Research on Distributed Dynamic Spectrum Access Based on Deep Reinforcement Learning

Yanchao Liu, Xiaohua Zhang, and Shubin Wang

Abstract Dynamic Spectrum Access (DSA) is a critical technology for Cognitive Wireless Sensor Network (CWSN). The main challenge of DSA is how Secondary Users (SUs) can quickly and accurately identify vacant spectrum, while ensuring that the service of the Primary User (PU) is not interrupted. The current DSA solutions do not satisfy the requirements of high throughput, low interference and fast convergence simultaneously for large scale multiple users and multiple channels access scenarios. In this paper, we propose a distributed DSA algorithm based on Deep Reinforcement Learning (DRL). First, we construct a Cognitive Wireless Sensor Network (CWSN) environment with multiple users and multiple channels. Next, based on the spectrum sensing results, each SU provides channel observations to our proposed Deep Q-Network (DQN) model for training in order to learn the optimal spectrum access policy. Finally, using the output of the DQN model, each SU intelligently accesses the appropriate channel. In order to improve the training accuracy and address the performance degradation problem caused by the network depth in deep neural networks, we added the Residual Network (ResNet) structure to the DQN. Simulation results show that the proposed algorithm achieves faster convergence speed, completely avoids collisions between SUs, greatly reduces the interference of SUs to PU, and significantly improves the success rate of channel access.

Keywords Dynamic spectrum access · Deep reinforcement learning · Spectrum allocation · Cognitive wireless sensor network · Deep Q-Network

Y. Liu · S. Wang (✉)
College of Electronic Information Engineering, Inner Mongolia University, Hohhot, China
e-mail: wangshubin@imu.edu.cn

X. Zhang
Department of Foreign Languages, Guizhou University of Commerce, Guiyang, China

W. Wang et al. (eds.), *Communications, Signal Processing, and Systems*, Lecture Notes in Electrical Engineering 1032, https://doi.org/10.1007/978-981-99-7505-1_20

1 Introduction

CWSN combines cognitive radio technology with Wireless Sensor Network (WSN) to address the problem of scarce spectrum resources by allowing a large number of sensor nodes as SUs to access the authorized spectrum. DSA is one of the key technologies in CWSN, and its task is to make a decision based on spectrum sensing data from cognitive sensor nodes to access a vacant spectrum licensed to a PU. However, when using this technique, the issues that need to be addressed are: how to minimise the interference to the PU while accessing and using the authorised spectrum, and how to avoid conflicts between SUs when multiple SUs try to access the same spectrum [1, 2].

Traditional optimization algorithms such as Game Theory, Particle Swarm Optimization and Genetic Algorithm have been used to address the DSA problem [3, 4]. Although these methods achieve spectrum reuse, their model design is complex, easily get trapped in local optima and less flexible and adaptive. In contrast, Reinforcement Learning (RL) can adaptively learn optimal strategies without a priori information in uncertain and dynamic complex environments. Therefore, in recent years, RL has been applied to DSA. In literature [5], a Q-learning based spectrum access algorithm is proposed to improve the transmission performance through intelligent utilisation of spectrum resources. Document [6] proposes a decentralised multi-intelligence reinforcement learning-based resource allocation scheme to address resource allocation problem without complete channel state information. The Q-learning used in the literatures [5, 6] performs well on small-scale models. However, it shows significant performance degradation when the state or action space is large. Deep Neural Network (DNN) is used in DRL to overcome this limitation. In literature [7], a centralised dynamic multichannel access framework based on DQN is proposed to minimise conflicts and optimise multi-user channel allocation through a centralised allocation policy. However, the centralised approach to spectrum access can lead to high communication overheads and may be difficult to implement in practice. In addition, the algorithm's performance may be limited as it doesn't account for imperfect spectrum sense that occur in real-world environments. Literature [8, 9] proposed using multi-intelligent deep reinforcement learning at medium access control layer for channel access. In this approach, users make transmission decisions through centralised training and decentralised execution to maximise the long-term average rates or to improve the performance of the network in terms of throughput, delay and jitter. However, this centralized training approach has single point of failure and necessitates high communication and computational resources, and decentralized execution requires transmission and synchronisation of parameters. In literature [10], a new DSA method is proposed for multichannel wireless networks that can find near-optimal policies in fewer iterations and can be applied to a wide range of communication environments. However, this method is limited as it targets at only one DSA user and does not consider the collision problem between SUs and PUs. The authors of [11, 12] employ reservoir computing or echo state networks, a type of Recurrent Neural Network (RNN), in DRL to enable distributed dynamic

spectrum access for multiple users. They mitigate the effects of spectrum sensing errors by taking advantage of the temporal correlation of RNNs, thereby reducing conflicts among users. Nonetheless, the Q-networks used are complicated and the convergence speed of the algorithm needs to be improved.

2 System Model and Problem Formulation

We consider a multi-user, multi-channel CWSN environment with N PUs and M SUs. Figure 1 depicts the intricate association of desired links and interfering links when PU_1, SU_1, and SU_2 operate on the same channel. We calculate the received signal of SU_i on channel m:

$$y_i^m = x_i^m \cdot h_{ii}^m + x_m^m \cdot h_{mi}^m + \sum_{j \in \Phi_m, j \neq i} x_j^m \cdot h_{ji}^m + z_i^m \tag{1}$$

where x_i^m represents the desired signal from SU_i on channel m, while x_m^m and x_j^m represent interfering signals from PU_m and SU_j, respectively. Similarly, the variables h_{ii}^m, h_{mi}^m, and h_{ji}^m represent the channel gain from the transmitter to SU_i at SU_i, PU_m, and SU_j, respectively. Additionally, z_i^m represents additive white Gaussian noise (AWGN). The corresponding signal to interference plus noise ratio (SINR) is:

$$SINR_i^m = \frac{p_i^m \cdot \left|h_{ii}^m\right|^2}{p_m^m \cdot \left|h_{mi}^m\right|^2 + \sum_{j \in \Phi_m, j \neq i} p_j^m \cdot \left|h_{ji}^m\right|^2 + B \cdot N_0} \tag{2}$$

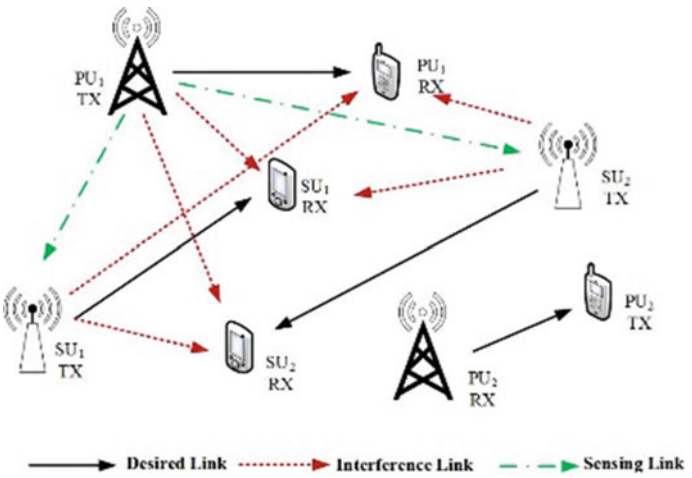

Fig. 1 System model

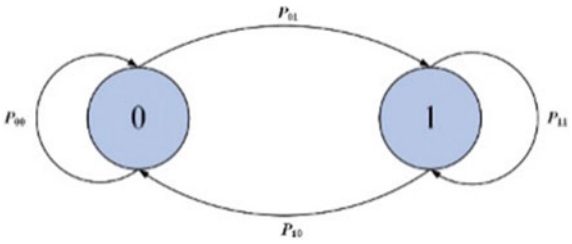

Fig. 2 Two-state Markov chain

where p_i^m, p_m^m and p_j^m denote the transmit power of users i, m and j on channel m. B and N_0 are the channel bandwidth and noise spectral density, respectively. The transmission rate C_i received by the receiver of SU_i is:

$$C_i = \log_2(1 + SINR_i) \tag{3}$$

Equations (2) and (3) show that optimal for only one SU to transmit on an inactive channel.

We divide the spectrum hole of the authorised channel into multiple time slots. The channel occupancy as a two-state Markov chain, as shown in Fig. 2, where 0 represents an occupied channel and 1 represents a vacant channel. The transition probability of the two-state Markov chain on the ith channel is:

$$p_i = \begin{bmatrix} p_{00}^i & p_{01}^i \\ p_{10}^i & p_{11}^i \end{bmatrix} \tag{4}$$

where $p_{xy} = \{\text{the next state is } x | \text{the current state is } y\}$, $(x, y \in \{0, 1\})$.

2.1 State

At the beginning of each time slot, SU_i conducts spectrum sensing on N channels to obtain information about the state of the channel. The state of the channel in the t-th time slot is expressed as follows:

$$s_i = \left[s_i^1, s_i^2, \cdots, s_i^N\right] \tag{5}$$

where $s_i^n = 1$ or $s_i^n = 0$. Since the spectrum detector is not perfect, the results of sensing the channel state may contain errors. We define the probability of sensing error for SU_i on channel n as P_i^n. Therefore, the probability of observing the true state o_i of the channel is given by:

$$\Pr(o_i) = s_i \cdot \left(1 - P_i^n\right) + (1 - s_i) \cdot P_i^n \tag{6}$$

The SU does not konw whether a spectrum sensing error will occur. Consequently, the observed results are mainly used as historical channel state data in this paper. The perception outcomes acquired by the SU in the presence of possible spectrum sensing errors are denoted as:

$$o_i = \left[o_i^1, o_i^2, \cdots, o_i^N\right] \tag{7}$$

2.2 Action

After spectrum sensing, the SU determines whether to access a channel based on the sensing result. The action of SU_i is denoted by $a_i \in \{0, 1, \cdots, N\}$, where $a_i = n(n > 0)$ indicates that at time slot t, SU_i chooses to transmit on the nth channel, while $a_i = 0$ indicates that SU_i chooses not to transmit. The action of each SU is denoted as:

$$A = \{a_1, a_2, \cdots, a_N\} \tag{8}$$

2.3 Reward

SUs receive rewards based on the actions they take. Principles for SU access to a channel include minimizing collisions with other SUs and avoiding interference with the PU to maximize their own transmission rate. The reward function is defined as:

$$r_i = \begin{cases} -\mathrm{C} & \text{, collision with PU} \\ 0 & \text{, no channel access} \\ \log_2(1 + SINR_i) & \text{, successful access} \end{cases} \tag{9}$$

Specifically, the reward is set to $-\ C\ (C > 0)$ when the SU collides with the PU, and 0 when the SU does not transmit data. Otherwise, the SU's reward is the transmission rate of its receiver.

2.4 Policy

SUs don't konw the probability of the channel state transmission and the sensing errors, so they use these rewards to form an access policy that maximizes their cumulative discounted returns, which can be expressed as:

$$R_i = \sum_{t=1}^{\infty} \gamma_{t-1} r_i(t+1) \tag{10}$$

where $\gamma \in [0, 1]$ is a discounted factor.

In summary, the ultimate goal of DSA is to maximise the reward as given in Eq. (10). The optimal Q value is calculated using the following equation to find the optimal policy π^*.

$$\pi^* = \underset{a_i \in A}{\operatorname{argmax}}\, Q_{\pi^*}(o_n, a_i) \tag{11}$$

3 Proposed DRL Algorithm

Since the efficiency of Q-learning deteriorates as the state and action space increases, we address the inefficiency of Q-learning by incorporating DNNs. The DQN architecture we use is shown in Fig. 3.

In the training phase of the DQN, as intelligent agent, each SU uses its observations at each time slot as input to the DQN evaluation network. The evaluation network selects actions using the ε-greedy strategy. After the SU takes action a_i, it receives a reward r_i from the environment and inputs channel observations $o_i\prime$ into the target network at the next time slot to obtain the next time slot action $a_i\prime$ and the target Q value $max_{a_i\prime} Q\left(o_i\prime, a_i\prime; \theta'\right)$. $(o_i, a_i, r_i, o_i\prime)$ represents an experience that is collected and stored in the experience pool by the ε-greedy strategy before training starts. The accumulated experiences in the experience pool are used to calculate the loss value during the DQN training:

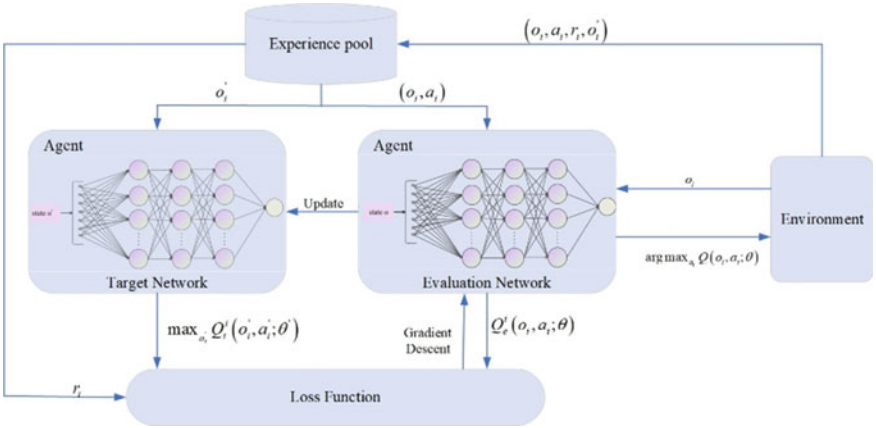

Fig. 3 The framework of DQN

$$loss = \left[r_i + \gamma \max_{a_i} Q_t^i\left(o_i\prime, a_i; \theta'\right) - Q_e^i(o_i, a_i; \theta) \right]^2 \quad (12)$$

The parameters θ of the evaluation network are updated using the calculated loss values through back propagation, and the parameters of the evaluation network are periodically copied to the target network to update its parameters θ'.

4 Simulation Results

We conducted simulation experiments in an environment where 2 SUs coexist with 6 PUs, and their positions were randomly set within a 150 m × 150 m area. The SUs were placed within range of 20–40 m from each other. We used the WINNER II and Rician models to calculate the path loss and channel model, respectively. We randomly selected p_{11} from the uniform distribution [0.7, 1] and p_{00} from [0, 0.3]. We then calculated $p_{10} = 1 - p_{11}$ and $p_{01} = 1 - p_{00}$. The parameters of the system model are shown in Table 1.

To improve the training accuracy and address the performance degradation of deep neural networks due to network depth, we designed the DNN structure in our DQN as a ResNet structure with four hidden layers, as shown in Fig. 4. Each hidden layer contains 64 neurons with Rectified Linear Unit (ReLU) as the activation function. In order to avoid sub-optimal decision strategies before gaining sufficient learning experience, we used the decaying ε-greedy algorithm with an initial value of ε set to 1. At each time slot, ε was decayed according to $\varepsilon \leftarrow \max\{0.995*\varepsilon, 0.005\}$. The hyperparameters are provided in Table 2.

We conducted simulations using Python and TensorFlow to evaluate the performance of our proposed algorithm DQN + MLP4 + ResNet against several other algorithms: myopic algorithm [13], DQN + RC [11], Q-learning, and DQN with only four fully connected layers (DQN + MLP4). We compared the algorithms based on their cumulative rewards, success rate, and conflicts with PUs and other SUs.

Our proposed algorithm has demonstrated superior performance compared to other algorithms, as shown in Figs. 5, 6, 7 and 8. Figure 5 shows that our algorithm achieved the highest average reward compared to other algorithms, while Fig. 6 shows

Table 1 Parameters of system model

Parameters	Value
Number of PUs N	6
Number of SUs M	2
Noise spectral density N_0	− 174 dBm/Hz
Transmission power of PU	40 mW
Transmission power of SU	20 mW

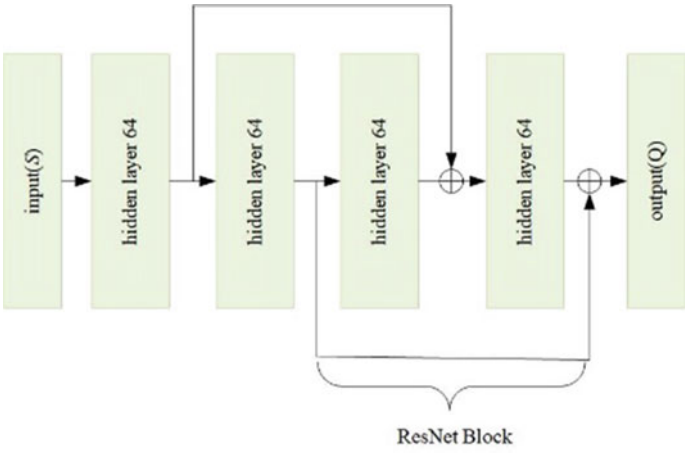

Fig. 4 The structure of deep neural networks for DQN algorithm

Table 2 Hyperparameters of DQN algorithm

Hyperparameters	Value
ε in ε-greedy policy	$1.0 \rightarrow 0.005$
Learning rate α	0.01
Discount rate γ	0.9
Activation function	ReLu
Memory size	2000
Optimzer	Adam
Target network update frequency	300

that our algorithm achieved a much higher access channel success rate, reaching approximately 95%. Figure 7 shows that all learning-based algorithms, except for the myopic policy, eventually reach a zero conflict rate with other SUs, indicating that they learn the access policies of other SUs by interacting with the environment. However, the myopic policy only accesses the channel that brings the maximum expected reward based on the known system channel information, and cannot learn the access policies of other SUs. To prevent conflicts with PUs, we set the reward to -2, and as depicted in Fig. 8, our proposed algorithm achieves the lowest collision rate with PUs, even lower than the myopic policy.

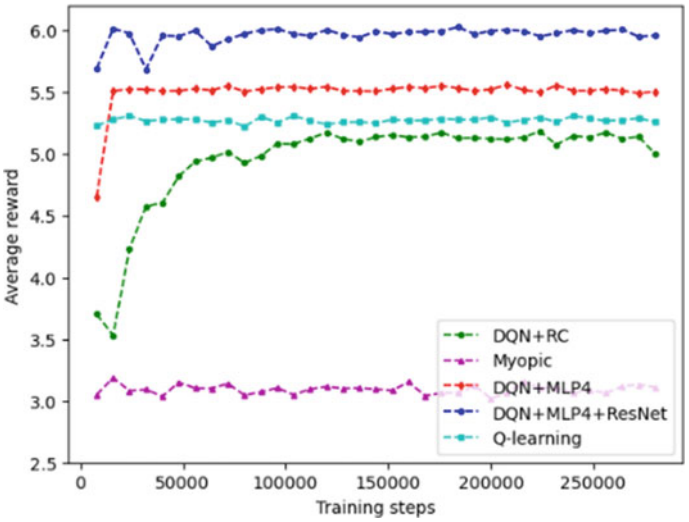

Fig. 5 The average reward

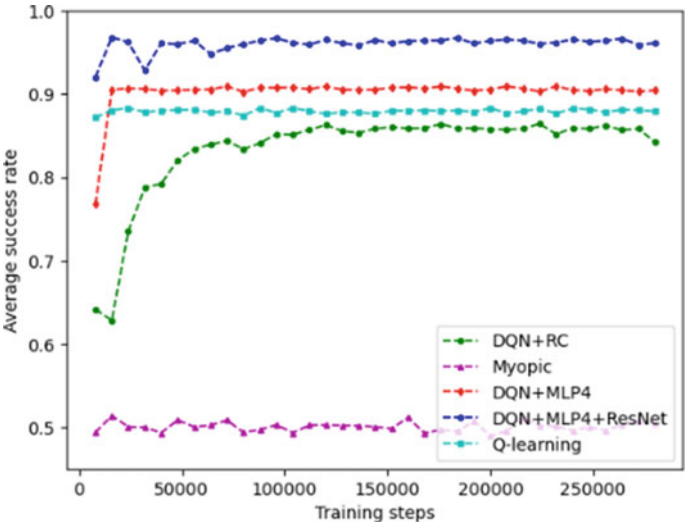

Fig. 6 The average success rate

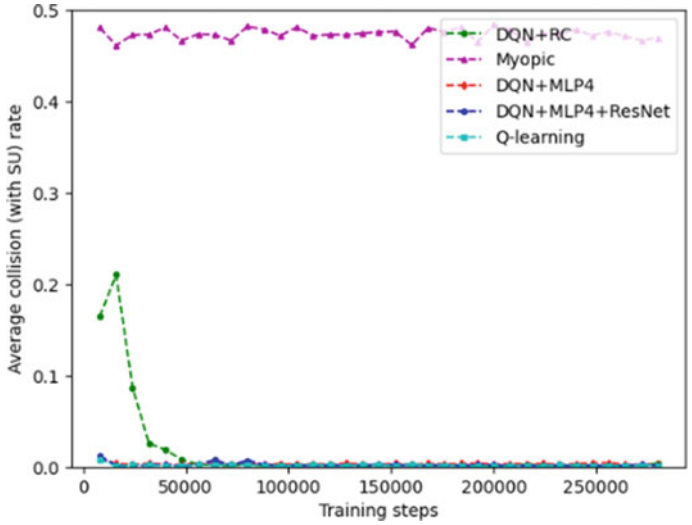

Fig. 7 The average collision with SU

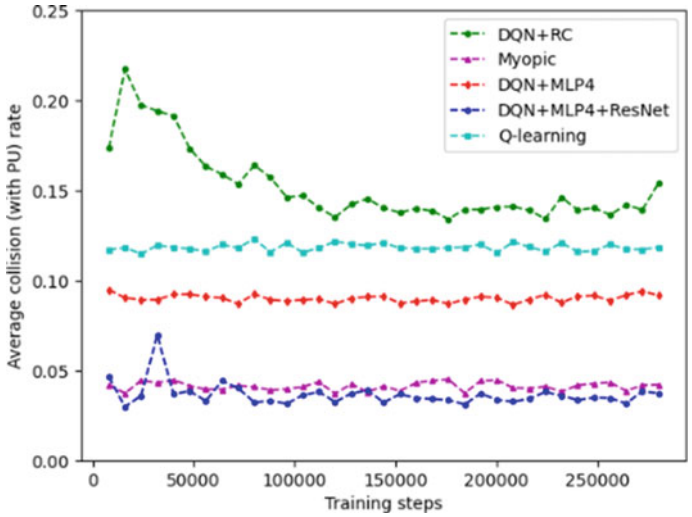

Fig. 8 The average collision with PU

5 Conclusion

This study addresses the spectrum access problem in distributed DSA networks with spectrum sensing errors, and proposes a DSA algorithm that combines DQN with ResNet. Simulation results demonstrate that the proposed DQN + MLP4 +

ResNet algorithm facilitates SUs to learn the optimal channel access policy more efficiently, improves spectrum access opportunities, and effectively reduces inter-user collisions when SUs have incomplete knowledge of the environment and face certain perception errors. In future work, we plan to consider more practical spectrum sharing scenarios and further improve the performance of the algorithm.

Acknowledgements Shubin Wang (wangshubin@imu.edu.cn) is the correspondent author and this work was supported by the National Natural Science Foundation of China (61761034).

References

1. Carie A, Li M, Marapelli B et al (2019) Cognitive radio assisted WSN with interference aware AODV routing protocol. J Ambient Intell Humaniz Comput 10:4033–4042
2. Song H, Liu L, Ashdown J et al (2021) A deep reinforcement learning framework for spectrum management in dynamic spectrum access. IEEE Internet Things J 8(14):11208–11218
3. Cai P, Zhang Y (2020) Intelligent cognitive spectrum collaboration: Convergence of spectrum sensing, spectrum access, and coding technology. Intelligent and Converged Networks 1(1):79–98
4. Qian B, Zhou H, Ma T et al (2020) Leveraging dynamic stackelberg pricing game for multi-mode spectrum sharing in 5G-VANET. IEEE Trans Veh Technol 69(6):6374–6387
5. Liu X, Sun C, Yu W et al (2021) Reinforcement-Learning-based dynamic spectrum access for software-defined cognitive industrial internet of things. IEEE Trans Industr Inf 18(6):4244–4253
6. Kaur A, Kumar K (2020) Imperfect CSI based intelligent dynamic spectrum management using cooperative reinforcement learning framework in cognitive radio networks. IEEE Trans Mob Comput 21(5):1672–1683
7. Cong Q, Lang W (2021) Double deep recurrent reinforcement learning for centralized dynamic multichannel access. Wirel Commun Mob Comput 2021:1–10
8. Doshi A, Yerramalli S, Ferrari L et al (2021) A deep reinforcement learning framework for contention-based spectrum sharing. IEEE J Sel Areas Commun 39(8):2526–2540
9. Guo Z, Chen Z, Liu P et al (2022) Multi-agent reinforcement learning-based distributed channel access for next generation wireless networks[J]. IEEE J Sel Areas Commun 40(5):1587–1599
10. Cong Q, Lang W (2021) Deep multi-user reinforcement learning for centralized dynamic multichannel access/2021. In: 6th international conference on intelligent computing and signal processing (ICSP). IEEE, pp 824–827
11. Chang HH, Song H, Yi Y et al (2019) Distributive dynamic spectrum access through deep reinforcement learning: A reservoir computing-based approach. IEEE Internet Things J 6(2):1938–1948
12. Chang HH, Liu L, Yi Y (2020) Deep echo state Q-network (DEQN) and its application in dynamic spectrum sharing for 5G and beyond. IEEE Trans Neural Netw Learn Syst 33(3):929–939
13. Li Y, Jayaweera SK, Bkassiny M et al (2012) Optimal myopic sensing and dynamic spectrum access in cognitive radio networks with low-complexity implementations. IEEE Trans Wireless Commun 11(7):2412–2423