Received May 11, 2022, accepted June 4, 2022, date of publication June 10, 2022, date of current version June 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3182021

Efficient Reinforcement Learning-Based Transmission Control for Mitigating Channel Congestion in 5G V2X Sidelink

LAN-HUONG NGUYEN[1,2], VAN-LINH NGUYEN[1,2], (Member, IEEE), AND JIAN-JHIH KUO[1,3], (Member, IEEE)

[1]Department of Computer Science and Information Engineering, National Chung Cheng University, Minhsiung, Chiayi 62102, Taiwan
[2]Department of Information Technology, Thai Nguyen University of Information and Communication Technology, Thai Nguyen 25000, Vietnam
[3]Advanced Institute of Manufacturing With High-Tech Innovations, National Chung Cheng University, Minhsiung, Chiayi 62102, Taiwan

Corresponding author: Jian-Jhih Kuo (lajacky@cs.ccu.edu.tw)

This work was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 107-2218-E-194-016-MY3, Grant MOST 108-2221-E-194-025-MY3, and Grant MOST 110-2811-E-194-501-MY2.

ABSTRACT Channel congestion has been an open challenge for vehicular networks due to the limited resource of communication channels. Explosion of channel access requests from a massive number of transmitter vehicles can exhaust bandwidth and then degrade transmission quality. The rapid drop of messages (because of the high bit error rate in the transmission congestion condition) can threaten the safety of connected vehicles. Maintaining congestion-free communications is then essential to improve the reliability for vehicular networks, including Cellular-V2X (C-V2X)-based cooperative intelligent transport systems and road-safety applications. In this work, we present a novel intelligent transmission control model, namely DEEPCUT, to automatically adjust the message broadcasting rate of a transmitter vehicle. DEEPCUT works based on a Double Deep Q-learning Networks with Prioritized Experience Relay framework. DEEPCUT encourages the transmitter vehicle to (1) reduce its broadcasting rate if the vehicle is maintaining a safe distance from its neighbors and (2) increase the rate if the vehicle is approaching the others at a high-risk distance, all done by using reward/punish strategies. The evaluation results show that DEEPCUT can cut up 16% redundant data while increasing 22% packet reception rate compared with baseline models, particularly in crowd vehicular communications. Our risk-based transmission control can be an excellent complement to address the congestion when the channel cannot satisfy every vehicle's resource requests. At best, the risk assessment-based approach in our congestion control method can provide a novel material to enhance Decentralized Congestion Control (DCC) for 5G V2X sidelink in the coming specifications.

INDEX TERMS Vehicular network congestion, transmission control, reinforcement learning.

I. INTRODUCTION

Channel congestion is an open challenge in vehicular communications [1], [2]. The problem is particularly worse in crowding contexts. For example, channel congestion can prevent many vehicles from accessing the network and sharing sensing data successfully for cooperative road-safety vehicular applications. Suppose the vehicles are moving in a crowded area with less visibility. In that case, the sharing interruption from the congestion can risk the safety of connected vehicles, e.g., tailgating, because of tracking loss. There are several reasons for the difficulty of addressing the congestion. First, the wireless spectrum and bandwidth are often limited, even with 5G Vehicle-to-Everything (V2X) [3], [4]. To reflect the latest updates of the vehicle movement, User Equipment (UE) in V2X-enabled vehicles must broadcast a large number of beacon messages, e.g., Cooperative Awareness Message (CAM) or Basic Safety Message (BSM) [5]). In the urban driving condition, the throughput can surpass 1GB per second if video streaming and LIDAR raw data are shared [6]. With a dozen vehicles sharing simultaneously, the network bandwidth can quickly become exhausted. In the future, the new demands for holographic infotainment applications can worsen *the congestion* [7]. An efficient

The associate editor coordinating the review of this manuscript and approving it for publication was Jie Gao.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

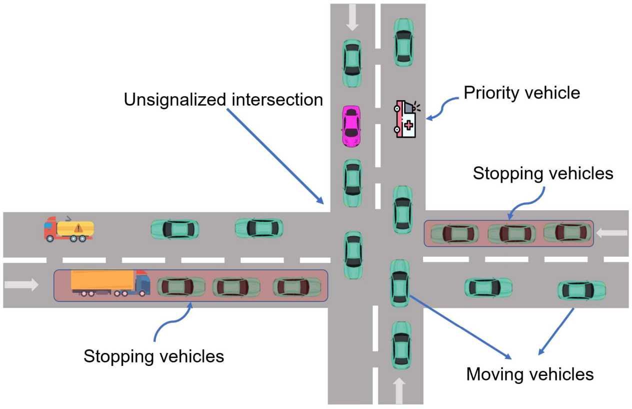


FIGURE 1. The illustration of assessing the risk of vehicles in the unsignalized intersection to cut down low-risk vehicles' sending rate when the V2V communication channel is already overloaded and congested. Stopping vehicles do not risk to the safety of the other vehicles since their fixed locations are noticed by the other vehicles in prior transmission. In this case, the stopping vehicles in the red zone can cut down their sending rate. By contrast, the moving vehicle should hold frequent updates to avoid a potential collision.

congestion control mechanism has been the target of many studies for years [8]–[11]. Generally, Dedicated Short-Range Communications (DSRC) IEEE 802.11p uses DCC mechanisms for congestion control [12], [13] while the congestion control mechanism for C-V2X/5G V2X is still being finalized. Besides, many scholars have proposed new approaches (e.g., Linear Message Rate Integrated Control (LIMERIC) in DSRC [12], [14], [15]) to enhance the standards. Recently, the machine learning-based congestion control approach has also started receiving much attention, e.g., [16], [17].

However, to the best of our knowledge, there is no work to exploit machine learning to assess the specific contexts of vehicular communications (e.g., collision risk of vehicles as illustrated in Figure 1) and then suggest a reasonable data rate for transmitter vehicles. There are two key challenges to pursuing this approach. First, there must have an efficient mechanism to *determine which vehicles are the ones to cut down their sending rate*. In a vehicular network, all vehicles should broadcast beacon messages to the neighbors periodically to maintain up-to-date information about the surrounding driving environment. By using fixed intervals for sending messages for every vehicle, the sharing from many vehicles in crowding areas can create a burst of traffic and lead to channel congestion. With the limited channel capacity, satisfying the requests for channel usage from all vehicles is impossible. Thus, if we can cut down the sending rate of several vehicles, the busy channel percentage can remarkably be reduced. Second, the data sharing must be adequate for fusing in the receiver vehicles; otherwise, improper rate cuts or long intervals can cause the receiver vehicles to fail to get the latest updates on surrounding movements and thus threaten safety. Besides, *safety requirement must be the priority*, i.e., the vulnerable-to-collision vehicles should be prioritized to use the channel. In summary, there is a trade-off between determining a proper broadcasting rate for improving channel usage and ensuring fairness/safety among different users. Maintaining the *trade-off* and *fairness/safety* in broadcasting information rate for different vehicle types in congestion situations is then a vital issue.

To address the challenges, this paper presents a novel efficient transmission control, namely DEEPCUT. The novelty of our system is to build intelligent Deep Reinforcement Learning (DRL)-based agents on the vehicles that can assess the collision risk during communications (small gap between the vehicles) and determine a proper sending rate for transmitter vehicles. Through the modeling, we show an important lesson that increasing of data sharing to reflect the latest updates of the vehicle movement does not always improve the overall safety of the driving. By contrast, the increase of data sharing into V2X networks can cause network congestion even worse and then threaten the safety of the vehicles due to tracking loss. Also, our risk-based intelligent transmission control model can maintain the vehicles' safety while not sacrificing valuable bandwidth for transmitting redundant data. Our main contributions are summarized as follows.

- Inspired by the urgency of improving the safety of V2X technologies for deploying in the coming years, we propose an efficient risk-based assessment method to suggest the proper data rate of broadcasting V2X messages. By measuring the risk of the driving context, the decentralized DRL agents on the vehicles can automatically adjust their sending rate according to the environment observation, thus mitigating pouring redundant data to V2X networks. Vehicles then act as intelligent machines to broadcast messages with the awareness of the safety changes in the surrounding environment.
- Our scheme can enhance channel-based congestion control – which is essential in vehicular communications. The evaluation results demonstrate the significant effects of the method in reducing the potential congestion for on-road safety applications, particularly maintaining fairness among different risk/non-risk vehicles. Precisely, DEEPCUT can also cut up 16% redundant data while increasing 22% packet reception rate compared with baseline models in congested traffic scenarios.
- The DRL-based risk management in our control can enhance the accuracy of the self-rate control and prevent the self-confidence of risk estimations through a multi-agent learning scheme. To the best of our knowledge, the safe-distance assessment from signal-based positioning techniques for transmission control is the first attempt.

The remainder of this paper is organized as follows. Section III presents our assumption and problem formulation. The details of our proposed risk management and rate control scheme are presented in Section IV. The evaluation results of the proposal are shown in Section V. Finally, the conclusion and future work are summarized in Section VI.

II. RELATED WORK

Mitigating channel congestion in vehicular networks has been well-studied for years. Our summary of typical congestion control mechanisms for vehicular networks and their features is presented in Table 1. Specifically, in cellular-based vehicular networks, the problem of congestion minimization can be resolved by maximizing the number of neighbors to receive exchange messages or packet reception ratio as defined in 3rd Generation Partnership Project (3GPP) specification [18]. The other way is to maximize the number of vehicles to access the channel through resource allocation and minimize communication delay/packet error rate (PER) [19], [20]. Generally, the methods are grouped into conventional model-based and machine learning-based. In the first type, in recent specifications, both Society of Automotive Engineers (SAE) and European Telecommunications Standards Institute (ETSI) organization indicates some baseline models, e.g., DCC mechanisms for DSRC IEEE 802.11p [12], [13]. DCC probes channel periodically to suggest a proper time interval for channel access, e.g., $1s$.

In the other studies, the authors in [21], [22] propose to adjust the transmission rate by evaluating the busy channel percentage. The location and direction of vehicles can be used as the measurement metric to prioritize the group with the same characteristics to access the channel [9]. Choudhury *et al.* [23] present a self-risk assessment for improving the safety of 802.11p based V2V Networks. Similarly, the authors in [24] use a dynamic distance-based evaluation to decrease the transmission rate of the vehicles in specific contexts (e.g., the vehicles are stopping or far from the others), thereby mitigating congestion. A summary of congestion control mechanisms can be found in the surveys [15], [25]. However, the lane for 802.11p-based vehicular communications is significantly narrowed after the recent decision of US Federal Communications Commission (FCC) decision [26]. Accordingly, in November 2020, the FCC unanimously approved to reallocating 45 MHz of the DSRC 5.9GHz spectrum to other unlicensed uses (e.g., WiFi). A spectrum of 30MHz is still kept for transportation-related services but expected to transition to C-V2X eventually.

For C-V2X networks, 3GPP layouts a general framework for access-layer congestion control for LTE-V2X and 5G New Radio V2X [8]. The control uses sensing-based semi-persistent scheduling (SB-SPS) mechanism to manage the congestion through scheduling resource reservation interval (RRI) [8], [27]. Accordingly, the RRI block can be $\{20, 50, 100, 200, \ldots, 1000\ ms\}$. The common points of the conventional-based congestion controls are: they are all or either to use channel characteristics, e.g., Channel Busy Ratio (CBR), packet dropping rate, for adjusting the channel access strategy (increasing RRI [20], [28], [29]). However, the disadvantage of these conventional-based approaches is the difficulty of knowing which vehicle should be the one being prioritized to use the channel. In urban areas, when many vehicles all move along and demand as much as possible connection resource for the sake of safety and their personal infotainment, it is challenging to satisfy such all requests ‘‘mechanically’’. As a result, the channel can quickly overload, and no vehicles will get their desire. Close to this work, the authors in [24] presented an aggressive risk-based transmission congestion control but the performance largely relied on the accuracy of the self-tracking engines. If the receivers cannot get any successful data, which is common in heavily congested situations, the system performance is significantly degraded.

Using machine learning for congestion control has received much attention recently due to its super performance [16], [30]. For example, Alperen *et al.* [19] propose an adaptive resource allocation for congestion control in 5G V2X communications by using a Multi-agent Deep Reinforcement Learning (MARL) model. In this model, each agent on the vehicle probes the channel state to adjust its transmission rate and maximize the packet reception ratio. However, relying on the channel estimation makes the method less robust to find the best solution if there is much noise or vehicles are moving in safe lanes. In another study, Lu *et al.* [31] enhance the channel access by tackling the power allocation problem and maximizing the sum rate performance by using a DRL framework. Such a centralized architecture of the framework is not suitable for V2V communications since building a consensus on the control plan is hard and infeasible due to dynamic changes in the network topology. Unlike prior work, the authors in [32] present a DRL-based data rate and transmission power control mechanism but specified for the old standard, i.e., DSRC V2V. Chen *et al.* [33] present a DRL-based method for radio access network information-assisted congestion control in 5G. However, the system is designed for Transmission Control Protocol (TCP) at the network layer. Similarly, Ma *et al.* [34] build a DRL-based method for TCP congestion control but specified for wired networks. Choi *et al.* [17] introduce a Deep Q-learning (DQN)-based congestion control to enhance DCC for C-V2X. However, using DQN may not be suitable for highly dynamic networks due to the instability learning of DQN models. In another work, Roshdi *et al.* [35] present a Deep Deterministic Policy Gradient (DDPG)-based congestion control model. However, the enhancement from work is merely to enhance DCC assessments. None of the existing works consider the collision risk of vehicles for improving congestion control.

In summary, to overcome the channel congestion in the vehicular networks, probing the channel state (e.g., using multiple factors as in DCC/LIMERIC) is an excellent way to cut down the sending rate. However, such a strategy is inadequate if we don’t consider the factor of safety. For example, an improper cutting on the sending rate for all vehicles can put vulnerable vehic‘es (moving at high speed and near each other) at high risk. In this case, *considering the new assessment factors, e.g., cut down the unnecessary access requests from low-risk vehicles as in our method, is a*

TABLE 1. Summary of several congestion control mechanisms for vehicular networks.

	Conventional-based approach				**Machine learning-based approach**			
Feature	[12]	[14]	[24]	[10]	[17]	[34]	[31]	**Our proposed scheme**
Methodology	DCC: keeping channel load below a target threshold by tunning multiple parameters such as power, transmit rate, data rate, sensitivity, transmit access	LIMERIC: Adaptive rate-control algorithm to adapt vehicle transmit rate in a way such that the total channel load converges to a target	RTC+: Transmit rate control in individual vehicles by collision-risk assessment	Enhance DCC by using age of information factor to tunning DCC parameter	Enhance DCC for C-V2 by using Deep Q-learning	Enhance DCC for C-V2X using DDPG	Enhancing radio access to support congestion control	Enhancing DCC by using risk-based assessment and DDQN models
Specified in standard	Yes	Mentioned	No	No	No	No	No	No
Vehicular technology	DSRC/ITS-G5, C-V2X (draft)	DSRC/ITS-G5	DSRC/C-V2X	DSRC/ITS-G5	C-V2X	C-V2X	5G	C-V2X
Simulation	NS3, Veins, OMNET++	Veins, NS2	Veins	PLEXE	LTEV2SIM	Unspecified	Unspecified	Veins
Measurement metric	PRR, CBR	CBR, IPG	PDR, CBR	CBR	CBR	PRR, CBR	CBR	PRR, CBR, CR
Major advantages	Consider multi-factors to control the channel load effectively	Effectively copes with stability and fairness issues	Effectively control the channel load by priority and risk impact	Enhancing channel load assessment in DCC	Support Tx power, packet Tx rate, and MCS control in a large scale	Can reache an optimal policy by utilizing CR adaptations	Applied for multiple protocols	The first idea to enhance DCC through considering the safety factor (risk assessment)
Major limitations	Unfairness data rate reduction, challenge to deal with multiple types of packets, low delivery rate in restrictive mode mode	Each user has full-precision information about the state of the congestion	Slow convergence, each vehicles to handle the congestion independently	Evaluating in platooning scenario cases only.	Instability in learning, requires a global info about the channel, no consider the risk of vehicles	Not suitable for large-scale V2V networks, no consider the risk of vehicles	Not specified for V2X	Low effectiveness for far distance vehicles

Decentralised Congestion Control (DCC), =Linear Message Rate Integrated Control (LIMERIC)
Packet Reception Rate (PRR), Channel Busy Ratio (CBR), Inter-Packet Gap (IPG), Collision Risk (CR)
Deep Deterministic Policy Gradient (DDPG), Double Deep Q-Learning Networks (DDQN)

promising approach. By considering the safety factor (risk assessment) in the data rate control for congestion resolution, our proposed scheme can provide a novel material to enhance DCC/LIMERIC. Our method is also different from the existing machine learning-based approaches. Specifically, our system considers the risk of vehicles in the data rate adjustment decision while the current methods do not.

III. SYSTEM MODEL & PROBLEM FORMULATION

In this work, we consider a 5G-based vehicular network with fixed bandwidth B to support a set of N connected and automated vehicles (CAVs), $N = \{1, 2, \ldots, |N|\}$. The vehicles are equipped with a V2X-enabled On-board Unit (OBU) to communicate with each other. Without loss of generality, we assume the antenna configuration in the vehicles is a Uniform Linear Array (ULA) type. The exchange messages are CAM [36] that can include the dynamic state of the vehicles, e.g., position, velocity, and heading. Vehicles communicate with each other through physical sidelink control channel in 5G NR V2X mode 2. In 5G V2V sidelink communications, the vehicles need to select a resource block from a set of available resources $R = \{1, 2, \ldots, |R|\}$. Intuitively, the network is at the non-congestion state if $|N| < |R|$, i.e., every vehicle can find a unique resource block for their access. However, in congested situations, $|N| \geq |R|$ and maintaining a unique resource block for each vehicle becomes impossible. Table 2 summarizes the notations used in this article.

Generally, according to [8], [27], C-V2X will likely rely on the congestion control mechanisms as in SAE J2945/1 and SAE J3161/1 if $|N| \geq |R|$. Since our method targets to work at the application layer, it has no impact on the existing MAC-layer-based congestion control mechanisms. At best, our work can be an extension to enhance the existing congestion control mechanisms in C-V2X. Finally, we assess vehicle states and risks from signal strength, which is cheap and easy to collect, to prioritize access to the channel.

TABLE 2. Notations used in this paper.

Symbol	Definition
N	Set of vehicles ($N \geq 1$)
N_p, N_v	Set of high-risk vehicles and low-risk vehicles
B	Bandwidth of V2V channel
N_0	The white Gaussian noise
R, R_i	Set of resource blocks (RB), RB for vehicle i
N_i^t	Neighbor vehicles of vehicle i
$d_{i,j}$	Distance between two vehicles (i, j)
d_{safe}	Minimum distance two vehicles (i, j) for safety
T	Observation period
$SINR_{i,j}^t$	SINR at vehicle j of the packet of vehicle i
DR_i^t	Broadcasting rate of vehicle i at the time t
DR_{sum}^t	Total data rate of all vehicles at the time t
DR_{max}, DR_{min}	Minimum, maximum data rate
PRR_i^t	Packet reception rate of vehicle i at the time t
$BLER_{i,j}^t$	Block error rate for the given $SINR_{i,j}^t$
RIS_i^t	Risk assessment of vehicle i at the time t

A. V2V CHANNEL MODEL

In V2V, light-of-sight(LOS) and non-light-of-sight (NLOS) links exist. For example, the vehicles may easily maintain LOS connections with each other in sparse areas. However, if the vehicles move into the areas with many obstacles, NLOS links likely dominate. Suppose that the LOS and NLOS path losses of a V2V link between the ith vehicle and the jth vehicle are $PL_{i,j}^{LOS}$ and $PL_{i,j}^{NLOS}$. The average path loss $PL_{i,j}$ over the probabilities of LOS path loss $PL_{i,j}^{LOS}$ and NLOS path loss $PL_{i,j}^{NLOS}$ is estimated as follows:

$$PL_{i,j}^{V2V} = p^{LOS} \times PL_{i,j}^{LOS} + (1 - p^{LOS}) \times PL_{i,j}^{NLOS}, \quad (1)$$

where p^{LOS} denotes the probability of having a LOS link between two vehicles. The LOS probability p^{LOS} varies, depending on the building and traffic density in the communication scenario. The details of p^{LOS} in some cities and highways can be found in [37].

For V2V LOS communications, according to [18], $PL^{LOS}_{i,j}$ is estimated in decibels (dB) in urban areas as follows:

$$PL^{LOS}_{i,j} = 38.77 + 16.7\log_{10} d_{i,j} + 18.2\log_{10}(f_v) \qquad (2)$$

where f_v is the carrier frequency of V2V (e.g., $f_v = 5.9GHz$), $d_{i,j}$ is the distance between the ith vehicle and the jth vehicle. $PL^{LOS}_{i,j}$ for V2V links on highways can be found in Section 6.2.1 of the specification [18]. For NLOS links, we use a standard model as follows:

$$PL^{NLOS}_{i,j} = 36.85 + 30\log_{10} d_{i,j} + 18.9\log_{10}(f_v) \qquad (3)$$

$PL^{NLOS}_{i,j}$ for other scenarios (e.g., open field (flat), suburban, rural, hills) can be found in 3GPP UMi street [38].

Suppose that vehicle i at the time t has a set of neighbor vehicles $N^t_i \subseteq N$. Then, the signal-to-interference-noise-ratio (SINR) at vehicle i of the packet of vehicle i can be expressed as follows:

$$SINR^t_{i,j} = \frac{|H^t_{i,j}|^2 P^t}{\underbrace{\sum_{k\in N^t_i : k\neq i,j} |H^t_{k,j}|^2 P^t}_{\text{Inference from all neighbors}} + PL^{V2V}_{i,j} + N_0}, \qquad (4)$$

where 1) $H^t_{i,j}$ is the channel gain between the transmitter vehicle i and the receiver vehicle j and 2) P^t is the transmit power of the vehicle OBU at the time t. We assume P^t is fixed for all vehicles and the same during the transmission. Notation N_0 is the additive white Gaussian noise, $N_0 \sim \mathcal{CN}(0, \delta^2)$. When two or more vehicles transmit at the same resource, we assume that the receiver will select the one with the highest SINR for decoding.

According to Shannon theory, the data sending rate of the unicast link between the vehicle i and vehicle j at the time t can be expressed by:

$$DR^t_{i,j} = B_{i,j} \times \log_2(1 + SINR^t_{i,j}) \qquad (5)$$

where $B_{i,j}$ is the V2V channel bandwidth. The sum rate with $|N|$ vehicles at the time t, namely DR^t_{sum}, are then calculated by:

$$\begin{aligned} DR^t_{sum} &= \underbrace{\sum_{i\in N_p}\sum_{j\in N^t_i} DR^t_{i,j}}_{\text{High-risk vehicles}} + \underbrace{\sum_{i\in N_v}\sum_{j\in N^t_i} DR^t_{i,j}}_{\text{Low-risk vehicles}} \\ &= \sum_{i\in N}\sum_{j\in N^t_i} DR^t_{i,j} \qquad (6) \end{aligned}$$

In beamforming-based systems, due to the difficulty of maintaining efficient beam scan on a large scope, we assume that the broadcasting rate of vehicle i at the time t, DR^t_i, as the data rate of an arbitrary link in the unicast transmission. Note that, in 5G mmWave networks, UE antennas support directional transmission instead of spherical radio wave propagation. With the limited resources of the V2V channel, to accommodate more vehicles for data exchange, we must reduce the data rate of each vehicle. However, that adjustment can impact safety negatively. In every case, the broadcasting rate from high-risk vehicles (defined in Section III-C, *collision risk*) should be prioritized to maintain unchanged.

Besides, each vehicle tries to maximize the number of surrounding vehicles that decode its packet to enhance safety. According to [18], the packet reception ratio (PRR) can be used to measure the ratio of the successful receptions among the total number of neighbors N^t_i of the transmitter vehicle i at the time t. At the time t, vehicle i calculates the average PRR, PRR^t_i, as follows:

$$PRR^t_i = \frac{1}{|N^t_i|}\sum_{j\in N^t_i}(1 - BLER^t_{i,j}), \qquad (7)$$

where $BLER^t_{i,j}$ is the block error rate for the given $SINR^t_{i,j}$ and modulation/coding scheme [19], $BLER^t_{i,j} \in [0, 1]$. Typically, the system can decode the received messages successfully if $BLER^t_{i,j} < 0.1$ (10%).

B. MOBILITY MODEL AND RELATIVE DISTANCE ESTIMATION

For mobility, we assume that the vehicle $i \in N$ can move with an arbitrary velocity v_i. The velocity v_i is limited by a threshold v_{max} (e.g., the maximum allowed speed of the road or at the intersections). Also, to update the neighbor vehicles on their presence, vehicle i periodically broadcasts beacon messages at a time interval Δ_i. The time interval Δ_i can be any value in the given range $[\Delta_{min}, \Delta_{max}]$. Adjusting time intervals can help to reduce the data pouring into the network. Depending on the velocity and the relative distance of the vehicles, the common value of Δ_{min} is often 20 milliseconds (ms) while Δ_{max} is $200ms$ [29].

Besides PRR and SINR, from the physical signals, one can estimate the relative distance between the transmitter and the receiver during their movements, e.g., by using the received signal strength indicator (RSSI) estimation. On modern OBUs, the transmitter-receiver distance estimation can be done via signal-based vehicular positioning methods due to the advance in signal processing techniques and the appearance of massive antenna arrays [39]. The common signal-based positioning techniques are Angle-of-Arrival (AoA), Angle-of-Departure (AoD), Time-Difference-of-Arrival (TDoA), and Phase-Difference-of-Arrival (PDoA)-based [39]. Due to the large errors of the RSSI-based positioning method in dynamic outdoor environments like V2X, in this work, we assume that the distance between the vehicle i and the vehicle j at the time t, namely $d^t_{i,j}$, is estimated via the signal-based vehicular positioning method. Accordingly, when a vehicle moves at 20 m/s (72km/h) speed and transmits at 10 Hz CAM, the average positioning error is 2m [39].

C. PROBLEM FORMULATION

As we defined above, the network is at the non-congestion state if $|N| \leq |R|$, i.e., every vehicle can find a unique resource block for their access. However, in congested situations, when the number of requests for resource blocks from N vehicles is much greater than available resource blocks, $|N| > |R|$, maintaining a unique resource block for each vehicle becomes impossible. Therefore, a congestion control system is designed to maximize the number of neighbor vehicles of each vehicle that can receive its broadcasting messages, i.e., PRR_i^t. This target is equal to the objective of the following optimization problem function:

$$\underset{DR_{i,j}^t}{\text{maximize}} \sum_{i \in N} PRR_i^t \tag{8a}$$

$$\text{subject to } DR_{i,j}^t \geq DR_{min}, \quad \forall i \neq j \in \mathcal{N},\ \forall t \in T \tag{8b}$$

$$DR_{sum}^t \leq DR_{max}, \quad \forall t \in T \tag{8c}$$

$$d_{i,j}^t \geq d_{safe}, \quad \forall i \neq j \in \mathcal{N},\ \forall t \in T \tag{8d}$$

where DR_{max} is the maximum data rate of the V2V channel. The constraint (8b) implies that, for safety control and QoS guarantee, $DR_{i,j}^t$ must be no less than a threshold DR_{min}. The constraint (8c) indicates that, with the limited sharing resources, the sum rate of the V2V channel at the time t, i.e., DR_{sum}^t in Eq. (6), is bounded by DR_{max}. The constraint (8d) means, under no circumstance, the vehicles can approach their neighbors at a high-risk distance.

Besides the methods of increasing the reuse ratio of resources by distance as suggested by 3GPP [18], to increase the probability of $|N|$ vehicles accessing the channel, in this work, we propose to reduce the data rate of low-risk vehicles and grant the portion of saved data rate for other vehicles. The adjustment is performed at the application layer and thus makes no impact on the MAC/PHY layer mechanisms. Notably, the adjustment should not risk the safety of the vehicles, which is the supreme goal of V2V communications. In the following paragraph, we introduce a new concept of collision risk evaluation that is the key metric for a follow-up adaptive sending rate adjustment mechanism.

We assume vehicle i and vehicle j are at risks of collision if $d_{i,j}^t - \alpha < d_{safe}$, where α is an expected error of the positioning. Note that $d_{i,j}^t$ is estimated through signal-based vehicular positioning methods as presented above. Depending on the speed of the vehicles, the safe distance gap d_{safe} also varies. For example, for the safety purpose (enough time to react, steering, and brake), two vehicles moving at 20 m/s (72 km/h) should maintain a minimum distance gap of 40 m. By contrast, if two vehicles stop, the safe distance may be only several meters. For simplicity, d_{safe} can be set by the distance of the driver can react over the current velocity, e.g., $d_{safe} = T \times v_i, T = 2(s)$ in default. In summary, the collision risk between two vehicles i, j at the time t, namely $RIS_{i,j}^t$, is expressed by:

$$RIS_{i,j}^t = \begin{cases} 1, & \text{if } d_{i,j}^t - \alpha < d_{safe}; \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Definition 1 (Collision Risk): The collision risk $RIS_{i,j}^t = 1$ means a high-risk case. Otherwise, $RIS_{i,j}^t = 0$ indicates a low-risk case, i.e., the vehicles still have time to react and brake if necessary.

In the low-risk case, the value of $d_{i,j}^t - \alpha - d_{safe}$ can be used to measure the urgency of the data update rate. In short, a higher positive value of $d_{i,j}^t - \alpha - d_{safe}$ means two vehicles are far from each other, and reducing the broadcasting rate can be acceptable. Reducing the broadcasting rate can significantly mitigate the busy channel ratio, and the channel can accept more vehicles to access. The following section introduces a deep reinforcement learning formulation for the packet reception optimization problem. Then, we present our approach to model, train, and test the DRL model.

IV. MULTI-AGENT DEEP REINFORCEMENT LEARNING-BASED TRANSMISSION CONTROL FOR CONGESTION MITIGATION

Unlike conventional optimization solutions, reinforcement learning considers the problem of a computational agent learning to make decisions by trial and error. By exploiting the power of the deep learning model, the DRL model can assist agents in learning from unstructured input data through millions of trial training on the state-action space. Deep RL algorithms can take in substantial inputs (e.g., sending rate options for all vehicles) and decide what actions to perform to optimize an objective (e.g., maximizing the packet reception ratio). The other advantage of DRL is that it can interact with the environment to adjust learning strategy and does not require labeled datasets for training. In this section, we first overview about DRL background and then transfer the problem (i.e., Eq. (8)) into the DRL problem. Finally, we detail the state-action-reward space and the training/testing process algorithm.

A. DEEP REINFORCEMENT LEARNING BACKGROUND

Generally, reinforcement learning (RL) is a machine learning method to solve the Markov decision process (MDP), which involves an agent interacting with the environment iteratively. Mathematically, an MDP can be specified by 4 tuple $< S, A, P, R >$, where 1) S is the state space, 2) A is the action space, 3) P is the state transition probability, with $P(s^{t+1}|s^t, a^t)$ specifying the probability of transiting to the next state $s^{t+1} \in S$ given the current state $s_t \in S$ after applying the action $a^t \in A$, and 4) $r^t(s^t, a^t)$ is the immediate reward received by the agent at the time t, usually denoted by $r(s^t, a^t)$ to show its general dependency on s^t and a^t. The agent's actions are governed by its policy $\pi : S \times A \to [0, 1]$, where $\pi(a^t|s^t)$ gives the probability of taking action $a^t \in A$ when in state $s^t \in S$. The goal of the agent is to improve its policy π based on its experience, so as to maximize its long-term expected return $\mathbb{E}[G] = \sum_{k=0}^{\infty} \gamma^k R_{t+k}$ the accumulated discounted reward from time step t onwards with a discount factor $0 \leq \gamma \leq 1$. A key metric of RL is the action-value function, denoted as $Q_\pi(s^t, a^t)$, which is the expected return starting from state s^t, taking the action a^t, and following policy π thereafter, i.e., $Q_\pi(s^t, a^t) = \mathbb{E}_\pi[G|s^t = s, A = a]$. Following the Bellman equation, the values and Q-values are related to each other. The value of the state depends on the value of the actions possible in that state $V_\pi(s)$, modulated by the probability that an action will be taken (i.e., the policy) as follows:

$$V_\pi(s^t) = \sum_{a^t \in A} \pi(a^t|s^t) Q_\pi(s^t, a^t) \tag{10}$$

$$Q_\pi(s^t, a^t) = \mathbb{E}[r(s^t, a^t) + \gamma V_\pi(s^{t+1})] \tag{11}$$

And the optimal action-value function is defined as follows:

$$V^*_\pi(s^t) = \max_{a^t \in A} Q^*_\pi(s^t, a^t) \tag{12}$$

$$Q^*_\pi(s^t, a^t) = \mathbb{E}[r(s^t, a^t) + \gamma \max_{a^{t+1} \in A} Q^*_\pi(s^{t+1}, a^{t+1})] \tag{13}$$

An essential task of RL is to obtain the optimal value functions. When the agent has no or incomplete prior knowledge about the MDP, it may apply the useful idea of temporal difference learning to improve its value function estimation by directly interacting with the environment. However, when the state-action space is large, a conventional RL is no longer suitable to apply. A common solution is to use deep reinforcement learning (DRL). When the action state space is large, a deep neural network (DNN) can be used to approximate the $Q(.)$ function. However, sometimes, a DNN may cause divergence, that is a negative result. To address this issue, DRL generally uses two major techniques: (i) experience replay and (ii) target network. However, in the experience replay buffer, uniform samples have been selected rather than importance-weighted samples, which may cause divergence with large state spaces. To address the issues of DQN, there are many solutions such as Double DQN, DQN with prioritized experience replay, and Rainbow [40].

In this work, we use the Double DQN (DDQN) [41], [42] as a result of our definition for the action space as a discrete vector. Firstly, instead of storing and updating the value functions for all state-action pairs, one only needs to learn the parameter θ, which typically has a much lower dimension than the number of state-action pairs. Secondly, function approximation enables generalization, i.e., the ability to predict the values even for those state-action pairs that have never been experienced, since different state-action pairs are coupled with each other via the function $Q(s^t, a^t; \theta)$ and parameter θ. The core of the DDQN algorithm is a Bellman equation as a simple value iteration update as follows:

$$y^{DDQN} \leftarrow r(s^t, a^t) + \gamma \max_{a^t \in A} Q(s^{t+1}, \arg\max_{a^{t+1} \in A} Q(s^{t+1}, a^{t+1}; \theta); \theta^-) \tag{14}$$

Therefore, the DDQN parameter θ can be updated by Eq. (15) to minimize the following loss function (16):

$$\theta \leftarrow \theta - \underbrace{\eta}_{\text{Learning rate}} \nabla_\theta Loss(\theta) \tag{15}$$

$$Loss(\theta) = \mathbb{E}[y^{DDQN} - Q(s^t, a^t; \theta)]^2 \tag{16}$$

Given the optimal function Q^*_π after a number of episodes, the i-th vehicles selects an action based on the following policy:

$$a_i^t = \begin{cases} \arg\max_{a_i^t \in (A)} Q^*(s_i^t, a_i^t; \theta_i) & \text{With } 1 - \epsilon \\ \text{random}(a_i^t) & \text{With } \epsilon, \end{cases} \tag{17}$$

where ϵ-greedy is a well-known method in DRL to select an optimal action by balancing between exploration and exploitation selection, $0 < \epsilon < 1$. The DDQN-based agents on the vehicles can independently update their transmission rate to reduce channel congestion. The following subsection details the basic parameters of the agent's state, action, and reward.

B. MULTI-AGENT DEEP REINFORCEMENT LEARNING-BASED TRANSMISSION CONTROL

Since V2V networks rely on the broadcast communication architecture, a multi-agent DRL is the best suitable model to maintain the convergence. In this model, a centralized training can be done offline after collecting neighbor vehicles' states through broadcasting messages. In the online execution, each agent can independently learn from the environment and adjust its policy to contribute to gaining the system's goal. In short, the state and action space of DRL in the congestion avoidance optimization problem are defined as follows:

Definition 2 (State Space): Each vehicle state at the time t, s_i^t, can be represented by (1) position p_i^{t-1}; (2) SINR^{t-1} patterns; (3) channel busy ratio CBR^{t-1}. The channel busy ratio CBR^{t-1} at the time slot $t - 1$ can be evaluated via the successful resource selection ratio or the ratio of the total number of received messages from neighbor vehicles and the channel capacity of the V2V network. Note that the parameters of the velocity and the relative distance between a vehicle and its preceding one can be excluded in the state space to reduce dimensions since they are already included in the risk assessment.

The state space for the multi-agent environment can be formulated as $s^t = \{s_1^t, \ldots, s_N^t\}$, $S \in \mathbb{R}^{3\times N}$.

Definition 3 (Action Space): At time slot t, the agent takes an action a_i^t to adjust transmission rate DR_i^t. Generally, we set the discrete values of $a_i^t = \{-\beta, 0, \beta\}$, where β is the step change for the transmission rate DR_i^t. The negative value indicates the agent will decrease DR_i^t while the agent will increase DR_i^t if selecting the positive value. $a_i^t = 0$ implies that the agent keeps the transmission rate unchanged compared to the prior state.

The action space for the multi-agent environment can be formulated as $a^t = \{a_1^t, \ldots, a_N^t\}$, $A \in \mathbb{R}^{3\times N}$.

The problem in Eq. (8) is transferred into the distributed optimization problem for each vehicle $i \in N$ as follows:

$$\arg\max_{a_i^t} \; r_i^t \tag{18a}$$

$$\text{subject to } DR_i^t \geq DR_{min}, \quad \forall t \in T \tag{18b}$$

$$RIS_i^t < 1, \quad \forall t \in T \tag{18c}$$

The goal of the DRL-based optimization problem in Eq. (18) is to optimize the policy on each agent $\pi(a_i^t|s_i^t)$ for maximizing the total number of all transmitter vehicles' neighbors that can decode the messages at time t. To achieve the goal, we need to set the reward for each step of action. We define the reward function for the action of adjusting the transmission rate of vehicle i as follows:

$$r_i^t = \begin{cases} \psi & \text{if } CBR^t > C_{thres} \text{ and } RIS_{i,j}^t = 1 \\ \delta & \text{else if } CBR^t > C_{thres} \text{ and } RIS_{i,j}^t = 0 \\ \lambda & \text{else if } CBR^t < C_{thres} \text{ and } RIS_{i,j}^t = 1 \\ \rho & \text{else if } CBR^t < C_{thres} \text{ and } RIS_{i,j}^t = 0, \end{cases} \tag{19}$$

where $\psi, \delta, \lambda, \rho$ are the weights of the reward or penalty, C_{thres} is the CBR threshold to determine the channel busy. The weights are used to utilize the cost of adjustment conditions. The reward also gets a penalty if the vehicles move near each other at a high-risk distance, i.e., $RIS_{i,j}^t = 1$. For fast convergence, the penalty is often a negative value and at the highest value if the vehicle chooses the action that leads to the congestion and fails to react to the collision risk. The penalty degrades if the action violates one of the constraints only. The values of these parameters are configurable. In this work, we set $\psi = -10, \delta = -2, \lambda = -4, \rho = 2$.

We consider a DRL system where each reinforcement learning agent on each vehicle simultaneously learns which resources to select for its transmission. At each time step t, each agent observes a state s_i^t locally, selects an action a_i^t (increase/decrease) from its policy $\pi(a_i^t|s_i^t)$, and receives a reward r_i^t from the environment. The sum of discounted rewards for the agent i for episodes of the observation interval T is $SR_i^t = \sum_{k=0}^{T} \gamma^{t+k} r_i^{t+k}$. Each vehicle aims to find a policy to maximize its expected accumulated discounted reward. Note that the reward of a vehicle i depends not only on the policy $\pi_i(a^t|s^t)$ but also on the policy of the other vehicles. The reward space for the multi-agent environment $R = \{r_1^t, \ldots, r_N^t\}, R \in \mathbb{R}^N$.

The goal of the problem in Eq. (18) is the cooperative learning among multiple agents to maximize their accumulated reward. Key algorithms of risk assessment and DDQN-based transmission rate control are presented in Algorithm 1 and Algorithm 2, respectively. Intuitively, before performing rate control, at the time t, each agent runs Algorithm 1 to build the list of neighbor vehicles (line 4) and evaluate risk by calculating the distance to each vehicle (line 5). The distance to the nearest neighbor vehicle $d_{i,j}^t$ is then selected for evaluating *RIS* (line 6-8). If the distance between vehicle i and the nearest vehicle j is less than a threshold, d_{safe}, the risk is identified ($RIS_x^t = 1$). Assume that x-axis and y-axis represent the vehicle driving direction and lane change direction, respectively. Due to the possibility of a sudden lane change of the vehicles, if the vehicles travel on the right/left side or the same lane ($|y_i - y_j| < 5$ in line 9), the target is still considered as "Risk identified".

Algorithm 1: Risk Assessment Algorithm

```
Data: $p_i^t$, $d_{safe}$
Result: $RIS_i^t$
1  Function RiskScore($p_i^t$,$d_{safe}$)
2  Initialize $RIS_i^t \leftarrow 1$; # default setting
3  $temp \leftarrow 0$; $\alpha \leftarrow 2$;
4  Extract received beacon messages to build the neighbor
     list $N_i^t$;
5  Calculate the distance to each vehicle through
     signal-based positioning;
6  Sort to find the nearest vehicle $j$ (by location) in the
     neighbor list $N_i^t$;
7  $d_{i,j}^t \leftarrow \sqrt{(p_i^t)^2 - (p_j^t)^2}$;
8  $temp \leftarrow d_{i,j}^t - \alpha - d_{safe}$;
9  $sideMov = |y_i - y_j| < 5?1 : 0$; # Traveling on the
     right/left side or the same lane.
10 if ($temp < 0 \&\& sideMov$) then
11   | $RIS_i^t \leftarrow 1$; # Risk identified
12 else
13   | $RIS_i^t \leftarrow 0$;
14 end
15 Embed $RIS_i^t$ into $i$'s beacon messages;
16 return $RIS_i^t$;
```

After obtaining risk assessment from the state information (e.g., p_i^t, CBR^t), the agent runs a training process to find the best sending rate DR_i^t (line 8-22 in Algorithm 2). For the training, system parameters such as learning rate η and memory size Ω (experience replay buffer) are initially set in default values, e.g., 0.01 and 10000. The state-action pairs are combined and stored in memory during the training process. Then, data are put to the main Q-network (see Figure 2), and the target network for training by batches. To avoid a potential delay in the rate control, the DDQN model can be pre-trained in advance through offline trials over different environments (different V2V use cases as illustrated in Figure 3). Then, in the testing phase, each agent can perform DR_i^t adjustment by using the trained model without waiting for a lengthy training process. At each time step t, if the channel is congested, the agent triggers to select an action with the maximum Q value by its trained Q network. Figure 2 illustrates our DDQN-based transmission rate control architecture. Each vehicle is supposed to equip with an agent to run its transmission rate control independently. The workflow of the components in the system is illustrated in Figure 4. Accordingly, the DDQN-based transmission rate control only activates if the channel is busy ($CBR^t > C_{thres}$). Based on the channel sensing and risk assessment, the pre-trained model can suggest a proper sending rate for the agent.

C. COMPLEXITY OF ALGORITHM AND THE WORST-CASE ANALYSIS

With a set of vehicles N, the time complexity of Algorithm 1 is $O(|N|^2)$ in the worst case, where all vehicles are the

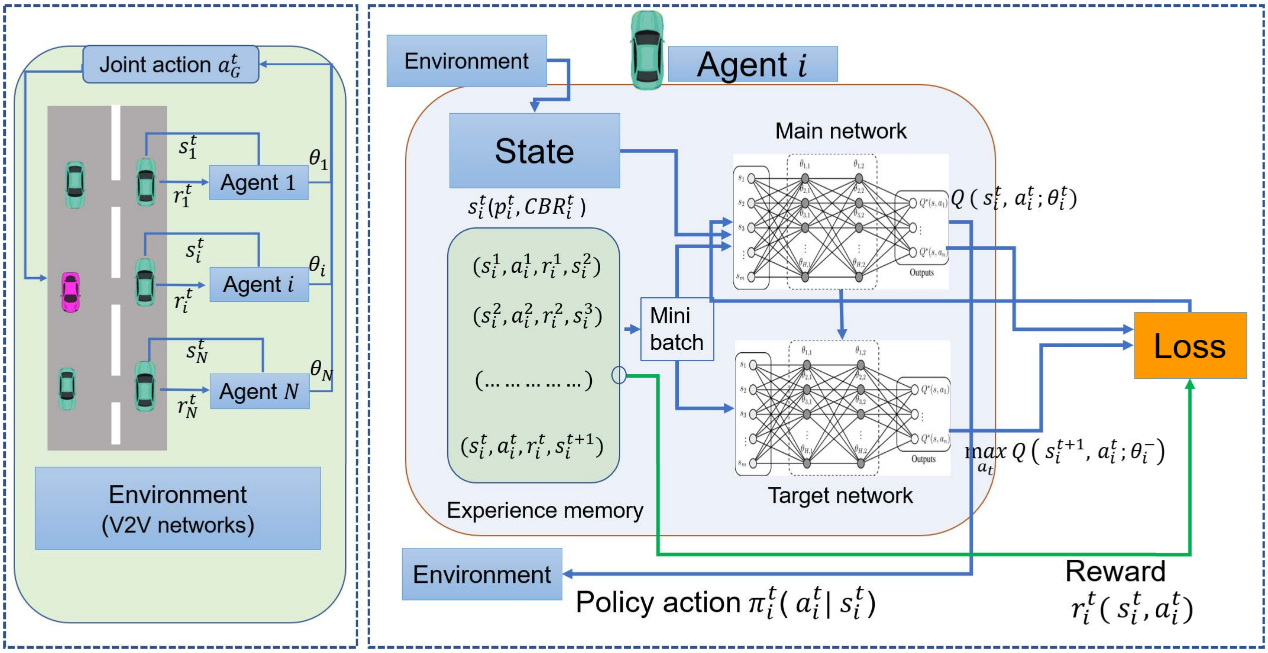


FIGURE 2. Our DDQN-based transmission rate control architecture.

neighbors of any vehicle in those. In this case, each vehicle must maintain $|N|(|N|-1)$ V2V links. Besides, in this work, the DDQN architecture is built based on a fully-connected structure. The computational complexity of DNN is estimated as follows. The computational complexity of DNN is estimated as follows. Let K^m and k_l^m (or K^c and k_l^c) denote the number of hidden layers and the number of the neurons in the l-th hidden layer of the main network (or the target network) of the DDQN model. Since the main/target network is fully connected, the computational complexity of training a DDQN (including the input layer, full-connected layers, and the output layer) can be written as $O^{DDQN} = J(2(3 \times N) \times k_1^m + \sum_{l=1}^{K^m-1} k_l^m k_{l+1}^m + \sum_{l=1}^{K^c-1} k_l^c k_{l+1}^c + k_{K^m}^m)$, where $J(\cdot)$ is the time complexity for updating the parameters of the fully-connected layers. The time complexity of the proposed scheme in Algorithm 2 for offline training is $O(M \times N \times T \times \Omega \times O^{DDQN})$. In the online execution, for each agent, the time complexity is $O(T \times \Omega \times O^{DDQN})$.

In the worst case, the number of vehicles in the congestion area is enormous, e.g., 500 vehicles over a short distance (traffic jam). Generally, the system reduces the low-risk vehicles' transmission rate for congestion avoidance. However, if the number of vehicles demanding channel access is still vast, cutting the transmission rate is likely impossible due to safety. We argue that, in this case, a novel control is required to activate for offloading. First, the nearby vehicles can be split into platoons. Figure 5 illustrates the platoons of vehicles after the splitting. Each platoon can manage the vehicles with a given length l (e.g., around 20 vehicles per platoon in the same lane). Second, the leaders of the platoons exchange the beacon messages to maintain the state of their regions. This groupcast communication can significantly mitigate the case of every vehicle requesting channel access.

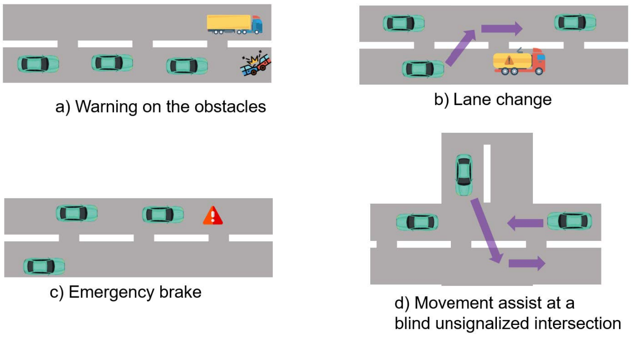

a) Warning on the obstacles
b) Lane change
c) Emergency brake
d) Movement assist at a blind unsignalized intersection

FIGURE 3. Training DDQN model through offline trials over different V2V-use cases can significantly help to accelerate the response tim for adjusting the transmission rate if the vehicle falls into one of the cases.

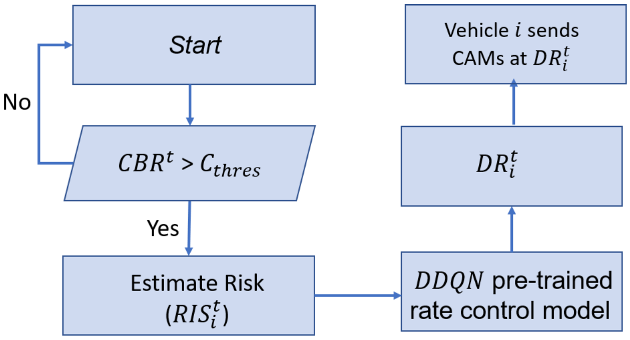

Start
No
$CBR^t > C_{thres}$
Yes
Estimate Risk (RIS_i^t)
$DDQN$ pre-trained rate control model
DR_i^t
Vehicle i sends CAMs at DR_i^t

FIGURE 4. The workflow of DDQN-based transmission rate control.

Algorithm 2: DDQN-Based Transmission Control Algorithm for Vehicles

```
Data: $RIS_i^t$, $\eta$, $DR_i^t$,$k$, $DR_{min}$,$N$,$\beta$
Result: $DR_i^t$
Function TransControl($RIS_i^t$, $\eta$, $DR_i^t$,$k$,$DR_{min}$,$N$, $\beta$)
Initialize replay memory $\Omega$ capacity $|N|$; mini-batch $k$; learning rate $\eta$,
Initialize action value step $\beta$, Q-function with random weights $\theta$
Initialize Target-$\hat{Q}$-function with the same weights $\theta^{-1} \leftarrow \theta$
for $i = 1$ to $N$ do
    Observe the state $s_i^0(p_i^0, CBR^0)$ and action $a_i^0$ ($DR_{min}$) with policy $\pi(a_i^0|s_i^0)$;
end
for $episode = 1$ to $M$ do
    for $i = 1$ to $N$ do
        for time step $t = 1$ to $T$ do
            Observe $s_i^t, a_i^t, r_i^t, \gamma^t$;
            Store transition $(s_i^{t-1}, a_i^t, r_i^t, \gamma^t, s_i^t)$ in replay memory $\Omega$;
            Select $a_i^t \leftarrow \arg\max_{a_i^t \in (A)} Q^*(s_i^t, a_i^t; \theta_i)$ in the target-$\hat{Q}$-network;
            Send messages at the rate $DR_i^t$ at the guide of $a_i^t$ and observe reward $r_i^t$;
            Calculate $s_i^{t+1}$;
            Store the experience $(s_i^t, a_i^t, r_i^t, \gamma^t, s_i^{t+1})$;
            Estimate the loss
              $Loss(\theta_i) \leftarrow \mathbb{E}_\pi[y_i^{DDQN} - Q(s_i^t, a_i^t; \theta_i))^2]$;
            Update $\theta_i \leftarrow \theta_i - \eta\nabla_{\theta_i} Loss(\theta_i)$;
            Update the target $\hat{Q}$-network with $\theta_i^{-1} \leftarrow \theta_i$;
        end
    end
end
```

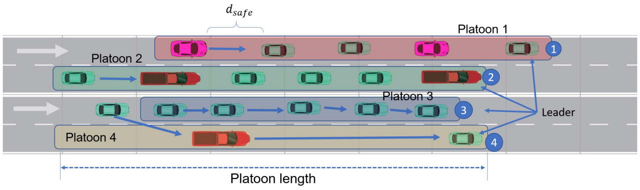


FIGURE 5. Illustration of splitting a large number of vehicles into platoons of vehicles and maintaining V2V communications for each group to reduce the congestion due to controlling channels for all vehicles.

V. PERFORMANCE EVALUATION

This section evaluates DEEPCUT performance in comparison with several baseline congestion control models such as (1) RTC+ [24], (2) DCC [18], (3) LIMERIC in DSRC [14]. The details of the models are summarized in Table 3. In this work, we use Veins, an open-source simulator with the third-party models for ETSI ITS-G5 (IEEE 802.11p) and 3GPP standard C-V2X via INET framework (OpenCV2X) [43] to validate our system performance and the baseline models. Veins integrated Simulation of Urban MObility (SUMO) for traffic engineering and vehicle mobility. Veins can accurately simulate traffic behavior and vehicular communications close to the real environment. Similar to [24], the road segment map of 2km with six lanes for simulation. All six lanes are available for the vehicles. Since this work is designed to reduce data flouring into the network, we perform two major traffic simulation cases which extremely require a strict transmission control: 1) high density (200 vehicles/km); and 2) congestion (300 vehicles/km). For low and medium traffic density, our system can be off if the network channel is not busy. For realistic mobility modeling, the two traffic simulation cases can be mapped to the traffic behavior traces at various times in Luxembourg city [44], e.g., high density (8:00 AM, 6:00 PM), congestion (traffic jam). Since the safety distance depends on the relative velocity of the vehicles, d_{safe} is dynamically set at the host vehicle velocity v_i and the observation period T, i.e., $d_{safe} = T \times v_i$ (defined in Section III-C). CBR starts at 80%. The other parameters for DEEPCUT training hyperparameters and V2X network configuration are summarized in Table 4.

TABLE 3. List of the congestion control models.

Model	Meaning	Sending rate
DEEPCUT	►DDQN-based rate control	Dynamic
RTC+ [24]	►Time-to-collision rate control	Dynamic
DCC [12]	►Channel busy ratio-based rate control	Dynamic
LIMERIC [14]	►Channel busy ratio-based rate control	Dynamic

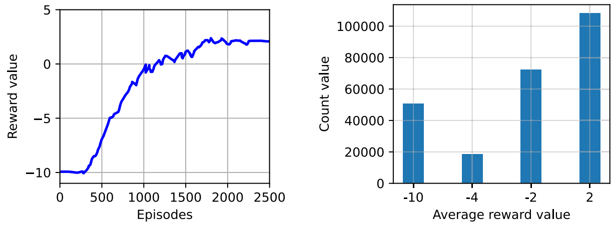
TABLE 4. The training hyperparameters and V2X network configuration.

Parameter	Value	Parameter	Value
Batch-size	512	V2X frequency	5.9GHz
Number of episodes	2500	V2X bandwidth	20MHz
Experience replay	1024	Subcarrier spacing	30kHz
Discount factor γ	0.9	Access scheme	OFDMA
Hidden layers	256 neurons	Transmit power	25dBm
Learning rate	1-e4	Message size	300 bytes
Activation function	ReLU	Observation period	2s
Optimizer	ADAM	Number of antennas	4

For measurement evaluation, Packet Reception Rate (PRR) is the critical metric to measure the ratio of the successful receptions among the total number of neighbors of a transmitter vehicle. This metric helps assess whether the system can increase the efficiency of using a network channel for many vehicles. To measure whether the reduction impacts safety, we used the Collision Risk (CR) metric. CR denotes the total times each pair of vehicles exceeds the safe distance d_{safe} at their relative speed. For on-road safety assessment, the maximum distance between the receivers and the transmitters is 300m. Finally, channel utilization can be assessed by using CBR metric.

A. TRAINING PERFORMANCE

We have trained our DEEPCUT model on the road segment map with mixed data of traffic simulation cases in 2500 episodes. As shown in Figure 6(a), the cumulative



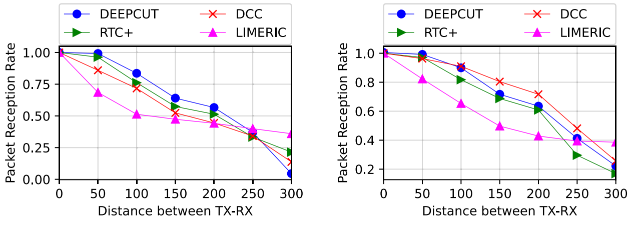
(a) (b)

FIGURE 6. The cumulative reward after 2500 training episode and the reward mean histogram: a) Rewards by episodes; b) Reward mean histogram. The weights of the reward and penalty for each action (in Eq. (19)) in default is $a = -10$, $b = -2$, $c = -4$, $d = 2$.

reward becomes convergence and remains stable after about 2000 episodes. The convergence speed slows if there are many vehicles and the step change for the transmission rate is small. This is because the state-action space from the matrix of all vehicles states and their possible actions is enormous. Figure 6(b) illustrates the reward histogram after training DEEPCUT with 2500 episodes. The reward distribution is changed based on the traffic density. In the worst case, high traffic density or traffic congestion produces many penalty values (negative) due to the difficulty of satisfying all channel requests of many risk-based-on-near-by-distance vehicles.

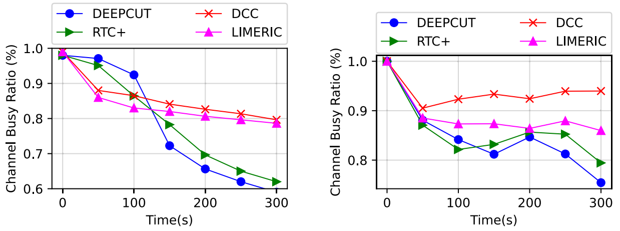
B. TESTING PERFORMANCE

Figure 7(a) and Figure 7(b) show the PRR performance results of our DEEPCUT system in comparison with the baseline methods in two traffic density cases. Accordingly, our method yields the best PRR performance (22% better) in all the methods if the distance between the transmitter (Tx) vehicle and the receiver (Rx) vehicle is no greater than 200m in Figure 7(a). We argue that, in this case, the risk assessment contributes significantly to reducing the data traffic. Compared with RTC+, a self-tracking-based approach [24], our system performs slightly better in the risk assessment and then improves the number of neighbor vehicles to receive data sharing (i.e., PRR). This improvement comes from the advantage of the DDQN training can find the best action configuration (transmission rate) from testing hundreds of thousands of configurations for all vehicles. That means the vehicles cooperate to adjust their sending rate to maximize the sum reward. By contrast, RTC+ lacks cooperation among vehicles to measure the risk and then adjust their sending rate together. Besides, DEEPCUT has a smaller benefit in PRR than a channel-busy-based approach like DCC or LIMERIC gains if the traffic is congested in far distances, as shown in Fig. 7(b). This is because if the Tx vehicles are far from the Rx vehicles (>200m and 100m in Figures 7(a) and 7(b)), potential collision risk is less critical. As a result, reducing the data rate for all transmitter vehicles as in 3GPP DCC or LIMERIC can be an efficient method to reduce the congestion immediately while having a less negative impact on vehicle safety (than doing that for near distances).



(a) (b)

FIGURE 7. The Packet Reception Rate performance of our DEEPCUT and three baseline models in two simulation cases: a) 200 vehicles/1km (heavy traffic); b) 300 vehicles/1km (congested traffic).

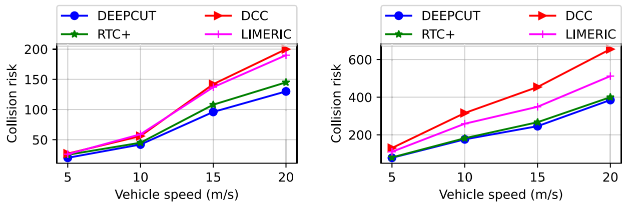


(a) (b)

FIGURE 8. The Channel Busy Ratio performance of our DEEPCUT and three baseline models in two simulation cases: a)200 vehicles/1km (heavy traffic); b) 300 vehicles/1km (congested traffic).

To measure the channel utilization, we used the CBR metric. As mentioned early, CBR can be measured by the number of successfully granted channel access over the total number of channel access requests or the ratio of the total number of received messages from neighbor vehicles and the channel capacity of the V2V network. The results in Figure 8(a) indicate that DEEPCUT can significantly reduce the ratio of channel overload during V2V high-density and traffic congestion cases compared with the baseline models. For example, DEEPCUT maintains an average of 20% better in cutting data pouring into V2V networks than the channel-busy-based transmission control approaches (DCC, LIMERIC) do. By exploiting cooperation in decentralized RL systems, our DEEPCUT system can improve channel usage's overall efficiency based on the vehicles' travel patterns. Specifically, CBR in Figure 8(b) shows that DEEPCUT can even maintain a little available channel space (∼4-8%) when a traffic jam occurs (150-*th* second). In this case, the risk of slow-moving vehicles is low. As a result, DEEPCUT-based agents on the vehicles suggest reducing the vehicle onboard units' sending rate without violating the safety or the urgency of V2V data sharing.

However, the results in Figure 8(a) also indicate the cutting speed on the sending rate of DEEPCUT is slower than that of the channel-busy-based congestion control approach (i.e., DCC/LIMERIC) is at the initial step. This is because DRL agents in DEEPCUT often take time to find an optimal configuration. In highly dynamic vehicular networks, the slow cutting can temporarily lead to a higher CBR than the immediate cutting approach as in LIMERIC. However, the case of



(a) (b)

FIGURE 9. The Collision rate performance of our DEEPCUT and three baseline models in two simulation cases: a)200 vehicles/1km (heavy traffic); b) 300 vehicles/1km (congested traffic).

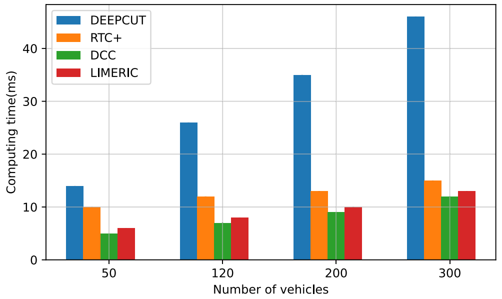


FIGURE 10. The processing speed of DEEPCUT in assessing vehicles and return a suggestion for sending rate, compared with three baseline models.

hundreds of vehicles moving at high speed in a short distance is uncommon in practice (e.g., racing). A solution, in this case, is to use groupcast transmission for multiple platoons instead of using a shared channel for all vehicles. And the leaders of the platoons can share the cutting strategies to converge quickly. We believe that finding a consensus configuration for many vehicles is a complicated issue with no perfect solution. In short, there is a trade-off between determining a proper broadcasting rate immediately and ensuring fairness among a large number of transmitter vehicles while maintaining safety.

To measure whether the data rate reduction impacts safety, we used the CR metric. As mentioned early, CR denotes the total times each pair of vehicles exceeds the safe distance d_{safe} at their relative speed. As shown in Figure 9, DEEPCUT can reduce the collision risk in all the cases of traffic density compared with the channel-based congestion control mechanisms. Even at high speed of the vehicles and high traffic density (Figure 9(a)), our system can assist in reducing up to 21% of the collision risk. This is because when our risk-based transmission model can accurately assess the risk patterns of nearby vehicles and suggest sending rate reduction for non-risk vehicles, which the channel-based congestion control cannot identify. Note that, in congested traffic cases, the vehicles move near each other, CR is high if there is improper acceleration. Besides, many obstacles ahead in the congested area can cause NLOS links that indirectly contribute to a high CR (as shown in Figure 9(b)).

In the worst case, i.e., the number of vehicles in the congestion area is huge, DEEPCUT-based agents take a slightly high cost of time to find the best configuration of the broadcasting rate. This is because the risk assessment for a massive number of neighbor vehicles is a time-consuming task. As shown in Figure 10, DEEPCUT's processing time to suggest a proper sending rate for the host vehicle is a little longer than the three baseline models if the traffic is heavy. This is because of the delay in finding a consensus on the data rate adjustment in our multi-agent learning model. Nonetheless, we argue that, since the system's latency for processing is lower than 50ms even in the worst case, the system is still applicable for many road-safety vehicular applications.

C. DISCUSSION

There is no perfect solution to solve all issues of channel congestion. Suppose the channel cannot satisfy every vehicle's resource request (sub-channel access through 5G default resource allocation functions) in crowd traffic cases. In that case, our risk-based assessment provides a new method to reduce redundant data transmission. Accordingly, the system will prioritize those who need to access the channel most (high-risk vehicles). By contrast, the channel-busy-based approach (DCC, LIMERIC) targets to reduce the transmission rate of all vehicles simultaneously if the channel is busy, regardless of the high-risk or low-risk state. This approach is potentially flawed for vehicle safety if the vehicles are at high-risk distances. In this case, our system has the advantages of guaranteeing both data transmission reduction and safety maintenance. For the best configuration, the two approaches should complement each other to assist in transmission congestion control, depending on the relative distance distribution of the vehicles.

Based on our baseline research in this work, several promising ideas can be conducted further. First, DCC is still being enhanced to become an official congestion control standard for C-V2X. At this point, an interesting study can be: integrating our risk-by-distance factor as an extensive parameter in DCC tunning parameters (e.g., power, transmission rate) to enhance the channel congestion in the various scenarios or for Vehicle-to-Infrastructure communications. However, a comprehensive evaluation of the positive/negative impact of the combination is critical. Second, the risk assessment in our method can be enhanced with new assessment models or factors, e.g., with assists of cameras or period feedback from neighbor vehicles. Finally, building the groups of vehicles in large-scale traffic cases and then applying hierarchy-based congestion control strategies for each group can be a potential direction for further conduct.

VI. CONCLUSION

In this work, we present an intelligent DDQN-based transmission control scheme, namely DEEPCUT, to dynamically adjust the broadcasting rate of beacon messages. DEEPCUT can evaluate the risks of surrounding vehicles and suggest reducing the transmission rate to the low-risk vehicles while increasing the channel access opportunities for the high-risk vehicles. The evaluation results show that our system can help cut up 16% redundant data while making no negative impact on overall packet delivery rate of the vehicles in the network. Besides, our method increases the packet reception rate up to 22%, indicating that more neighbor vehicles can receive data from the transmitter vehicles. In this way, our risk-based approach contributes a new method to deal with the channel congestion issue in crowding vehicular communications, besides using conventional congestion control mechanisms.

REFERENCES

[1] S. Kuhlmorgen, H. Lu, A. Festag, J. Kenney, S. Gemsheim, and G. Fettweis, "Evaluation of congestion-enabled forwarding with mixed data traffic in vehicular communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 233–247, Jan. 2020.

[2] G. Thandavarayan, M. Sepulcre, and J. Gozalvez, "Cooperative perception for connected and automated vehicles: Evaluation and impact of congestion control," *IEEE Access*, vol. 8, pp. 197665–197683, 2020.

[3] S. Husain, A. Kunz, A. Prasad, E. Pateromichelakis, K. Samdanis, and J. Song, "The road to 5G V2X: Ultra-high reliable communications," in *Proc. IEEE Conf. Standards Commun. Netw. (CSCN)*, Oct. 2018, pp. 1–6.

[4] H. Zhou, W. Xu, J. Chen, and W. Wang, "Evolutionary V2X technologies toward the Internet of Vehicles: Challenges and opportunities," *Proc. IEEE*, vol. 108, no. 2, pp. 308–323, Feb. 2020.

[5] *Study on Enhancement of 3GPP Support for 5G V2X Services*, 3GPP, document TR 22.886 V16.1.1, 2018.

[6] R. Liu, J. Wang, and B. Zhang, "High definition map for automated driving: Overview and analysis," *J. Navigat.*, vol. 73, no. 2, pp. 324–341, Mar. 2019.

[7] *V2X Functional and Performance Test Procedures—Selected Assessment of Device to Device Communication Aspects*, 5G Automot. Assoc., Munich, Germany, 2018.

[8] *Study on LTE-Based V2X Services (v14.0.0, Release 14)*, 3GPP, Technical Specification Group Radio Access Network, document 4, 2016.

[9] P. Sewalkar and J. Seitz, "MC-COCO4V2P: Multi-channel clustering-based congestion control for vehicle-to-pedestrian communication," *IEEE Trans. Intell. Vehicles*, vol. 6, no. 3, pp. 523–532, Sep. 2021.

[10] N. Lyamin, B. Bellalta, and A. Vinel, "Age-of-information-aware decentralized congestion control in VANETs," *IEEE Netw. Lett.*, vol. 2, no. 1, pp. 33–37, Mar. 2020.

[11] C. B. Math, H. Li, S. H. de Groot, and I. G. Niemegeers, "V2X application-reliability analysis of data-rate and message-rate congestion control algorithms," *IEEE Commun. Lett.*, vol. 21, no. 6, pp. 1285–1288, Jun. 2017.

[12] *Intelligent Transport Systems (ITS); Decentralized Congestion Control Mechanisms for Intelligent Transport Systems Operating in the 5 GHz Range; Access Layer Part*, ETSI, Standard ETSI TS 102 687 V1.2.1, 2018.

[13] *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, ETSI, Standard ETSI-EN 302 637-2, 2014.

[14] G. Bansal, J. B. Kenney, and C. E. Rohrs, "LIMERIC: A linear adaptive message rate algorithm for DSRC congestion control," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, pp. 4182–4197, Nov. 2013.

[15] A. Balador, E. Cinque, M. Pratesi, F. Valentini, C. Bai, A. A. Gómez, and M. Mohammadi, "Survey on decentralized congestion control methods for vehicular communication," *Veh. Commun.*, vol. 33, Jan. 2022, Art. no. 100394.

[16] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for V2V communications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3163–3173, Apr. 2019.

[17] J.-Y. Choi, H.-S. Jo, C. Mun, and J.-G. Yook, "Deep reinforcement learning-based distributed congestion control in cellular V2X networks," *IEEE Wireless Commun. Lett.*, vol. 10, no. 11, pp. 2582–2586, Nov. 2021.

[18] *Study on Evaluation Methodology of New Vehicle-to-Everything (V2X) use Cases for LTE and NR (Release 15)*, 3GPP, document TR 37.885, 2019.

[19] A. Gündoğan, H. M. Gürsu, V. Pauli, and W. Kellerer, "Distributed resource allocation with multi-agent deep reinforcement learning for 5G-V2 V communication," in *Proc. 21st Int. Symp. Theory, Algorithmic Found., Protocol Design Mobile Netw. Mobile Comput.* New York, NY, USA: Assoc. Comput. Machinery, Oct. 2020, pp. 357–362.

[20] S. Rene, O. Ascigil, I. Psaras, and G. Pavlou, "A congestion control framework based on in-network resource pooling," *IEEE/ACM Trans. Netw.*, vol. 30, no. 2, pp. 683–697, Apr. 2022.

[21] V. Mannoni, V. Berg, S. Sesia, and E. Perraud, "A comparison of the V2X communication systems: ITS-G5 and C-V2X," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–5.

[22] T. Shimizu, B. Cheng, H. Lu, and J. Kenney, "Comparative analysis of DSRC and LTE-V2X PC5 mode 4 with SAE congestion control," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Dec. 2020, pp. 1–8.

[23] B. Choudhury, V. K. Shah, A. Dayal, and J. H. Reed, "Joint age of information and self risk assessment for safer 802.11 p based V2 V networks," in *Proc. IEEE Conf. Comput. Commun.*, May 2021, pp. 1–10.

[24] L.-H. Nguyen, V.-L. Nguyen, and J.-J. Kuo, "Risk-based transmission control for mitigating network congestion in vehicle-to- everything communications," *IEEE Access*, vol. 9, pp. 144469–144480, 2021.

[25] A. Mekrache, A. Bradai, E. Moulay, and S. Dawaliby, "Deep reinforcement learning techniques for vehicular networks: Recent advances and future trends towards 6G," *Veh. Commun.*, vol. 33, Jan. 2022, Art. no. 100398.

[26] FCC. (2021). *Use of the 5.850–5.925 GHz Band—Final Rule*. [Online]. Available: https://www.govinfo.gov/content/pkg/FR-2021-05-03/pdf/2021-08802.pdf

[27] Qualcomm. (2021). *C-V2X Congestion Control Study*. [Online]. Available: https://www.qualcomm.com/media/documents/files/c-v2x-congestion-control-study.pdf

[28] *Intelligent Transport Systems (ITS); Access Layer Part; Congestion Control for the Cellular: V2X PC5 Interface*, ETSI, Standard ETSI TS 103 574 V0.3.1, 2018.

[29] B. McCarthy and A. O'Driscoll, "Congestion control in the cellular-V2X sidelink," 2021, *arXiv:2106.04871*.

[30] W. Abera, T. Olwal, Y. Marye, and A. Abebe, "Learning based access class barring for massive machine type communication random access congestion control in LTE—A networks," in *Proc. Int. Conf. Electr., Comput. Energy Technol. (ICECET)*, Dec. 2021, pp. 1–7.

[31] Z. Lu, C. Zhong, and M. C. Gursoy, "Dynamic channel access and power control in wireless interference networks via multi-agent deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 71, no. 2, pp. 1588–1601, Feb. 2022.

[32] J. Aznar-Poveda, A.-J. Garcia-Sanchez, E. Egea-Lopez, and J. Garcia-Haro, "Simultaneous data rate and transmission power adaptation in V2 V communications: A deep reinforcement learning approach," *IEEE Access*, vol. 9, pp. 122067–122081, 2021.

[33] M. Chen, R. Li, J. Crowcroft, J. Wu, Z. Zhao, and H. Zhang, "RAN information-assisted TCP congestion control using deep reinforcement learning with reward redistribution," *IEEE Trans. Commun.*, vol. 70, no. 1, pp. 215–230, Jan. 2022.

[34] H. Ma, D. Xu, Y. Dai, and Q. Dong, "An intelligent scheme for congestion control: When active queue management meets deep reinforcement learning," *Comput. Netw.*, vol. 200, Dec. 2021, Art. no. 108515.

[35] M. Roshdi, S. Bhadauria, K. Hassan, and G. Fischer, "Deep reinforcement learning based congestion control for V2X communication," in *Proc. IEEE 32nd Annu. Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2021, pp. 1–6.

[36] *Specification of Cooperative Awareness Basic Service*, ETSI, Standard ETSI EN 302 637-2 V1.4.0, 2018.

[37] M. Boban, X. Gong, and W. Xu, "Modeling the evolution of line-of-sight blockage for V2 V channels," in *Proc. IEEE 84th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2016, pp. 1–7.

[38] *Study on Channel Model for Frequencies From 0.5 to 100 GHz*, 3GPP, document ETSI TR 138 901, 2019.

[39] S.-W. Ko, H. Chae, K. Han, S. Lee, D.-W. Seo, and K. Huang, "V2X-based vehicular positioning: Opportunities, challenges, and future directions," *IEEE Wireless Commun.*, vol. 28, no. 2, pp. 144–151, Dec. 2021.

[40] B. McCarthy and A. O'Driscoll, "Congestion control in the cellular-V2X sidelink," 2021, *arXiv:2106.04871*.

[41] H. V. Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. 30th AAAI Conf. Artif. Intell.* Menlo Park, CA, USA: AAAI Press, 2016, pp. 2094–2100.

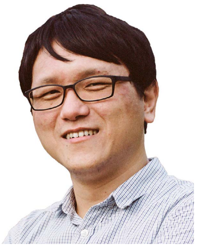
[42] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," 2015, *arXiv:1511.05952*.

[43] B. McCarthy, A. Burbano-Abril, V. R. Licea, and A. O'Driscoll, "OpenCV2X: Modelling of the V2X cellular sidelink and performance evaluation for aperiodic traffic," 2021, *arXiv:2103.13212*.

[44] L. Codecà, R. Frank, S. Faye, and T. Engel, ''Luxembourg SUMO traffic (LuST) scenario: Traffic demand evaluation,'' *IEEE Intell. Transp. Syst. Mag.*, vol. 9, no. 2, pp. 52–63, May 2017.

LAN-HUONG NGUYEN received the M.Sc. degree in computer science from the VNU University of Engineering and Technology, Vietnam, in 2016. She is currently pursuing the Ph.D. degree in computer science and information engineering with the National Chung Cheng University, Chiayi, Taiwan. Her research interests include vehicular networks, mobile edge computing, and network optimization.

VAN-LINH NGUYEN (Member, IEEE) received the Ph.D. degree in computer science and information engineering from the National Chung Cheng University (CCU), Taiwan, in 2019. He is currently working as a Postdoctoral Fellow. He is also an Assistant Professor with the Thai Nguyen University of Information and Communication Technology (TNU-ICTU), Vietnam. His research interests include network intelligence, edge intelligence, cyber security, and autonomous driving.



JIAN-JHIH KUO (Member, IEEE) received the Ph.D. degree in computer science from the National Tsing Hua University, Taiwan, in 2014. He is currently an Assistant Professor with the Department of Computer Science and Information Engineering, National Chung Cheng University, Taiwan. He was a Postdoctoral Fellow with the Institute of Information Science, Academia Sinica, Taiwan. His research interests include mobile edge computing, distributed computing, and cloud computing.