^1,s}^t \) are the sensing frequency and the second moment of transmission time of information \( d^1 \), respectively. On the other hand, the information \( d^n \) has the lowest uploading priority. Since it requires that \( \rho_{d^n}^t < 1 \) to ensure the queue is stable and the queuing time is finite, we have

\[
\rho_{d^n}^t = \sum_{\forall d^n \in D_{d^n,s}} \lambda_{d^n,s}^t \alpha_{d^n,s}^t < \sum_{\forall d^n \in D_{d^n,s}} \lambda_{d^n,s}^t \alpha_{d^n,s}^t = \rho_{d^n}^t < 1
\]

Similarly, \( \rho_{d^n}^t + \lambda_{d^n,s}^t \alpha_{d^n,s}^t < 1. \) The mean queuing time of transmission time of information \( d^n \) can be obtained when \( n \) tends to infinity due to \( \lim_{n \to \infty}(1 - \rho_{d^n}^t) \to 0 \), similarly, \( \lim_{n \to \infty}(1 - \rho_{d^n}^t - \lambda_{d^n,s}^t \alpha_{d^n,s}^t) \to 0 \).

\[
\lim_{n \to \infty}(\bar{q}_{d^n,s}^t) = \frac{\lambda_{d^n,s}^t \rho_{d^n}^t}{2(1 - \rho_{d^n}^t)} + \frac{\rho_{d^n}^t}{1 - \rho_{d^n}^t} \to \infty
\]

where \( \lambda_{d^n,s}^t \) and \( \beta_{d^n,s}^t \) are the sensing frequency, the mean and second moment of transmission time of information \( d^n \), respectively.

Then, we model the data uploading via V2I communications based on the Shannon theory. The SNR of V2I communications between vehicle \( s \) and edge node \( e \) at time \( t \) is denoted by \( \text{SNR}_{s,e}^t \), which is computed by [31]

\[
\text{SNR}_{s,e}^t = \frac{1}{N_0} |h_{s,e}|^2 \zeta \delta_{s,e}^t e^{-\psi} \pi_s
\]

where \( N_0 \) is the additive white Gaussian noise [31], which is a common noise model that can simplify the complexity of system modeling and performance analysis, and suitable to the vehicular network scenario where vehicles cannot interfere with each other; \( h_{s,e} \) is the channel fading gain; \( \zeta \) is a constant that depends on the antennas design, and \( \psi \) is the path loss exponent. Then, the V2I transmission rate between vehicle \( s \) and edge node \( e \) at time \( t \), denoted by \( z_{s,e}^t \), is computed by

\[
z_{s,e}^t = b_{s,e}^t \log_2(1 + \text{SNR}_{s,e}^t)
\]

where \( b_{s,e}^t \) is the bandwidth allocated to vehicle \( s \) at time \( t \). Note that given the transmission power \( \pi_s \) of vehicle \( s \), the SNR of V2I communications between vehicle \( s \) and edge node \( e \) at time \( t \) can be obtained by Eq. 14. Then, the transmission rate can be obtained by Eq. 15.

Assume vehicle \( s \) is scheduled to upload \( t \) at time \( t \), and \( d \) will be transmitted after a certain queuing time \( \bar{q}_{d,s}^t \). Then, we denote the moment when vehicle \( s \) starts to transmit \( d \) as \( k_{d,s}^t = t + \bar{q}_{d,s}^t \). The amount of data transmitted from time \( k_{d,s}^t \) to \( k_{d,s}^t + j \) can be obtained by \( \int_{k_{d,s}^t}^{k_{d,s}^t + j} z_{s,e}^t \, dt \) bits, where \( j \in \mathbb{R}^+ \) and \( z_{s,e}^t \) is the transmission rate at time \( t \). If the amount of data could be transmitted during the entire transmission is...
larger than the size of $d$, the uploading would be complete. Therefore, the transmission time of information $d$ from vehicle $s$ to edge node $e$ is denoted by $g_{d,s,e}^t$, which is computed by

$$g_{d,s,e}^t = \inf_{j \in \mathbb{R}^+} \left\{ k_{d,s}^{-j} \right\} \text{subject to } d_r \geq |d| \right\} \quad (16)$$

A successful transmission requires that the received SNR is above a certain threshold called SNR wall [32] during the packet transmission, which is obtained by

$$\text{SNR}_{\text{wall}} = \frac{\sigma^2 - 1}{\sigma} \quad (17)$$

where $\sigma = 10^{\nu/10}$, and $\nu$ is a parameter measured by dB that quantifies the size of the noise uncertainty, and $(\nu^2 - 1) N_0 = \pi_y v$. Thus, the successful transmission indicator, which indicates whether information $d$ is successfully transmitted from vehicle $s$ to edge node $e$, is denoted by

$$c_{d,s,e}^t = \left\{ \begin{array}{cl} 1, & \forall t^* \in [k_{d,s}^{-j}, k_{d,s}^{-j} + \delta_{d,s,e}], \text{SNR}_{t^*,e} > \text{SNR}_{\text{wall}} \\ 0, & \forall t^* \in [k_{d,s}^{-j}, k_{d,s}^{-j} + \delta_{d,s,e}], \text{SNR}_{t^*,e} \leq \text{SNR}_{\text{wall}} \end{array} \right\} \quad (18)$$

Therefore, the set of information transmitted by vehicle $s$ and received by edge node $e$ is denoted by $D_{t^*,e} = \{ d \mid c_{d,s,e}^t = 1, \forall d \in D_s \}, D_{t^*,e} \subseteq D_s, \forall s \in S, \forall e \in E$.

C. Heterogeneous Information Fusion Model

The set of physical elements in vehicular networks such as vehicles, pedestrians and roadside infrastructures is denoted by $O$. For each element $o \in O$, the edge node can construct a logical mapping $\delta_o$ based on the corresponding sensed information, and the set of information is denoted $D_o$. A view $v$ may contain multiple elements of the logical mapping based on specific ITS requirement, which is defined as follows.

**Definition 1 (View $v$):** View $v$ is a set of logical mapping of physical elements constructed at the edge node, which is constructed by fusing the sensed information of vehicles based on particular ITS application requirement and represented by

$$v = \{ o' \mid \gamma_{o,v} = 1, \forall o \in O \} \quad (19)$$

where $\gamma_{o,v} \in \{0, 1\}$ indicates whether the logical mapping of the physical element $o$ is included in view $v$.

The set of information required by view $v$ is denoted by $D_v$, which is the mapping of physical traffic elements required by particular ITS application, and it is represented by

$$D_v = \bigcup_{\gamma_{o,v} = 1} D_o \quad (20)$$

The number of required elements in view $v$ is denoted by $|D_v|$. Denote the set of views in the system as $V$, and denote the set of views required by edge node $e$ at time $t$ by $V^t_e \subseteq V$. Thus, the set of information received by edge node $e$ and required by view $v$ is represented by

$$\text{D}_{v,e} = \bigcup_{v \in V^t_e} \text{D}_{v,e} \quad (21)$$

and $|\text{D}_{v,e}|$ is the number of information that received by edge node $e$ and required by view $v$. Then, we define the three characteristics of heterogeneous information fusion, including the timeliness, completeness, and consistency of the view as follows.

First, the heterogeneous information is time-varying, and information freshness is essential for modeling the quality of views. Thus, we define the timeliness of information $d$ in vehicle $s$ as follows.

**Definition 2 (Timeliness of information $d$):** The timeliness $\xi_{d,s} \in (0, +\infty)$ of information $d$ in vehicle $s$ is defined as the sum of the inter-arrival time, queuing time, and transmission time of the information $d$.

$$\xi_{d,s} = a_{d,s}^t + c_{d,s}^t + g_{d,s,e}^t, \forall d \in D_s, \forall s \in S$$

Second, vehicular networks have several intrinsic characteristics including the high mobility of vehicles, restricted network resources, and unreliable wireless communications. The view may be incomplete due to the disconnection between the vehicle and the edge node, or the loss of the packet. Therefore, we define the completeness of a view as follows.

**Definition 3 (Completeness of view $v$):** The completeness $\Phi_v \in [0, 1]$ of view $v$ is defined as the ratio of the number of information actually received by edge node $e$ to the number of total required information.

$$\Phi_v = \frac{|D_{v,e}|}{|D_v|}, \forall v \in V^t_e, \forall e \in E$$

Third, since different types of information have their own sensing frequencies and uploading priorities, it is important to
keep the versions of different types of information as close as possible when constructing a view. Therefore, we define the consistency of a view as follows.

**Definition 5 (Consistency of view v):** The consistency \( \Psi_v \in (0, +\infty) \) of view \( v \) is defined as the quadratic sum of the difference between the receiving time of information \( d \) and average receiving time of information required by the view.

\[
\Psi_v = \sum_{\forall v \in \mathcal{S}} \sum_{\forall d \in \mathcal{D}_v \cap \mathcal{D}_v'} \left| q_{d,s}^v + g_{d,s,e}^v - \psi_v \right|^2, \forall v \in \mathcal{V}_e', \forall e \in \mathcal{E}
\]

where \( \psi_v \) is average receiving time of information required by view \( v \), which is computed by

\[
\psi_v = \frac{1}{|\mathcal{D}_v|} \sum_{\forall v \in \mathcal{S}} \sum_{\forall d \in \mathcal{D}_v \cap \mathcal{D}_v'} \left( q_{d,s}^v + g_{d,s,e}^v \right), \forall v \in \mathcal{V}_e', \forall e \in \mathcal{E}
\]

Finally, we give the formal definition of the age of view, which synthesizes the timeliness, completeness, and consistency of the view.

**Definition 6 (Age of View, AoV):** The age of view AoV \( v \) is defined as the weighted average of normalized timeliness, completeness, and consistency of the view \( v \).

\[
\text{AoV}_v = w_1 \text{\hat{E}}_v + w_2 \Phi_v + w_3 \Psi_v, \forall v \in \mathcal{V}_e', \forall e \in \mathcal{E}
\]

where \( \text{\hat{E}}_v \in (0, 1) \), \( \Phi_v \in (0, 1) \) and \( \Psi_v \in (0, 1) \) denote the normalized timeliness, normalized completeness, and normalized consistency of view \( v \), respectively. Note that since the dimensions of the timeliness, completeness, and consistency of view are different, in order to form a uniform representation of AoV, they are normalized to (0, 1) based on the min-max scaling as follows.

\[
\begin{align*}
\text{\hat{E}}_v & = \frac{\text{\hat{E}}_v}{\delta_{\text{\hat{E}}} |\mathcal{D}_v|} \\
\Phi_v & = 1 - \Phi_v \\
\Psi_v & = \frac{\Psi_v}{\delta_{\Psi} \max_{\forall d \in \mathcal{D}_v \cap \mathcal{D}_v'} \left( q_{d,s}^v + g_{d,s,e}^v - \psi_v \right)^2}
\end{align*}
\]

where \( \delta_{\text{\hat{E}}} \in (0, 1) \) and \( \delta_{\Psi} \in (0, 1) \) are the data scaling factors of the timeliness and consistency, respectively. They are utilized to avoid concentrating most of the values in a small range by scaling down the theoretical maximum of the timeliness and consistency in the min-max scaling.

The weighting factors of \( \text{\hat{E}}_v \), \( \Phi_v \), and \( \Psi_v \) are denoted by \( w_1, w_2, \) and \( w_3 \), respectively, and \( w_1 + w_2 + w_3 = 1 \). These three weighting factors can be tuned accordingly based on the different requirements of ITS applications. For example, for the speed advisory application at the road intersection, the vehicles are expected to receive instructions of real-time velocity from the edge node so as to pass the intersection safely and smoothly. In such a case, the timeliness factor (e.g., real-time traffic light status) is more important to be modeled in the view compared with the completeness factor (e.g., the pedestrians to be modeled in the view). Note that the lower value of AoV \( v \) indicates higher quality of the constructed view.

**D. Quality of VCPS**

Given the above metric AoV, which evaluates the quality of views individually, we further define the quality of VCPS at the system level as follows.

**Definition 7 (VCPS Quality):** The quality of VCPS \( \gamma \in (0, 1) \) is defined as the average of the complement of AoV for each view \( v \) in edge nodes during the scheduling period \( T \).

\[
\gamma = \frac{\sum_{\forall v \in \mathcal{V}_e} \sum_{\forall e \in \mathcal{E}} \sum_{\forall v \in \mathcal{V}_e'} (1 - \text{AoV}_v)}{\sum_{\forall v \in \mathcal{V}_e} \sum_{\forall e \in \mathcal{E}} |\mathcal{V}_e'| (29)}
\]

Given a solution \( (A, P, B) \), where \( A \) denotes the determined sensing frequencies, \( P \) denotes the determined uploading priorities, and \( B \) denotes the determined V2I bandwidth allocation, which are represented by

\[
\begin{align*}
A & = \{ \lambda_{d,s}^l | \forall d \in \mathcal{D}_s', \forall s \in \mathcal{S}, \forall t \in \mathcal{T} \} \\
P & = \{ p_{d,s}^l | \forall d \in \mathcal{D}_s', \forall s \in \mathcal{S}, \forall t \in \mathcal{T} \} \\
B & = \{ b_{l,s,e}^v | \forall s \in \mathcal{S}_s', \forall e \in \mathcal{E}, \forall t \in \mathcal{T} \}
\end{align*}
\]

the problem is to maximize the quality of VCPS, which is expressed as follows:

\[
\begin{align*}
\max_{A, P, B} & \gamma \\
\text{s.t.} \ C1: & \lambda_{d,s}^l \in \left[ \lambda_{d,s}^\text{min}, \lambda_{d,s}^\text{max} \right], \forall d \in \mathcal{D}_s', \forall s \in \mathcal{S}, \forall t \in \mathcal{T} \\
C2: & p_{d,s}^l \neq p_{d,s}^l, \forall s \in \mathcal{S}_s', \forall d \in \mathcal{D}_s', \forall t \in \mathcal{T} \\
C3: & b_{l,s,e}^v \in [0, b_e], \forall s \in \mathcal{S}_s', \forall e \in \mathcal{E}, \forall t \in \mathcal{T} \\
C4: & \sum_{\forall d \in \mathcal{D}_s'} \lambda_{d,s}^l \cdot a_{d,s}^l < 1, \forall s \in \mathcal{S}, \forall t \in \mathcal{T} \\
C5: & \sum_{\forall e \in \mathcal{E}} b_{l,s,e}^v \leq b_e, \forall s \in \mathcal{S}, \forall t \in \mathcal{T}
\end{align*}
\]

Constraint C1 requires that the sensing frequencies of information \( d \) in vehicle \( s \) at time \( t \) should meet the requirement of its sensing ability. C2 guarantees the uploading priority of information \( d \) in vehicle \( s \) at time \( t \). C3 states that the V2I bandwidth allocated by the edge node \( e \) for vehicle \( s \) at time \( t \) cannot exceed its bandwidth capacity \( b_e \). C4 guarantees the queue steady-state during the scheduling period \( T \). C5 requires that the sum of V2I bandwidth allocated by the edge node \( e \) cannot exceed its capacity \( b_e \).

**V. Proposed Solution**

In this section, we propose MDRAC-VBA algorithm for cooperative sensing and heterogeneous information fusion in detail. We introduce the multi-agent action determination of MDRAC-VBA for vehicles at the beginning of the section. Then, the V2I bandwidth allocation scheme of MDRAC-VBA for edge nodes is proposed. Finally, we present the detailed training process of MDRAC-VBA.

**A. Multi-Agent Action Determination for Vehicles**

As shown in Fig. 3, the solution model consists of \( S \) vehicles, edge node \( e \), the VCPS environment, and an
experience replay buffer. First, each vehicle consists of four fully-connected neural networks, i.e., a local actor, a target actor, a local critic, and a target critic network. The parameters of the local actor and local critic networks of vehicle \( s \) are denoted by \( \theta_s^\mu \) and \( \theta_s^Q \), respectively. The parameters of target actor and target critic networks are denoted by \( \theta_s^\mu \) and \( \theta_s^Q \), respectively. The parameters of actor and critic networks of vehicles are initialized randomly. The parameters of local actor and local critic networks of vehicles are initialized at the beginning. In particular, the parameters of local actor and local critic networks of vehicles are initialized as the same as the corresponding local network,

\[
\theta_s^\mu \leftarrow \theta_s^\mu, \forall s \in S \\
\theta_s^Q \leftarrow \theta_s^Q, \forall s \in S
\]

(32)

Then, the action space of vehicle \( s \) consists of the sensing frequencies and uploading priorities of the sensed information at time \( t \), which is denoted by

\[
a_s^i = \{\lambda_{d,s}^i, p_{d,s}^i | \forall d \in D_s^i\}, \forall s \in S
\]

(33)

where \( \lambda_{d,s}^i \) and \( p_{d,s}^i \) are the sensing frequency and uploading priority of information \( d \) in vehicle \( s \) at time \( t \), respectively. The set of vehicle actions is denoted by \( a^i_S = \{a_s^i | \forall s \in S\} \).

On the other hand, edge node \( e \) broadcasts its view requirements and cached information periodically. Thus, the local observation of the system state in vehicle \( s \) at time \( t \) is denoted by \( o_s^i \) and represented by

\[
o_s^i = \{D_s^i, D_e^i, V_e^i\}, \forall s \in S
\]

(34)

where \( D_s^i \) represents the set of information sensed by vehicle \( s \) at time \( t \), \( D_e^i \) represents the set of cached information in edge node \( e \) at time \( t \), and \( V_e^i \) represents the set of views required by edge node \( e \) at time \( t \). The system state at time \( t \) is denoted by

\[
o^i = \{D^i, \cdots, D_s^i, \cdots, D_e^i, V_e^i\}
\]

(35)

Further, vehicle \( s \) decides its action \( a_s^i \) by utilizing the local actor network of vehicle \( s \) with the input of the local observation \( o_s^i \) of the system state. In a specific, a random process \( \mathcal{N} \) is initialized for exploration at the beginning of each iteration. The action of vehicle \( s \) at time \( t \) is obtained by the local actor network based on its local observation of the system state.

\[
a^t_s = \mu_s (o^t_s | \theta_s^\mu) + \mathcal{N}
\]

(36)

where \( \mathcal{N} \) is an exploration noise to increase the diversity of vehicle actions.

B. V2I Bandwidth Allocation Scheme for Edge Node

The V2I bandwidth allocation (VBA) scheme is proposed for edge node \( e \) to decide its action on allocating V2I bandwidth for vehicles within its radio coverage based on predicted vehicle trajectories and view requirements. In particular, the action of edge node is denoted by \( a_e^t \), which is represented by

\[
a_e^t = \{b_{e,s}^t | \forall s \in S^t\}
\]

(37)

where \( b_{e,s}^t \) is the V2I bandwidth allocated by edge node \( e \) for vehicle \( s \) at time \( t \). First, the mobility patterns of vehicles are predicted by edge node \( e \) using the expectation-maximization (EM) method [33] based on historical distances between vehicles and the edge node. Then, the trajectory of vehicle \( s \) at future \( h \) time slots is predicted based on the EM-based mobility patterns prediction, which is denoted by \( \text{Traj}_s^t = (\hat{x}_{s,t}^{t+1}, \hat{x}_{s,t}^{t+2}, \cdots, \hat{x}_{s,t}^{t+h}) \), where \( \hat{x}_{s,t}^{t+1} \) is the predicted location of vehicle \( s \) at time \( t+1 \). Thus, the average distance of vehicle between edge node is computed by

\[
\bar{d}_{s,e}^t = \frac{1}{h} \sum_{i=1}^{h} \hat{d}_{s,e}^{t+i}
\]

(38)

where \( \hat{d}_{s,e}^{t+i} \) is the distance between the predicted location of vehicle \( s \) and the edge node, and \( \bar{d}_{s,e}^t \) is distance between vehicle \( s \) and the edge node.

Then, the set of information sensed by vehicle \( s \) and required by view \( v \) at time \( t \) is represented by

\[
D_{s,v}^t = \{d | d \in D_s^t \cap D_v\}
\]

(39)
Thus, the set of information sensed by vehicle \( s \) and required by views in edge node \( e \) at time \( t \) is denoted by
\[
D_{s,v}^t = \{ d \mid d \in \bigcup_{v \in V_e} D_{s,v}^t \}
\]
(40)
and its size is denoted by \(|D_{s,v}^t|\), which is computed by
\[
|D_{s,v}^t| = \sum_{v \in V_e} |d|
\]
(41)
Finally, the V2I bandwidth allocated by edge node \( e \) for vehicle \( s \) is calculated by
\[
b_{s,e}^t = \frac{b_e}{\omega + \text{rank}_s}
\]
(42)
where \( \omega \) is a constant, and \( \text{rank}_s \) is the sort ranking of vehicle \( s \) by the sequence of \(|D_{s,v}^t|\) in the descending order and \( \text{dis}_{s,e} \) in the ascending order.

C. Training Process of MDRAC-VBA

The training process of MDRAC-VBA consists of two parts, i.e., 1) acquiring replay experiences for training and 2) updating network parameters. First, the vehicles and edge nodes continuously interact with the environment to obtain corresponding replay experiences. Specifically, after determining the actions of vehicles and the edge node, the environment obtains the system reward according to the actions \( \{a_i^1, \ldots, a_i^t, \ldots, a_i^j\} \). The system reward is the achieved VCPs quality in edge node \( e \) at time \( t \). A DR-based credit assignment is adopted to divide the system reward into the difference rewards \( \{r_s^1, \ldots, r_s^t, \ldots, r_s^j\} \), where \( r_s^t \) is utilized to evaluate the contribution of vehicle \( s \) on view construction. In particular, the system reward is defined as the achieved VCPs quality in edge node \( e \) according to the actions \( \{a_i^1, \ldots, a_i^t, \ldots, a_i^j\} \) in the system state \( o^t \) at time \( t \), which is computed by
\[
r(a_i^1, a_i^t | o^t) = \frac{1}{|V_e^t|} \sum_{v \in V_e^t} (1 - \text{Ao}_v^t)
\]
(43)
The system reward reflects the entire system performance, which is contributed by both vehicles and the edge node. It is expected to assign the system reward into individual rewards for vehicles to evaluate their contributions. The DR-based credit scheme use the difference between the system reward and the reward achieved without the agent action to represent the individual reward of the agent, so that it can further improve the performance of MDRAC-VBA by accurately evaluating the action of each agent separately. Accordingly, the difference reward of vehicle \( s \) is represented by [34]
\[
r_s^1 = r(a_i^1, a_i^t | o^t) - r(a_i^{S|s}, a_i^t | o^t)
\]
(44)
where \( r(a_i^{S|s}, a_i^t | o^t) \) is the system reward achieved without the contribution of vehicle \( s \), and it can be obtained by setting null action set for vehicle \( s \). The set of the difference reward for vehicles is denoted by \( r_S^1 = \{r_s^1 \mid \forall s \in S\} \). Then, the interaction experiences including the system state \( o^t \), vehicle actions \( a_i^t \), difference reward \( r_s^t \), and next system state \( o^{t+1} \), are stored in the experience replay buffer \( B \), which is initialized with a maximum size \(|B|\).

Second, a minibatch of \( M \) transitions is sampled from experience replay buffer to train the actor and critic networks in vehicles. The transition of the \( M \) minibatch is denoted by \((o^m, a_i^m, r_s^m, o^{m+1})\). The loss function of the local critic network of vehicle \( s \) is computed by
\[
\mathcal{L}(\theta_s^Q) = \frac{1}{M} \sum_{m} \left( \eta_m - Q_s(o^m, a_i^m | \theta_s^Q) \right)^2
\]
(45)
where \( \eta_m \) is the target value generated by the target critic network \( \eta_m = r_s^m + \tau Q_s(o_i^{m+1}, a_i^{m+1} | \theta_s^Q) \), and \( \tau \) is the discount rate. The action of vehicle \( s \) at time \( m+1 \) is obtained by the target actor network based on the local observation of next system state, i.e., \( a_i^{m+1} = \mu_s(o_i^{m+1} | \theta_s^\mu) \). The parameters of the local actor network of vehicle \( s \) are updated via policy gradient.
\[
\nabla_{\theta_s^\mu} J \approx \frac{1}{M} \sum_m \nabla a_i^m Q_s(o^m, a_i^m | \theta_s^Q) \nabla_{\theta_s^\mu} \mu_s(o_i^{m+1} | \theta_s^\mu)
\]
(46)
Finally, vehicles update the parameters of target actor and critic networks,
\[
\theta_s^{\mu'} \leftarrow \eta_s \theta_s^{\mu} + (1 - \eta_s) \theta_s^{\mu'}, \forall s \in S
\]
\[
\theta_s^{Q'} \leftarrow \eta_s \theta_s^{Q} + (1 - \eta_s) \theta_s^{Q'}, \forall s \in S
\]
(47)
with \( \eta_s \ll 1, \forall s \in S \). The procedure of MDRAC-VBA is shown in Algorithm 1.

VI. PERFORMANCE EVALUATION

A. Settings

In this section, we implement a simulation model by using Python 3.9 and PyTorch 1.11.0 to evaluate the performance of MDRAC-VBA. The simulation model\(^1\) is based on a Ubuntu 20.04 server with an AMD Ryzen 9 5950X 16-core processor @ 3.4 GHz, two NVIDIA GeForce RTX 3090 graphic processing units, and 64 GB memory. In particular, we have examined three traffic scenarios using real-world vehicle trajectories\(^2\) collected from Didi GAIA open data set [35], including 1): a 3 km × 3 km area of Qingyang District, Chengdu, China, from 8:00 to 8:05, on 16 Nov. 2016; 2): the same area from 23:00 to 23:05, on 16 Nov. 2016; 3): a 3 km × 3 km area of Beilin District, Xian, China, from 8:00 to 8:05, on 27 Nov. 2016. Detailed statistics including the total number of vehicle traces, average dwell time (ADT) of vehicles, the variance of dwell time (VDT), the average number of vehicles (ANV) in each second, the variance of the number of vehicles (VNV), the average speed of vehicles (ASV), and the variance of speeds of vehicles (VSV) are summarized in Table II. Fig. 4 shows the heat maps of vehicle distribution within the scheduling period \( T \) to better exhibit the traffic characteristic under different

\(^1\)The code of the simulation model can be found at https://github.com/neardws/Multi-Agent-Deep-Reinforcement-Learning.

\(^2\)The code of vehicle trajectory processing can be found at https://github.com/neardws/Vehicular-Trajectories-Processing-for-Didi-Open-Data.
Algorithm 1 MDRAC-VBA
1 Initialize the parameters $\theta_1^\mu$, $\theta_2^Q$, $\forall s \in S$, randomly;
2 $\theta_1^\mu \leftarrow \theta_1^\mu$, $\theta_2^Q \leftarrow \theta_2^Q$, $\forall s \in S$;
3 Initialize the experience replay buffer $B$;
4 for iteration $= 1$ to max-iteration-number do
   Initialize a random process $N$ for exploration;
   Receive the initial system state $o_1$;
   for $t = 1$ to $T$ do
      for vehicle $s = 1$ to $S$ do
         Receive a local observation $o_s^t$;
         Select a action $a_s^t = \mu_s (o_s^t | \theta_1^\mu) + N_t^s$;
         Predict the mobility pattern via EM-based method with historical relative distances;
         Predict the future locations $\text{Traj}_s^t$;
         Compute the average distance $d_{\text{dis}}^{i,t}$;
         Obtain the information set $D_{s,t}^i$;
      end for
      for vehicle $s = 1$ to $S$ do
         Allocate the bandwidth $b_{s,c}^t$ to $s$ via the VBA scheme $b_{s,c}^t = \frac{1}{n_s} \text{rank}_c$;
         Receive the system reward $r (a_s^t, a_{\bar{s}}^t | o^t)$ and the next system state $o^{t+1}$;
         Divide the system reward into the difference reward $r_{s,c}^t$ for vehicles;
         Store $(o_s^t, a_{s,c}^t, r_{s,c}^t, o^{t+1})$ in replay buffer;
      end for
   Sample $M$ transitions from replay buffer randomly;
   Update the local critic network based on the loss function
   $L (\theta_2^Q) = \frac{1}{B} \sum_m (\eta_m - Q_s (a_m^s, a_{\bar{s}}^m | \theta_2^Q))^2$;
   Update the local actor network via policy gradient
   $\nabla_{\theta_1^\mu} J \approx \frac{1}{B} \sum_m \nabla a_m^s \bar{Q}_s (a_m^s, a_{\bar{s}}^m | \theta_2^Q) \nabla \mu_s (a_m^s | \theta_1^\mu)$;
   Update the target actor networks of vehicles
   $\theta_1^\mu \leftarrow n_s \theta_1^\mu + (1 - n_s) \theta_1^{\mu_i}$, $\forall s \in S$;
   Update the target critic networks of vehicles
   $\theta_2^Q \leftarrow n_s \theta_2^Q + (1 - n_s) \theta_2^{Q_i}$, $\forall s \in S$;
end for

TABLE II
Traffic Characteristics of Each Scenario

| Scenario | Map   | Number of Traces | Time  | Date         | ADT | VDT | ANV | VNV | ASV | VSV |
|----------|-------|------------------|-------|--------------|-----|-----|-----|-----|-----|-----|
| 1        | Chengdu | 718              | 08:00 | Nov. 16, 2016 | 198.3(s) | 123.8 | 474.6 | 11.6 | 5.22(m/s) | 2.61 |
| 2        | Chengdu | 359              | 23:00 | Nov. 16, 2016 | 173.7(s) | 124.1 | 207.9 | 3.93 | 7.30(m/s) | 3.16 |
| 3        | Xian   | 206              | 08:00 | Nov. 27, 2016 | 145.5(s) | 114.7 | 99.9  | 7.65 | 8.06(m/s) | 3.70 |

It is observed that the vehicle distribution is totally different in Fig.4(c), which is extracted from another city.

The parameter settings are described as follows. The data sizes of information are uniformly distributed in the range of [100B, 1MB]. The transmission power of each vehicle is set to 1 mW. The additive white Gaussian noise and the path loss exponent of V2I communications are set to -90 dBm and 3, respectively [31]. The channel fading gains of V2I communications follow the Gaussian distribution with a mean of 2 and a variance of 0.4, and the bandwidth of the edge node is set to 3 MHz. The noise uncertainty is set to follow uniformly distributed in the intervals [0, 3] dB [32].

For the implementation of MDRAC-VBA, the architectures and hyperparameters of the actor and critic networks are described as follows. The local actor network is a four-layer fully connected neural network with two hidden layers, where the numbers of neurons are 64, and 32, respectively. The architecture of the target actor network is the same as the local actor network. The local critic network is a four-layer fully connected neural network with two hidden layers, where the numbers of neurons are 128, and 64, respectively. The architecture of the target critic network is the same as the local critic network. The Rectified Linear Unit (ReLU) is utilized as the activation function and the Adam optimizer is used to update network weights with a learning rate of 0.001, and the discount factor is set to 0.996. The size of the experience replay buffer $|B|$ is set to 100000, and the size of minibatch $M$ is set to 512. Four comparable algorithms are implemented as follows.

- **RA**: it randomly selects one action on determining the sensing frequencies, uploading priorities, and V2I bandwidth allocation.
- **C-DDPG** [26]: it implements an agent at the edge node to determine the sensing frequencies, uploading priorities, and V2I bandwidth allocation in a centralized way based on the system state. Meanwhile, the system reward is received by the agent to evaluate its contribution.
- **MAC** [28]: it implements agents in vehicles to decide the sensing frequencies and uploading priorities based on local observation of the physical environment, and an agent in the edge node to decide the V2I bandwidth allocation. The system reward is received by each agent to evaluate their contributions, which is the same for each agent.
- **MAC-VBA**: To enable MAC to better allocate V2I bandwidth, we further design a variant called MAC-VBA, where edge nodes allocate V2I bandwidth based on predicted vehicle trajectories and view requirements. Furthermore, the following metrics are designed for performance evaluation.
Fig. 4. Heat map of the distribution of vehicles under different scenarios.

Fig. 5. Convergence comparison.

- **Cumulative Reward (CR):** it is the cumulative system reward during the scheduling period $T$, which is computed by $\sum_{t \in T} r\left(a^t_S, a^t_e | o^t\right)$.

- **Composition of Average Reward (CAR):** it is defined as the percentage of the normalized timeliness, completeness, and consistency in the average reward and formulated by $\frac{\sum_{t \in T} \left(1 - \hat{\Sigma}_v, \frac{2}{1 - \hat{\Phi}_v}, \frac{3}{1 - \hat{\Psi}_v}\right)}{S} >$.

- **Average Queuing Time (AQT):** it is defined as the sum of queuing time of the sensed information divided by the number of information during the scheduling period $T$, which is computed by $\sum_{t \in T} \left(\frac{\sum_{s \in S} \left(\sum_{d \subseteq D} q^t_{d,s} \right)}{|D^t_s|} / |S| / T$.}

- **Service Ratio (SR):** it is defined as the number of views, which satisfy the completeness requirement, over the total number of required views during the scheduling period $T$, which is computed by $\sum_{t \in T} \sum_{v \in V_t} I\left(\phi_v \geq \Phi_{threshold}\right) / \sum_{t \in T} |V^t_c|$, and $\Phi_{threshold}$ is the completeness threshold.

B. Results and Analysis

1) **Algorithm Convergence:** Fig. 5 compares the CR of the five algorithms in terms of convergence speed. As noted, MDRAC-VBA converges the fastest (around 660 iterations) and achieves the highest CR value (around 357). In contrast, C-DDPG, MAC and MAC-VBA converge after around 4500, 950, and 870 iterations and achieve the CR around 307, 290 and 315, respectively. RA achieves a CR of around 241 as the baseline. We note that MDRAC-VBA achieves about 16.3%, 23.1%, and 13.3% improvement with respect to CR and about 6.8x, 1.4x, and 1.3x improvement with respect to convergence speed compared with C-DDPG, MAC, and MAC-VBA, respectively. The primary reason is that MDRAC-VBA is designed to maintain a stable communication environment for vehicles, which makes the training of actor and critic networks in vehicles more efficient. On the other hand, due to the smaller action space of MDRAC-VBA model, MDRAC-VBA converges much faster than C-DDPG, which decides the actions on determining the sensing frequencies, uploading priorities, and V2I bandwidth allocation simultaneously using a DDPG agent.

2) **Effect of Traffic Scenarios:** Fig. 6 compares the performance of the five algorithms under different traffic scenarios. Fig. 6(a) compares the VCPS quality of the five algorithms. As demonstrated, MDRAC-VBA achieves the highest VCPS quality under all scenarios. In particular, MDRAC-VBA improves the VCPS quality by 58.0%, 27.1%, 19.1%, and 12.5% on average over RA, C-DDPG, MAC, and MAC-VBA, respectively, under different traffic scenarios. Fig. 6(b) shows the average AoV of the five algorithms. It is expected that MDRAC-VBA achieves the lowest average AoV under all the scenarios. Fig. 6(c) compares the CR of the five algorithms. As noted, the CR of MDRAC-VBA is higher than RA, C-DDPG, MAC, and MAC-VBA. Meanwhile, the CR of MDRAC-VBA and MAC-VBA are similar under scenario 3. The reason is that the lower vehicle density and higher traffic dynamic in scenario 3 make the data uploading more difficult than that in scenarios 1 and 2.

Fig. 6(d) breaks down the average reward into three parts, which demonstrates the proportion of timeliness, completeness, and consistency, respectively. It is observed that the timeliness and consistency of the five algorithms are very small under scenario 3. The reason is mainly because the requirements of timeliness and consistency can hardly be satisfied when the view is incomplete. Figs. 6(e) and 6(f) compare the AQT and SR of the five algorithms under different traffic scenarios.
Fig. 6. Performance comparison under different traffic scenarios.

It demonstrates that MDRAC-VBA achieves the lowest AQT, and maintains the highest SR under all scenarios.

3) Effect of V2I Bandwidths: Fig. 7 compares the performance of the five algorithms under different V2I bandwidths. In this set of experiments, we consider the V2I bandwidth of edge nodes increases from 1 MHz to 5 MHz. A larger bandwidth represents that more information can be uploaded via V2I communications. Fig. 7(a) compares the VCPS quality...
Fig. 8. Performance comparison under different requirements on views.

of the five algorithms. With the increasing bandwidth, the VCPS quality of all algorithms increases accordingly. The VCPS quality of MDRAC-VBA outperforms RA, C-DDPG, MAC, and MAC-VBA by around 72.9%, 28.3%, 17.8% and 9.3%, respectively, under different bandwidths of the edge node. Fig. 7(b) compares the average AoV of the five algorithms. In particular, MDRAC-VBA achieves the lowest average AoV across all cases. Fig. 7(c) compares the CR of the five algorithms. As expected, the performance of all five algorithms gets better when the bandwidth increases. Specifically, MDRAC-VBA achieve around 75.1%, 29.4%, 22.7%, and 10.6% improvement in CR than RA, C-DDPG, MAC, and MAC-VBA, respectively.

Fig. 7(d) compares the CAR of the five algorithms. MDRAC-VBA achieves a better performance than the other four algorithms, particularly, in terms of timeliness and consistency of average reward. This is because the cooperation of sensing and uploading information among vehicles is more efficient in MDRAC-VBA under the limited bandwidth. Fig. 7(e) compares the AQT of the five algorithms. As noted, the AQT of MDRAC-VBA maintains the lowest under different edge bandwidths, which reflects that the designed proposed solution can allocate the bandwidth more efficiently. The advantage can be further justified by Fig. 7(f), which shows the SR of the five algorithms. The SR of MDRAC-VBA remains at the highest level across all cases.

4) Effect of View Requirements: Fig. 8 compares the performance of the five algorithms under different requirements on views, in which the average size of views required by ITS applications increases from 0.25× to 4×, and the average view size of 1× is around 6.46 MB. Fig. 8(a) compares the VCPS quality of the five algorithms. As expected, the performance of all five algorithms gets worse when the average view size increases. MDRAC-VBA outperforms RA, C-DDPG, MAC, and MAC-VBA by around 68.1%, 23.5%, 27.9% and 4.9%, respectively, in terms of maximizing the VCPS quality under different application requirements on views. Figs. 8(b) and 8(c) compare the average AoV and CR of the five algorithms. When the average view size is small (i.e., around 1.62 MB), the average AoV in MDRAC-VBA is slightly lower than that in MAC and MAC-VBA. Meanwhile, note that the CR of MDRAC-VBA, MAC, and MAC-VBA are similar. The reason is that a smaller data size has a higher probability of being successfully uploaded. Fig. 8(d) compares the CAR of the five algorithms. It is observed that the performance difference between MDRAC-VBA and MAC-VBA is small when the average view size increases from 0.25× to 0.5×. The reason is the scheduling effect is not significant when there are sufficient resources to meet the requirements of a smaller average view size (i.e., around 1.62 MB and 3.23 MB). Figs. 8(e) and 8(f) compare the AQT and SR of the five algorithms, showing that MDRAC-VBA can remain the lowest AQT, and meanwhile achieve the highest SR in most cases. It is noted that MAC-VBA achieves the lowest AQT and the highest SR when the average view size is 2×, which reflects that the proposed VBA scheme can allocate the bandwidth more efficiently.

VII. CONCLUSION AND FUTURE WORK

In this paper, a new metric AoV was designed to evaluate the quality of the logical view constructed at the edge in terms of timeliness, completeness, and consistency of heterogeneous
information in VCPS. On this basis, the problem was formulated to maximize the VCPS quality. Further, a tailored solution based on multi-agent DRL was proposed, in which vehicles act as independent agents to determine the sensing frequencies and uploading priorities. Edge nodes allocated the V2I bandwidth based on VBA scheme by considering vehicle trajectories and view requirements. The DR-based credit assignment scheme was adopted to divide the system reward based on the difference between the system reward and achieved reward without the action of vehicle, which are utilized to evaluate the individual contributions of vehicles. Finally, a comprehensive performance evaluation was conducted to demonstrate the significance of the newly designed metric AoV and the superiority of MDRAC-VBA. In particular, MDRAC-VBA outperforms RA, C-DDPG, MAC, and MAC-VBA by around 61.8%, 23.8%, 22.0%, and 8.0%, respectively, in terms of maximizing the VCPS quality. Meanwhile, compared with C-DDPG, MAC, and MAC-VBA, MDRAC-VBA speeds up the convergence by around 6.8×, 1.4×, and 1.3×, respectively.

As an early stage of exploring cooperative sensing and information fusion in VCPS, this work focused on the AoV modeling and evaluation within the coverage of a single edge node. In our future work, the cooperation among the edge nodes will be further investigated to extend the supported ITS applications as well as enhance overall system performance. Also, the transmission power control and allocation will be further investigated when considering the overall power consumption and spatial reusability of V2V communication.

REFERENCES

[1] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, “A survey on plateau-based vehicular cyber-physical systems,” IEEE Commun. Surveys Tuts., vol. 18, no. 1, pp. 263–284, 1st Quart., 2016.
[2] K. Liu, X. Xu, M. Chen, B. Liu, L. Wu, and V. C. S. Lee, “A hierarchical architecture for the future Internet of Vehicles,” IEEE Commun. Mag., vol. 57, no. 7, pp. 41–47, Jul. 2019.
[3] X. Deng et al., “A review of 6G autonomous intelligent transportation systems: Mechanisms, applications and challenges,” J. Syst. Archit., vol. 142, Jan. 2023, Art. no. 102929.
[4] W. Wu, X. Deng, P. Jiang, S. Wan, and Y. Guo, “CrossFuser: Multimodal feature fusion for end-to-end autonomous driving under unseen weather conditions,” IEEE Trans. Intell. Transp. Syst., vol. 24, no. 12, pp. 14378–14392, Mar. 2023.
[5] K. Liu, C. Liu, G. Yan, V. C. S. Lee, and J. Cao, “Accelerating dnn inference with reliability guarantee in vehicular edge computing,” IEEE-ACM Trans. Netw., early access, 2023, doi: 10.1109/TNET.2023.3279512.
[6] A. Singh, G. S. Aujla, and R. S. Bali, “Intent-based network for data dissemination in software-defined vehicular edge computing,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 8, pp. 5310–5318, Jun. 2020.
[7] K. Liu, K. Xiao, P. Dai, V. C. S. Lee, S. Guo, and J. Cao, “Fog computing empowered data dissemination in software defined heterogeneous VANETs,” IEEE Trans. Mobile Comput., vol. 20, no. 11, pp. 3181–3193, Nov. 2021.
[8] K. Zhang, J. Cao, S. Maharjan, and Y. Zhang, “Digital twin empowered content caching in social-aware vehicular edge networks,” IEEE Trans. Comput. Soc. Syst., vol. 9, no. 1, pp. 239–251, Apr. 2022.
[9] Y. Dai, D. Xu, K. Zhang, S. Maharjan, and Y. Zhang, “Deep reinforcement learning and permissioned blockchain for content caching in vehicular edge computing and networks,” IEEE Trans. Veh. Technol., vol. 69, no. 4, pp. 4312–4324, Apr. 2020.
[10] Z. Su, Y. Hai, Q. Xu, T. Yang, J. Liu, and Y. Jia, “An edge caching scheme to distribute content in vehicular networks,” IEEE Trans. Veh. Technol., vol. 67, no. 6, pp. 5346–5356, Jun. 2018.
[11] B. Shang, L. Liu, and Z. Tian, “Deep learning-assisted energy-efficient task offloading in vehicular edge computing systems,” IEEE Trans. Veh. Technol., vol. 70, no. 9, pp. 9619–9624, Sep. 2021.
[12] H. Liao et al., “Learning-based intent-aware task offloading for air-ground integrated vehicular edge computing,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 8, pp. 5127–5139, Aug. 2021.
[13] L. Zhao et al., “MESON: A mobility-aware dependent task offloading scheme for urban vehicular edge computing,” IEEE Trans. Mob. Comput., early access, 2023, doi: 10.1109/TMC.2023.3260911.
[14] Y. Zhang, L. Chu, Y. Ou, C. Guo, Y. Liu, and X. Tang, “A cyber-physical system-based velocity-profile prediction method and case study of application in plug-in hybrid electric vehicle,” IEEE Trans. Cybern., vol. 51, no. 1, pp. 40–51, Jan. 2021.
[15] T. Zhang, Y. Zou, X. Zhang, N. Guo, and W. Wang, “Data-driven based cruise control of connected and automated vehicles under cyber-physical system framework,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 10, pp. 6307–6319, Oct. 2021.
[16] C. Li, H. Zhang, T. Zhang, J. Rao, L. Y. Wang, and G. Yin, “Cyber-physical scheduling for predictable reliability of inter-vehicle communications,” IEEE Trans. Veh. Technol., vol. 69, no. 4, pp. 4192–4206, Apr. 2020.
[17] Y. Lian, Q. Yang, W. Xie, and L. Zhang, “Cyber-physical system-based heuristic planning and scheduling method for multiple automatic guided vehicles in logistics systems,” IEEE Trans. Ind. Inform., vol. 17, no. 11, pp. 7882–7893, Nov. 2021.
[18] X. Hu, H. Wang, and X. Tang, “Cyber-physical control for energy-saving vehicle following with connectivity,” IEEE Trans. Electron. Dev., vol. 64, no. 11, pp. 8578–8587, Nov. 2017.
[19] C. Lv, X. Hu, A. Sangiovanni-Vincentelli, Y. Li, C. M. Martinez, and D. Cao, “Driveing-style-based codesign optimization of an automated electric vehicle: A cyber-physical system approach,” IEEE Trans. Ind. Electron., vol. 66, no. 4, pp. 2965–2975, Apr. 2019.
[20] J. Shi, P. Cong, L. Zhao, X. Wang, S. Wan, and M. Guizani, “A two-stage strategy for UAV-enabled wireless power transfer in unknown environments,” IEEE Trans. Mobile Comput., early access, 2023, doi: 10.1109/TMC.2023.3240763.
[21] J. Liu, V. C. S. Lee, J. Y. W. Ng, J. Chen, and S. H. Son, “Temporal data dissemination in vehicular cyber-physical systems,” IEEE Trans. Intell. Transp. Syst., vol. 15, no. 6, pp. 2419–2431, Mar. 2014.
[22] S. T. Rager, E. N. Ciftcioglu, R. Ramanathan, T. F. La Porta, and R. Govindan, “Scalability and satisfiability of quality-of-information in wireless networks,” IEEE-ACM Trans. Netw., vol. 26, no. 1, pp. 398–411, Dec. 2017.
[23] D. D. Yoon, B. Ayalew, and G. G. M. N. Ali, “Performance of decentralized cooperative perception in V2V connected traffic,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 7, pp. 6850–6863, Jan. 2022.
[24] Y. Zhao and C. H. Liu, “Social-aware incentive mechanism for vehicular crowdsensing by deep reinforcement learning,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 4, pp. 2314–2325, Aug. 2020.
[25] J. Dong et al., “Spatio-weighted information fusion and DRL-based control for connected autonomous vehicles,” in Proc. IEEE Int. Conf. Intell. Transp. Syst. (ITSC), Sep. 2020, pp. 1–6.
[26] Z. Mitka and S. Cherkassky, “Deep deterministic policy gradient to minimize the age of information in cellular V2X communications,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 12, pp. 23597–23612, Jul. 2022.
[27] X. Xu et al., “Joint task offloading and resource optimization in noma-based vehicular edge computing: A game-theoretic DRL approach,” J. Syst. Archit., vol. 134, Jan. 2023, Art. no. 102780.
[28] Y. He, Y. Wang, F. R. Yu, Q. Lin, J. Li, and V. C. Leung, “Efficient resource allocation for multi-beam satellite-terrestrial vehicular networks: A multi-agent actor-critic method with attention mechanism,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 3, pp. 2727–2738, Dec. 2021.
[29] Z. Qian, F. Wu, J. Pan, K. Srinivasan, and N. B. Shroff, “Minimizing age of information in multi-channel time-sensitive information update systems,” in Proc. IEEE Conf. Comput. Commun. (INFOCOM), Jul. 2020, pp. 446–455.
[30] T. Takine, “Queue length distribution in a fifo single-server queue with multiple arrival streams having different service time distributions,” Queueing Syst., vol. 39, no. 4, pp. 349–375, 2001.
[31] A. K. Sadek, Z. Han, and K. J. R. Liu, “Distributed relay-assignment protocols for coverage expansion in cooperative wireless networks,” IEEE Trans. Mobile Comput., vol. 9, no. 4, pp. 505–515, Apr. 2010.
[32] R. Tandra and A. Sahai, “SNR walls for signal detection,” IEEE J. Sel. Topics Signal Process., vol. 2, no. 1, pp. 4–17, Feb. 2008.
[33] T. Hofmann, “Unsupervised learning by probabilistic latent semantic analysis,” Mach. Learn., vol. 42, no. 1, pp. 177–196, Jan. 2001.
[34] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, “Counterfactual multi-agent policy gradients,” in Proc. AAAI Conf. Artif. Intell., Apr. 2018, vol. 32, no. 1, pp. 2974–2982.
[35] Data Source: Didi Chuxing Gaia Open Dataset Initiative. Accessed: Mar. 14, 2020. [Online]. Available: https://outreach.didichuxing.com/research/opendata/en/

Xincao Xu (Member, IEEE) received the B.S. degree from the North University of China, Taiyuan, China, in 2017, and the Ph.D. degree from Chongqing University, Chongqing, China, in 2023. He is currently a Post-Doctoral Research Fellow with the Shenzhen Institute for Advanced Study, University of Electronic Science and Technology of China, Shenzhen, China. His research interests include vehicular networks, vehicular cyber-physical systems, edge computing, and deep reinforcement learning.

Kai Liu (Senior Member, IEEE) received the Ph.D. degree in computer science from the City University of Hong Kong in 2011. From 2010 to 2011, he was a Visiting Scholar with the Department of Computer Science, University of Virginia, Charlottesville, VA, USA. From 2011 to 2014, he was a Post-Doctoral Fellow with Nanyang Technological University, Singapore; the City University of Hong Kong; and Hong Kong Baptist University, Hong Kong. He is currently a Full Professor with the College of Computer Science, Chongqing University, China. His research interests include mobile computing, pervasive computing, intelligent transportation systems, and the Internet of Vehicles.

Penglin Dai (Member, IEEE) received the B.S. degree in mathematics and applied mathematics and the Ph.D. degree in computer science from Chongqing University, Chongqing, China, in 2012 and 2017, respectively. He is currently an Associate Professor with the School of Computing and Artificial Intelligence, Southwest Jiaotong University, Chengdu, China. His research interests include intelligent transportation systems and vehicular cyber-physical systems.

Ruitao Xie received the B.Eng. degree from the Beijing University of Posts and Telecommunications in 2008 and the Ph.D. degree in computer science from the City University of Hong Kong in 2014. She is currently an Assistant Professor with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. Her research interests include AI networking and mobile computing, distributed systems, and cloud computing.

Jingjing Cao (Member, IEEE) received the Ph.D. degree in computer sciences from the City University of Hong Kong, Hong Kong, China, in 2013. She is currently an Associate Professor with the School of Transportation and Logistics Engineering, Wuhan University of Technology, Hubei, China. Her current research interests include machine learning, pervasive computing, and their applications in transportation and logistics.

Jiangtao Luo (Senior Member, IEEE) received the B.S. degree from Nankai University in 1993 and the Ph.D. degree from the Chinese Academy of Sciences in 1998. He was a Visiting Scholar with the University of Hamburg, Germany, and the University of Southern California, Los Angeles, CA, USA, from 2015 to 2016. He is currently a Full Professor, a Ph.D. Supervisor, and the Deputy Dean of the Electronic Information and Networking Research Institute, Chongqing University of Posts and Telecommunications (CQUPT). He has published more than 150 articles and owned 30 patents in these fields. His research interests include network data analysis, visual big data, and future internet architecture. He received the Chongqing Provincial Award of the Scientific and Technological Progress in 2007 and 2010, the Chongqing Science and Technology Award for Youth in 2010, and the Chinese State Award of the Scientific and Technological Progress in 2011.