Computer Communications 221 (2024) 54–65

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Joint mode selection and resource allocation for cellular V2X communication using distributed deep reinforcement learning under 5G and beyond networks

Shalini Yadav[1,*], Rahul Rishi[1]

Computer Science and Engineering, UIET, Maharishi Dayanand University, Rohtak, Haryana, India

ARTICLE INFO

Keywords:
V2X
V2V
V2R
MDP
DDQN
DRL
QoS

ABSTRACT

Vehicle-to-everything (V2X) communication via cellular networks is a promising technique for 5G and beyond networks. The cars interact directly with one another, as well as with the infrastructure and various vehicles on the road, in this mode. It enables the interchange of time-sensitive and safety-critical data. Despite these benefits, unstable vehicle-to-vehicle (V2V) communications, insufficient channel status information, high transmission overhead, and the considerable communication cost of centralized resource allocation systems all pose challenges for defense applications. To address these difficulties, this study proposes a combined mode selection and resource allocation system based on distributed deep reinforcement learning (DRL) to optimize the overall network sum rate while maintaining the reliability and latency requirements of V2V pairs and the data rate of V2R connections. Because the optimization issue is non-convex and NP-hard, it cannot be solved directly. To tackle this problem, the defined problem is first translated into machine learning form using the Markov decision process (MDP) to construct the reward function and decide whether agent would conduct the action. Following that, the distributed coordinated duelling deep Q-network (DDQN) method based on prioritized sampling is employed to improve mode selection and resource allocation. This approach learns the action-value distribution by estimating both the state-value and action advantage functions using duelling deep networks. The results of the simulation show that the suggested scheme outperforms state-of-the-art decentralized systems in terms of sum rate and QoS satisfaction probability.

1. Introduction

Vehicle-to-everything (V2X) transmission is a method of wireless communication between automobiles and road infrastructure. According to [1], it is one of the fundamental technologies that increases transport infrastructure, traffic efficiency, entertainment services, and travel pleasure. Using a range of on-board sensors, essential information about the environment around the cars may be detected, communicated, and integrated with other adjacent vehicles. According to [2], the two primary forms of V2X communication are vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V). Vehicles connect with the base station (BS) or roadside equipment in V2I mode (RSU). This mode is used to improve the overall network's QoS and has been proven to be excellent for increasing entertainment services due to its high bandwidth. In contrast, in V2V mode, the vehicles communicate directly with or without the assistance of the RSU. It often requires modest throughput while maintaining rigorous latency and reliability constraints. It is employed in real-time traffic control and vehicle interchange of road information, according to [3].

There are presently two prominent V2V communication standards: Cellular-V2X (C-V2X) and Dedicated Short-Range Communications (DSRC). C-V2X outperforms DSRC in aspects of big coverage and good interference robustness, according to [4]. C-V2X has attracted the attention of both academics and industry in this regard [5,6]. Furthermore, due to vehicle movement, vehicular networks differ significantly from standard cellular networks in terms of dynamic channel conditions and network topologies. According to [7], these characteristics make developing effective resource allocation algorithms for V2V communication problematic. In general, centralized systems are employed to tackle optimization issues, but they have high transmission overhead and insufficient channel state information owing to vehicle mobility. Furthermore, these issues grow exponentially with network size, limiting their utility to large-scale networks. To address these

[*] Corresponding author.
E-mail addresses: shalini.rs.uiet@mdurohtak.ac.in (S. Yadav), rahulrishi@mdu.ac.in (R. Rishi).

[1] All authors are contributed equally in this paper.

https://doi.org/10.1016/j.comcom.2024.04.015
Received 27 October 2022; Received in revised form 3 June 2023; Accepted 9 April 2024
Available online 12 April 2024

challenges, model-free reinforcement learning (RL) approaches are being investigated [8].

RL is a subset of machine learning that is used to guide policy and promote decision making. It is concerned with the incentives and policies used in making decisions in order to get the intended result. Furthermore, each agent in RL selects which action to take is ideal based on the learnt policy. Despite these advantages, traditional RL algorithms have a sluggish convergence rate and perform poorly when dealing with problems with a large state or action space. Deep reinforcement learning (DRL) is being researched as a possible solution to the identified problem, according to [9].

DRL is a cutting-edge technique that combines deep learning with reinforcement learning. DRL trains the learning process using Deep Neural Networks (DNNs), which leads to faster learning and greater performance than RL algorithms. DRL is capable of resolving complicated network issues. This allows the entities to watch each other and choose the optimum approach by making observations locally and exchanging little or no information. In this way, the DRL technique decreases communication overhead and delay while increasing dependability [9].

1.1. Related work

This section analysed the literature on centralized and decentralized V2X communication systems. The authors of [10] investigated the relationship between rapidly shifting channels and channel uncertainty due by delayed CSI reporting to the BS. Using a hybrid channel allocation and power control method, the system throughput was maximized while still fulfilling the latency and reliability criteria of each V2V connection. The worst-case delay of V2V transmissions is reduced by adopting data from high traffic and a pair resource allocation approach proposed in [11]. The transmit power was then changed on a very tiny time scale for each V2V link in line with CSI. [12] used the spatio-temporal traffic pattern to create a centralized power control technique. This approach not only solves the latency and reliability requirements of V2V services, but it also significantly reduces the cost of CSI reporting to the BS. Lyapunov optimization was used to build an energy-efficient resource allocation solution in [13], subject to the dependability and waiting time constraints for each vehicle. [14] employs graph theory to boost system performance. All of the existing systems discussed above rely on V2V connections to supply local data, and when additional cars are added to the route, the signalling overhead increases significantly. Non-linear constraint resource distribution problems, on the other hand, have been characterized as combinatorial optimization problems. These issues are difficult to address using typical optimization techniques. As a result, the researchers investigated decentralized solutions to address these concerns.

Decentralized approaches for cellular V2X communications have been developed to alleviate the burden of gathering global CSI and processing complexity in the above centralized systems. So every V2V transmitter acts as an independent agent that makes these decisions based on their local observations in [15], which creates a DRL-based decentralized resource allocation system for V2V communications. According to [16], who address mixed centralized/distributed V2X connectivity, the topic of power control and resource allocation phase sifters is investigated when varied network load conditions are present. Two strategies are provided for low and high network load scenarios to improve vehicle QoS in regard to packet prioritized and communication connection quality.

Only V2V transmission is employed in the described publications for the transmission of safety-critical messages between vehicles. When the blocking effect is considered, however, the V2V network loses dependence, limiting the usefulness of Data transmission [17]. To tackle this issue, a V2I-based queuing solution could be used. Its endurance performance could be improved at the expense of increased relay delay and lower spectrum utilization, according to [18]. To meet QoS constraints and maximize spectrum use, transmission mode choice and resource allocation for wireless V2X transmissions should be changed concurrently. Further binary channel allocation variables, on either hand, render the joint optimization issue unsolvable by the existing optimization approaches.

Another of the most successful machine learning methods, reinforcement learning (RL), has recently been utilized to enhance improved decision — making in wireless networks. Wu et al. [19] constructed a Q-learning oriented route selection strategy and investigated multi-hop V2I connection to obtain high throughput while maintaining low latency. [20] presents a distributed strategy of phase shifters and RB allocation for possible device-to-device (D2D) couplings in a D2D equipped cloud wireless connectivity network, in which D2D couples updated their strategies utilizing the RL process. Yan et al. [21] propose a Q-learning strategic entry mode selection technique and a convex minimization based spectrum sharing strategy to manage network transmission reliability and front-haul efficiency in fog virtualization automobile networks.

However, various sensing elements and precise channel gains give a large uninterrupted state space, rendering Q-learning ineffective. [22, 23] provided as motivation for DRL and are willing to address the above difficulties. The uninterrupted state may be received as a real access by the deep neural network (DNN), a description of the Q-table in DRL. Atallah et al. [24] use the DRL framework to determine the appropriate transmission mode decision model for systems of rechargeable cars. Because of the tremendously high mobility and time-varying frequency band states in intelligent radio-based car networks, [25] developed a DRL-based optimum data transmission scheduling strategy to minimize transmission costs while still keeping data QoS standards. Zhang et al. [26] developed a DRL-based optimum task offloading solution for compute dumping in routing protocol with changing conditions of many edge nodes and diverse vehicle discharging modes.

A rational design is frequently utilized to learn the DRL algorithms in the work discussed above. In reality, vehicles are constantly getting classification model, resulting in greater internet use and less anonymity. Furthermore, time-varying speedy disappearing channels in V2X transmission are never recognized due to their high characteristics. To support automated driving decision-making, a decentralized DRL architecture is necessary. Finally, the DRL model's substantial performance is hampered by a lack of localized learning algorithm for each vehicle [27]. The use of DRL for joint mode selection and resource allocation has shown great potential, but its application to cellular V2X communication is not without challenges. The first of these challenges is the fact that time-varying fast fading channels are always unknown at vehicles due to their high dynamics, which contradicts the assumptions made in [28,29]. Secondly, to enable vehicles to make autonomous decisions, a decentralized DRL framework is essential. Additionally, the limited local training data available on each vehicle poses a significant obstacle to the robust learning of the DRL model. Finally, improper federated clusters can significantly degrade the performance of federated learning. Thus, addressing these challenges is crucial to the successful deployment of DRL in cellular V2X communication.

In existing schemes from [19–27], attempts to learn joint action in a centralized manner were found to be impractical in large-scale wireless networks. On the other hand, independent learning, while a distributed scalable algorithm, was found to have poor performance. In order to overcome these limitations, a priority sampling distributed coordinated multi-agent DDQN framework has been developed. The framework allows each cell to coordinate with a few neighbour cells, provided they have overlapped areas. If they do not interact, the cells can act independently with respect to one another. The proposed framework optimizes device association, spectrum allocation, and power allocation in a heterogeneous network. In this manner, both the learning convergence efficiency and performance enhancement can be jointly optimized.

1.2. Contributions

To address the challenges posed by varying QoS parameters and unreliable V2V interconnections, we provide in this paper a randomized mode selecting and revenue management approach for cellular V2X transmissions based on DRL. The following are the key contributions of this paper:.

- V2V pairs use the V2I-based relaying mode to reduce the negative consequences of insecure V2V connections. According to the actual network parameters, each V2V pair decides whether to use the V2V or V2I mode. In order to improve the overall network throughput while maintaining the delay and security needs of V2I and V2V links, a combined issue of transmission method selection and resource distribution for wireless V2X communications is described.
- We present a DRL-based distributed solution and represent the issue as a Markov decision process (MDP). Each V2V pair, in addition, acts as a DRL agent, making adaptive decisions based on nearby measurements such as interruption levels, large-scale channel amplification, and traffic. To ensure that the reliability requirement is met, the reward function employs an average availability threshold.
- A PS-DC-DDQN approach is created to learn the optimal resource provisioning provide that to the network's high dynamical and diversity, as well as its large state and action spaces. To do this, the DQN is modified by integrating competitive deep neural networks with a distributed coordinated learning technique. After segregating the state-value standard deviation and the initiative regression line using two main types of hidden layers, the DDQN approach includes the action advantage product to create the action values. This dramatically accelerates convergence and improves learning efficacy.
- Finally, a multi-agent decentralized synchronized DDQN framework is created. In the case of no interaction, this structure allows each V2V pair to function independently of all the other V2V sets. Furthermore, V2V pairs interact with surrounding cells if their areas overlap. The approach in this scenario co-optimizes throughput improvement and learning converging efficiency.
- The simulation results show that the new DRL algorithm improves the known distributed algorithms and outperforms the controlled method. Furthermore, the proposed PS-DC-DDQN techniques provide quick convergence for V2V pairings.

1.3. Organization

The article is arranged as follows. The system model and problem statements are provided in Section second. The principles of DRL and the randomized algorithm based on DRL are described in Section third. In addition, the PS-DC-DDQN method is recommended for V2V link RB and power allocation. The fourth section offers simulation results and analysis. Finally, Section fifth brings the paper to a close.

2. System model and problem formulation

Consider a vehicular communication model underlaying heterogeneous networks as shown in Fig. 1. The model is composed of one BS, $\mathcal{R} = \{1, 2, \dots, r, \dots, R\}$ road side units (RSUs), and $\mathcal{V} = \{1, 2, \dots, v, \dots, V\}$ electric vehicular user equipments (EVUEs). Let us assume that the set of $\mathcal{V}$ EVUEs is classified into two modes, i.e., V2R and V2V mode as shown in Fig. 2. The V2V mode is further divided into two parts: Direct Mode and Hop Mode. The set of V2R and V2V mode are represented as $\mathcal{U} = \{1, 2, \dots, u, \dots, U\}$ and $\mathcal{W} = \{1, 2, \dots, w, \dots, W\}$, respectively. Let B is the total bandwidth that is divided into $\mathcal{S} = \{1, 2, \dots, s, \dots, S\}$ sub-channels (SCs). Therefore, the bandwidth allocated to each SC is denoted as $\Theta = \frac{B}{S}$. It is assumed that each V2R uses a single SC for uplink transmission and there are $S \geq N$ unused SCs. It is presumed that the number of SCs S is more than the number of N V2R.

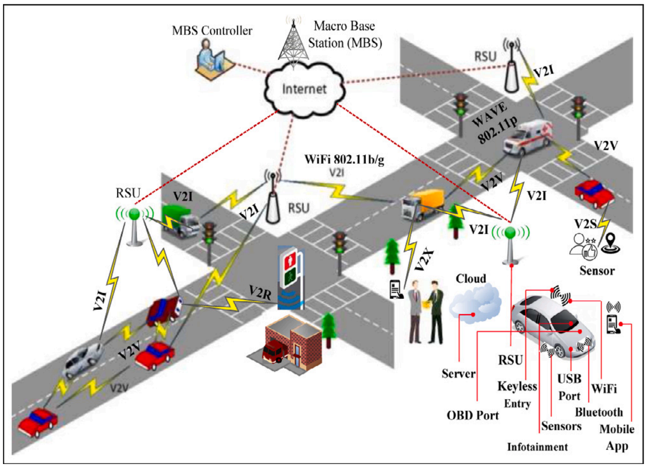

Fig. 1. System model.

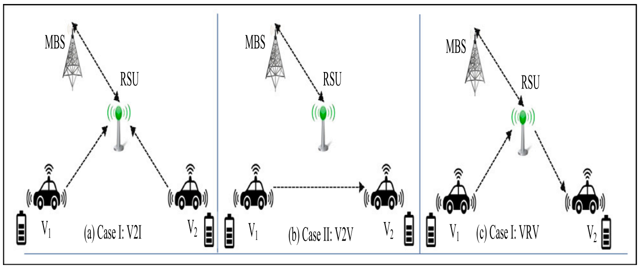

Fig. 2. Mode selection data transmission model: (a) V2R (b) V2V (c) VRV.

Each V2V pair reuse one SC with V2R and the remaining unused SCs in order to maximize the use of available spectrum. Also, it is consider that the multiple V2V pairs reuse the same SC. Moreover, we assume that only large-scale channel gain, including path loss and shadowing fading, is known at the BS and vehicles due to the high mobility of VUEs. Also, the channel used follow either line of sight or non-line of sight depending on the obstruction of neighbouring vehicles and infrastructure. It is also assumed that each vehicle is also enabled with global positioning system (GPS) and PC5 interface. For V2V and V2R communications, the dedicated short range communication protocol is used for data sharing, and to dramatically reduce the number of fatal roadway accidents by providing early warnings.

2.1. Mode selection and data transmission model

In this section, the signal-to-interference noise (SINR) is estimated for the two modes, i.e., V2R and V2V.

2.1.1. V2R mode

In this mode, the uplink data is transmitted from EVUE to RSU. The signal received at the r RSU from the u EVUE over the s SC is estimated as follows [30,31]:

$$Y^{s}_{u,r} = \underbrace{\sqrt{P^{s}_{u}} G^{s}_{u,r} X^{s}_{u,r}}_{\text{Desired Data Signal from } u \text{ to } r}$$

$$+ \underbrace{\sum_{w=1}^{W} \sum_{s=1}^{S} \alpha^{s}_{u,w} \sqrt{P^{s}_{w}} G^{s}_{w,r} X^{s}_{w,r}}_{\text{Interference Signal from } w \text{ to } r} + \underbrace{\xi_{0},}_{\text{AWGN}} \tag{1}$$

where P^{s}_{u} = transmitted power of the u EVUE, P^{s}_{w} = transmitted power of the w EVUE, $X^{s}_{u,r}$ = transmitted signal from u EVUE to r RSU, $X^{s}_{w,r}$

= interference signal from w EVUE to r RSU, $G^s_{u,r}$ = channel gain from the u EVUE to r RSU, $G^s_{w,r}$ = channel gain from the w EVUE to r RSU, and $\xi_0 \in \{0,1\}$ is additive white Gaussian noise (AWGN) with mean = 0 and variance ($\sigma^2 = 1$), $\alpha^s_{u,r}$ = SC allocation for V2R mode. The value of $\alpha^s_{u,w}$ and $\beta^s_{w,r}$ are defined as follows:

$$\alpha^s_{u,w} = \begin{cases} 1, & (w \text{ reuse the SC allocated to } u), \\ 0, & \text{Otherwise.} \end{cases} \tag{2}$$

Now, according to (1) the signal-to-interference noise (SINR) is calculated as follows:

$$\Gamma^s_{u,r} = \left[\frac{P^s_u |G^s_{u,r}|^2}{\sum_{w \neq u}^{W} \sum_{s=1}^{S} \alpha^s_{u,w} P^s_w |G^s_{w,r}|^2 + \sigma^2} \right]. \tag{3}$$

2.1.2. V2V mode

This mode consists of two parts: (a) Direct Mode (b) Hop Mode.

(a) Direct Mode: In this mode, the w_1 EVUE transmit data directly to the w_2 EVUE. The signal received at the w_2 EVUE from the w_1 EVUE is given as follows [30]:

$$Y^{s,D}_{w_1,w_2} = \underbrace{\delta^s_w \sqrt{P^s_{w_1}} G^s_{w_1,w_2} X^s_{w_1,w_2}}_{\text{Desired Data Signal from } w_1 \text{ to } w_2}$$

$$+ \underbrace{\sum_{w' \neq w, w=1}^{W} \sum_{s=1}^{S} \beta^s_{w',w_2} \sqrt{P^s_{w'}} G^s_{w',w_2}, X^s_{w',w_2}}_{\text{Interference Signal from } w' \text{ to } w_2} + \xi_0, \tag{4}$$

where $X^s_{w_1,w_2}$ = transmitted signal from w_1 to w_2, X^s_{w',w_2} = transmitted signal from w' vehicle to w_2, $G^s_{w_1,w_2}$ = channel gain from w_1 to w_2, and G^s_{w',w_2} = channel gain from w' to w_2, α^s_{w',w_2} represents the SC indicator with V2V mode, and δ^s_w = mode selection indicator. The values of β and δ are defined as follows [32]:

$$\beta^s_{w',w} = \begin{cases} 1, & (\text{if } w' \text{ reuse the SC allocated to } w), \\ 0, & \text{Otherwise.} \end{cases} \tag{5}$$

$$\text{and } \delta^s_w = \begin{cases} 1, & \text{Direct Mode,} \\ 0, & \text{Hop Mode.} \end{cases} \tag{6}$$

Now, according to (4) the SINR from w_1 to w_2 is calculated as follows.

$$\Gamma^{s,D}_{w_1,w_2} =$$

$$\left[\frac{\delta^s_w P^s_{w_1} |G^s_{w_1,w_2}|^2}{\sum_{u=1}^{U} \alpha^s_{u,w} P^s_u |G^s_{u,r}|^2 + \sum_{w' \neq w_1}^{W} \beta^s_{w',w} P^s_{w'} |G^s_{w',w_2}|^2 + \sigma^2} \right]. \tag{7}$$

(b) Hop Mode: In this mode, firstly, the data transmitted in an uplink mode from w_1 EVUE to r RSU, and then the data is transmitted in downlink mode from r RSU to the w_2 EVUE. Note that only unused SCs can be allocated to V2V pairs in hop mode, and each unused SC can be allocated to at most one V2V pair in hop mode

- **Case-I: Data Transmission from w_1 to r**: The uplink data transmission from w_1 EVUE to r RSU is estimated as follows [33]:

$$Y^{s,H}_{w_1,r} = \underbrace{\delta^s_w \sqrt{P^s_{w_1}} G^s_{w_1,r} X^s_{w_1,r}}_{\text{Desired Data Signal from } w_1 \text{ to } r}$$

$$+ \underbrace{\sum_{w' \neq w_1, w=1}^{W} \sum_{s=1}^{S} \beta^s_{w',r} \sqrt{P^s_{w'}} G^s_{w',r} \; X^s_{w',r}}_{\text{Interference signal from } w' \text{ to } r} + \xi_0, \tag{8}$$

Now, according to (8), the SINR in from w_1 to r is calculated as follows.

$$\Gamma^{s,H}_{w_1,r} = \left[\frac{\delta^s_w P^s_{w_1} |G^s_{w_1,r}|^2}{\sum_{w' \neq w_1}^{W} \sum_{s=1}^{S} \beta^s_{w',r} P^s_{w'} |G^s_{w',r}|^2 + \sigma^2} \right]. \tag{9}$$

- **Case-II: Data Transmission from r to w_2**: The downlink data transmission from r RSU to w_2 EVUE is estimated as follows [33].

$$Y^{s,H}_{r,w_2} = \underbrace{\delta^s_w \sqrt{P^s_r} G^s_{r,w_2} X^s_{r,w_2}}_{\text{Desired Data Signal from } r \text{ to } w_2}$$

$$+ \underbrace{\sum_{r' \neq r}^{R} \sum_{s=1}^{S} \kappa^s_{r',w_2} \sqrt{P^s_{r'}} G^s_{r',w_2} \; X^s_{r',w_2}}_{\text{Interference Signal from } r' \text{ to } w_2} + \xi_0, \tag{10}$$

where x^s_{r,w_2} = transmitted signal from r to w_2, x^k_{r',w_2} = interference signal from r' to w_2, G^s_{r,w_2} = channel gain from r to w_2, and G^s_{r',w_2} = channel gain from r' to w_2, and κ^s_{r',w_2} represents the SC indicator whose value is defined as follows [34]:

$$\kappa^s_{r',w_2} = \begin{cases} 1, & (r' \text{ reuse the SC allocated to } r), \\ 0, & \text{Otherwise.} \end{cases} \tag{11}$$

Now, according to (10), the SINR from r to v_2 is calculated as follows:

$$\Gamma^{s,H}_{r,w_2} = \left[\frac{\delta^s P^s_r |G^s_{r,w_2}|^2}{\kappa^s_{r',w_2} P^s_{r'} |G^s_{r',w_2}|^2 + \xi_0} \right]. \tag{12}$$

Now, according to (9) and (12), the SINR from w_1 to w_2 using r RSU is estimated as follows:

$$\Gamma^{s,H}_{w_1,w_2} = \min \left[\Gamma^{s,H}_{w_1,r}, \; \Gamma^{s,H}_{r,w_2} \right]. \tag{13}$$

2.2. Data rate and sum rate estimation of mode selection and data transmission model

2.2.1. V2R mode

According to (3), the data rate of V2R mode is given as follows:

$$\mathfrak{D}^s_{u,r} = \Theta \log_2 \left(1 + \Gamma^s_{u,r} \right), \tag{14}$$

where Θ represents the bandwidth allocated to each SC.

2.2.2. V2V mode

The data rates of V2V modes using (7) and (13) are given as follows:

(a) Direct Mode:

$$\mathfrak{D}^{s,D}_{w_1,w_2} = \frac{1}{2} \Theta \log_2 \left(1 + \Gamma^{s,D}_{w_1,w_2} \right). \tag{15}$$

(b) Hop Mode:

$$\mathfrak{D}^{s,H}_{w_1,w_2} = \frac{1}{2} \Theta \log_2 \left(1 + \Gamma^{s,H}_{w_1,w_2} \right). \tag{16}$$

The total data rate of the V2V mode is given as follows:

$$\mathfrak{D}^s_{w_1,w_2} = \left(\mathfrak{D}^{s,D}_{w_1,w_2} + \mathfrak{D}^{s,H}_{w_1,w_2} \right). \tag{17}$$

The total data rate of the overall vehicular network over the s SC using the Eqs. (14), (15), and (16) is estimated as follows:

$$\mathfrak{D}^s_T = \mathfrak{D}^s_{u,r} + \left(\mathfrak{D}^{s,D}_{w_1,w_2} + \mathfrak{D}^{s,H}_{w_1,w_2} \right). \tag{18}$$

Now, the sum rate of the overall vehicular network over the kth SC using Eq. (18) is estimated as follows:

$$\mathfrak{R}^s_O = \sum_{r=1}^{R} \sum_{w=1}^{W} \sum_{s=1}^{S} \mathfrak{D}^s_T \tag{19}$$

2.3. Reliability and latency requirements of mode selection and data transmission model

In vehicular communication networks, there are numerous types of applications which works as per the demand of their data rate requirements. Therefore, the QoS requirement of each mode is estimated with respect to their applications.

2.3.1. V2R mode

V2R mode is generally used to perform bandwidth-intensive entertainment or traffic applications. Therefore, in V2R mode, QoS requirement is equal to the minimum data rate requirements to ensure a comfortable experience which is defined as follows:

$$\mathfrak{D}^{s}_{u,r} \geq \mathfrak{D}^{s,\min}_{u,r} \tag{20}$$

2.3.2. V2V mode

V2V mode is used to broadcast safety-critical signals such as alerts about cooperative operations in real time. The failure of this mode would jeopardize road safety. Therefore, reliability and latency is used in this mode to fulfil the QoS requirements.

- **Reliability Requirement**: In V2V mode, outage probability is used to calculate the reliability requirement among V2V pairs. The outage probability of the V2V pair is calculated as follows:

  $$\mathbb{P}\left(\Gamma^{s}_{w_1,w_2} \leq \Gamma_O\right) \leq \mathbb{P}_O, \tag{21}$$

  $$\text{where} \quad \Gamma^{s}_{w_1,w_2} = (1-\delta^{s}_{w})\sum_{s=1}^{S}\Gamma^{s,D}_{w_1,w_2} + \delta^{s}_{w}\sum_{s=1}^{S}\Gamma^{s,H}_{w_1,w_2} \tag{22}$$

  Now, the reliability constraints formulated in (22) can be updated with respect to Rayleigh fading as follows:

  $$\Gamma^{s}_{w_1,w_2} \leq \Gamma^{s}_{EOT} = \frac{\Gamma_O}{\ln\left[\frac{1}{1-\mathbb{P}_O}\right]} \tag{23}$$

- **Latency Requirements** The safety-related packets are produced at a constant rate of Ω bps, and it is assumed that the transmitter of a V2V link has a finite-length buffer. According to V2V mode, the following data rate for the w_1 to w_2 V2V pair is possible [35]:

  $$\mathfrak{D}^{s}_{w_1.w_2} = (1-\delta^{k}_{s})\mathfrak{D}^{s,D}_{w_1.w_2} + \delta^{k}_{s}\mathfrak{D}^{s,H}_{w_1.w_2} \tag{24}$$

  The data rate formulated in (24) can vary depending on the slot, and there may be a discrepancy between instantaneous throughput and packet generation. As a result, the queues congestion increases at the transmitters of the V2V pairs due to which queueing delays increases.

  At the beginning of slot t, the queue length of the wth V2V pair is estimated as follows [36]:

  $$\lambda^{s}_{w}(t) = \max\{0, \lambda^{s}_{w}(t-1) + \tau\Omega - \tau\mathfrak{D}^{s}_{w}(t)\}, \tag{25}$$

  where $\lambda^{s}_{w}(t-1)$ = transmitter queue length at $(t-1)$ slot, $\tau\Omega$ = number of bits arrived at the queue per slot, and $\tau\mathfrak{D}^{s}_{w}(t)$ = number of bits sent to the corresponding receiver at $(t-1)$ slot.

  The average queuing latency is linear to the queue length, according to Little's Law [13]. Let $\mathbb{L}^{s,\max}_{w}(t)$ stand for the maximum tolerable transmission delay for V2V packets, and the number of packets encounter delays greater than $\mathbb{L}^{s,\max}_{w}(t)$ is written as follows:

  $$\lambda^{s,\max}_{w}(t) = \Omega(\mathbb{L}^{s,\max}_{w}(t)) \tag{26}$$

  Now, to guarantee a steady-state queue length with a tolerable probability threshold, the delay constraint of the wth V2V pair can be expressed as follows:

  $$\mathbb{P}\left(\mathbb{L}^{s}_{w}(t) \geq \mathbb{L}^{s,\max}_{w}(t)\right) = \mathbb{P}\left(\lambda^{s}_{w}(t) \geq \mathbb{Q}^{s,\max}_{w}(t)\right) \leq \mathbb{P}_O, \tag{27}$$

  Now, by using the Markov's inequality, i.e., $\mathbb{P}(\mathbb{X} > \Phi) \leq \mathbb{E}[\mathbb{X}]/\Phi$, the Eq. (27) can be reformulated as follows:

  $$\bar{\lambda}^{s,\max}_{w}(t) = \lim_{t\to\infty}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\lambda^{s}_{w}(t)] \leq \mathbb{P}_O\lambda^{s,\max}_{w}(t) \tag{28}$$

  Eq. (28) implies that if the time-averaged queue length of the transmitter is upper bounded, then the V2V packets can be delivered on time.

2.4. Problem formulation

The main objective of the paper is to maximize the sum rate of the overall network while ensuring the data rate, reliability and latency requirements of EVUE pairs. Mathematically, the formulated problem can be represented as follows.

$$\begin{aligned}
\mathbb{P}.\mathbb{F}. &= \max_{\delta,\alpha,P}\mathfrak{R}^{s}_{O}, \qquad (29)\\
s.t.\ C_1 &: \mathfrak{D}^{s}_{u,r} \geq \mathfrak{D}^{s,\min}_{u,r}, \mathfrak{D}^{s,D}_{w_1,w_2} \geq \mathfrak{D}^{s,\min}_{w_1,w_2}, \mathfrak{D}^{s,H}_{w_1,w_2} \geq \mathfrak{D}^{s,\min}_{w_1,w_2}\\
C_2 &: \Gamma^{s}_{w_1,w_2} \leq \Gamma^{s}_{EOT}\\
C_3 &: \bar{\lambda}^{s,\max}_{w}(t) \leq \mathbb{P}_O\mathbb{Q}^{s,\max}_{w}(t)\\
C_4 &: \lambda^{s}_{w}(t) \leq \mathbb{Q}^{s,\max}_{w}(t)\\
C_5 &: \delta^{s}_{w} \in \{0,1\}, \forall w \in W\\
C_6 &: \alpha^{s}_{u,w}, \beta^{s}_{w',w}, \kappa^{s}_{r',w_2} \in \{0,1\}, \forall s \in S\\
C_7 &: \alpha^{s}_{u,w}, \beta^{s}_{w',w} < 1, \forall s \in S,\\
C_8 &: P^{s}_{u} \leq P^{s,\max}_{u}, P^{s}_{w} \leq P^{s,\max}_{w}, \forall u \in U, \forall w \in W
\end{aligned}$$

The constraints described in Eq. (29) are defined as follows. Constraints C_1 represents the data rate requirements of EVUEs during V2R and V2V mode. The latency and reliability specifications for EVUE pairs in V2V mode are shown in the constraints C_2–C_4. Each vehicle can choose either the V2I mode or the V2V mode, according to C_5. Constraint C_6 demonstrates that each V2V pair can be assigned to a single SC and that SC can be shared by several V2V pairs. Constraint C_7 states that each SC may be assigned to a maximum of one V2V pair. Constraints C_8 ensures that no EVUE pair's transmit power exceed its maximum amount.

The problem formulated in (28) is in a mixed integer non-linear programming form and cannot be solved directly due to the following reasons: (i) the mode selection and SC assignment indicator are in combinatorial form because they are binary in nature. (ii) the transmit power, the optimization object and constraints C_1–C_3 are in non-convex form due to presence of interference and continuous variables. So, to solve these types of problems typically centralized solutions are studied but in practical wireless networks, the QoS requirements, the channel conditions and services of devices change dynamically. Hence, to overcome these problems, we used the model-free RL. Following, we convert the formulated optimization problem into a multi-agent RL (MARL) problem.

3. Proposed solution

3.1. Markov decision process

In this section, first of all, the optimization problem formulated in (29) is modelled into machine learning form using MDP concept. The MDP consists of five tuples $(\mathbb{J}, \mathbb{S}, \mathbb{A}, \mathbb{P}, \mathbb{R})$, where $\mathbb{J}$ is the agent, $\mathbb{S}$ represents the state space, $\mathbb{A}$ presents the action space, $\mathbb{P}$ represents the state transition probability, and $\mathbb{R}$ is the reward function. The detailed description of each of the aforementioned components are described as follows:

- **Agent** ($\mathbb{J}$): V2V pair
- **State Space**: The state space s of the w EVUE pair over the s SC at the t time frame is defined as follows:

$$s_w^t = \{\mathbb{I}_w^{s,t-1}, \mathbb{I}_r^{s,t-1}, \mathbb{N}_w^{s,t-1}, G_{w,r}^{s,t}, G_{w_1,w_2}^{s,t}, \lambda_w^{s,t}\} \tag{30}$$

where I_w^{t-1} is the interference received at the w EVUE pair over each SC at the $(t-1)$ time frame; I_r^{t-1} is the interference received at the r RSU over each SC at the $(t-1)$ time frame; $\mathbb{N}_w^{s,t-1} = \{\mathbb{N}_1^{s,t-1}, \mathbb{N}_2^{s,t-1}, \dots, \mathbb{N}_W^{s,t-1}\}$ is the number of neighbouring V2V pairs over each SC at the $(t-1)$ time frame; $G_{w,r}^{s,t}$ denotes the channel gain from w EVUE to r RSU; $G_{w_1,w_2}^{s,t}$ represents the channel gain from w_1 to w_2 EVUE in the V2V mode; and $\lambda_w^{s,t}$ is the current state queue length at the transmitter of the wth EVUE pair.
- **Action Space**: The action space of wth EVUE pair depends upon the three factors: mode selection, SC allocation, and power control, which is given as $j_w^{s,t} = \{\delta, \alpha, P\}$, where $\delta \in \{0,1\}$, $\alpha \in \{0,1\}$, and $P = \{0, \frac{1}{N_P} P_w^{s,\max}, \frac{2}{N_P} P_w^{s,\max}, \dots, P_w^{s,\max}\}$. The transmit power of EVUEs is considered to have $N_P + 1$ levels, and we used a discrete power control technique [9]. Also, it is assumed that if the transmit power of EVUE is 0, then no packet is transferred over a EVUE pair. As a result, the size of $\mathbb{A}$ is set to be $S \times (N_P + 1)$.
- **State Transition Probability**: $\mathbb{P}(s'|s,a)$ represents the state transition probability from the present state s_w^t to a new state $s_w'^t$ using an action a_w^t.
- **Reward Function**: To maximize the sum rate of EVUEs lies under V2R mode while ensuring the reliability and latency requirements of V2V pairs, the reward function is defined as follows:

$$\mathbb{R}_w^t = \varphi_1 \mathfrak{R}_O^s + \sum_{u=1}^{U} \varphi_2 \Psi\left(\mathfrak{D}_{u,r}^s - \mathfrak{D}_{u,r}^{s,\min}\right)$$
$$+ \sum_{w=1}^{W} \varphi_3 \Psi\left(\mathfrak{D}_{w,r}^s - \mathfrak{D}_{w,r}^{s,\min}\right) + \sum_{w=1}^{W} \varphi_4 \Psi\left(\lambda_w^s - \lambda_w^{s,\max}\right)$$
$$+ \sum_{w=1}^{W} \varphi_5 \Psi\left(\Gamma_{w_1,w_2}^s - \Gamma_{EOT}^s\right) \tag{31}$$

Here, $\Psi(x)$ levies a penalty if any part of the reward function is not satisfied as promised, described as follows:

$$\Psi(x) = \begin{cases} M, & x \geq 0, \\ x, & x \leq 0, \end{cases} \tag{32}$$

where $M > 0$ is a positive integer used to represents the revenue, and $\varphi_1, \varphi_2, \varphi_3, \text{and} \varphi_4$ denotes the weights and its function is to balance the revenue and penalty.

At the start of time frame t, each EVUE pair observes their state s_w^t autonomously and then executes mode selection, SC allocation and power control based on the established action value function $\mathbb{Q}(s_w^{s,t}, a_w^{s,t}, \phi_w^{s,t})$. Now, according to RL, the value of expected reward is equals to the action value which is defined as follows:

$$\mathbb{Q}(s,a,\phi) = \mathbb{E}\left[\sum_{t'=t}^{T} \lambda^{t'-t} \mathbb{R}_w^{t'} | s = \mathbb{S}, a = \mathbb{A}, \phi\right] \tag{33}$$

where T is the terminal step of each epoch, and $0 < \lambda < 1$ is the discount factor. Now, on using the Bellman function, Eq. (33) is rewritten as follows:

$$\mathbb{Q}(s,a,\phi) = \mathbb{E}\Bigg[\mathbb{R}(s,a,\phi)$$
$$+\theta \sum_{s' in \mathbb{S}} \mathbb{P}(s'|s,a) \sum_{a' in \mathbb{A}} \pi(s',a',\phi)\mathbb{Q}(s,a,\phi)\Bigg] \tag{34}$$

The optimal policy π^* must be learned using Q-learning in order to produce the optimized Q-function, which is given as follows:

$$\mathbb{Q}^*(s,a,\phi) = \mathbb{R}(s,a,\phi) + \theta \sum_{s' in \mathbb{S}} \mathbb{P}(s'|s,a) \max_{a' in \mathbb{A}} \mathbb{Q}^*(s,a,\phi) \tag{35}$$

Now, we used the deep Q network (DQN) to calculate the Q-function, i.e., $\mathbb{Q}(s,a) \sim \mathbb{Q}(s,a,\phi)$, where ϕ represents the deep neural network parameter. The DQN is a model-free RL technique in which the agent selects the state and perform the action with respect to the change in environment only. In this, the agent does not need to know the state transition probability. Furthermore, to maintain the overall learning performance, the DQN adjusts its parameters by minimizing the loss function, which is specified as follows:

$$\mathbb{L}(\phi) = \mathbb{E}\left[(y^{\mathrm{DQN}} - \mathbb{Q}(s,a,\phi))^2\right] \tag{36}$$

where $y^{\mathrm{DQN}} = \mathbb{R} + \theta \max_{a' \in \mathbb{A}} \hat{\mathbb{Q}}(s', a', \phi^-)$. Here, ϕ^- is the DNN parameter of the target Q network $\hat{\mathbb{Q}}(s', a', \phi^-)$.

3.2. Priority sampling based distributed coordinated DDQN (PS-DC-DDQN) algorithm

The suggested PS-DC-DDQN approach for allocating subchannels and power is described in this section. Next, the abiding reward of every learning agent in the proposed network will then be maximized using the DDQN-based technique. Finally, the distributed learning procedure among various agents and the implementation of the algorithm are presented.

3.2.1. PS-DC-DDQN framework for smart resource allocation

The DQN algorithm's rate of convergence is nevertheless limited as a result of the overestimation of optimizer that takes place during the learning process, particularly in large-scale wireless networks where the state and action spaces are rather enormous. Furthermore, if the number of states is enormous, the chosen action may not have an impact on many states. Therefore, various decision-making policies may have the same value functions. Additionally, the centralized DRL algorithm may not be appropriate for large-scale heterogeneous networks because to the enormous state and action spaces, which result in impractically long training times and excessively sluggish convergence. To overcome these issues, the DDQN algorithm is integrated with a distributed coordinated learning technique to improve the training convergence rate and efficiency.

Fig. 3 shows the process of training framework. Fig. 3 have two sections: left-hand section (LHS) and right-hand section (RHS). The environment and $J + 1$ agents are in the LHS, while the prioritized experience replay (PER), primary DDQN (P-DDQN), target DDQN (T-DDQN), and environment are in the RHS. All agents in the LHS execute the DDQN algorithm (the RHS) to construct a joint resource allocation and power allocation scheme. Each agent j develops their decision-taking skills according to the noticed state information. However, small portions of decision data such as resource allocation and power allocation will be shared with neighbouring agents in a distributed and coordinated fashion. Each agent J sees its current state s_j^t during the tth time slot and chooses an action a_j^t by optimizing the utility function. After performing the action a_j^t the agent communicates a portion of its decision information with only neighbouring agents. Then agent move into a new state and get a reward. When the gain of all links is known during one slot time, then the key aspect influencing the reward are interferences caused by neighbour devices. Therefore, not all information needs to be shared; only the sub-carrier and power allocation information of those devices (creating interference) is required.

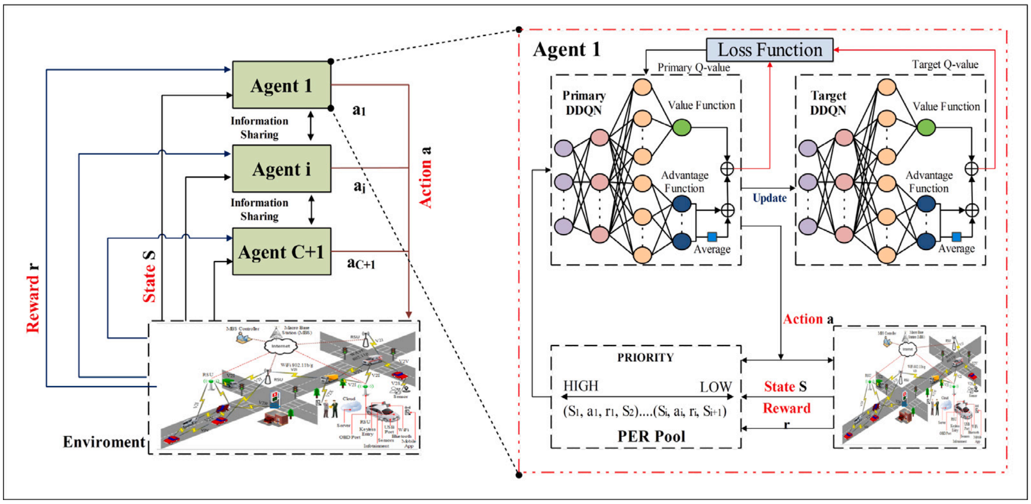

Fig. 3. DDQN Model.

3.2.2. Algorithm based on DDQN

As shown in RHs of Fig. 3, the action-value function is not directly assessed in the DDQN method. The two streams of completely connected layers are used to determine the value and advantage function. The effectiveness of the learning policy in a particular state is decided by a value function. On the other hand, the correlation between one action and other actions is established by the advantage function. The Q- value is then generated in the final layer by combining these functions [37].

The state-value function for a particular state s and action a in DDQN is defined as:

$$\mathbb{Q}_{\pi}(s_j^t, a_j^t) = \widehat{\mathbb{E}}\big(\mathbb{Q}_t | s_j^t, a_j^t, \pi\big). \tag{37}$$

In Eq. (37), $\widehat{\mathbb{E}}$ denotes the expectation operator and π is the known policy.

The value-function for a particular state is given as follows:

$$\mathbb{V}_{\pi}(s_j^t) = \widehat{\mathbb{E}}_{a\sim\pi(a)}\big(\mathbb{Q}(s_j^t, a_j^t)\big). \tag{38}$$

Now, according to Eqs. (37) and (38) the advantage function can be computed as :

$$\mathbb{F}_{\pi}(s_j^t, a_j^t) = \mathbb{Q}_{\pi}(s_j^t, a_j^t) - \mathbb{V}_{\pi}(s_j^t) \tag{39}$$

In Eq. (39), the function $\mathbb{Q}(s_j^t, a_j^t)$ assesses the Q-value according to the action a_j^t and state s_j^t, a, whereas the function $\mathbb{V}(s_j^t)$ is utilized to determine the performance quality in a certain state s_j^t. Defining $\widehat{\mathbb{E}}_{a_j^t\sim\pi(a_j^t)}\big(\mathbb{F}_{\pi}(s_j^t, a_j^t)\big) = 0$ as in [38]. Furthermore, for a known deterministic policy $\hat{a}_t^n = \arg\max_{a_j'^t \in \mathcal{A}} Q(s_j^t, a_j'^t)$, $Q(s_j^t, \hat{a}_j^t) = \mathbb{V}(s_j^t)$ and therefore, $\mathbb{A}(s_j^t, \hat{a}_j^t) = 0$.

In D3QN the $\mathbb{V}_{\pi}(s_j^t)$ return single value for every state. On the other hand, $\mathbb{F}_{\pi}(s_j^t, a_j^t)$ layer return N actions for every action in particular state. The sum of these two functions is used to calculate the Q-value of executing an action in a specific state and can be expressed as follows:

$$\mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu) = \mathbb{V}(s_j^t; \theta, \mu) + \mathbb{F}(s_j^t, a_j^t; \theta, \varsigma), \tag{40}$$

In Eq. (40), θ represents convolution layer parameters, whereas α and μ are the weights for value-function and advantage function, respectively. However, the Eq. (40) exhibits poor performance due to an unidentifiable issue. The reason behind this problem is that if the same constant is added to $\mathbb{V}(s_j^t; \theta, \mu)$ and subtracted from $\mathbb{F}(s_j^t, a_j^t; \theta, \varsigma)$ then the same Q-value is obtained. To tackle this issue, the Q-value is further modified as follows:

$$\mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu) = \mathbb{V}(s_j^t; \theta, \mu)$$

$$+\Big(\mathbb{F}(s_j^t, a_j^t; \theta, \varsigma) - \arg\max_{a_j'^t \in \mathcal{A}} \mathbb{F}(s_j^t, a_j'^t; \theta, \varsigma)\Big) \tag{41}$$

Assume that the value of the advantage function for best action is 0, and in this case, $\mathbb{Q}(s_j^t, \hat{a}_t^n; \theta, \varsigma, \mu) = \mathbb{V}(s_j^t; \theta, \mu)$. Now, optimal action can be computed as follows:

$$\hat{a}_t^n = \arg\max_{a_j'^t \in \mathcal{A}} \mathbb{Q}(s_j^t, a_j'^t; \theta, \varsigma, \mu) = \arg\max_{a_j'^t \in \mathcal{A}} \mathbb{F}(s_j^t, a_j'^t; \theta, \varsigma, \mu). \tag{42}$$

The Q-value can be further modified by taking the difference between the advantage function of each individual action and the average advantage function over all actions. Therefore, the modified Q-value can be expressed as follows:

$$\mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu) = \mathbb{V}(s_j^t; \theta, \mu)$$

$$+\Big(\mathbb{F}(s_j^t, a_j^t; \theta, \varsigma) - \frac{1}{|\mathcal{A}|} \sum_{a_j'^t} \mathbb{F}(s_j^t, a_j'^t; \theta, \varsigma)\Big) \tag{43}$$

However, if DQN evenly selects each transition $e_t^n = \big(s_j^t, a_j^t, r_j^t, s_j^{t+1}\big)$ from the experience replay buffer, then the model may be inefficiently trained and result in divergence rather than convergence. This is due to the fact that various experience samples have variously significant effects on the learning policy. A PER method has been suggested as a result to improve experience efficiency [38]. In the PER method, the concept of priority is introduced. According to this concept, the highest priority samples in the experience replay buffer are given the highest sampling probability. As a result, the sample's priority is determined using the temporal difference error (TDE), which is stated as:

$$\delta(t) = r_j^t + \gamma \max_{a_j'^t \in \mathcal{A}} \hat{\mathbb{Q}}(s_j'^t, a_j'^t; \theta^-, \varsigma^-, \mu^-) - \mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu) \tag{44}$$

The probability of selecting a qth sample from the experience replay buffer is given by:

$$p(q) = \left(\frac{1}{rank(q)}\right)^{\eta_1} \Big/ \sum_{q'} \left(\frac{1}{rank(q')}\right)^{\eta_1}, \tag{45}$$

where $rank(q)$ represents the position of qth sample in experience replay buffer. Exponent η_1 represents the value of regulation for the prioritization process of sampling. If sampling is done uniformly, then $\eta_1 = 0$. The difference between the targeted Q-value and the predicted Q-value is represented by the TDE. In (45), if TDE for selecting qth sample is large then predicted accuracy has considerable amount of room for improvement, hence this sample requires more learning.

It is possible that samples with large values of TDE are more likely to be sampled repeatedly, which increases the frequency of access to certain states and, as a result, introduces bias. We might use dominant-sampling (DS) weights in the ERB to eliminate the bias. DS weights can be expresses as:

$$\mathbb{W}(q) = N_E p(q)^{\eta_2} \; / \; \max_{i'} \mathbb{W}(q') \tag{46}$$

In Eq. (46), N_E represents the count of samples in the ERB and exponent η_2 represents the value of regulation for the sampling rectification process. Now, the loss function is commutated with the help of DS weights. Also, PER is integrated with DDQN, and finally, the loss function of (36) is modified as follows:

$$\mathbb{L}(\theta, \varsigma, \mu) = \frac{1}{D} \sum_{q=1}^{D} \left(\mathbb{W}_q L_q(\theta, \varsigma, \mu) \right), \tag{47}$$

where D is the tiny chunk size of ERB. As a result, the gradient descent approach [7] can be used to estimate the updated value of the parameter θ as follows:

$$\theta \leftarrow \theta + \nu \nabla \left(L_q(\theta, \varsigma, \mu) \right), \tag{48}$$

where β is the parameter θ's learning-rate.

The RHS of Fig. 3 consist of P-DDQN and T-DDQN. The main objective of P-DQNN is to produce Q-values for pairs of state–action. T-DDQN, on the other hand, aims to generate target Q-values. To keep the Q- values around the desired Q-values, the P-DDQN can be updated continuously. After that, the learning agent updates the value of θ^- in T-DDQN architecture with the help of parameters θ in P-DDQN architecture for each training slot.

3.2.3. PS-DC-DDQN algorithm

Centralized techniques are suitable for small-scale wireless networks where agent learns about the global joint directly. However, these techniques are not appropriate for ultra-massive wireless networks. One solution is self-managed learning, in which agents learn through a distributed and scalable approach. In the proposed network architecture, each agent interacts with a small number of its neighbours and functions irrespective of the others.

To achieve the target of maximizing the anticipated reward, a small group of a few nearby agents learn in a distributed cooperative learning fashion. The aggregate anticipated reward for mth small group can be given as:

$$r_m^t = \sum_{m=1}^{J_m} r_{m,j}^t. \tag{49}$$

In (49) r_j^t stands for jth agent's reward function and J_m stands for total agents in the mth small group. The mth group's predicted Q-value of executing an action in a specific state is represented in a coordinated fashion as:

$$\mathbb{Q}(s_j^t, a_j^t) = \mathbb{E}_\pi \left[\sum_{q=0}^{\infty} \sum_{j=1}^{J_m} r_{m,j}^t \right] \tag{50}$$

Since each agent in a small group of devices cooperates with one another, the desired joint action a_m is expressed as:

$$\hat{a}_m^t = \arg\max_{a_m^t \in \mathcal{A}_m} \hat{\mathbb{Q}}(s_m^t, a_m^t). \tag{51}$$

In Eq (51) s_m^t and $\mathcal{A}_m$ denotes the mth group's combined state and action space.

The details of the distributed coordinated PS-DC-DDQN Algorithm is shown in Algorithm 1. In this, each agent keeps track of the environment's current state throughout each learning stage. The environment's current state includes preceding obtained values of SINR, sub-carrier allocation, and details of all associated devices' channel gain. The training model is then trained using the observed state data that is supplied to the DDQN. Now, ϵ greedy policy is executed to select which action is performed by the agent. The ϵ greed policy requires an agent to choose the action with the highest Q value and probability 1-ϵ. On the other hand, a random action is chosen to have a probability ϵ. The chosen action a_j^t consists of sub-carrier allocation as well as power allocation. After completing the chosen action, an agent receives a reward and also records the next state s_j^{t+1} through the environment

Algorithm 1: PS-DC-DDQN Algorithm for Sub-Carrier and Power Allocation .

Input

- Environment: Cellular V2X Communication.
- $\Gamma_{u,r}^{s} \geq \Upsilon_{u,r}^{s,\min}$: Minimum SINR requirement of V2R links.
- $\Gamma_{w_1,w_2}^{s,D} \geq \Gamma_{w_1,w_2}^{s,D,\min}$: Minimum SINR Requirements of V2V links during direct mode.
- $\Gamma_{w_1,w_2}^{s,H} \geq \Gamma_{w_1,w_2}^{s,H,\min}$: Minimum SINR Requirements of V2V links during hop mode.

Initialization:

- P-DDQN with parameters α and μ.
- T-DDQN with parameters α^- and μ^-.
- D = ERB of fixed size and mini-batch.

```
for Each episode = $1,\ldots,\Theta$ do
    Each agent in the network notes its starting state $s_j^t$;
    for Each time-slot $t = 0,1,2,3,\ldots,\mathbb{T}$ do
        Select action $a_j^t$ with state $s_j^t$ by employing $\epsilon$
        greed policy ;
        $a_j^t = \arg\max_{a_j^t \in \mathcal{A}} \mathbb{Q}(s_j^t, a_j^t, \varsigma, \mu)$, when probability is $1-\epsilon$;
        $a_j^t$= randomly $\{a_j^t\}_{a_j^t \in \mathcal{A}}$, when probability is $\epsilon$;
        Execute the action $a_j^t$, and get a reward $r_t^n$ and
        records the next state $s_{t+1}^j$;
        Save $e_j^t = \left(s_j^t, a_j^t, r_j^t, s_j^{t+1}\right)$ in the PER pool;
        for $q = 0,1,2,3,\ldots,D$ do
            Compute TDE value $\delta(q)$ using (44);
            Compute the probability $p(q)$ of selecting
            a $q^{th}$ sample from the ERP using (45);
            Calculate the DS weights $\mathbb{W}(q)$ using (46);
            Modify the transition's priority q
            with $\frac{1}{rank(q)} \leftarrow \delta(q)$:
        end for
        Computes $\mathbb{V}(s_j^t;\theta,\mu)$ and $\mathbb{F}(s_j^t, a_j^t;\theta,\varsigma)$, and
        adds these two values to estimates of
        $\mathbb{Q}(s_j^t, a_j^t;\theta,\varsigma,\mu)$ in (40);
        if $(s_j^{t+1} = s_{final}^{t+1})$ then
            Renew target Q-value as $y^{DQN} = r_j^t$;
        else
            $y^{DQN} = r_j^t + \gamma \max_{a_j^{t+1} \in \mathcal{A}} \hat{\mathbb{Q}}\left(s_j^{t+1}, a_j^{t+1}; \theta^-, \varsigma^-, \mu^-\right)$
        end if
        Modify $\mathbb{L}(\theta,\varsigma,\mu)$ and $\theta$ using (47) and (48);
        Initialize parameter $\theta^- = \theta$;
    end for
    Each agent exchanges small portions of decision
    data such as resource and power allocation with
    neighbouring agents in a distributed and
    coordinated fashion;
end for
Return to the DDQN learning framework;
Output: $\delta, \alpha, P$.
```

Table 1
Simulation parameters.

Parameters	Values	Parameters	Values
RSU's Transmission power	40 dB	AN learning rate	0.001
DDP link distance	10–50 m	CN learning rate	0.001
Number of V2R links	50	DF	0.98
Number of RBs	50	Starting exploration	1
Number of V2V links	$20, 40, 60, \dots, 180$	Finishing exploration	0.01
Each RB bandwidth	180 KHz	Total exploratory steps	1000
Frequency of carrier	5 MHz	Replay storage size	3000
Noise power spectrum density	−174 dBm/Hz	Mini-batch size	32
V2V link path loss exponent	4	Steps in a epoch	20
Vehicle velocity	40 km/hr	Reward function weight	1, 1, 1, 1
Multipath fading	Unit mean	Level of discrimination	10
Shadowing standard deviation	8 dB	Weight of renew duration	10
Maximum transmit power of EVUE	25 dB	Optimizer	Adam
SINR threshold,	8 dB	Activation function	ReLu
V2R link path loss (LOS)	$38.40 + 21\log_{10}(d)$	V2R link path loss (NLOS)	$38.40 + 31.89\log_{10}(d)$
V2V link path loss (LOS)	$44.23 + 16.7\log_{10}(d)$	V2V link path loss (NLOS)	$42.52 + 30\log_{10}(d)$
Radius of EVUEs transmit power	200 m	Training episodes	0, 1, 2,. . . , 5000

interaction. The gathered experience samples $e_j^t = (s_j^t, a_j^t, r_j^t, s_j^{t+1})$ are then saved in ERB of capacity D. The PER technique is utilized to select the experience samples from the ERB in order to improve the effectiveness and efficiency of experience replay. Then, TDE values of each sample are determined using (44)–(46) are used to calculate p(q) and its associated DS weights $\mathbb{W}(q)$ using TDE values. The learning model then computes $\mathbb{V}(s_j^t; \theta, \mu)$ and $\mathbb{F}(s_j^t, a_j^t; \theta, \varsigma)$ and adds these two values to estimates of $\mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu)$ in (40). Then the agent renews the DDQN's parameter θ at the completion of every step by applying the GD approach and exchanging small portions of decision data such as resource and power allocation with neighbouring agents in a distributed and coordinated fashion. Lastly, the learning framework is trained to find a combined smart resource allocation strategy according to the chosen action $\hat{a}_j^t = \underset{a_j^t \in \mathcal{A}}{\arg\max}\ \mathbb{Q}(s_j^t, a_j^t; \theta, \varsigma, \mu)$.

4. Performance evaluation

In this section, the performances of the proposed scheme is evaluated and discussed. This section consists of three parts: (i) Baseline Schemes (ii) Simulation settings, and (iii) Results and Discussion.

4.1. Baseline schemes

In this section, we discussed the baseline schemes with which the performance of the proposed scheme is compared.

- Centralized DRL (CDRL) Scheme [14]: In this paper, the greedy scheme is used to determine the transmission mode, and the Hungarian algorithm and closed-form solution are used to calculate the best transmit power and RB for each V2V pair. It should be noted that this procedure is carried out centrally by the BS, which is supposed to have global CSI.
- Single Agent DRL (SADRL) Scheme [26]: In this paper, the direct V2V communication mode is used. Here, the each V2V pair individually determines its RB and transmit power depending on the local DRL model.
- Random Scheme [2]: In this paper, only the V2V mode is adopted. Here, the V2V pair chooses the RB at random from a pool of RBs. Also, the maximum transmission power is used to reduce the lower interference.

4.2. Numerical settings

In this subsection, we define the numerical variables that will be used to simulate the recommended approach. The presented scheme's functionality is modelled on the Pytorch using a CPU (Intel(R) Xeon(R) Gold 6134M) and a GPU (NVIDIA Quadro P5000). The simulated memory is expected to be 32 GB. To estimate the LOS state, insertion loss, shading, and rapid fading characteristics, we employed the urban scenario specified in 3GPP TR 37.885. Consider a two-lane (1 Km × 1 Km) intersection situation in which the cars follow a spatial Dynamical system and move randomly on the road. Let the BS to be in the network's center. Assume 15 cars with or before specific loading for interactions with the RSU, whereas each activated V2V transmitter maintains a V2V link also with vehicle situated at the far end of its access to the public. Each vehicle regularly transmits defense transmissions, and the current maximum transmitting delay of a V2V package is set approximately 100 ms. The DQN utilized in the model is a fully integrated neural network composed of an input nodes, a convolution layer, and an output layer. The deep neural network design has three layers, each with 512, 512, and 256 neurons. The activation function and optimizer are the ReLu and dynamic moment estimation techniques, respectively. Table 1 lists the additional parameters relating to V2X transmission and DQN.

4.3. Results and discussion

4.3.1. Comparative metric

In this part, the whole network's sum rate is empirically tested and analysed in relation to various factors.

Fig. 4(a) illustrates the relationship between the total network's sum rate and network size. The results show that as number of nodes increases, the sum rate falls. This occurred because the influence of co-channel and cross-channel congestion rises with network size. Despite this, the results show that the suggested scheme achieves 17.25%, 36.72%, and 49.75% greater total rates than the baseline methods. The reason for this is that the suggested technique uses cross interference problems to manage the power of EVUEs. On the other hand, contact across nearby agents increases the learning rate, which reduces co-channel interference.

Fig. 4(b) illustrates the relationship between the entire network's sum rate and V2V links The results reveal that when the number of V2V links grows, so does the overall sum-rate. This occurred because co-channel congestion increased as the number of V2V lines increased. In addition, the graph shows that the suggested system achieves a greater total rate than the baseline designs. The suggested technique employs distributed coordinated DQN, which causes this condition. The multi-agent dulling deep-Q network-based technique was integrated with distributed coordinated training to learn the mode decision and resource management policy successfully. As a consequence, co-channel interference is reduced, and the QoS of the V2R and V2V links is improved.

Fig. 4(c) displays the relationship between the entire network's summation rate and the quantity of V2R links The finding suggests that as

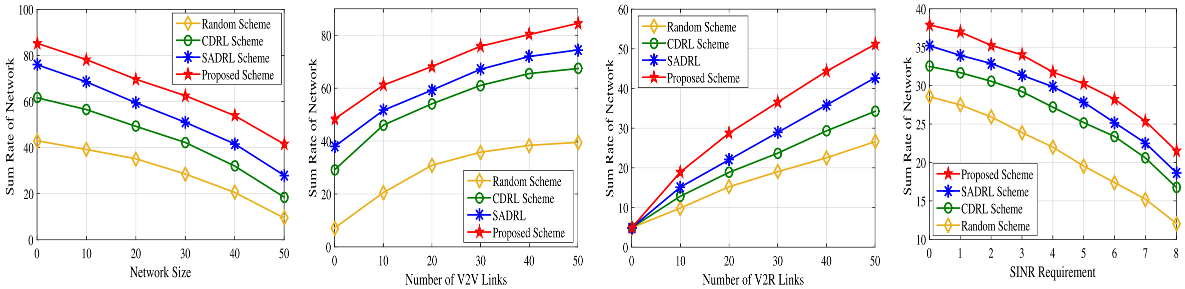

Fig. 4. Comparative analysis (a) sum rate of overall network v/s network size (b) sum rate of overall network v/s number of V2V links (c) sum rate of overall network v/s number of V2R links (d) sum rate of overall network v/s interference threshold.

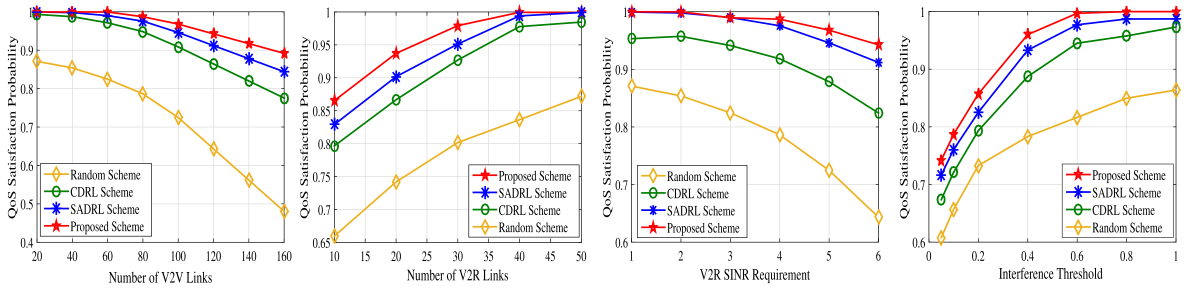

Fig. 5. Comparative analysis (a) QoS satisfaction probability v/s number of V2V links (b) QoS satisfaction probability v/s number of V2R links (c) QoS satisfaction probability v/s V2R SINR requirement (d) QoS satisfaction probability v/s interference threshold.

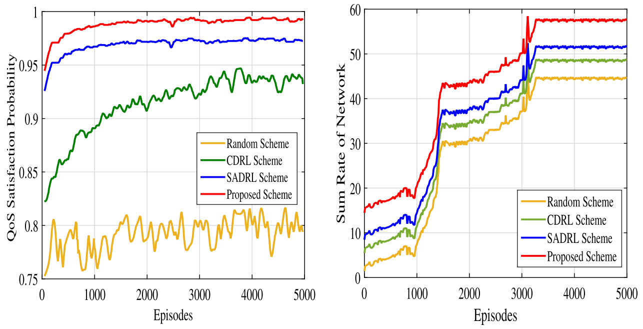

Fig. 6. Convergence analysis (a) QoS satisfaction probability v/s episodes (b) sum rate of network v/s episodes.

the CUs rise, so does the total rate. This occurred because as the volume of V2R links increased, the effect of co-channel interference decreased as the resource count increased. Furthermore, the results show that the suggested scheme outperforms the baseline schemes in terms of sum rate. The reason for this is that in the proposed architecture, each V2V pair not just to performs a distributed choice based on nearby findings, but also shares this knowledge with surrounding agents.

The connection between the entire system sum rate and the required SINR needs of V2V links is depicted in Fig. 4(d). This means that when the SINR requirement grows, the network sum rate falls. This happened because when the Signal power requirement increased, a greater number of resources were required to meet the demand for V2V pairings. It is also more difficult to supply services to automobiles with low channel gains. Despite these limitations, it can be shown in the figure that the suggested system outperforms the baseline methods. This occurred because the suggested approach coordinated with nearby cars, resulting in less co-channel interference.

4.3.2. QoS satisfaction probability

In this subsection, the suggested scheme's QoS probability is compared to the standard systems. The following equation is used to determine the QoS probability:

$$\text{QoS Probability} = \frac{\text{Number of V2V links}}{\text{Number of V2R and V2V links}} \tag{52}$$

Fig. 5(a) shows how the amount of V2V links affects the probability of QoS. The graph indicates that when the number of V2V connections in a cell is low, the QoS is good. When the volume of V2V connections in a cell approaches 60, the QoS begins to deteriorate. This occurred because as the amount of V2V lines increased, so did the amount of resources and co-channel interference. To overcome this issue, we employed the PS-DC-DDQN to adjust the power of the EVUE pair, which resulted in less co-channel interference. As a result, the suggested scheme achieved greater QoS than the baseline methods.

Fig. 5(b) depicts the relationship between QoS probability and the volume of V2R links (b). The data implies that as the amount of V2R links in a cell grows, so does the likelihood of QoS success. The reasoning behind this is because the size of V2R links is proportionate to the available resources.

Fig. 5(c) shows the variation in QoS likelihood in relation to the minimum V2R SINR requirement The results show that as the minimum rate requirement of V2R connections grows, so does the likelihood of QoS. The reason for this behaviour is that when the SINR need grows, so does the EVUE's transmission power, resulting in increased co-channel interference across the V2R lines. As a result, the QoS probability of V2V connections steadily increases while that of V2R links gradually decreases, resulting in a fall in the likelihood of overall QoS networks. Despite this, the suggested system outperforms the standard schemes in terms of QoS since we developed the PS-DC-DDQN scheme, which not only eliminates co-channel disruption but also preserves the QoS probability of the V2R connections throughout each SC.

Fig. 5(d) depicts the change in QoS probability as a function of the interference threshold (d). The results show that as the interference threshold improves, so does the likelihood of QoS, but after a certain point, the probability stays constant. The rationale for this pattern is that as the threshold grows, so does the transmission power of EVUE during V2V mode towards the V2R lines, reducing their performance. Regardless, the suggested system outperforms the baseline methods because to the inclusion of DDQN across the V2R connections, which mitigates the influence of interference created by the V2V links.

4.3.3. Convergence analysis

In this section, we analyse the convergence behaviour of the suggested scheme regarding both QoS probability and network sum rate. Also, our aim is to demonstrate the rationale behind the proposed approach's faster convergence compared to the baseline schemes.

The convergence performance of Algorithm 1 is showcased through the graph depicted in Fig. 6(a). From the graph, it can be inferred that the proposed scheme achieves network sum rate convergence within a maximum of 4000 episodes. This is attributed to the optimization of power of both the RSUs and EVUEs, which works towards reducing interference. To achieve the desirable outcome, each agent was subjected to multiple rounds of training and had to rely on the previously trained agents to expedite the process of finding the appropriate policy. This approach has been successful in ensuring that the proposed scheme converges at an accelerated pace, leading to improved network sum rate. The results obtained from this study could potentially serve as a foundational basis for future research in the field of communication networks. Overall, the proposed scheme's convergence performance demonstrates its potential for practical implementation and the capability to improve network performance.

Fig. 6(b) displays the convergence of the network sum rate for the proposed scheme, SADRL, CDRL, and random schemes, with varying numbers of episodes. The presented figure demonstrates that an increase in the number of episodes results in the network sum rate of the proposed scheme, SADRL, CDRL, and random schemes reaching their respective maximum values and converging at approximately 2500 episodes. Also, it has been observed that the proposed scheme achieves rapid convergence and outperforms SADRL, CDRL, and random schemes. This is attributed to the proposed scheme's ability to achieve optimal power allocation faster than SADRL, CDRL, and random schemes, respectively, as a result of its capacity to handle large dimensional action spaces in less time.

5. Conclusion

To achieve mode identification and approval process for cellular V2X transmission, an inter duelling deep-Q system technique combined with decentralized coordinated learning is suggested in this research. The primary goal is to optimize the total network data rates while maintaining the security and delay requirements of V2V couples and the data rate of V2R connections. Because the optimization issue is non-convex and NP-hard, it cannot be solved immediately. To address this issue, the MDP framework is first built to represent the defined problem. Each V2V pairing in this model chooses the appropriate communication protocol and allocates resources depending on local findings. In addition, to deal with the large states and actions regions, a multi-agent DRL-based distributed learning technique, PS-DC-DDQN, has been created to learn the best resource allocation strategy with a rapid convergence time. The simulation results show that the suggested scheme outperforms the state-of-the-art methods in terms of performance. In the future, we will extend our work to successfully perform handover operations utilizing parametrized DQN (P-DQN) while preserving both EVUE pair connection dependability and V2R data throughput. We used P-DQN because it proposes a framework that can directly work on the discrete-continuous hybrid action space without approximation or relaxation.

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.

Data availability

The data that has been used is confidential.

References

[1] M. Boban, A. Kousaridas, K. Manolakis, J. Eichinger, W. Xu, Connected roads of the future: Use cases, requirements, and design considerations for vehicle-to-everything communications, IEEE Veh. Technol. Mag. 13 (3) (2018) 110–123.

[2] S. Chen, J. Hu, Y. Shi, L. Zhao, LTE-V: A TD-LTE-based V2X solution for future vehicular network, IEEE Internet Things J. 3 (6) (2016) 997–1005.

[3] N. Lu, N. Cheng, N. Zhang, X. Shen, J.W. Mark, Connected vehicles: Solutions and challenges, IEEE Internet Things J. 1 (4) (2014) 289–299.

[4] S. Han, Y. Huang, W. Meng, C. Li, N. Xu, D. Chen, Optimal power allocation for SCMA downlink systems based on maximum capacity, IEEE Trans. Commun. 67 (2) (2019) 1480–1489.

[5] P. Liu, C. Wang, T. Fu, Y. Ding, Exploiting opportunistic coding in throwbox-based multicast in vehicular delay tolerant networks, IEEE Access 7 (2019) 48459–48469.

[6] S. Sharma, B. Kaushik, A survey on Internet of Vehicles: Applications, security issues & solutions, Veh. Commun. 20 (2019) 100182.

[7] S. Tanwar, S. Tyagi, I. Budhiraja, N. Kumar, Tactile Internet for Autonomous Vehicles: Latency and reliability analysis, IEEE Wirel. Commun. 26 (4) (2019) 66–72.

[8] S. Aradi, Survey of deep reinforcement learning for motion planning of autonomous vehicles, IEEE Trans. Intell. Transp. Syst. 17 (8) (2020) 1683–1692.

[9] I. Budhiraja, N. Kumar, S. Tyagi, Deep-reinforcement-learning-based proportional fair scheduling control scheme for underlay D2D communication, IEEE Internet Things J. 8 (5) (2021) 3143–3156.

[10] L. Liang, J. Kim, S.C. Jha, K. Sivanesan, G.Y. Li, Spectrum and power allocation for vehicular communications with delayed CSI feedback, IEEE Wirel. Commun. Lett. 6 (4) (2017) 458–461.

[11] H. Yang, K. Zheng, L. Zhao, L. Hanzo, Twin-timescale radio resource management for ultra-reliable and low-latency vehicular networks, IEEE Trans. Veh. Technol. 69 (1) (2019) 1023–1036.

[12] H. Yang, L. Zhao, L. Lei, K. Zheng, A two-stage allocation scheme for delay-sensitive services in dense vehicular networks, in: IEEE International Conference on Communications Workshops, ICC Workshops, Paris, France, IEEE, May, 2017, pp. 1358–1363.

[13] M.I. Ashraf, C.-F. Liu, M. Bennis, W. Saad, Towards low-latency and ultra-reliable vehicle-to-vehicle communication, in: European Conference on Networks and Communications, EuCNC, Shengai, China, IEEE, Aug., 2017, pp. 1–5.

[14] L. Liang, S. Xie, G.Y. Li, Z. Ding, X. Yu, Graph-based resource sharing in vehicular communication, IEEE Trans. Wireless Commun. 17 (7) (2018) 4579–4592.

[15] H. Ye, G.Y. Li, B.-H.F. Juang, Deep reinforcement learning based resource allocation for V2V communications, IEEE Trans. Veh. Technol. 68 (4) (2019) 3163–3173.

[16] X. Li, L. Ma, R. Shankaran, Y. Xu, M.A. Orgun, Joint power control and resource allocation mode selection for safety-related V2X communication, IEEE Trans. Veh. Technol. 68 (8) (2019) 7970–7986.

[17] X. Hou, Z. Ren, J. Wang, W. Cheng, Y. Ren, K.-C. Chen, H. Zhang, Reliable computation offloading for edge-computing-enabled software-defined IoV, IEEE Internet Things J. 7 (8) (2020) 7097–7111.

[18] M. Harounabadi, D.M. Soleymani, S. Bhadauria, M. Leyh, E. Roth-Mandutz, V2X in 3GPP standardization: NR sidelink in release-16 and beyond, IEEE Commun. Stand. Mag. 5 (1) (2021) 12–21.

[19] C. Wu, T. Yoshinaga, Y. Ji, Y. Zhang, Computational intelligence inspired data delivery for vehicle-to-roadside communications, IEEE Trans. Veh. Technol. 67 (12) (2018) 12038–12048.

[20] Y. Sun, M. Peng, H.V. Poor, A distributed approach to improving spectral efficiency in uplink device-to-device-enabled cloud radio access networks, IEEE Trans. Commun. 66 (12) (2018) 6511–6526.

[21] S. Yan, X. Zhang, H. Xiang, W. Wu, Joint access mode selection and spectrum allocation for fog computing based vehicular networks, IEEE Access 7 (2019) 17725–17735.

[22] Y. Sun, M. Peng, S. Mao, Deep reinforcement learning-based mode selection and resource management for green fog radio access networks, IEEE Internet Things J. 6 (2) (2018) 1960–1971.

[23] Y. Sun, M. Peng, Y. Zhou, Y. Huang, S. Mao, Application of machine learning in wireless networks: Key techniques and open issues, IEEE Commun. Surv. Tutor. 21 (4) (2019) 3072–3108.

[24] R.F. Atallah, C.M. Assi, M.J. Khabbaz, Scheduling the operation of a connected vehicular network using deep reinforcement learning, IEEE Trans. Intell. Transp. Syst. 20 (5) (2019) 1669–1682.

[25] K. Zhang, S. Leng, X. Peng, L. Pan, S. Maharjan, Y. Zhang, Artificial intelligence inspired transmission scheduling in cognitive vehicular communications and networks, IEEE Internet of Things J. 6 (2) (2019) 1987–1997.

[26] K. Zhang, Y. Zhu, S. Leng, Y. He, S. Maharjan, Y. Zhang, Deep learning empowered task offloading for mobile edge computing in urban informatics, IEEE Internet Things J. 6 (5) (2019) 7635–7647.

[27] Y. Yuan, G. Zheng, K.-K. Wong, K.B. Letaief, Meta-reinforcement learning based resource allocation for dynamic V2X communications, IEEE Trans. Veh. Technol. 70 (9) (2021) 8964–8977.

[28] H. Yang, X. Xie, M. Kadoch, Intelligent resource management based on reinforcement learning for ultra-reliable and low-latency IoV communication networks, IEEE Trans. Veh. Technol. 68 (5) (2019) 4157–4169.

[29] I. Budhiraja, N. Kumar, H. Sharma, M. Elhoseny, Y. Lakys, J.J. Rodrigues, Latency-energy tradeoff in connected autonomous vehicles: A deep reinforcement learning scheme, IEEE Trans. Intell. Transp. Syst. (2022).

[30] I. Budhiraja, N. Kumar, S. Tyagi, ISHU: Interference reduction scheme for D2D mobile groups using uplink NOMA, IEEE Trans. Mob. Comput. 21 (9) (2022) 3208–3224.

[31] H. Sharma, I. Budhiraja, P. Consul, N. Kumar, D. Garg, L. Zhao, L. Liu, Federated learning based energy efficient scheme for MEC with noma underlaying UAV, in: Proceedings of the 5th International ACM Mobicom Workshop on Drone Assisted Wireless Communications for 5G and beyond, 2022, pp. 73–78.

[32] I. Budhiraja, N. Kumar, S. Tyagi, Q.-V. Pham, S. Tanwar, Energy efficient mode selection scheme for wireless powered D2d communications with NOMA underlaying UAV, in: IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops, INFOCOM WKSHPS, IEEE, 2020, pp. 877–882.

[33] I. budhiraja, N. Kumar, S. Tyagi, Energy-delay tradeoff scheme for NOMA-based D2D groups with WPCNs, IEEE Syst. J. 15 (4) (2021) 4768–4779.

[34] V. Vishnoi, I. Budhiraja, S. Gupta, N. Kumar, A deep reinforcement learning scheme for sum rate and fairness maximization among D2D pairs underlaying cellular network with NOMA, IEEE Trans. Veh. Technol. (2023).

[35] H. Sharma, N. Kumar, I. Budhiraja, A. Barnawi, Secrecy rate maximization in THz-aided heterogeneous networks: A deep reinforcement learning approach, IEEE Trans. Veh. Technol. (2023).

[36] P. Consul, I. Budhiraja, R. Chaudhary, D. Garg, FLBCPS: Federated learning based secured computation offloading in blockchain-assisted cyber-physical systems, in: 2022 IEEE/ACM 15th International Conference on Utility and Cloud Computing, UCC, IEEE, 2022, pp. 412–417.

[37] N. Van Huynh, D.T. Hoang, D.N. Nguyen, E. Dutkiewicz, Optimal and fast real-time resource slicing with deep dueling neural networks, IEEE J. Sel. Areas Commun. 37 (6) (2020) 1455–1470.

[38] X. Tao, A.S. Hafid, DeepSensing: A novel mobile crowdsensing framework with double deep Q-network and prioritized experience replay, IEEE Internet Things J. 7 (12) (2020) 11547–11558.