Telecommunication Systems
https://doi.org/10.1007/s11235-024-01188-5

A dynamic spectrum access algorithm based on deep reinforcement learning with novel multi-vehicle reward functions in cognitive vehicular networks

Lingling Chen[1,2] · Ziwei Wang[1] · Xiaohui Zhao[3] · Xuan Shen[1] · Wei He[1]

Accepted: 11 June 2024

Abstract
As a revolution in the field of transportation, the demand for communication of vehicles is increasing. Therefore, how to improve the success rate of vehicle spectrum access has become a major problem to be solved. The case of a single vehicle accessing a channel was only considered in the previous research on dynamic spectrum access in cognitive vehicular networks (CVNs), and the spectrum resources could not be fully utilized. In order to fully utilize spectrum resources, a model for spectrum sharing among multiple secondary vehicles (SVs) and a primary vehicle (PV) is proposed. This model includes scenarios where multiple SVs share spectrum to maximize the average quality of service (QoS) for vehicles. And the condition is considered that the total interference generated by vehicles accessing the same channel is less than the interference threshold. In this paper, a deep Q-network method with a modified reward function (IDQN) algorithm is proposed to maximize the average QoS of PVs and SVs and improve spectrum utilization. The algorithm is designed with different reward functions according to the QoS of PVs and SVs under different situations. Finally, the proposed algorithm is compared with the deep Q-network (DQN) and Q-learning algorithms under the Python simulation platform. The average access success rate of SVs in the IDQN algorithm proposed can reach 98%, which is improved by 18% compared with the Q-learning algorithm. And the convergence speed is 62.5% faster than the DQN algorithm. At the same time, the average QoS of PVs and the average QoS of SVs in the IDQN algorithm can reach 2.4, which is improved by 50% and 33% compared with the DQN algorithm, and improved by 60% and 140% compared with the Q-learning algorithm.

Keywords Spectrum access · Cognitive vehicular networks · Deep reinforcement learning (DRL) · Quality of service (QoS)

1 Introduction

With the emergence and implementation of large-scale Internet of things (IoT) [1], Internet of vehicles (IoV) [2] and other technologies, a wider range of resources need to be occupied by wireless communication services [3]. Although the static spectrum allocation method can effectively avoid the conflict and interference between different wireless services, it cannot fully exploit the distribution characteristics of radio signals in the time domain, frequency domain and air domain. Dynamic spectrum access (DSA) [4] is a key technology to realize the effective use of spectrum resources in cognitive radio (CR) [5], which can solve the above problems. In DSA technology, secondary user (SU) is allowed to access the current frequency band without affecting the quality of service (QoS) [6] of the normal communication of the primary user (PU). The spectrum hole of the current frequency band is perceived by the SU and is accessed by the

✉ Lingling Chen
cll807900@163.com

Ziwei Wang
2239626672@qq.com

Xiaohui Zhao
xhzhao@jlu.edu.cn

Xuan Shen
sx_15195855815@163.com

Wei He
1738739555@qq.com

[1] College of Information and Control Engineering, Jilin Institute of Chemical Technology, Jilin 132000, China

[2] College of Communication Engineering, Jilin University, Changchun 130012, China

[3] Key Laboratory of Information Science, College of Communication Engineering, Jilin University, Changchun 130012, China

Published online: 28 June 2024

SU changing the access parameters under certain conditions by DSA technology, so that the frequency band utilization can be improved when the number of users is greater than the number of channels [7].

Efficiently resolving the problem of lacking spectrum resources caused by the explosive growth of vehicle communication services has become a matter of significant concern. In recent years, cognitive vehicular networks (CVNs) [8] have emerged as a promising solution, combining cognitive radio (CR) technology with vehicular networks [9]. While progress has been made in spectrum sensing technology for the cognitive Internet of vehicles (CIoV), research on spectrum access technology is still in the developmental stage, drawing attention from numerous scholars [10]. The fundamental idea behind CIoV's spectrum access is to detect the state of the PU spectrum and enable cognitive radios (CRs) to opportunistically utilize idle portions of the PU spectrum to improve overall spectrum utilization [11, 12]. As the PU's spectrum state dynamically changes, the spectrum access of CIoV, based on CR technology, must adapt accordingly to avoid interfering with normal communication among primary vehicles (PVs) [13]. This adaptability is termed dynamic spectrum access (DSA) of CIoV. DSA allows secondary vehicles (SVs) in CIoV to access available spectrum in a timely manner. Idle spectrum can be efficiently occupied by CIoV through spectrum coverage, while busy spectrum can be accessed with limited interference power through spectrum superposition [14]. Consequently, CIoV's dynamic spectrum access effectively addresses the problem of insufficient spectrum resources in IoV communication [15] To achieve spectrum efficiency in CIoV, it is crucial to study dynamic spectrum access to adapt to varying spectrum states and sensing results of the PU [16].

2 Related work

As dynamic spectrum access progressively becomes a challenge in extensive state spaces, the complex computational issues stemming from large-scale state and action domains are effectively addressed by deep reinforcement learning (DRL) [17] agents. These agents obtain eigenvalues from high-latitude raw data to formulate optimal action strategies [18]. Consequently, within the framework of spectrum resource allocation following Markov decision processes (MDP), DRL has been widely adopted to address issues such as low spectrum access rates [19]. Among these efforts, an innovative resource allocation algorithm based on multi-agent reinforcement learning (MARL) was introduced in [20] and [21]. This algorithm aims to improve data packet reception rates and tackle decentralized radio resource management challenges within 5G vehicle networks. Employing actor-critic RL techniques, the algorithm facilitates optimal time budget (TB) selection by individual agents. Furthermore, a centralized training mechanism enables the sharing of observations across agents, effectively mitigating non-stationarity in the multi-agent environment. In contrast to conventional approaches, this algorithm enhances packet reception rates by 18% and reward rates by 33% when compared to the advanced MARL-based method [21].

Additionally, [22] introduces a hybrid strategy aimed at maximizing network utility through efficient dynamic spectrum access. A distributed deep reinforcement learning (DRL)-based scheme is proposed, leveraging a deep recursive reinforcement learning network with an integrated gated recurrent unit (GRU) layer to optimize the network utility function.

Furthermore, [23] proposes a distributed dynamic power allocation scheme based on multi-agent deep reinforcement learning (MADRL). This scheme caters to computational complexity and instantaneous cross-channel state information (CSI) demands. Each transmitter collects CSI and quality of service information from multiple neighbors, thereby facilitating individual transmit power adjustments. To enhance the service quality of users in licensed networks while minimizing interference, [24] suggests a distributed dynamic spectrum access communication framework based on MARL. In this approach, multiple units in a multi-user multiple-input multiple-output (MU-MIMO) network act as proxies, utilizing the average Signal-to-Interference plus Noise Ratio (SINR) value as a reward to maximize average SINR.

Some studies emphasize optimizing throughput [25–29]. For example, [30] proposes a QoS-aware decentralized resource allocation approach for vehicle-to-everything (V2X) communication. Employing DRL, this approach aims to maximize throughput, resulting in improved system and device-to-device (D2D) throughput. The challenges posed by severe interference between D2D and cellular users motivate comprehensive considerations in [30]. The proposed spectrum allocation scheme is built on distributed learning, allowing D2D users to independently select spectrum resources. This decision-making process focuses on maximizing D2D user throughput while minimizing interference with cellular users. While existing studies demonstrate the efficacy of the DRL algorithm in dynamic spectrum allocation, many efforts focus on model improvement or optimization targets. In contrast, [31] introduces a shared model and optimizes both throughput and link payload transfer rates, addressing spectrum sharing in vehicular networks based on MARL. By modeling resource sharing as a MARL problem, multiple vehicle-to-vehicle (V2V) links can reuse the spectrum occupied by vehicle-to-infrastructure (V2I) links. The approach leverages a fingerprint-based deep Q-network (DQN) for solving this challenge, effectively learning a distributed cooperative strategy among multiple V2V agents.

This approach results in enhanced V2I link capacity and V2V link payload transfer rates.

Nonetheless, certain shortcomings remain in the sharing model outlined in previous studies. For instance, the limitation of allowing only one secondary vehicle (SV) to access the spectrum when occupied by a primary vehicle (PV) [32] fails to consider the QoS for PVs and SVs. The advantages and disadvantages of related literature are compared as shown in Table 1. Some literature does not consider shared spectrum nor QoS of vehicles. Some literature only considers a single SV and a PV sharing model and only considers the QoS of secondary vehicles. To address these limitations and enhance vehicle QoS, this paper refines and expands the system model proposed in [31]. It permits multiple SVs to access the same spectrum, while also comprehensively considering the QoS of PVs and SVs. QoS in this context primarily comprises throughput and transmission delay. This research introduces a dynamic spectrum sharing scheme for cognitive radio-enabled vehicular ad hoc networks (CR-VANETs) based on multi-vehicle DRL. This innovative approach allows multiple SVs to access the same channel, significantly contributing to efficient spectrum utilization. In summary, the paper's key contributions are as follows, and the explanation of terms is shown in Table 2.

(1) First, compared with fewer SVs, considering that more SVs access the same spectrum can make full use of existing spectrum resources. Secondly, as there are more and more vehicles and spectrum resources are limited, it is necessary to consider that as many vehicles as possible can communicate from the perspective of shared spectrum. Model the spectrum access of multiple SVs as a cognitive radio-enabled vehicular ad hoc networks (CR-VANETs) spectrum sharing problem, permitting multiple SVs to share the same channel by setting an interference threshold to enhance spectrum utilization.

(2) Design separate QoS functions for PVs and SVs, ensuring communication quality in a spectrum-sharing context. This involves adjusting the SVs' spectrum selection strategy to maximize QoS. The average QoS of PVs and the average QoS of SVs in a deep Q-network method with a modified reward function (IDQN) algorithm can reach 2.4, which is improved by 50% and 33% compared with the DQN algorithm, and improved by 60% and 140% compared with the Q-learning algorithm.

(3) Improve the reward function based on the QoS functions of PVs and SVs to better integrate the proposed algorithm with the model. The average access success rate of SVs in the IDQN algorithm proposed can reach 98%, which is improved by 18% compared with the Q-learning algorithm. And the convergence speed of IDQN is 62.5% faster than the DQN algorithm.

The rest of this paper is organized as follows. The system model is detailed in Sect. 3. An IDQN-based spectrum access algorithm for CR-VANETs is proposed by us in Sect. 4. Our experimental results are given in Sect. 5. Conclusions are made in Sect. 6. And a final extensions is made in Sect. 7.

Table 1 Comparison of relevant literature

References	Advantages	Disadvantages
[22]	Maximize network utility	No consideration of QoS
[23]	Considered QoS of SVs	Only the QoS of SVs is considered
[24]	Considered SINR	No consideration of QoS
[25]	Taking into account the throughput between D2D	Only the QoS of D2D users is considered
[30]	Consider D2D user throughput and interference	Only the QoS of D2D users is considered
[31]	Introduced sharing model and optimized throughput and link payload transfer rate	No consideration of QoS
[32]	Shared models considered	Only a model shared between SVs and PVs was considered, and QoS was not considered

3 System model

Previous studies on spectrum sharing for CR-VANETs have mostly focused on scenarios involving a single PV sharing a channel with a single SV or two SVs sharing a channel, leading to suboptimal spectrum resource utilization. To address this limitation, our study considers scenarios where PVs share channels with multiple SVs share channels. Figure 1 illustrates a CR-VANETs environment comprising M PVs, M channels, N SVs, and one base station (BS). Figure 1 shows two communication scenarios sharing spectrum. When PVs occupy spectrum for communication, SVs and PVs accessing the spectrum interfere with each other in a scenario where a PV shares a channel with multiple SVs. And at the same time, interference also occurs between SVs. When PVs do not occupy the spectrum, interference occurs between SVs accessing the spectrum in a scenario where multiple SVs share a channel. It is necessary to limit mutual interference to ensure the communication quality of PVs and SVs. Our analysis assumes the use of a single antenna for all transceivers in the CR-VANETs environment. Additionally, we presume that the spectrum with fixed transmission power has already been pre-allocated by M PVs, where the mth PV occupies the mth channel. In this network, both PVs and SVs have the capability to communicate with the BS. In order to enhance spectrum utilization, SVs are allowed to utilize authorized channels, enabling them to share resources with

Table 2 Explanation of terms

Abbreviation of terms	Definition of terms	Full name of term
DSA	Dynamic spectrum access is a technique by which a radio system dynamically adapts to the local radio spectrum environment by determining the spectrum available at specific locations and at a specific time	Dynamic spectrum access
CR	Cognitive radio technology studies how secondary users with sensing capabilities can intelligently utilize spectrum holes without interfering with the communication quality of primary users	Cognitive radio
PU	In cognitive radio communication systems, users who have priority to use a certain fixed frequency band are defined as Primary users	Primary user
SU	In cognitive radio communication systems, users equipped with sensing devices are defined as secondary users	Secondary user
CVN	Introducing CR technology into vehicular network can effectively solve the problem of spectrum resource shortage in vehicular network	Cognitive vehicular networks
CIoV	Introducing CR technology into Internet of vehicles can effectively solve the problem of spectrum resource shortage	Cognitive Internet of vehicles
CR-VANETs	By applying cognitive radio technology to vehicle-mounted ad hoc networks, cognitive wireless vehicle-mounted ad hoc networks can alleviate the problem of spectrum resource scarcity and effectively improve the spectrum resource utilization of vehicle-to-vehicle communications	Cognitive radio-enabled vehicular ad hoc networks

PVs and achieve higher channel capacity. This paper focuses on ensuring optimal spectrum sharing for vehicular communication, particularly when the interference generated by SVs is lower than that produced by the PV. The paper is structured around the examination of two key scenarios as follows.

3.1 In a scenario where a PV shares a channel with multiple SVs

This scenario is shown in Fig. 2, where multiple SVs and PVs share the same channel. As can be seen from Fig. 2, when a PV occupies the mth channel, other SVs and PVs accessing the channel interfere with each other, and SVs also interfere with each other. If the interference generated by the SV is less than the maximum interference that the PV can withstand, the SV accesses the channel successfully, otherwise the access fails. First, to satisfy the interference without affecting the normal communication of PVs, multiple SVs can gain access to the PVs' channel, given that they adhere to the maximum allowable interference threshold $\xi_m[m]$. This criterion can be succinctly denoted using the following formula.

$$\sum_{n=1}^{N} I_n[m] \le \xi_m[m] \tag{1}$$

where $I_n[m]$ is defined as the interference to the PV generated by the n th SV accessing the m th channel. And it is specifically expressed as the following formula.

$$I_n[m] = \rho_n[m] P_n[m] H_n^{\sim}[m] \tag{2}$$

where $\rho_n[m]$ is the binary spectrum allocation indicator. $\rho_n[m]$ = 1 is defined as the authorized channel of the mth PV is occupied by the nth SV. Otherwise, $\rho_n[m]$ = 0. and are respectively the transmission power of the nth SV on the mth channel and the interference channel gain of the nth SV on the mth channel.

In order to satisfy the interference temperature constraints, the SINR of the PV in the mth channel and SVs in the mth channel can be expressed as (3) and (4).

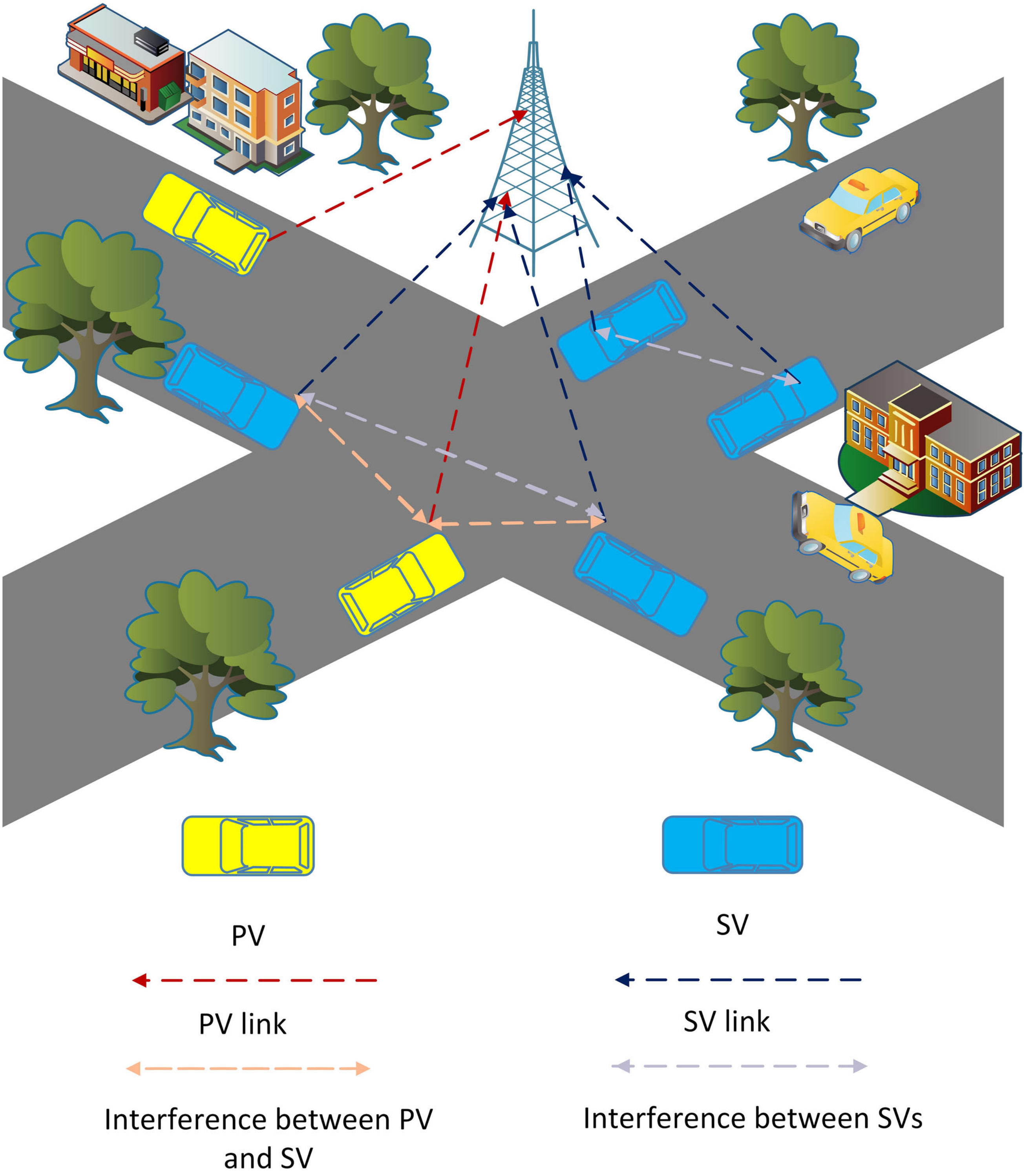


Fig. 1 CR-VANETs communication scenario

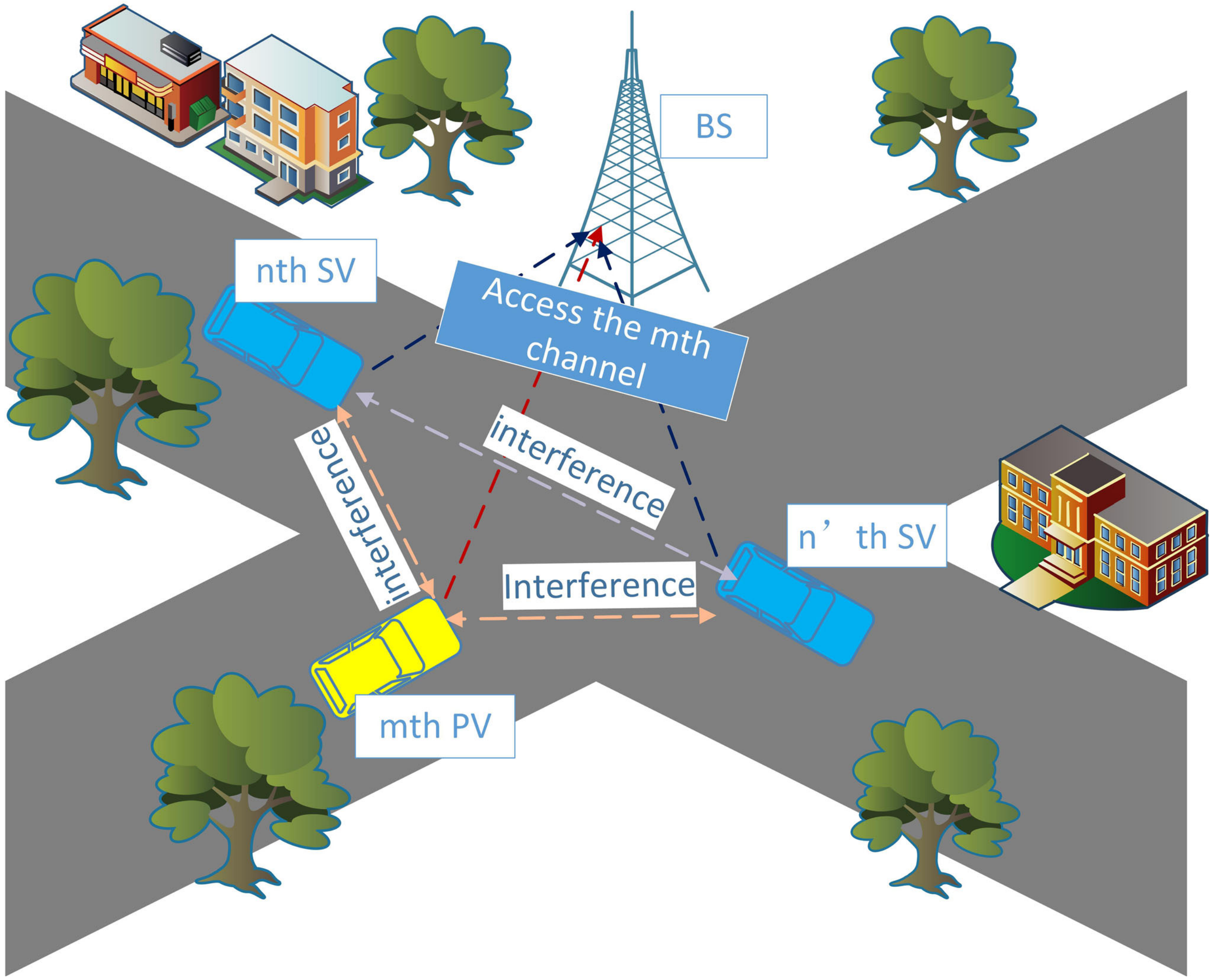


Fig. 2 A scenario where a PV shares a channel with multiple SVs

$$\gamma_m[m] = \frac{P_m[m]H_m[m]}{B_0 N_0 + \sum_{n=1}^{N} I_n[m]} \tag{3}$$

$$\gamma_n[m] = \frac{P_n[m]H_n[m]}{B_0 N_0 + I_{m,n}[m] + I_{n',n}[m]} \tag{4}$$

where $P_m[m]$ is denoted as the transmission power of the PV occupying the mth channel. $H_m[m]$ is denoted as the channel gain of the PV in the mth channel. B_0 is defined as the bandwidth of the channel. N_0 is defined as the power spectral density of the background noise in the channel. $H_n[m]$ is the channel gain of the nth SV accessing the mth channel. $I_{m,n}[m]$ represents the interference caused by the PV occupying the mth channel to the nth SV occupying the mth channel. And $I_{n',n}[m]$ is defined as the total interference of other SVs in the channel to the nth SV. The details are as the following formula.

$$I_{m,n}[m] = \rho_n[m]P_m[m]H_m^{\sim}[m] \tag{5}$$

$$I_{n',n}[m] = \sum_{n' \neq n} \rho_{n'}[m]\rho_n[m]P_{n'}[m]H_{n'}^{\sim}[m] \tag{6}$$

where $H_m^{\sim}[m]$ corresponds to the interference channel gain of the PV in the mth channel. $\rho_{n'}[m]$ is the binary spectrum allocation indicator. $\rho_{n'}[m] = 1$ is defined as the authorized channel of the mth PV is reused by the n'th SV. Otherwise, $\rho_{n'}[m] = 0$. $P_{n'}[m]$ is the transmission power of the n'th SV occupying the mth channel, and $H_{n'}^{\sim}[m]$ denotes the interference channel gain of the n'th SV occupying the mth channel.

Then, the throughput of the PV in the mth channel and SVs in the mth channel can be obtained according to Shannon's theorem. The specific formulas are as (7) and (8).

$$C_m[m] = B_0 \log_2(1 + \gamma_m[m]) \tag{7}$$

$$C_n[m] = B_0 \log_2(1 + \gamma_n[m]) \tag{8}$$

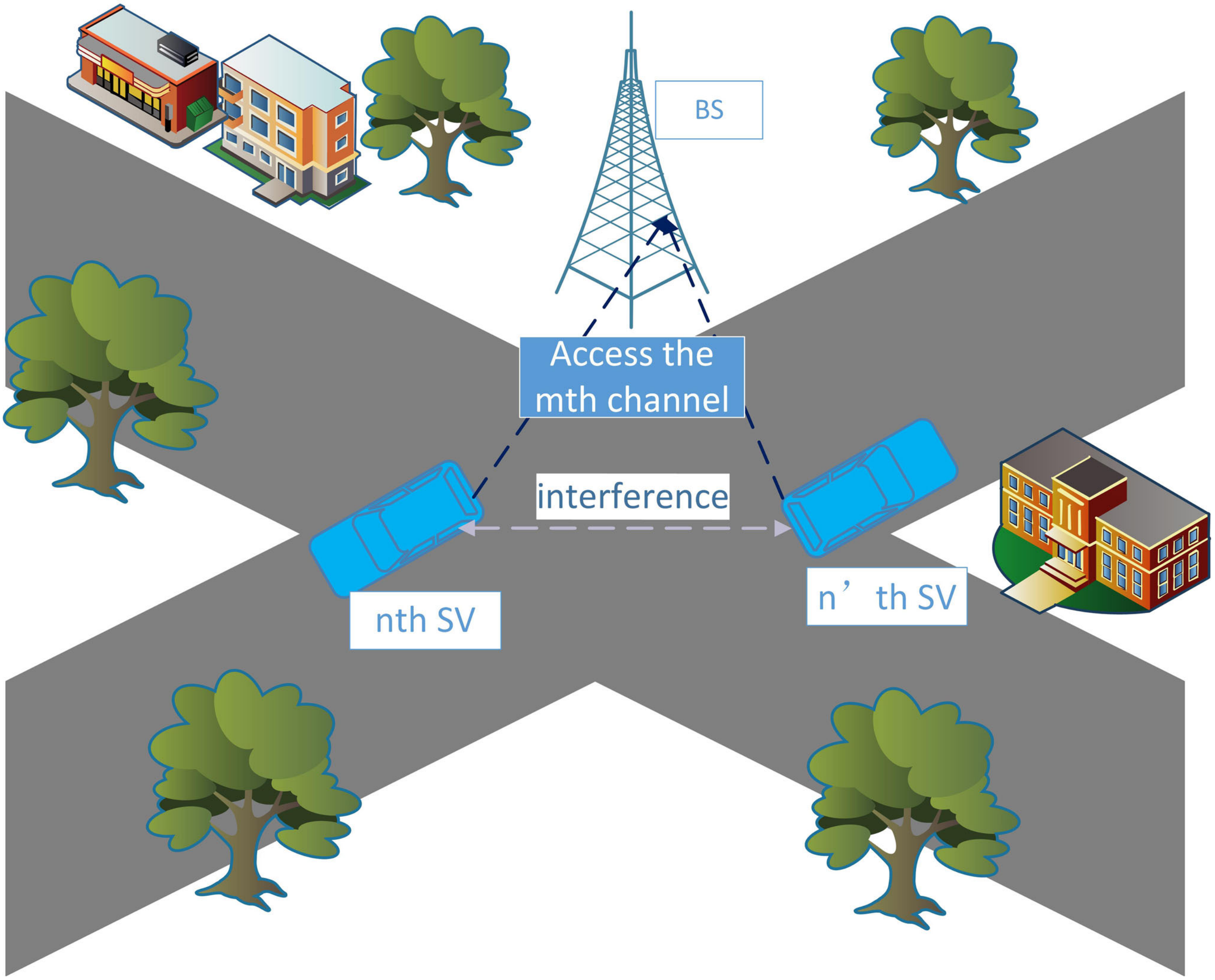


Fig. 3 A scenario where multiple SVs share a channel

Finally, in order to guarantee the QoS of the PV and SVs in this channel, the QoS function $QoS_m[m]$ of the PV and the QoS function $QoS_n[m]$ of SVs can be obtained according to (7) and (8). And they are composed of the transmission delay and throughput. The specific formulas are as (9) and (10).

$$QoS_m[m] = \theta \frac{E_m[m]}{R_m[m]} + \eta C_m[m] \tag{9}$$

$$QoS_n[m] = \theta \frac{E_n[m]}{R_n[m]} + \eta C_n[m] \tag{10}$$

where θ and η are defined as the preference parameters of the two terms, which are used to unify the units and balance the weights. $E_m[m]$ and $R_m[m]$ represent the load of the PV transmission data and data transmission rate in the mth channel, respectively. $E_n[m]$ and $R_n[m]$ represent the payload and data transmission rate of the nth SV transmission data in the mth channel, respectively.

3.2 In a scenario where multiple SVs share a channel

This scenario is shown in Fig. 3, where multiple SVs share the same channel. As can be seen from Fig. 3, multiple SVs interfere with each other when accessing the channel. If the interference generated by the SV is less than the maximum interference that the SV can withstand, the SV accesses the channel successfully, otherwise the access fails. Under the premise of satisfying the interference constraint, the SINR of accessing the nth SV in the mth channel can be expressed as the following formula.

$$\gamma_n^{'}[m] = \frac{P_n[m]H_n[m]}{B_0 N_0 + I_{n',n}[m]} \tag{11}$$

Multiple SVs can gain access to the PVs’ channel, given that they adhere to the maximum allowable interference threshold $\xi_n[m]$. This criterion can be succinctly denoted using the following formula.

$$\sum_{n=1}^{N} I_n^{'}[m] \le \xi_n[m] \tag{12}$$

Then, according to Shannon's theorem, the throughput of accessing the nth SV in the mth channel can be obtained as the following formula.

$$C_n^{'}[m] = B_0 \log_2(1 + \gamma_n^{'}[m]) \tag{13}$$

Finally, the QoS function $QoS_n^{'}[m]$ of the PV in this channel can be obtained as the following formula.

$$QoS_n^{'}[m] = \theta \frac{E_n[m]}{R_n[m]} + \eta C_n^{'}[m] \tag{14}$$

In this environment, the QoS functions of PVs and SVs are designed to ensure the communication quality of vehicles and improve the reliability of vehicle information transmission. the access failure rate of SVs can be reduced and the spectrum utilization can be improved by designing a proper reward function. Furthermore, the QoS of PVs and SVs can be maximized of the method proposed in this paper. More details are discussed in Sect. 4.

4 Spectrum access based on improved DRL

In the depicted VANETs spectrum access scenario illustrated in Fig. 1, numerous SVs endeavor to utilize the constrained spectrum resources. This intricate process can be modeled as a multi-vehicle deep reinforcement learning (DRL) problem. DRL, a fusion of deep learning (DL) and reinforcement learning (RL), offers a potent framework for effective implementation of the spectrum resource allocation mechanism. DL is adept at resolving the modeling problem between value functions and policies, while RL is used to effectively define problem and optimizes objectives. As can be seen from Fig. 4, reinforcement learning mainly consists of environment, agent, current state, action selection, reward value and new state. The specific learning process can be expressed as follows.

First, in the current system environment, the agent perceives the state of the current environment and obtains all sensing results as a state set. Secondly, after obtaining the sensing state, the agent selects actions based on the action selection strategy. Then, after the action is selected, the state of the environment changes. At the same time, the environment also feeds back a reward to the agent based on the action chosen by the agent as a way to judge the quality of the action. And the environment state changes from the current state to the new state. Finally, after receiving the reward, the agent combines the state and action to iteratively update the action selection strategy. The multi-vehicle spectrum access problem may be seen as a competitive game. But, it is transformed into a mutual cooperative problem by designing different rewards for all SVs in different situations for the overall network performance. Each SV acquires experiences through interactions within unknown communication environments, with channel selection being influenced by these acquired experiences.

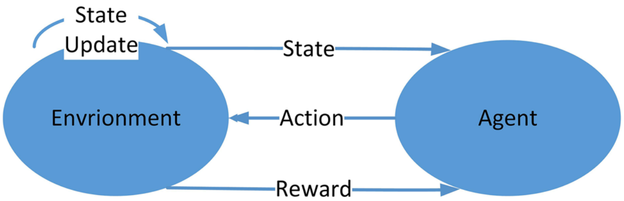


Fig. 4 The learning process of RL

This section unfolds in two distinct segments. The first part proposed the improved reinforcement learning (IRL) algorithm and designed the foundational elements of the model. This encompasses defining the state and action spaces, as well as formulating a novel reward function. The design of this reward function is primarily rooted in the objective function of QoS established within this paper. In the second part, the focus shifts to introduce the deep Q-learning algorithm. This involves solving the mapping relationship between observations and value functions, and deducing an optimization strategy.

4.1 Improved reinforcement learning

As illustrated in Fig. 5, the RL framework comprises both the vehicle and the environment, which can dynamic interact with each other. Throughout this interaction process, at time t, the channel occupancy state in the CR-VANETs environment is denoted by s_t^n and is observed by the nth SV. And a channel is selected for access represented a_t^n. SVs can continuously learn according to the environment, the optimal action is selected, and finally the dynamic spectrum access of the CR-VANETs is completed. The pivotal components of the IRL model are outlined below.

4.1.1 State and state space

For CR-VANETs spectrum access, the agent can perceive the external environment and generate its state based on its interactions with the environment. This is specifically expressed as follows.

$$s_t = \{I_t[m], H_t[m], T_t[m], E_t[m]\}_{m \in M} \tag{15}$$

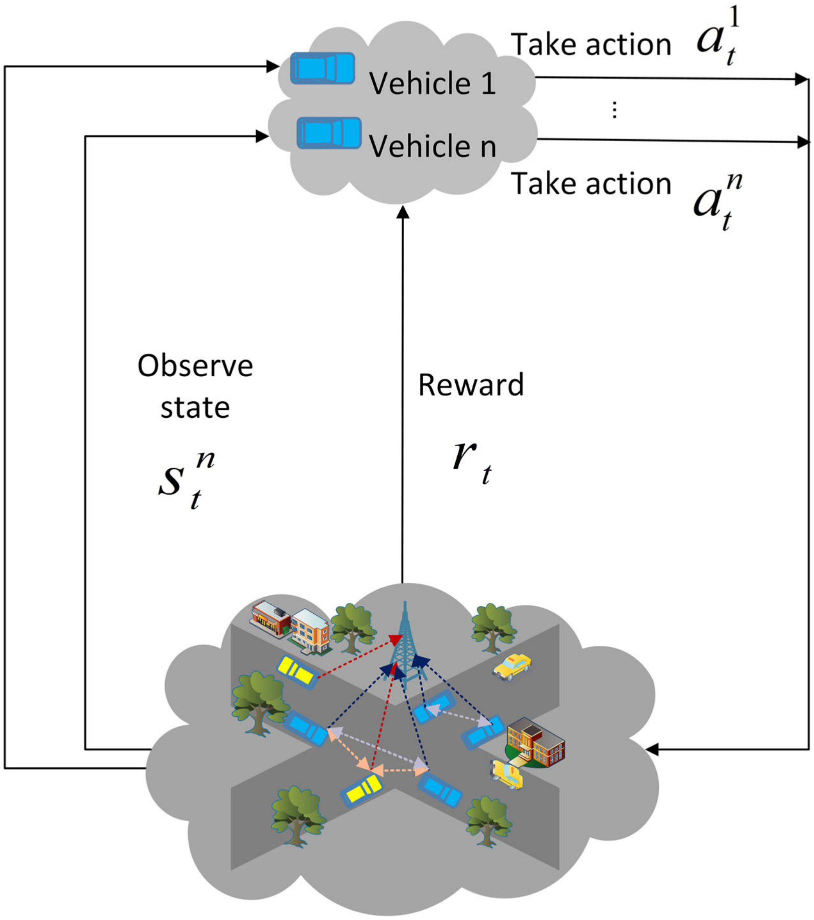


Fig. 5 Vehicle-environment interaction for RL

where $I_t[m] = \left\{I_n[m], I_{m,n}[m], I_{n',n}[m]\right\}$ represents the total interference generated in the mth channel at time t. $H_t = \left\{H_m[m], H_{\tilde{m}}[m], H_n[m], H_{\tilde{n}}[m], H_{\tilde{n}'}[m]\right\}_{m \in M}$ represents the channel gain at time t. $T_t[m]$ and $E_t[m] = \{E_m[m], E_n[m]\}$ represent the duration and load of SVs' transmission information at time t, respectively. At different time periods, different states are observed by each SV. And the state space S is composed of all possible states. The size of the state space is $|S| = M * N$.

4.1.2 Set of action

In the CR-VANETs, a channel is chosen to be accessed by the SV at time slot t. The state of the environment is observed by the SV, and the action at $a_t \in A$ is chosen according to policy π. Policy π is defined as the mapping function from the state space S to the action space A, and the action selection is determined by the policy π in each state. And a_t represents the selected channels, which can be expressed as follows.

$$a_t = m \tag{16}$$

where $m \in \{0, 1, 2, \ldots m, \ldots, M\}$, $m = 0$ means that no channel is selected. After the complexity of DQN and the needs of SVs are comprehensively considered, the case where the vehicle does not take any action must be considered. In different time periods, each SV takes different actions, and the action space A is composed of all possible actions. Therefore, the size of the action space $|A|$ (i.e., the number of different actions) can be expressed as follows.

$$|A| = M + 1 \tag{17}$$

After an agent takes an action, the environment is affected by that action. The state of the environment changes from s_t to s_{t+1}, and an immediate reward r_{t+1} is fed back to the agent. The details are as the following formula.

$$(s_{t+1}, a_{t+1}, r_{t+2}, s_{t+2}) \longleftarrow (s_t, a_t, r_{t+1}, s_{t+1}) \tag{18}$$

4.1.3 Improved reward function

To identify the impact of selected actions on the system, the reward function is defined as the weighted QoS of PVs and SVs by us. Not just QoS or throughput related to agent. Since the algorithm needs to be combined with the model, the performance of vehicle communication must be linked to the algorithm. Because the paper is to ensure the communication service quality of the vehicle, the QoS of the vehicle is set as the reward value of the algorithm. In this way, the QoS of the vehicle is positively related to the reward value. Therefore, the immediate reward r_t are denoted as r_{t1}, r_{t2} and r_{t3} respectively. The details are expressed as the following two scenarios.

(1) In a scenario where a PV shares a channel with multiple SVs, the reward function obtained by choosing SVs with different actions are expressed as a, b and c below.

a. The QoS of both PVs and SVs needs to be considered in this scenario. When the interference generated by the accessed SVs to the PV is less than the interference threshold, the SVs are successfully accessed. The reward function at time t is set as the following formula.

$$r_{t1} = \lambda QoS_m^t[m] + (1 - \lambda) QoS_n^t[m] \tag{19}$$

where $\lambda \in [0, 1]$ is the weight factor. QoS_m^t $[m]$ represents the QoS of the PV in the mth channel at time slot t. And QoS_m^t $[m]$ represents the QoS of SVs occupying the mth channel at time slot t.

b. When the interference generated by the accessed SVs to the PV exceeds the interference threshold, the SVs are access failed. Setting the reward value to a negative value is equivalent to setting a penalty for SVs. The reward function is set as the following formula.

$$r_{t2} = -\left\{\lambda QoS_m^t[m] + (1 - \lambda) QoS_n^t[m]\right\} \tag{20}$$

c. If no channel is selected by SVs, the reward function is set as the following formula.

$$r_{t3} = 0 \tag{21}$$

(2) In a scenario where multiple SVs share a channel, the reward function obtained by choosing SVs with different actions are expressed as d, e and f below.

d. Because the channel is not occupied by PVs, only the QoS of SVs needs to be considered in this scenario. When the interference generated by the accessed SVs to the SV is less than the interference threshold, the SVs are successfully accessed. The reward function at time slot t is set as the following formula.

$$r_{t1} = QoS_n^t[m] \tag{22}$$

e. When the interference generated by the accessed SVs to the SV exceeds the interference threshold, the SVs are access failed. Setting the reward value to a negative value is equivalent to setting a penalty for SVs. The reward function is set as the following formula.

$$r_{t2} = -QoS_n^t[m] \tag{23}$$

f. If no channel is selected by SV, the reward function is set as the following formula.

$$r_{t3} = 0 \tag{24}$$

In the realm of RL, it becomes imperative to account not only for immediate rewards but also for the long-term average cumulative rewards. The average cumulative rewards are defined as the cumulative sum of rewards earned by all vehicles over a significant period of time. This consideration holds particular significance within the CR-VANETs context, where sustaining stable long-term rewards is instrumental in securing the enduring success rate of secondary vehicle access. Therefore, the ultimate goal of this paper is to obtain long-term cognitive vehicle success access rates based on the average cumulative rewards of optimized SVs. However, within the CR-VANETs scenario, the absence of a definitive final state within the environment results in the total reward stretching to infinity. To address this quandary, we introduce a discount factor γ, which allows us to regulate the weighting of long-term rewards. This leads us to define the average cumulative rewards $\Re_n$ as the following formula.

$$\Re_n = \sum_{t=1}^{\infty} \gamma^{t-1} r_{t+1}^n, \, (r_{t+1} \in r_{t1}, r_{t2}, r_{t3}) \tag{25}$$

where r_{t+1}^n represents the reward value of the nth SV at different time $t + 1$, $\gamma \in [0, 1]$. When γ is close to 1, long-term rewards are considered more important by the agent. And when γ is close to 0, the current reward becomes more important. The performance of the system is controlled in RL by designing a reward function, and learning a policy to maximize the expected discounted reward is regarded as the goal.

4.2 Deep Q-learning algorithm

Many effective algorithms have been proposed to achieve the goal of RL, and the Q-learning algorithm is currently one of the commonly used algorithms. The problem of channel selection for CR-VANETs is solved using Q-learning. However, if there are too many vehicles and channels, the vehicle using Q-learning may not be able to select the optimal action. The Q-value table for Q-learning is replaced by the neural network in this section. The strategy π is optimized by Q-learning using the Q-value. The Q-value is closely related to the observed state s_t of the vehicle and the selected channel action a_t, denoted as $Q(s_t, a_t)$. It can be approximated as the expected total rewards for an SV choosing an action in state

s_t. The action with the largest Q-value is chosen to update the policy π by the SV. The Q-value is then updated using the new policy. And this process is repeated until the Q-value converges to the optimal Q-value Q^*. Once Q^* is obtained, the optimal strategy π^* can be found. The iterative formula of the Q-value is defined as the following formula.

$$Q(s_t, a_t) \longleftarrow Q(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \tag{26}$$

where α is denoted as the learning rate. In Q-learning, Q-values are stored in a Q-value table, and the size of the Q-value table is $|A|^{|S|}$. As the state-action space increases, the size of the Q-value table also increase significantly. In the spectrum sharing problem of SVs communication, the state space $|S|$ is large and uncertain. So, classical Q-learning cannot be applied to solve the problem in the proposed CR-VANETs environment. This problem can be effectively solved by using a neural network. The neural network consists of an input layer, a hidden layer, and an output layer, as shown in Fig. 6. In CR-VANETs, the observed state of SVs is considered as the input to the neural network, and the Q-value of each action is outputted from the output layer of the neural network through the hidden layer of the neural network. s_t^m represents the state of the mth channel observed by SVs at time t, and a_n represents the nth action taken. Q-value tables are replaced by neural networks that can be called Q-networks.

In the spectrum access problem for CR-VANETs, actions $a_t \in A$ are discrete and finite. The network structure of the deep Q-network is shown in Fig. 6. The output of the Q-network can be expressed as the following formula.

$$Q_\phi(s_t) = \begin{bmatrix} Q_\phi(s_t, a_t^1) \\ \vdots \\ Q_\phi(s_t, a_t^n) \end{bmatrix} \tag{27}$$

where ϕ is denoted as the weight value in the Q-network. $Q_\phi(s_t)$ refers to the Q-value output by the Q-network whose weight is ϕ, and a_t^n refers to the action taken by the nth SV. The Q-network is trained to ensure that $Q_\phi(s_t)$ is close to the true Q-value. And it is guaranteed to be close to the real Q-value through learning. There are two problems in the learning process. One reason is that the target is unstable, and the target of parameter learning depends on the parameters themselves. The other reason is that there is a strong correlation between training samples. Therefore, the DQN algorithm needs to be adopted to solve these two problems. In order to solve the spectrum access problem of multiple SVs, the DQN algorithm is improved into the IDQN algorithm proposed in this paper. Specifically, the reward functions of all SVs are designed according to the different QoS in different situations. Each SV takes an action according to the policy π after sensing the environment state in each time slot, and stores the action, environment state, reward value and new environment state into the experience pool of reinforcement learning. In order to obtain the maximum reward value, some samples are taken from the experience pool for training at regular intervals, and SVs make optimal actions based on the trained optimal strategy. In the long run, SVs and PVs can successfully share channels to improve spectrum utilization. The learning process of IDQN is described in Algorithm 1.

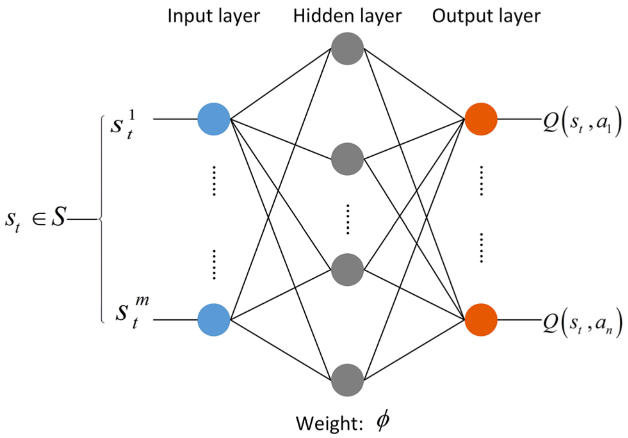


Fig. 6 Structure of deep Q-network

In the IDQN training process, the ε-greedy method is adopted to fully explore the channel occupancy for SVs in a CR-VANETs environment. The agent selects the action with the largest Q-value with a probability of $1-\varepsilon$, and randomly selects an action from A with a probability of ε. Specifically, it is denoted by the following formula.

$$a_{t+1} = \begin{cases} \arg\max_a Q(s_t, a_t), \ \mathit{with\ probability}\ 1-\varepsilon \\ \mathit{Choose\ random\ action},\ \mathit{with\ probability}\ \varepsilon \end{cases} \tag{28}$$

where a_{t+1} represents the action taken by SVs in time slot $t+1$. $\arg\max_a Q(s_t, a_t)$ represents the action a_t with the largest Q-value in state s_t. ε is a decimal number between 0 and 1. And the Q-value and strategy are iteratively updated to gradually converge to the optimal strategy. Upon completion of training, the Q-value reaches convergence. Within the spectrum access framework, actions taken by one SV in the present time frame remain unknown to other SVs. Consequently, multiple SVs might inadvertently reuse the same channel, leading to increased interference resulting from channel congestion. Excessive interference detrimentally impacts communication quality, leading to reduced rewards and hampering overall system performance. To mitigate this challenge, an approach is adopted where the effects of an agent's chosen actions on the environment are observed by other vehicles across various time instances. This dynamic

Table 3 Algorithm 1

Algorithm 1: The process of IDQN algorithm

```
Input: State space $S$, action space $A$, discount
    rate $\gamma$, learning rate $\alpha$
Output: Deep Q-network
Initialize simulation environment and
parameters about Q-networks;
Initialize replay memory $D$ to capacity $O$;
Randomly initialize the weights $\phi$ of the
Q-network;
Randomly initialize the weights of the target
Q-network $\widehat{\phi} = \phi$;
for episode = 1 : $j$ do
  Initialize state $s_t$ for each $n \in N$;
  for step = 1 : $i$ do
    for $n \in N$
      In state $s_t$, select action $a_t$ with policy $\pi$;
      Take action $a_t$, observe the reward $r_{t+1}$
      and a new state $s_{t+1}$;
      Save $s_t, a_t, r_{t+1}, s_{t+1}$ into $D$;
      Uniformly sample mini-batches from $D$;
      $y_t = r_t + \gamma \max_{a_{t+1}} Q_{\widehat{\phi}}(s_{t+1}, a_{t+1})$;
      Train the deep Q-network with the loss
      function $Loss(\phi) = (y_t - Q_\phi(s_t, a_t))^2$;
    end for
    Every C steps $\widehat{\phi} \leftarrow \phi$;
  end for
end for
```

Table 4 Parameter settings for communication environment

Parameter	Environment 1	Environment 2
Number of PVs M	20	20
Number of SVs N	20	20,25,30
Transmit power of PV	40 mW	40 mW
Transmit power of SV	20 mW	20 mW
Carrier frequency	5.9 Ghz	5.9 Ghz
Bandwidth B_0	10^8 Hz	10^8 Hz
Noise spectral density N_0	10^{-8} mW/Hz	10^{-8} mW/Hz

Table 5 Algorithm parameter settings

Parameter	Environment 1 and 2
Number of neuron	64
Memory size	500
Batch size	100
Epsilon update period	200
Total episode	30,000
Discount rate γ	0.9
Learning rate α	0.001
θ, η and λ	0.5, 0.5 and 0.5

observation of actions allows SVs to make decisions based on their environmental perception, thereby mitigating the occurrence of multiple vehicles simultaneously occupying a single channel, thereby enhancing rewards and avoiding undue channel congestion.

In summary, the spectrum access predicament within scenarios involving multiple SVs can be effectively addressed through the utilization of the IDQN algorithm introduced in this paper. A comprehensive performance analysis is elaborated in Sect. 5.

5 Simulation analysis

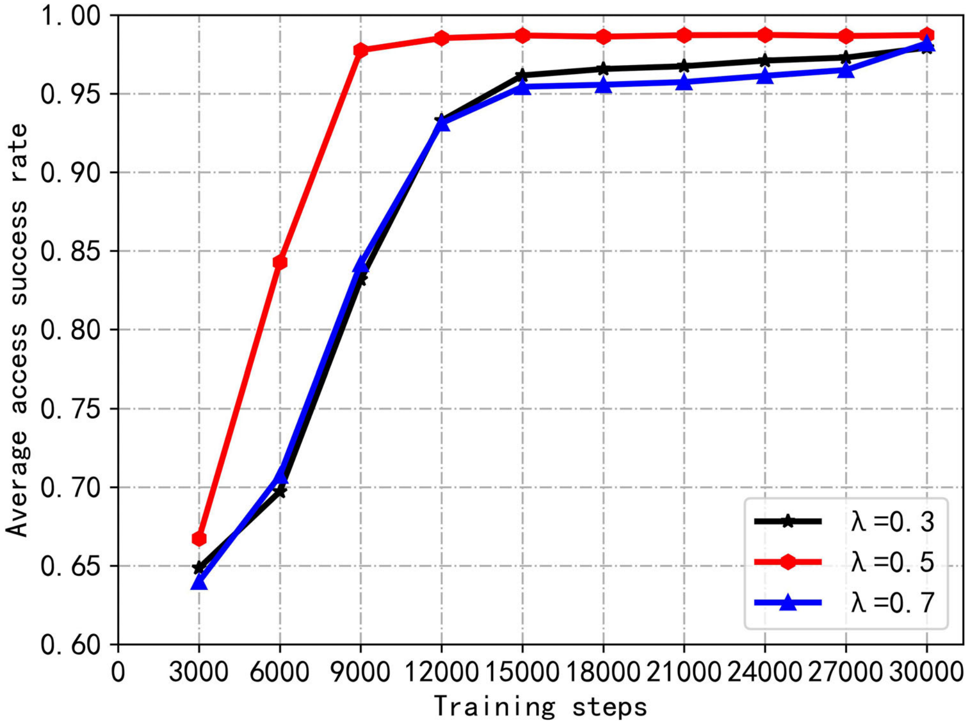
In this section, we evaluate the performance of the IDQN-based dynamic spectrum access method proposed, comparing it with DQN and Q-learning. Overall, the proposed IDQN method can guarantee maximum QoS for communication quality of PVs and SVs. Furthermore, the IDQN method's strong adaptability to dynamic environments is verified through simulation.

5.1 Simulation analysis

The spectrum access problem in the IDQN-based CR-VANETs communication environment is studied in this paper. To verify the superiority and adaptability of the IDQN method in dynamic environments, the following two environments are studied by us. Environment 1 is a communication environment in which the number of channels is fixed, but the number of SVs remains constant. Environment 2 is a communication environment in which the number of channels is fixed, but the number of SVs changes. In the two communication environments, the bandwidth is set to 1×10^8 Hz, and the noise spectral density is 1×10^{-8} mW/Hz. The neuron value of the IDQN method is set to 64, and the learning rate is 0.001. The specific parameter settings are shown in Table 4 and Table 5 below. As shown in Figs. 7, 8 and 9, the effect of the IDQN algorithm is compared when the parameter λ is set to 0.3, 0.5 and 0.7. It can be found that when λ is 0.5, the proposed algorithm has the best effect.

Figure 7 shows the average access success rate obtained by the IDQN algorithm when λ is different values. Figure 8 shows the average cumulative reward value obtained by the IDQN algorithm when λ is different values. Figure 9 shows the average QoS value obtained by the IDQN algorithm when λ is different values. It can be seen from these three figures

Fig. 7 Access success rate for different λ



that when λ is 0.5, the convergence speed and convergence value of the IDQN algorithm are optimal. In particular, the convergence speed is the fastest, because the reward function set in this paper equally considers the QoS values of PVs and SVs. Regardless of whether λ is high or low, one of PVs and SVs is not fully considered. This affects the better combination of IDQN algorithm and model. For example, when λ is 0.3, the system's average cumulative reward takes more into account the QoS of PVs, causing SVs to ignore their own QoS values in order to obtain higher rewards. When λ is 0.7, the system's average cumulative reward considers the QoS of SVs more, causing SVs to ignore the QoS value of PVs in order to get higher rewards.

5.2 Performance analysis

5.2.1 Environment 1

The dynamic spectrum access for multiple SVs in CR-VANETs is discussed in this section. In a real environment, multiple vehicles simultaneously access the spectrum for communication. So, in the current situation, a CR-VANETs scenario with 20 channels and 20 SVs is considered. And this part of the training time took 4546 s. The feasibility of the model established and the objective function set in this paper has been verified from the following six aspects, which are average access success rate, average access failure rate, average cumulative reward, average QoS, average QoS of PVs, and average QoS of SVs. And as shown in Figs. 10, 11, 12, 13, 14 and 15, the DQN method and Q-learning method are compared with the proposed IDQN method in order to verify the effect of the proposed IDQN method.

As shown in Fig. 10, it can be clearly seen that the average access success rate of the IDQN algorithm is higher than that of the DQN algorithm and the Q-learning algorithm. After the training times reach about 9000, the best average access success rate convergence value of 98% was obtained by SVs using the IDQN algorithm, while the Q-learning algorithm only achieved an average access success rate of 80%. Because the IDQN algorithm is set with different reward functions based on the QoS obtained by the SV's access. In order to obtain a superior reward value, all vehicles choose to access the optimal spectrum, thus avoiding access failures and improving the average access success rate. As shown in Equation (29), when the number of channels remains unchanged, the higher the access success rate of each SV, the higher the average access success rate obtained by the final IDQN algorithm. This allows SVs to choose the optimal action in order to obtain higher reward values. Although the final convergence value of the DQN algorithm achieved 97%, the convergence speed is very slow. The Q-learning algorithm can reach the optimal value in a short time but falls into a local optimum as the training time increases. In addition, the effect of IDQN is worse than that of Q-learning before training 6000 times. Because a good policy was not trained by the IDQN method to select a more correct channel before training 6000 times. However, IDQN has been fully trained, and the optimal channel can be selected for access

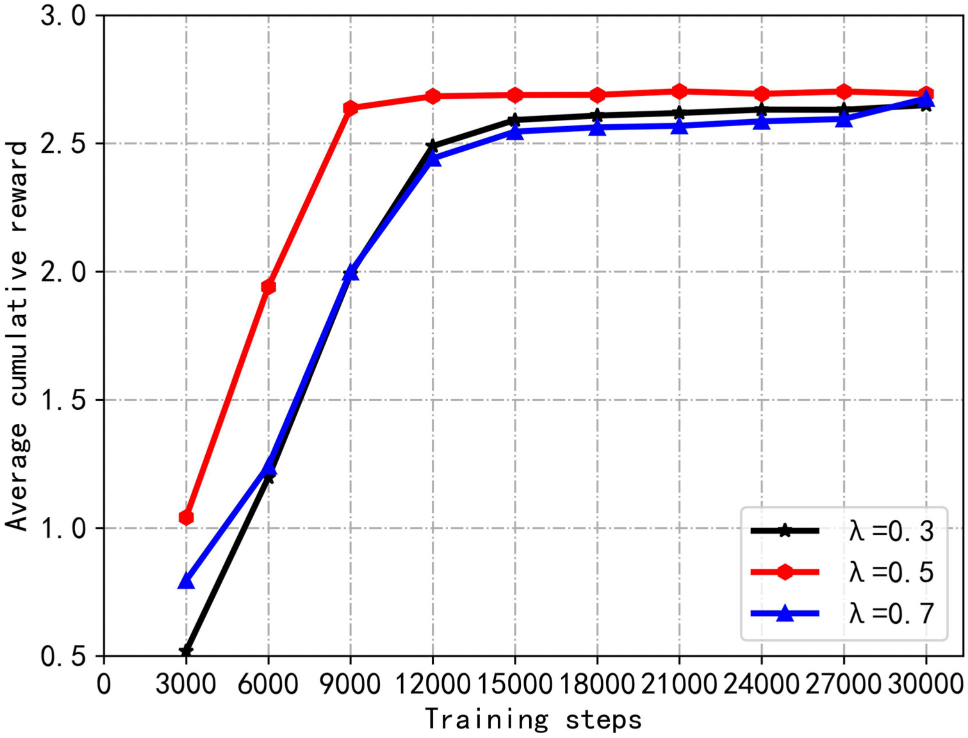


Fig. 8 Average cumulative rewards for different λ

after training 6,000 times. The action with the largest Q-value selected by Q-learning according to the Q-value table is not the optimal action. Therefore, the average access success rates of the IDQN and Q-learning methods are equal at 6000 times. And the formula is expressed as follows.

$$AS = \frac{1}{bs}\sum_{bs=1}^{bs}\frac{1}{k}\sum_{k=1}^{k}k_{success} \tag{29}$$

where $k_{success}$ represents the sum of successful accesses in a single training. bs represents the batch size, and k represents the number of SVs.

As can be seen from Fig. 11, the average access failure rate of the IDQN algorithm is lower than that of the DQN algorithm and the Q-learning algorithm in the case where multiple vehicles access the same channel considered in this paper. Because channel state information can be quickly learned by SVs using the IDQN algorithm to avoid sharing failures. The details are as follows. After training 9,000 times, the average access failure rate rapidly converges to 2% of the IDQN algorithm, while the Q-learning algorithm only achieved an average access failure rate of 20%. Because the IDQN algorithm is set with different reward functions based on the QoS obtained by the SV's access. This allows SVs to avoid collisions for higher reward values. Although the final access failure rate of the DQN algorithm is also very low, the convergence speed is very slow. The Q-learning algorithm can reach the optimal value in the fastest time, but the convergence value becomes worse as the training time increases. But before training 6000 times, the IDQN method is not fully trained, and the optimal channel cannot be selected by SVs. Therefore, the effect of IDQN was worse than that of Q-learning previously. However, the failure rate of spectrum sharing between SVs and PVs, as well as spectrum sharing among multiple SVs, is effectively reduced because the IDQN algorithm is fully trained after training for 6000 times. So the effect of IDQN is clearly better. And the formula is expressed as follows.

$$AF = \frac{1}{bs}\sum_{bs=1}^{bs}\frac{1}{k}\sum_{k=1}^{k}k_{failure} \tag{30}$$

where $k_{failure}$ represents the sum of failed accesses in a single training.

As shown in Fig. 12, first of all, the average reward value obtained by the IDQN algorithm is higher than that of the DQN algorithm and the Q-learning algorithm in the CR-VANETs environment after training 9000 times. In the end, the average cumulative reward value obtained by the IDQN algorithm was 2.7, while the DQN and Q-learning algorithms only obtained 2.0 and 0.4. Because the failure rate of spectrum sharing between SVs and PVs, as well as spectrum sharing between multiple SVs, can be effectively avoided by SVs using the IDQN algorithm to achieve the fastest convergence speed and highest reward value. The final average cumulative reward value obtained by the DQN algorithm is also very high, but the convergence speed is very slow. The Q-learning algorithm can obtain the optimal average

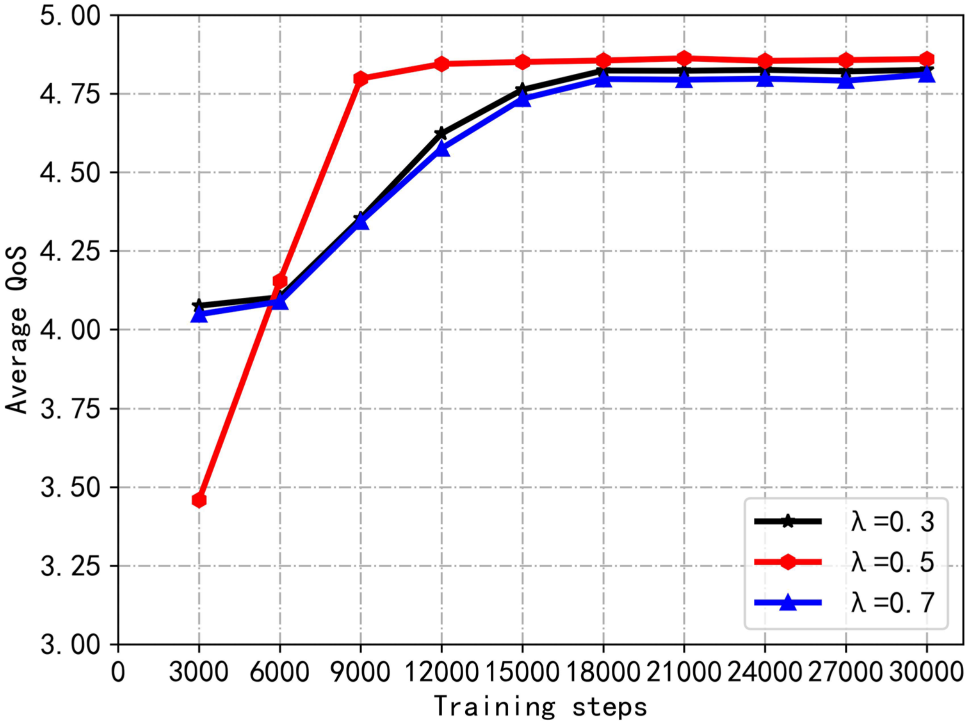


Fig. 9 Average QoS for different λ

cumulative reward value in the fastest time, but the final convergence value is very low. Secondly, the reward values obtained by the IDQN and DQN algorithms are equal when training 3000 times. Because both algorithms' strategies are in the exploratory stage. Finally, when training 6000 times, the action selection strategy of the Q-learning method was changed for the first time. However, the effect of Q-learning on selecting actions by looking up the Q-value table of large states is not good. It can be seen that the highest reward and fastest convergence speed can be obtained by applying the proposed IDQN algorithm to the CR-VANETs spectrum sharing model established in this paper.

In order to ensure the communication quality of all vehicles, the optimization of QoS should also be considered in the process of spectrum access. The average QoS obtained by the IDQN algorithm is higher than that of the DQN algorithm and the Q-learning algorithm, as shown in Fig. 13. According to formula (31), the final average QoS obtained by the IDQN algorithm is 4.9, while the DQN and Q-learning algorithms only obtain 3.4 and 2.5. Since the IDQN algorithm is set with different reward functions based on the QoS obtained by the SV's access. This allows SVs to choose the optimal action in order to obtain higher QoS. The average QoS finally obtained by the DQN algorithm is also very good, but the convergence speed is very slow. The QoS obtained by the Q-learning algorithm is very stable, but the final convergence value is relatively low. Since the best channel can be selected by the SV using the IDQN method, the highest QoS is obtained by vehicles. Therefore, the average QoS obtained by the IDQN algorithm begins to converge when training 9000 times, while the average QoS obtained by the DQN algorithm begins to converge when training 21000 times. In addition, the same QoS was obtained by the DQN and Q-learning when the training times reached approximately 6000. Because SVs randomly select channels to access using the DQN algorithm at this time. It can be seen from this performance index that the proposed IDQN algorithm is applied to the spectrum sharing model for CR-VANETs established in this paper, ensuring the communication quality of vehicles.

$$\overline{QoS} = \frac{1}{bs}\sum_{bs=1}^{bs}\left(\frac{1}{M}\sum_{m=1}^{M} QoS_m + \frac{1}{N}\sum_{n=1}^{N} QoS_n\right) \tag{31}$$

where QoS_m represents the QoS of the mth PV in a single training. QoS_nrepresents the QoS of the nth SV in a single training.

As shown in Figs. 14 and 15, respectively, both the QoS of PVs and the QoS of SVs should be considered in this paper.

The relationship between the average QoS of PVs and the training time of SVs in CR-VANETs is shown in Fig. 14. It can be seen from the figure that the IDQN algorithm outperforms both the DQN algorithm and the Q-learning algorithm in terms of convergence speed and the average QoS convergence value for PVs. The final average QoS of PVs obtained by the IDQN algorithm is 2.4, while the DQN and Q-learning algorithms only obtain 1.6 and 1.5. In order to obtain the optimal reward value and avoid sharing failure with PVs, the interference caused by SVs is relatively small, so PVs obtains

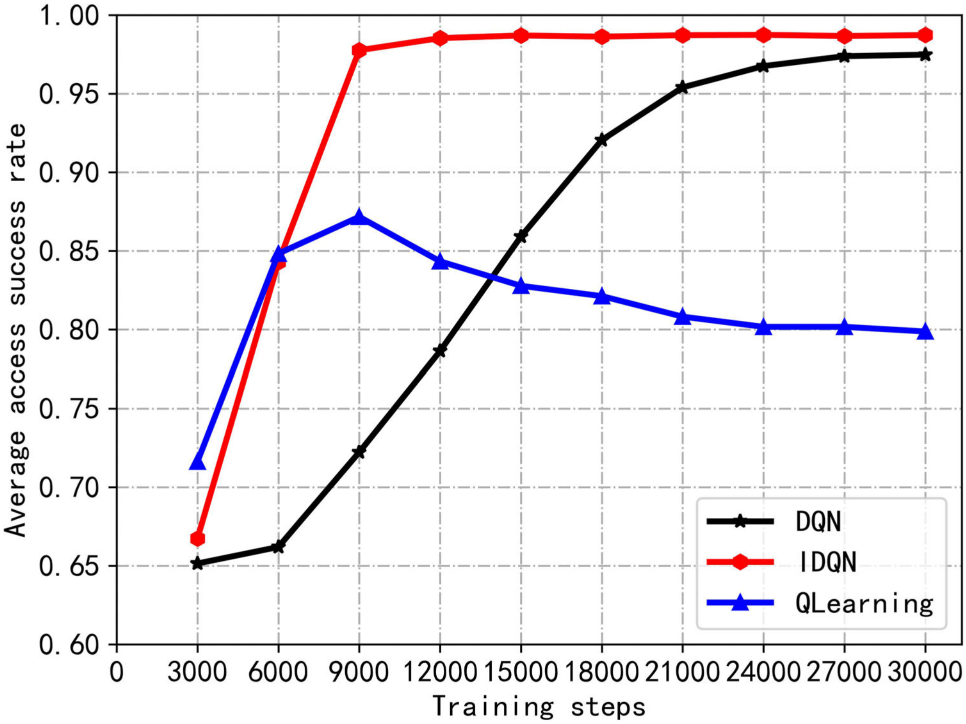


Fig. 10 Access success rate for different algorithms

the optimal average QoS. As shown in formula (32), if the number of channels remains unchanged, the greater the QoS obtained by every PV, the greater the QoS of the average SVs obtained by the final IDQN algorithm. There is no sudden change in the entire training process of the DQN algorithm, but the convergence speed is very slow. The training process of the Q-learning algorithm changes relatively slowly, but the values fluctuate high and low and the final convergence value is relatively low. And when training 9000 times, IDQN has learned the optimal policy and started to converge. The proposed IDQN algorithm guarantees the average QoS of PVs. And the formula is expressed as follows.

$$\overline{QoS_{PV}} = \frac{1}{bs}\sum_{bs=1}^{bs}\frac{1}{M}\sum_{m=1}^{M}QoS_m \tag{32}$$

As shown in Fig. 15, first, the average QoS of SVs obtained by the IDQN algorithm converges to the optimal value during training 9000, which is higher than that of the DQN algorithm and the Q-learning algorithm. The final average QoS of SVs obtained by the IDQN algorithm is 2.4, while the DQN and Q-learning algorithms only obtain 1.6 and 1.0. In order to obtain the optimal reward value and avoid sharing failure, the interference caused by SVs is relatively small, so the SVs themselves obtain the optimal average QoS. As shown in formula (33), if the number of channels remains unchanged, the greater the QoS obtained by every SV, the greater the QoS of the average SVs obtained by the final IDQN algorithm. The entire training process of the DQN algorithm has no mutations, but the convergence speed is very slow. The Q-learning algorithm can obtain the optimal value quickly after training, but the final convergence value is low. Secondly, the average SVs QoS of the DQN algorithm remains unchanged before training 6,000 times. From 6,000 to 21,000 times, the average SVs QoS rapidly rises to convergence. Finally, the QoS of the SVs obtained through the Q-learning method reached its highest level when it was trained 6000 times. And the formula is expressed as follows.

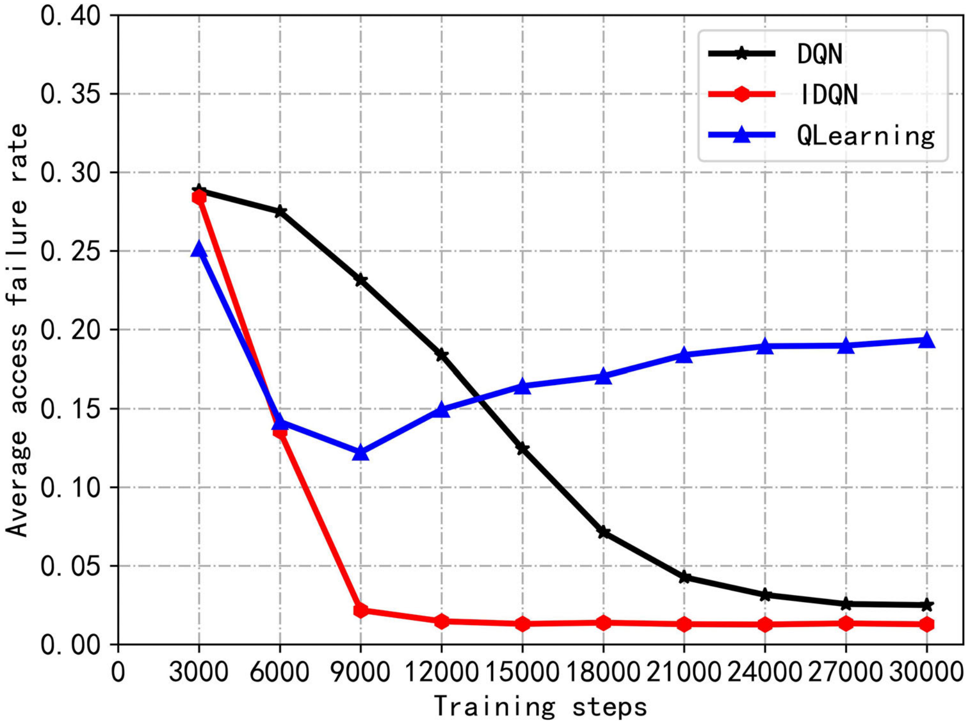
$$\overline{QoS_{SV}} = \frac{1}{bs}\sum_{bs=1}^{bs}\frac{1}{N}\sum_{n=1}^{N}QoS_n \tag{33}$$

5.3 Environment 2

In the scenario where the number of channels and the number of SVs is fixed at 20, the proposed IDQN algorithm outperforms the DQN algorithm and the Q-learning algorithm. This is evident from the various indicators displayed in the simulation diagram above. The performance change of the IDQN algorithm is analyzed in the following simulation section when the number of channels is fixed at 20, and the number of SVs varies from 20 to 25 and 30. The specific details are shown in Figs. 16, 17, 18, 19, 20 and 21.

Overall, it can be seen from Fig. 16 that the convergence speed and average access success rate of the IDQN method decrease as the number of SVs increases. Specifically, when the number of SVs increases from 20 to 25, the average access success rate of the IDQN algorithm drops from 98% to 87%

Fig. 11 Access failure rate for different algorithms



due to the increased probability of multiple SVs selecting the same channel. And when the number of SVs increases to 30, the average access success rate of the IDQN algorithm drops to 77%. Therefore, there are more cases where SVs in the channel generate excessive interference, resulting in a decrease in the average access success rate of SVs. When the number of SVs increases to 30, this phenomenon becomes more apparent, resulting in increased interference caused by SVs. Finally, it is evident that when the number of SVs increases from 25 to 30, the average access success rate of the IDQN method remains high at 77%, with only a slight decrease of about 11%. The strong applicability of the IDQN method in dynamic environments is also verified by this figure.

Figure 17 illustrates a trend in which the convergence speed of the IDQN method slows down and the average access failure rate increases as the number of SVs increases. Specifically, when the number of SVs increases from 20 to 25, the average access failure rate of the IDQN algorithm increases from 2% to 11% due to too many SVs selecting the same channel. When the number of SVs increases to 30, the average access failure rate of the IDQN algorithm rises to 20%. When the number of SVs increases from 20 to 25, multiple SVs end up selecting the same channel for access, resulting in channel access failure for these SVs. When the number of SVs increases to 30, the probability of channel access failure for SVs also increases. But when the number of SVs increases from 20 to 25, the average access failure rate of the IDQN method only increases by approximately 9%.

Figure 18 shows that the average cumulative reward value obtained by SVs in the CR-VANETs environment deteriorates as the number of SVs increases. First, when the number of channels is 20 and the number of SVs increases from 20 to 25, the average cumulative reward value obtained by the SVs decreases. Specifically, when the number of SVs increases from 20 to 25, the average cumulative reward value of the IDQN algorithm drops from 2.7 to 2.0. When the number of SVs increases to 30, the average cumulative reward value of the IDQN algorithm drops to 1.3. As the number of SVs increases, the probability of spectrum sharing failure between SVs and PVs and between multiple SVs also increases. Therefore, the reward value obtained by SVs is low. When the number of SVs increases to 30, the probability of failure for SVs accessing the channel also increases, resulting in a decrease in the average cumulative reward value obtained. However, when the number of SVs increases from 20 to 25, the final average cumulative reward by the IDQN method decreases by only approximately 12%.

As can be seen from Fig. 19, the average QoS obtained by the IDQN method decreases as the number of SVs increases. When the number of SVs increases from 20 to 25, the average QoS of the IDQN algorithm drops from 4.9 to 4.1. When the number of SVs increases to 30, the average QoS of the IDQN algorithm drops to 3.0. Specifically, when the number of SVs increases from 20 to 25, there is an increase in the number of SVs accessing the same channel. As a result, the total inter-

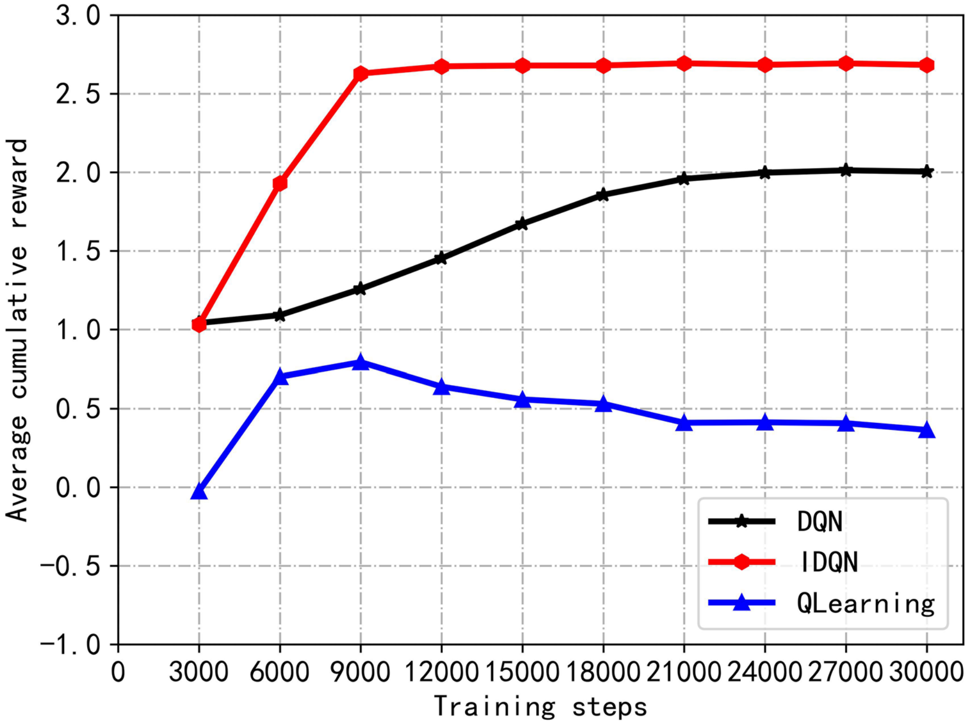


Fig. 12 Average cumulative rewards for different algorithms

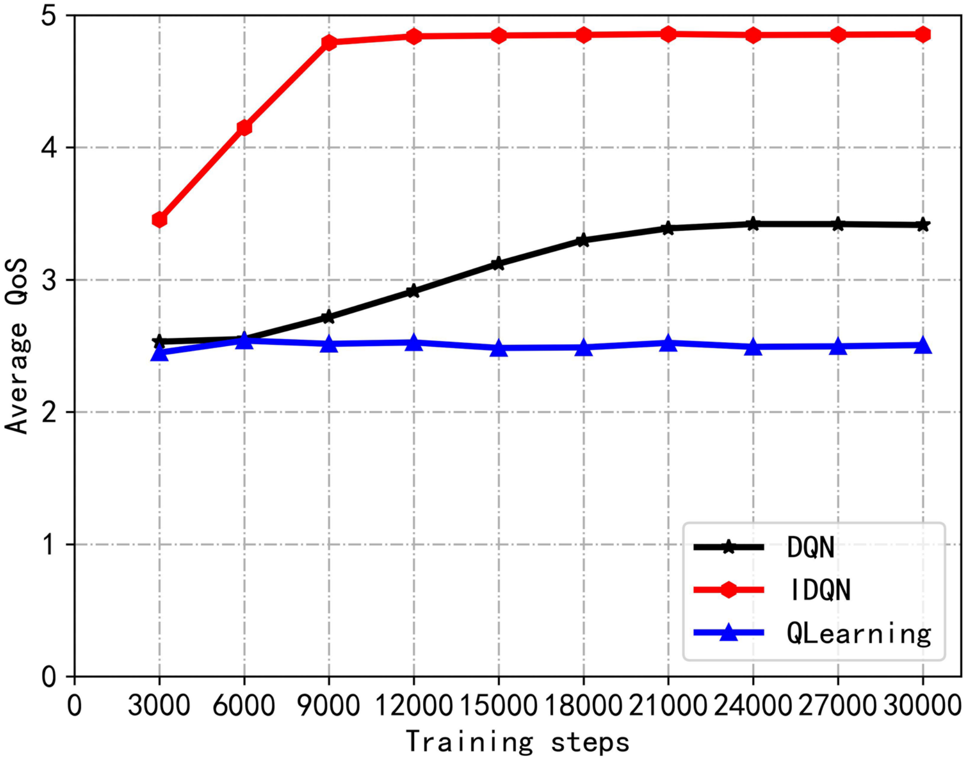


Fig. 13 Average QoS for different algorithms

Fig. 14 Average QoS of PVs for different algorithms

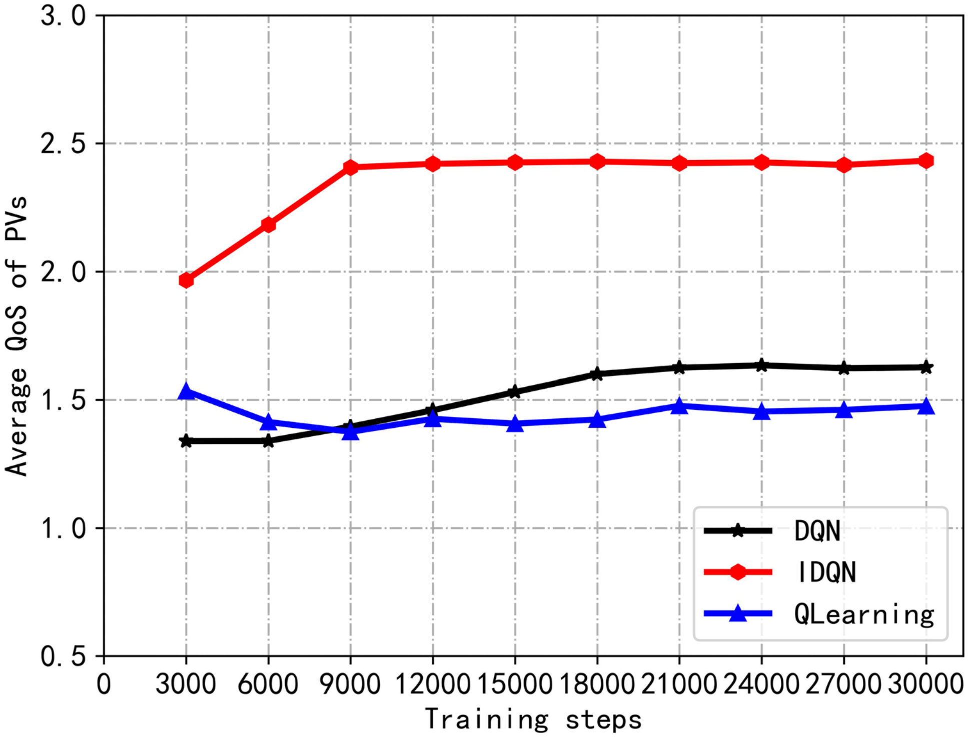
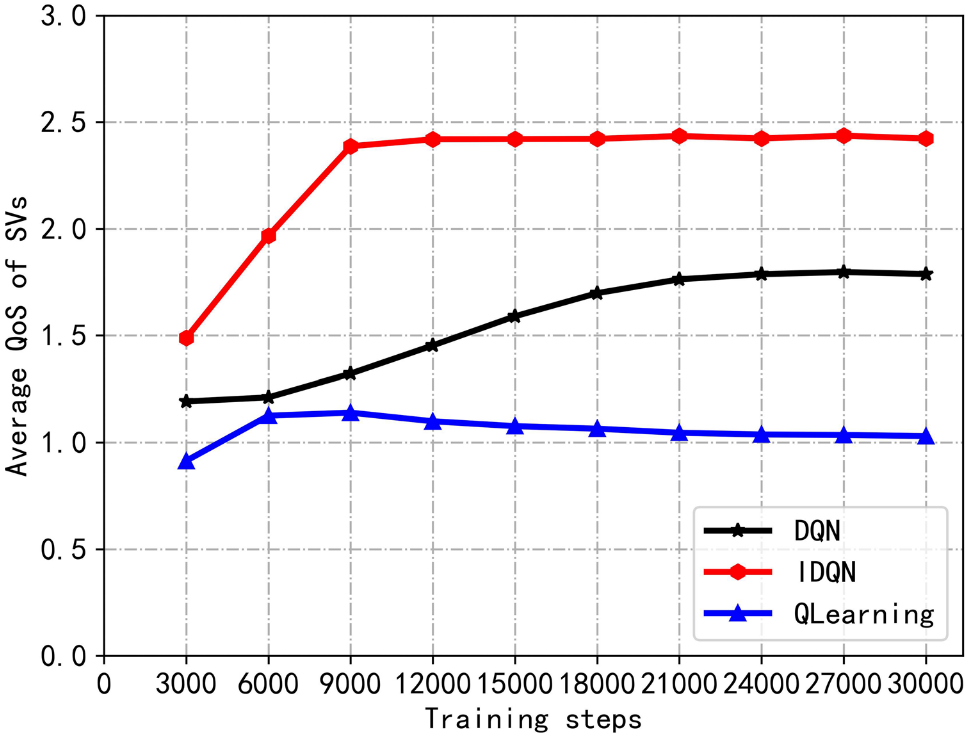


Fig. 15 Average QoS of SVs for different algorithms



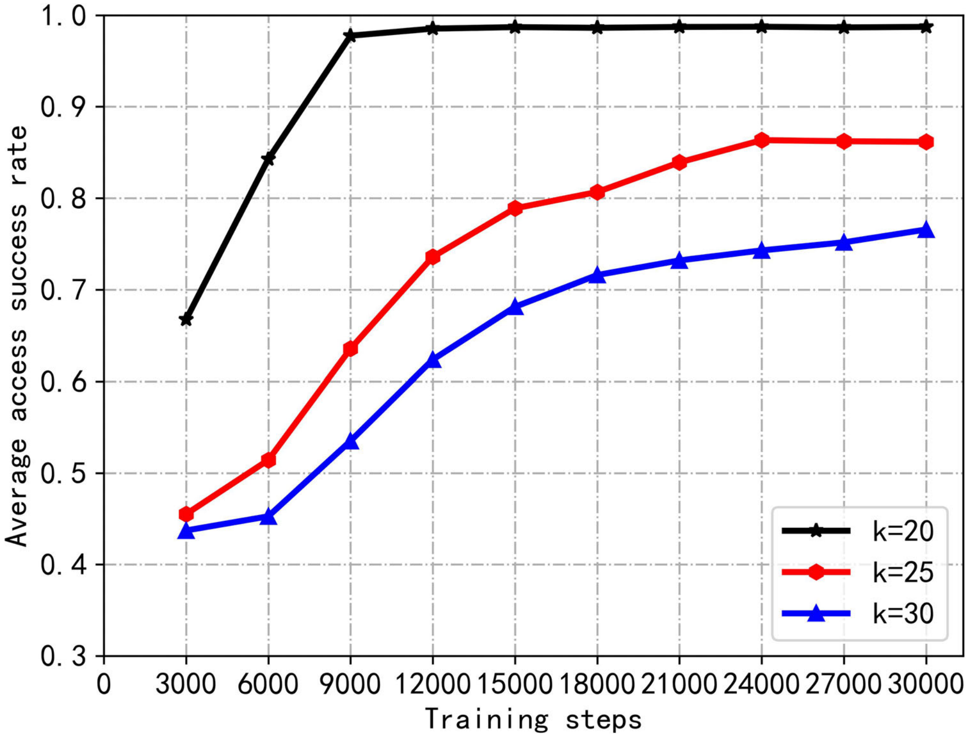


Fig. 16 Average access success rate (M = 20, N = 20, N = 25, N=30)

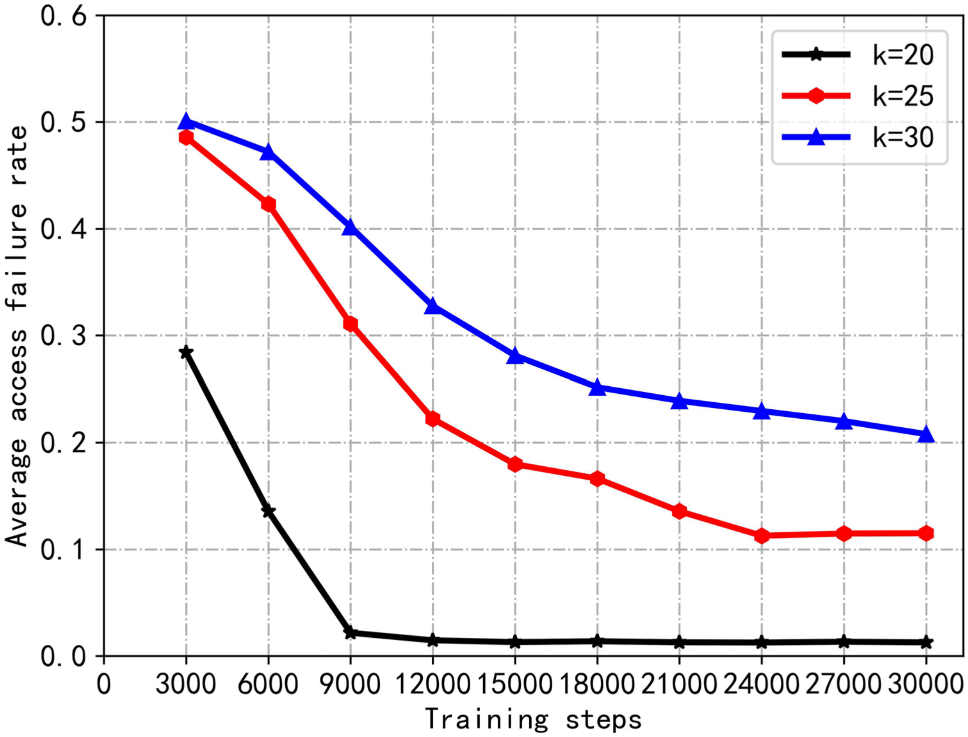


Fig. 17 Average access failure rates (M = 20, N = 20, N = 25, N = 30)

Fig. 18 Average cumulative rewards (M = 20, N = 20, N = 25, N = 30)

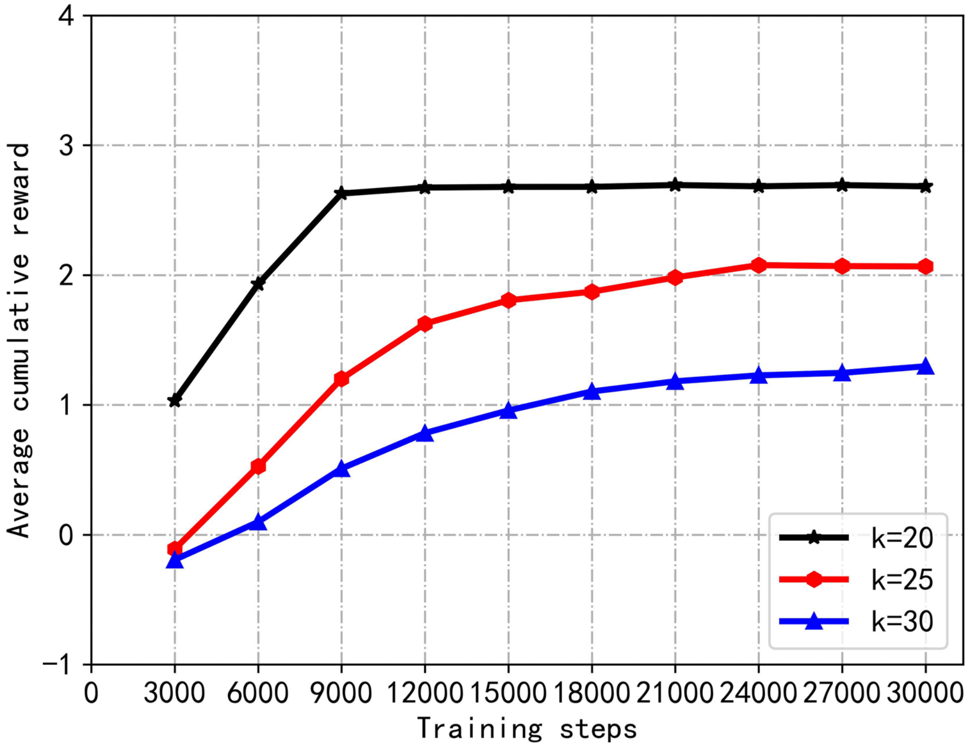


Fig. 19 Average QoS of vehicles (M = 20, N = 20, N = 25, N = 30)

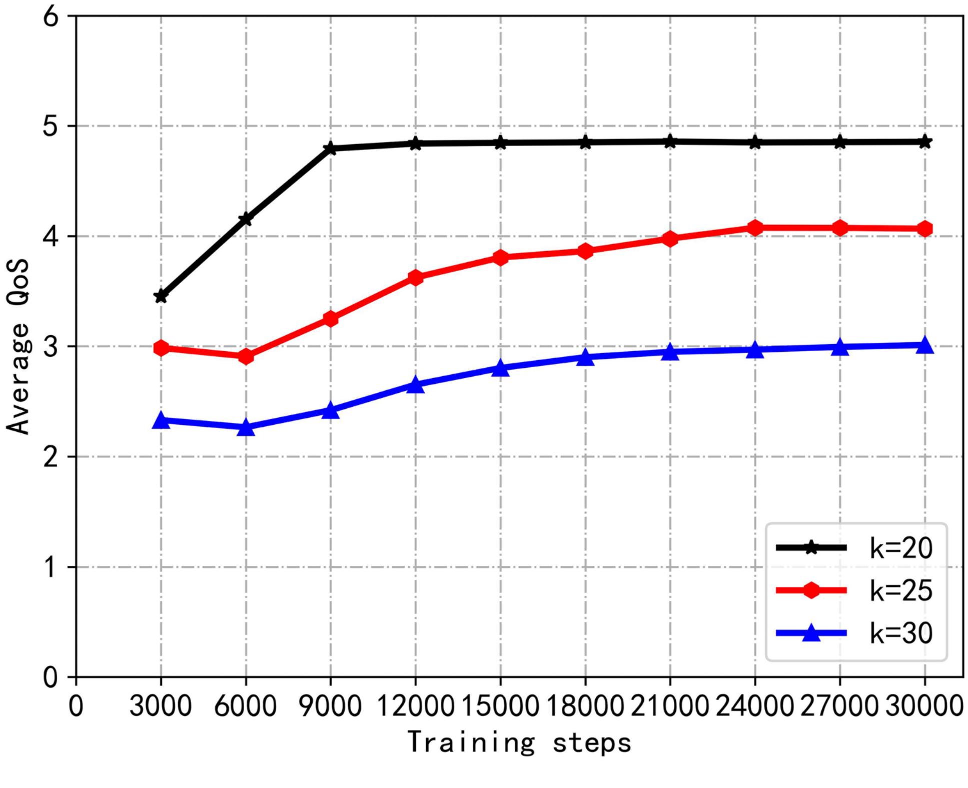


Fig. 20 Average QoS of PVs (M = 20, N = 20, N = 25, N = 30)

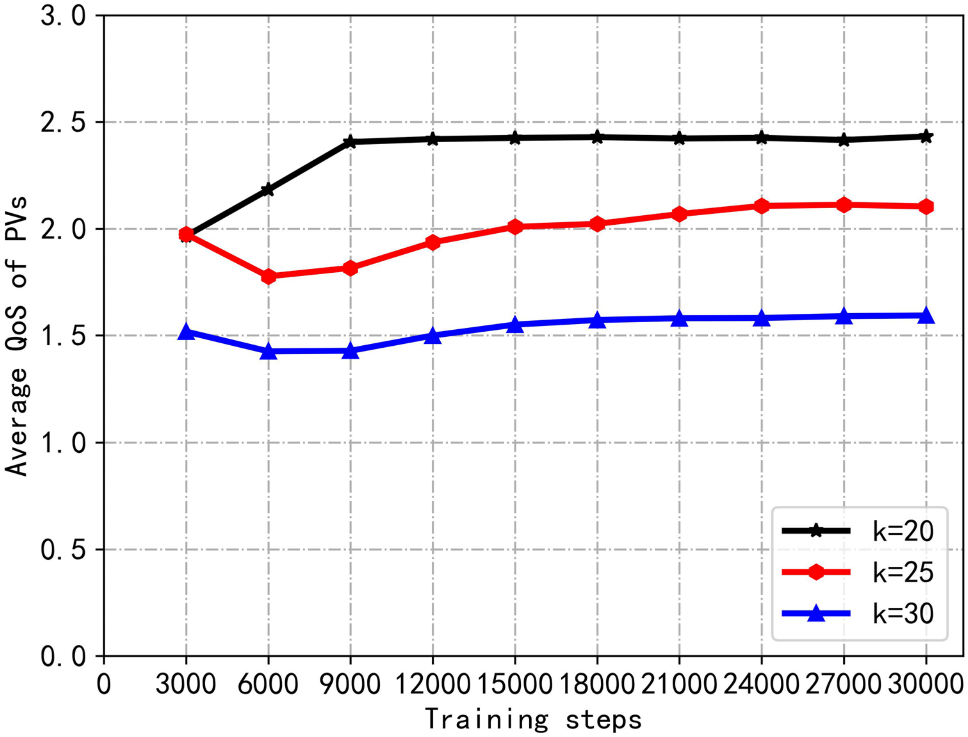
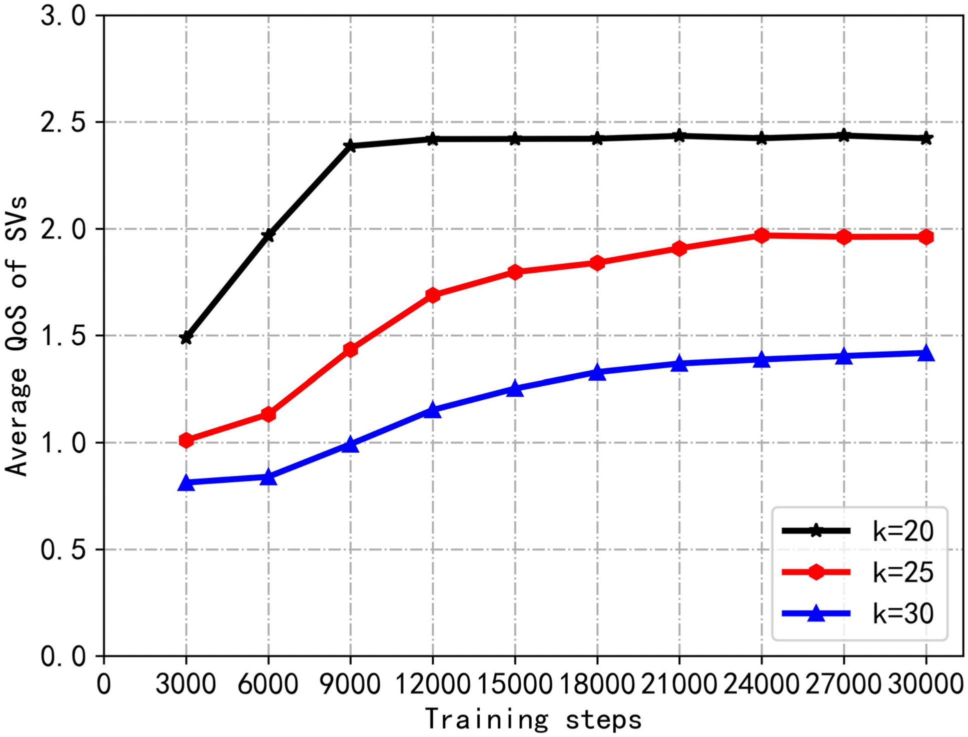


Fig. 21 Average QoS of SVs (M = 20, N = 20, N = 25, N = 30)



ference on the same channel increases, leading to a decrease in the SINR of PVs. When the number of SVs increases to 30, the total interference accessing the same channel also increases, leading to a decrease in SINR for PVs. This situation ultimately results in a lower average QoS received by vehicles. But when the number of SVs increases from 20 to 30, the average QoS of the IDQN method decreases by only approximately 38%.

As shown in Fig. 20, the average QoS of PVs in the CR-VANETs scenario decreases as the number of SVs increases. When the number of SVs increases from 20 to 25, the average QoS of PVs by the IDQN algorithm drops from 2.4 to 2.1. When the number of SVs increases to 30, the average QoS of PVs by the IDQN algorithm drops to 1.6. The increase in the number of SVs leads to more SVs accessing the same channel. Such a phenomenon results in increased interference to PVs, which decreases the SINR of PVs. Thus, the convergence value of the average QoS of the final PVs decreases. But when the number of SVs increases from 20 to 25, the average QoS of PVs decreases by only approximately 12%.

It can be seen from Fig. 21 that the convergence speed and convergence value of the IDQN method deteriorate as the number of SVs increases. When the number of SVs increases from 20 to 25, the average QoS of SVs by the IDQN algorithm drops from 2.4 to 2.0. When the number of SVs increases to 30, the average QoS of SVs by the IDQN algorithm drops to 1.4. Because multiple SVs access the same channel. Therefore, the interference between the SVs becomes larger, resulting in a decrease in the convergence value of the average SVs' QoS. However, when the number of SVs increases to 1.5 times, the QoS of SVs only decreases by about 40%. The strong applicability of the IDQN method in dynamic environments is further verified by this figure.

6 Conclusions

Aiming at the problem of spectrum resource shortage, a spectrum resource sharing model based on multi-SVs is proposed in the CR-VANETs scenario, where a PV and SVs share spectrum and multiple SVs share spectrum. It satisfies the interference constraint condition that the interference generated by SVs accessing the same channel is less than PVs or SVs. In order to achieve the purpose of improving spectrum utilization, the QoS functions of PVs and the QoS functions of SVs are designed separately. An IDQN method is proposed to solve the problem of low success rate of SVs accessing the channel. The failure probability of SVs accessing the spectrum can be effectively reduced by the IDQN method with designing four reward functions related to QoS functions for SVs. In order to prove the effectiveness of this algorithm, it was compared with the DQN algorithm and Q-learning algorithm under the Python platform. The results show that when compared to the other two methods, not only the dynamic environment can be better adapted to the proposed IDQN method, but also the average access success rate and average QoS performance of SVs can be improved. In particular, the average access success rate of SVs has reached 98%, which is improved by 18% compared with the Q-learning algorithm. And the convergence speed is 62.5% faster than the DQN algorithm. At the same time, the average QoS of PVs and the average QoS of SVs in the IDQN algorithm can reach 2.4, which is improved by 50% and 33% compared with the DQN algorithm, and improved by 60% and 140% compared with the Q-learning algorithm. Moreover, when the number of SVs in the communication environment increases from 20 to 30, the average access success rate of the IDQN method only decreases by 20%. And the average QoS of PVs only dropped by 33%.

7 Extensions

Although the proposed IDQN algorithm to improve the QoS and success rate of SVs spectrum access in the established CR-VANETs model, the impact of the distance between vehicles on spectrum access is not carefully considered in this paper. Since there is often a distance between driving vehicles in real life, and the distance changes in real time. Communication quality is certainly affected by the distance between vehicles, so the distance between vehicles will be taken into account in spectrum access in future work. And in complex CR-VANETs scenarios, the IDQN effect decreases. In future work, we will consider whether the IDQN algorithm can achieve better results by changing the network structure in more complex CR-VANETs scenarios. For example, the IDQN algorithm adds a neural network with a hidden layer to the Q-learning algorithm. So whether increasing the number of hidden layers can adapt to more complex environments requires further research. In addition, the improved IDQN algorithm is compared with advanced professional methods in the field to study the advantages and disadvantages of this method.

Acknowledgements This work is supported in part by the National Natural Science Foundation of China under grant no. 61571209 and 61501059, Science and Technology Department of Jilin Provincial, China (Grant No. YDZJ202201ZYTS653), Science and Technology Department of Jilin Provincial, China (Grant No. 20180101336JC).

Funding The authors have not disclosed any funding.

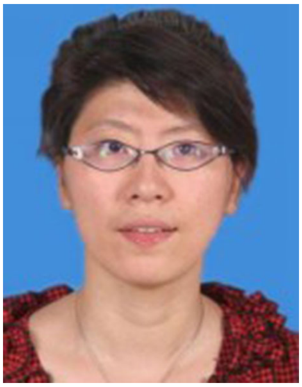
Declarations

Conflict of 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.

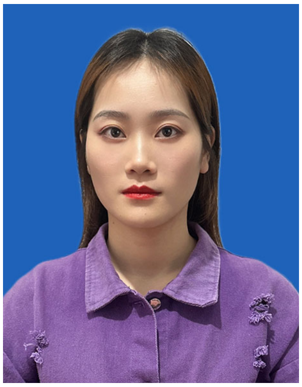
References

1. Hussain, I. (2024). Secure, sustainable smart cities and the internet of things: Perspectives, challenges, and future directions. *Sustainability, 16*(4), 1390.
2. Knari, A., Derfouf, M., Koulali, M. A., et al. (2024). Multi-agent deep reinforcement learning for content caching within the internet of vehicles. *Ad Hoc Networks, 152*, 103305.
3. Chuang, C. L., Chiu, W. Y., & Chuang, Y. C. (2021). Dynamic multiobjective approach for power and spectrum allocation in cognitive radio networks. *IEEE Systems Journal, 15*(4), 5417–5428.
4. Gnanaselvam, R., & Vasanthi, M. S. (2024). Dynamic spectrum access-based augmenting coverage in narrow band Internet of Things. *International Journal of Communication Systems, 37*(1), e5629.
5. El-Sherif, M. F., Rabia, S. I., & Abd El-Malek, A. H., et al. (2024). Age of information minimization in hybrid cognitive radio networks under a timely throughput constraint. *Perform Evaluation* 102407.
6. Kumar, S. S., Kumar, P. K. M., Panimalar, S. A., et al. (2024). QoS based soft computing techniques for evaluating efficient web service recommendation. *International Journal of System Assurance Engineering and Management, 15*(1), 205–215.
7. Gao, X., Dou, Z., & Qi, L. (2020). A new distributed dynamic spectrum access model based on DQN. *IEEE Transactions on Signal Processing, 1*, 351–355.
8. Sirait, R., Hardjawana, W., & Wibisono, G. (2023). Performance of downlink NOMA for a massive IoT network over a Nakagami-m fading channel with optimized power allocation. *IEEE Access., 11*, 67779–67790.
9. Zhang, H., & Guo, C. (2019). Cognitive communication device for vehicular networking. In *IEEE global conference on signal and information processing* (pp. 1–5).
10. Ding, H., Li, X., Cai, Y., & Lorenzo, B. (2018). Intelligent data transportation in smart cities: A spectrum-aware approach. *IEEE/ACM Transactions on Networking, 26*(6), 2598–2611.
11. Ejaz, W., & Ibnkahla, M. (2018). Multiband spectrum sensing and resource allocation for IoT in cognitive 5G networks. *IEEE Internet of Things Journal, 5*(1), 150–163.
12. Liu, X., & Zhang, X. (2019). Noma-based resource allocation for cluster-based cognitive industrial internet of things. *IEEE Transactions on Industrial Informatics, 16*(8), 5379–5388.
13. Zakariya, A. Y., Tayel, A. F., Rabia, S. I., et al. (2020). Modeling and analysis of cognitive radio networks with different channel access capabilities of secondary users. *Simulation Modelling Practice and Theory, 103*, 102096.
14. Benomarat, I., Madini, Z., & Zouine, Y. (2018). Enhancing Internet of vehicles (IOVs) performances using intelligent cognitive radio principles. In *International conference on electronics, control, optimization and computer science* (pp. 1–4).
15. Yao, W., Yahya, A., Khan, F., Tan, Z., & Ur Rehman, A. (2019). A secured and efficient communication scheme for decentralized cognitive radio-based Internet of vehicles. *IEEE Access* **7**, 160889–160900.
16. Hill, E., & Sun, H. (2018). Double threshold spectrum sensing methods in spectrum-scarce vehicular communications. *IEEE Transactions on Industrial Informatics, 14*(9), 4072–4080.
17. Sun, D., Chen, Y., & Li, H. (2024). Intelligent vehicle computation offloading in vehicular ad hoc networks: A multi-agent LSTM approach with deep reinforcement learning. *Mathematics, 12*(3), 424.
18. Yavas, M. U., Kumbasar, T., & Ure, N. K. (2023). Toward learning human-like, safe and comfortable car-following policies with a novel deep reinforcement learning approach. *IEEE Access, 11*, 16843–16854.
19. Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310–1318).
20. Hu, Y., Fu, J., Wen, G., et al. (2024). Distributed entropy-regularized multi-agent reinforcement learning with policy consensus. *Automatica, 164*, 111652.
21. Urmonov, O., Aliev, H., & Kim, H. (2023). Multi-agent deep reinforcement learning for enhancement of distributed resource allocation in vehicular network. *IEEE Systems Journal, 17*(1), 491–502.
22. Kaur, A., Thakur, J., Thakur, M., Kumar, K., Prakash, A., & Tripathi, R. (2023). Deep recurrent reinforcement learning-based distributed dynamic spectrum access in multichannel wireless networks with imperfect feedback. *IEEE Transactions on Cognitive Communications and Networking, 9*(2), 281–292.
23. Nasir, Y. S., & Guo, D. (2019). Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks. *IEEE Journal on Selected Areas in Communications, 37*(10), 2239–2250.
24. Albinsaid, H., Singh, K., & Biswas, S. (2021). Multi-agent reinforcement learning-based distributed dynamic spectrum access. *IEEE Transactions on Cognitive Communications and Networking, 8*(2), 1174–1185.
25. Bhadauria, S., & Shabbir, Z. (2020). QoS based deep reinforcement learning for V2X resource allocation. In *International black sea conference on communications and networking* (pp. 1–6).
26. Tan, X., Zhou, L., Wang, H., Sun, Y., & Zhao, H. (2022). Cooperative multi-agent reinforcement-learning-based distributed dynamic spectrum access in cognitive radio networks. *IEEE Internet of Things Journal, 9*(19), 19477–19488.
27. Khuntia, P., & Hazra, R. (2018). An actor-critic reinforcement learning for device-to-device communication underlaying cellular network. *IEEE Region 10 Conference* 0050–0055.
28. Sohaib, M., Jeong, J., & Jeon, S. W. (2022). Dynamic multichannel access via multi-agent reinforcement learning: Throughput and fairness guarantees. *IEEE Transactions on Wireless Communications, 21*(6), 3994–4008.
29. Sharma, S., & Singh, B. (2019). Weighted cooperative reinforcement learning-based energy-efficient autonomous resource selection strategy for underlay D2D communication. *IET Communications, 13*(14), 2078–2087.
30. Zia, K., Javed, N., Sial, M.N., & Ahmed, S. (2018). Multi-agent RL based user-centric spectrum allocation scheme in D2D enabled hetnets. In *IEEE 23rd International Workshop on Computer-Aided Modeling and Design of Communication Links and Networks* (pp. 1–6).
31. Liang, L., Ye, H., & Li, G. Y. (2019). Spectrum sharing in vehicular networks based on multi-agent reinforcement learning. *IEEE Journal on Selected Areas in Communications, 37*(10), 2282–2292.
32. Liu, X., Sun, C., Zhou, M., Lin, B., & Lim, Y. (2021). Reinforcement learning based dynamic spectrum access in cognitive internet of vehicles. *China Communications, 18*(7), 58–68.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Lingling Chen received her Bachelor and Ph.D. degrees both in Communication Engineering from Jilin University, China, in 2004 and 2015 respectively. Currently, she is a professor of College of Information and Control Engineering, Jilin Institute of Chemical Technology. Her research interests are in nonlinear optimization, mobile computing and cognitive radio networks.



Ziwei Wang received the B.Eng. degree in Huainan Normal University, Anhui, China. She has been working towards the M.Sc. degree in College of Information and Control Engineering, Jilin Institute of Chemical Technology. Her research interests are in the spectrum access in cognitive vehicular network.

Xiaohui Zhao received his Bachelor and Master degrees both in Electrical Engineering from Jilin University of Technology, China, in 1982 and 1986 respectively, and his Ph.D. degree in control theory from University de Technology de Compiegne, in 1993, France. Currently, he is a Professor of College of Communication Engineering, Jilin University. His research interests are signal processing, nonlinear optimization and wireless communication

Xuan Shen received the B.Eng. degree in Nanjing Institute of Technology, Jiangsu, China. She has been working towards the M.Sc. degree in College of Information and Control Engineering, Jilin Institute of Chemical Technology. Her research interests are in the security of cognitive radio networks and spectrum sensing in cognitive radio.

Wei He received the B.Eng. degree in Nanjing University of Posts and Telecommunications, Nanjing, China. He has been working towards the M.Sc. degree in College of Information and Control Engineering, Jilin Institute of Chemical Technology. His research interests are in the power control in cognitive vehicular network