Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9446418, 14 pages
https://doi.org/10.1155/2022/9446418

Research Article

A Dueling Deep Recurrent *Q*-Network Framework for Dynamic Multichannel Access in Heterogeneous Wireless Networks

Haitao Chen, Haitao Zhao, Li Zhou, Jiao Zhang, Yan Liu, Xiaoqian Pan, Xingguang Liu, and Jibo Wei

College of Electronic Science and Technology, National University of Defense Technology, Changsha, Hunan 410073, China

Correspondence should be addressed to Haitao Zhao; haitaozhao@nudt.edu.cn

Received 15 July 2022; Accepted 31 August 2022; Published 1 October 2022

Academic Editor: Xiaojie Wang

This paper investigates a deep reinforcement learning algorithm based on dueling deep recurrent *Q*-network (Dueling DRQN) for dynamic multichannel access in heterogeneous wireless networks. Specifically, we consider the scenario that multiple heterogeneous users with different MAC protocols share multiple independent channels. The goal of the intelligent node is to learn a channel access strategy that achieves high throughput by making full use of the underutilized channels. Two key challenges for the intelligent node are (i) there is no prior knowledge of spectrum environment or the other nodes' behaviors; (ii) the spectrum environment is partially observable, and the spectrum states have complex temporal dynamics. In order to overcome the aforementioned challenges, we first embed the long short-term memory layer (LSTM) into the deep *Q*-network (DQN) to aggregate historical observations and capture the underlying temporal feature in the heterogeneous networks. And second, we employ the dueling architecture to overcome the observability problem of dynamic environment in neural networks. Simulation results show that our approach can learn the optimal access policy in various heterogeneous networks and outperforms the state-of-the-art policies.

1. Introduction

With the development of mobile communication [1], internet of things (IoT) [2], and virtual reality (VR) [3], the global mobile data traffic increases seven-fold between 2016 and 2021 with a compound annual growth rate of 46%, and it will reach 160 exabytes per month in 2025. Efficient spectrum utilization is crucial for future wireless networks to cater to the explosive growth of mobile data traffic. Dynamic spectrum access (DSA) is believed to be one of the key technologies to improve spectrum efficiency in the limited frequency bands [4]. However, most DSA technologies either rely on the prior information of the network or cannot effectively adapt to the real wireless network environment with complex and dynamic features. Motivated by these considerations, we propose a deep reinforcement learning (DRL) framework for dynamic multichannel access in heterogeneous wireless networks.

At the first step, we consider the efficient spectrum utilization in a multichannel heterogeneous wireless network. On one hand, with the development of network technology and the opening of unlicensed frequency bands, the next-generation wireless networks will become more heterogeneous and complex due to the types of emerging radio access networks (RANs) they integrate and the types of numerous applications they support. On the other hand, in order to further improve the capacity and performance of the wireless network, multiband aggregation will be a significant trend in the next-generation network deployment. Based on these premises, the first goal of this paper is to investigate a clean-slate design in which the intelligent node shares the spectrum with these nodes from different networks in a dynamic way. There is no cooperation mechanism and information exchange between the intelligent node and other nodes. The intelligent node needs to learn the optimal channel access policy by a chain of observations and actions without any prior knowledge about the system statistics (e.g., the MAC mechanisms of other nodes and the state transition probability of each channel).

Since the intelligent node can only obtain the state of channels chosen by itself, the whole spectrum environment is partially observable. The multichannel access problem is a partially observable Markov decision process (POMDP) for the intelligent node. In theory, POMDP is polynomial-space-hard (PSPACE-hard) [5], and the increase in the dimension of states will result in double-exponential growth in complexity. Hence, the traditional DRL method, e.g., the deep Q-network (DQN), may be unable to find an optimal access policy. Thus, in this paper, the long short-term memory [6], a powerful variant of the recurrent neural network (RNN), is embedded into the DQN to capture the related state information from the historical observations and learn the dynamics of the spectrum environment. Further, considering the observability problem about the dynamic environment in DSA, we introduce the dueling architecture [7] in our neural network, which can determine the best action for each state in the large dynamic unknown environments. We refer to our proposed DRL algorithm as Dueling DRQN.

The main contributions in this work can be summarized as follows:

(i) We propose a DRL-based framework for dynamic multichannel access in heterogeneous wireless networks. In contrast to traditional DSA technologies, our scheme does not rely on any prior information of the network and can intelligently select underutilized channels by itself in each time slot

(ii) We introduce the LSTM into our neural network to overcome the partially observation problem about the spectrum environment. In addition, we add the dueling architecture in our neural network to evaluate comprehensive states and make the best decision

(iii) For the sake of avoiding Dueling DRQN falling into the local optimum, we propose an adaptive ε-greedy method to balance exploitation and exploration

(iv) Extensive simulation results demonstrate that the proposed Dueling DRQN algorithm can achieve higher throughput performance and lower collision rate compared with the baseline algorithms. Moreover, our scheme performs good adaptiveness when the spectrum environment dynamically changes over time

1.1. Related Work. Over the past decade, a variety of methods have been investigated for the DSA problem in wireless networks [8–16]. They are divided into two mainstreams in general: model-dependent methods and model-free methods. For model-dependent methods, it relies on establishing the accurate system model, based on which an effective access policy can be acquired. The myopic policy [8] and whittle index policy [9] are two classic model-dependent methods. When channels are independent and identically distributed (i.i.d.), the myopic policy performs well. But the myopic policy does not have any performance guarantee when channels are correlated or follow different distributions. The whittle index policy has similar problem. According to [9], when all channels are independent and the two-state Markov chain transition matrix is known, the whittle index policy can be regarded as a close-form solution. When all channels are identical, the whittle index policy has the same performance as the myopic policy. However, if there are some channels that cannot be modelled in the form of two-state Markov transition matrices, the performance of these two model-dependent methods will deteriorate.

Instead, for model-free methods, the optimal policy is acquired through interacting with the environment. The reinforcement learning (RL), a typical representative of model-free methods, is widely applied to investigate the DSA problems [12–16]. The most popular algorithm is Q -learning since it is easy to implement and has better performance than traditional algorithms. However, Q-learning needs to maintain a tabulation to store and update the Q -value during the interaction with the environment. When the state space or action space becomes larger, or the environment is partially observable, the complexity of tabulating the Q-values is fairly high, and the performance of Q -learning becomes inefficient. Fortunately, by introducing the deep neural network in RL, DRL can deal with these problems with very large state and action spaces, which has attracted more and more attention in recent years [17, 18].

Many existing works have adopted DRL to solve the DSA problems. For example, authors in [19] examined the correlated channels in DSA, which is a pioneer work using the DRL for the channel access. In [20], a deep actor-critic reinforcement learning based framework for dynamic multichannel access was proposed in both the single-user and multiple-user scenarios. The simulation results showed that the proposed framework had good performance in handling a large number of channels. In addition, a methodology that utilized DRL to implement a fair multichannel access strategy in a distributed wireless network (DWN) was developed in [21]. The authors removed the assumption that all nodes in DWN are in the saturated mode, which is closer to the practical scenarios. In [22–25], the authors concentrated on the DRL-based DSA strategies in cognitive radio network (CRN). However, all the network scenarios considered in these works are homogeneous. Unlike [19–25], in this paper, we pay more attention to the coexistence of nodes adopting various MAC protocols in the shared multichannel heterogeneous network. The spectrum environment is more complicated than these works. In addition, our scheme needs to keep flexible and efficient spectrum access strategies so that it can live in harmony with other nodes.

There are a few works that investigate the DSA problem in heterogeneous networks, such as [26–29]. However, the authors in [26, 27] only considered a single channel that is shared by the intelligent node and other nodes. The intelligent node only needs to decide whether transmit or not, which greatly reduces the difficulty of the DSA problem. Both [28, 29] investigated the joint user association and spectrum allocation issue in multichannel heterogeneous networks and proposed the multiagent DRL schemes to solve the computationally expensive optimization problem with the large state and action space. However, they hold an assumption that each user can obtain the global state space about other users in the network and ignore the partial observability about external environment. Actually, real-world tasks often feature incomplete state information caused by partial observability [30]. It is extremely difficult to collect the global state information because of the large communication overhead and processing cost. While in this paper, our proposed scheme does not rely on the global state information, and the intelligent node strives to achieve optimal performance based on limited information about spectrum environment. Table 1 summarizes the differences of our proposed scheme, comparing to some existing works that previously reported.

The rest of the paper is organized as follows. Section 2 introduces the system model and formulates the DSA problem by POMDP. Section 3 proposes the Dueling DRQN algorithm and explains it in detail. Simulation results and performance analyses are presented in Section 4. Finally, we conclude this paper in Section 5.

2. System Model and Problem Formulation

2.1. System Model. As shown in Figure 1, the system model we consider in this paper is a time-slotted multichannel (including N orthogonal channels) heterogeneous wireless network where multiple radio networks coexist with each other. Each radio network is independent of each other, and the channels for different networks have been preallocated by AP, which means that different radio networks do not interfere with each other. We assume that all the nodes are backlogged, i.e., they always have packets to transmit. A node can transmit at the beginning of a time slot and must finish transmission within that time slot. For the AP, it can receive data from different radio networks on different channels in each time slot simultaneously. Thereafter, AP can broadcast corresponding ACK information on different channels to indicate whether the transmissions are successful or not.

The entire network is heterogeneous because different radio networks may adopt different time-slotted MAC protocols. Specifically, we assume that each node chooses one of N orthogonal channels at each time slot to transmit its data packets. The types of nodes and their transmission manners are described below:

(i) *Authorized Node*. Occupies a fixed channel and transmits all the time

(ii) *TDMA Node*. Occupies a fixed channel, but transmits in X specific time slots within each frame of Y time slots in a repetitive manner from frame to frame

(iii) *Hopping Node*. Dynamically occupies several channels which change at each time slot following a fixed regular pattern

(iv) *q -ALOHA Node*. Occupies a fixed channel and transmits with a probability q in each time slot in an i.i.d. manner

(v) *Stochastic Node*. Occupies a fixed channel based on a probability distribution such as a two-state Markov chain

As depicted in Figure 2, numerous nodes belonging to different radio networks and their own transmission manners jointly lead to a complex and dynamic spectrum environment. Meanwhile, we can find there still exist many idle spectrums that are not fully utilized due to the fixed channel preallocation method. An intuitive motivation of this paper is how to improve the spectral efficiency of the network on the premise of overcoming the complex and dynamic external spectrum environment. Specifically, the DRL node is an intelligent node that implements our proposed Dueling DRQN algorithm. The DRL node needs to adopt an appropriate access strategy to make use of the idle spectrum and avoid collisions with other nodes.

However, there are two key challenges for the DRL node. First, the prior system information, such as the MAC mechanism of different nodes and the channel state information, is not known by the DRL node. Second, the DRL node can only obtain the state of the selected channel through the ACK signal from the AP. In other words, the whole channels' states are partially observable from the perspective of the DRL node. The above challenges make these classical approaches for DSA ineffective and even difficult to implement. Thus, a new online learning method is needed.

2.2. Problem Formulation. Because the prior system statistics are unknown and the full channel state information is partially observable by the DRL node, we use the POMDP to model the dynamic multichannel access process, which is described by the action, state, reward, and state transition function.

2.2.1. Action. The action of the DRL node is to select which channel to access in each time slot, and all possible actions constitute the action space $\mathscr{A}$, $a_t \in \mathscr{A}$. The action is denoted as a_t at time slot t, $a_t = \{a_{1,t}, a_{2,t}, \cdots, a_{N,t}\}$, where

$$a_{i,t} = \begin{cases} 1, & \text{if the DRL node chooses channel } i \text{ to access,} \\ 0, & \text{other.} \end{cases} \tag{1}$$

2.2.2. State. Due to the different transmission strategies of these existing nodes, the channel states will change at each time slot. We define the full channel states at time slot t as Ω_t, which is

$$\Omega_t = \{\omega_{1,t}, \omega_{2,t} \cdots, \omega_{i,t}, \cdots, \omega_{N,t}\}, 1 \le i \le N, \tag{2}$$

where $\omega_{i,t}$ is the binary representation of the state of channel i: occupied (1) or idle (-1). After taking action a_t, the DRL

Table 1: Differences between the proposed scheme and existing works in DSA problem.

	Network scenarios	Model-dependent or model-free	Multichannel access	Partial observability	Time-varying environment
[8–11]	Homogeneous networks	Model-dependent	Yes	No	No
[12–16]	Homogeneous networks	Model-free	Yes	No	No
[19]	Homogeneous networks	Model-free	Yes	Yes	No
[20]	Homogeneous networks	Model-free	Yes	Yes	Yes
[21]	Heterogeneous networks	Model-free	No	Yes	No
[22, 23, 25]	Homogeneous networks	Model-free	Yes	No	No
[24]	Homogeneous networks	Model-free	Yes	Yes	No
[26–27]	Heterogeneous networks	Model-free	No	No	No
[28–29]	Heterogeneous networks	Model-free	Yes	No	No
Ours	Heterogeneous networks	Model-free	Yes	Yes	Yes

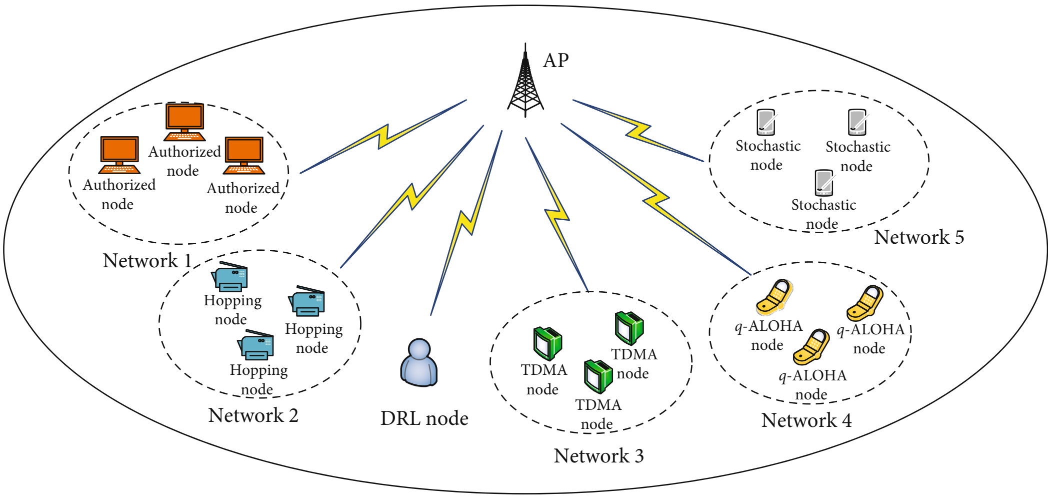

Figure 1: The architecture of heterogeneous wireless networks.

node gets an observation $o_t = \{o_{1,t}, o_{2,t} \cdots, o_{i,t}, \cdots, o_{N,t}\}$. If the DRL node chooses channel i to access,

$$o_{i,t} = \begin{cases} 1, & \text{transmission is successful} \\ -1, & \text{transmission is collided} \end{cases} \tag{3}$$

for $j \neq i$, $o_{j,t} = 0$, which represents the state of channel j is unknown by the DRL node. The state of the DRL node at time slot $t+1$ is defined as the past l -length action-observation pairs up to the time slot t, i.e.,

$$s_{t+1} = (a_{t-l+1}, o_{t-l+1}, a_{t-l+2}, o_{t-l+2}, \cdots a_t, o_t). \tag{4}$$

2.2.3. Reward. After performing the action a_t, the transition from state s_t to s_{t+1} generates a reward r_{t+1}. The immediate reward r_{t+1} is based on the successful transmission or conflicting transmission. Specifically, the reward r_{t+1} is defined as

$$r_{t+1} = \begin{cases} 1, & \text{if } o_{i,t} = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Note that the object of our proposed scheme is to improve the spectrum utilization of the network while avoiding collisions. However, the spectrum efficiency is related to the number of successful transmissions directly.

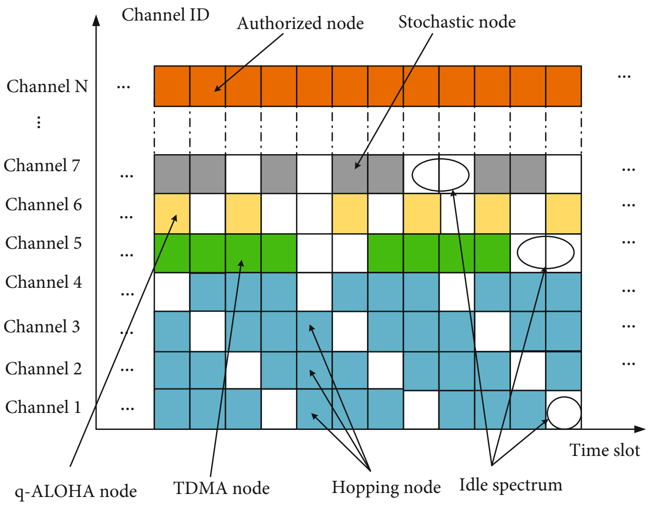

Figure 2: Complex and dynamic spectrum environment.

Thus, we pay more attention to the successful transmission in the following sections of this paper. In addition, the transmission is failed if the collision occurs.

2.2.4. State Transition Function. At time slot t, the state transition function is denoted as ψ_t, and ψ_t is the probability that the DRL node executes action a_t at the state s_t and moves to the next state s_{t+1}. As interaction times increase, ψ_t will gradually converge to the optimal ψ_t^* by maximizing the long-term accumulated discounted reward, i.e.,

$$\psi_t^* \longleftarrow \max \sum_{t=0}^{\infty} \gamma^t r_{t+1}, \tag{6}$$

where $0 \le \gamma \le 1$ is a discounted factor indicating the important degree of future reward to the current reward.

3. The Proposed Dueling DRQN

The framework of Dueling DRQN includes experience replay, deep neural networks, and reinforcement learning, which is shown in Figure 3.

3.1. Experience Replay. During the learning process, the experience-replay pool $\mathscr{D}$ [31] stores the current state, action, reward, and the next state as a tuple $(s_t, a_t, r_{t+1}, s_{t+1}$), which is referred to as experience E_t at time slot t. The experience-replay pool $\mathscr{D}$ accumulates the history experiences for many time slots. When training the Q-network, the network is trained by a minibatches M_E of experiences that are sampled from $\mathscr{D}$ randomly, rather than only using current experience. Experience replay can increase data efficiency through the reuse of experience samples. More importantly, it reduces the correlation among the training data by randomly sampling data in experience-replay pool.

3.2. Deep Neural Networks. The mapping from states to actions follows the policy π, i.e., $a_t \sim \pi(s_t)$. The performance of policy π is evaluated by the state-action value function $Q^\pi(s, a)$, which is denoted as the expected accumulated discount reward for executing action a at the state s following the policy π, i.e.,

$$\mathscr{R}_t = \sum_{t=0}^{\infty} \gamma^t r_{t+1},$$
$$Q^\pi(s, a) = \mathbb{E}[\mathscr{R}_t | s_t = s, a_t = a, \pi]. \tag{7}$$

Moreover, deep neural networks are used to fit the state-action value function by $Q^\pi(s, a; \theta)$, rather than computing and storing the true $Q^\pi(s, a)$ in Q-table, i.e., $Q^\pi(s, a; \theta) \approx Q^\pi(s, a)$. The parameter θ is the set of weights in the deep neural networks. The estimation network and the target network have the same network structure but different hyperparameters.

Generally, deep neural networks have the strong nonlinear mapping and fitting ability. The reason is that the networks contain multiple hidden layers. However, for traditional deep neural networks (e.g., FNN), the output of each hidden layer depends only on the current input. In order to accommodate the partial observation, we should make full utilization of historical observations and infer the true channel state based on the historical experiences. Therefore, we improve the deep neural networks in our proposed algorithm by adding an LSTM layer and the dueling architecture. Figure 4 shows the detailed structure of deep neural networks in our proposed algorithm.

3.2.1. Long Short-Term Memory Network (LSTM). LSTM is a variant of the recurrent neural network, which overcomes the gradient exploding problem or the gradient vanishing problem in RNNs by introducing a gating mechanism. The LSTM layer maintains an intermediate state and aggregates observations over time, which is capable of capturing the information about historical observations. Therefore, we add an LSTM layer to estimate the true channel states and learn the dynamic pattern of the networks.

In LSTM, the intermediate state C_t at time slot t is used to retain historical information about channel states from initial time slot to time slot t. For each time slot t, C_t is determined by both the forget gate F_t and input gate I_t. First, the forget gate determines how much information stored in C_{t-1} needs to be forgotten, which is given by

$$F_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right), \tag{8}$$

where $\sigma = 1/(1 + e^{-x})$ is the logistic function [32]. x_t is the current input matrix. h_{t-1} is the hidden state at time slot $t - 1$, which can be regarded as the short-term memory. W_f is the weight matrices of F_t. b_f is the bias term of F_t. Second, the input gate I_t controls how much information stored in the candidate state $\tilde{C}_t$ should be retained to update C_t. We calculate the input gate values I_t and the candidate state $\tilde{C}_t$ as

$$I_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i), \tag{9}$$

$$\tilde{C}_t = \tanh\,(W_c \cdot [h_{t-1}, x_t] + b_c), \tag{10}$$

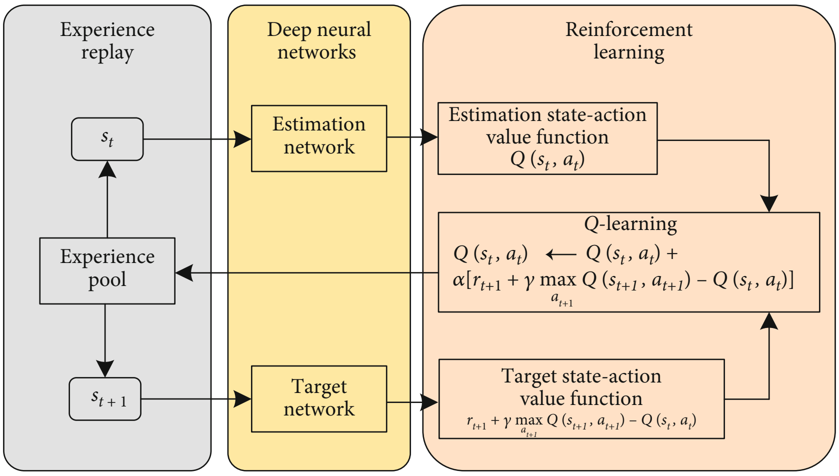
Experience replay
Deep neural networks
Reinforcement learning
s_t
Estimation network
Estimation state-action value function $Q(s_t, a_t)$
Q-learning
$Q(s_t, a_t) \longleftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$
Experience pool
s_{t+1}
Target network
Target state-action value function
$r_{t+1} + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)$

FIGURE 3: The framework of Dueling DRQN.

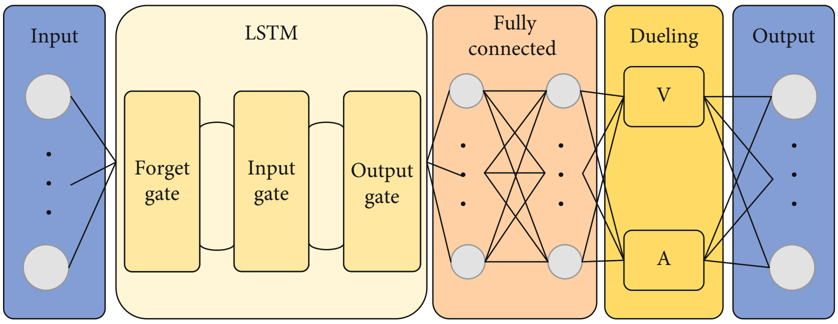

FIGURE 4: The structure of deep neural networks for Dueling DRQN algorithm.

where W_i and W_c are the weight matrices of I_t and $\tilde{C}_t$, respectively. b_i and b_c are the bias terms of I_t and $\tilde{C}_t$, respectively. $\tanh = (e^x - e^{-x})/(e^x + e^{-x})$ denotes the hyperbolic tangent function [33]. Given the values of F_t and I_t, the intermediate state C_t can be updated as

$$C_t = F_t \odot C_{t-1} + I_t \odot \tilde{C}_t, \quad (11)$$

where the symbol $\odot$ is the element multiplication.

The output gate O_t is used to control the amount of the information in C_t exported to the hidden state h_t at time slot t. O_t and h_t can be obtained as

$$O_t = \sigma(W_o[h_{t-1}, x_t] + b_o), \quad (12)$$

$$h_t = O_t \odot \tanh(C_t), \quad (13)$$

where W_o is the weight of O_t, and b_o is the bias of O_t.

In the LSTM model, C_t can capture the key information from x_t at each time slot and save it for a certain time interval. Based on C_t, the DRL node can predict the true channel state well. Finally, we can get the output of the LSTM as

$$y_t = W_y \cdot C_t + b_y, \quad (14)$$

where W_y and b_y are the output weights and the output bias, respectively.

3.2.2. Dueling Architecture. Factually, there is an observability problem about dynamic environment in the neural network. The state may be good or bad regardless of the taken action when there are multiple idle channels. Therefore, it is desirable to estimate the state-action value function $Q(s_t, a_t; \theta)$ by the value of state (i.e., $V(s_t; \theta)$) and the advantage of each action (i.e., $A(s_t, a_t; \theta)$) [7], which can be expressed as

$$Q(s_t, a_t; \theta) = V(s_t; \theta) + A(s_t, a_t; \theta), \quad (15)$$

where $V(s_t; \theta)$ estimates the expected value of state with respect to the taken action, and $A(s_t, a_t; \theta)$ denotes the advantage of each action minus the average value of the state with respect to the taken actions, i.e., $A(s_t, a_t; \theta) - (1/|\mathscr{A}|) \sum_{a' \in \mathscr{A}} A(s_t, a'; \theta)$. Copying the output streams from the fully connected layer and leading them to the value layer (V layer) and the advantage layer (A layer), we will get the expected value of the state by V layer and the comparative advantage of each action by A layer, which can evaluate the state-action value function more precisely and reduce redundant operations in action sampling.

Through the LSTM and dueling architecture, the estimation network approximates the estimation state-action value function eval_$Q = Q^{\pi}(s_t, a_t\,; \theta)$ with a set of weights θ, and the target network approximates a target state-action value functiontarget_$Q = r_{t+1} + \gamma \max_{a_{t+1} \in \mathscr{A}} Q^{\pi}(s_{t+1}, a_{t+1}\,; \theta^{-})$ with a set of weights θ^{-}. The loss function of Dueling DRQN is given as

$$L(\theta) = \mathbb{E}_{s_t, a_t, r_t, s_{t+1}}\left[(\text{target\_}Q\text{-eval\_Q})^2\right]. \tag{16}$$

The parameter of the estimation network θ is updated at every time slot t by

$$\theta \longleftarrow \theta + \alpha[\text{target\_}Q\text{-eval\_Q}]\nabla_{\theta}\text{eval\_Q}, \tag{17}$$

where $\alpha \in (0, 1]$ is the learning rate. ∇_{θ} denotes the gradient of θ. Finally, the parameter of the target network θ^{-} is updated to the parameter of the estimation network θ every K training times.

3.3. Q-Learning. Through the deep neural networks mentioned above, the output layer generates the state-action value functions corresponding to different actions, i.e., $Q^{\pi}(s_t, \forall a_t \in \mathscr{A};;\theta)$. For the sake of avoiding the DRL node falling into local optimal solution, it is necessary to trade off the exploitation (using the best known action) and the exploration (learning new, possibly better actions). The ε – greedy policy is often used to keep this balance, which is described as

$$a_t = \begin{cases} \arg\max\limits_{\tilde{a} \in \mathscr{A}} Q^{\pi}(s_t, \tilde{a}), & \text{with probability } 1\text{-}\varepsilon, \\ \text{random action}, & \text{with probability } \varepsilon, \end{cases} \tag{18}$$

where the DRL node selects an exploitation action with probability $1 - \varepsilon$ and selects an exploration action with probability ε.

The exploration probability is a key hyperparameter that directly affects the DRL node's decision. If ε is a small value, the DRL node becomes conservative, and it prefers to select the best known action based on the state-action value function. On the contrary, if ε is a large value, the DRL node becomes aggressive, and it prefers to select the new and unknown action. In the initial stage of algorithm implementation, due to the large state and action space, the DRL node should explore different possible states and actions frequently to enrich its knowledge about the environment. However, with the increasing number of iterations, the exploitation probability should increase accordingly so that the DRL node can make the best decision. Therefore, we design an adaptive ε – greedy policy, which is expressed as

$$\varepsilon = \varepsilon_{\min} + (\varepsilon_{\max} - \varepsilon_{\min})e^{-\varsigma t}, \tag{19}$$

where $\varepsilon_{\max}$ and $\varepsilon_{\min}$ are the maximal value and the minimal value of ε, respectively. ς is the decay factor.

After acquiring the action a_t by the adaptive ε – greedy policy, the DRL node obtains the corresponding reward r_{t+1} and the new state s_{t+1} from the environment. The Q -learning will update $Q^{\pi}(s_t, a_t\,; \theta)$ following Bellman equation

$$\begin{aligned} Q^{\pi}(s_t, a_t\,; \theta) \longleftarrow{} & Q^{\pi}(s_t, a_t\,; \theta) \\ & + \alpha\left[r_{t+1} + \gamma \max_{a_{t+1} \in \mathscr{A}} Q^{\pi}(s_{t+1}, a_{t+1}\,; \theta^{-}) - Q^{\pi}(s_t, a_t\,; \theta)\right]. \end{aligned} \tag{20}$$

With the training times increase, the state-action value function gradually converges to the optimal, and the policy π approaches the best policy π^* by maximizing the long-term accumulated discounted reward

$$\pi^* \longleftarrow \max \sum_{\tau=\tau_0}^{t} Q^{\pi^*}(s_{\tau}, a_{\tau}\,; \theta). \tag{21}$$

3.4. Dueling DRQN Training. The pseudocode of Dueling DRQN is given in Algorithm 1. To be specific, the training process is given as follows. Step 1: input the current state s_t into the estimation network and obtain the state-action value function for different actions. Step 2: choose an action a_t according to the adaptive ε – greedy policy. Step 3: adopt this action to obtain observation o_t and the reward r_{t+1}. Step 4: generate the next state s_{t+1} and store the experience tuple $(s_t, a_t, r_{t+1}, s_{t+1})$ into experience pool $\mathscr{D}$. Step 5: compute the target_Q and train the estimation network. The hyperparameters of the estimation network are updated in real time. Step 6: update the hyperparameters of the target network every K time slots.

4. Experimental Results and Discussion

In this section, we investigate and analyze the performance of the proposed Dueling DRQN algorithm through simulations with Tersorflow1.19.0 on Python3.7 platform. We first describe the simulation settings, which are summarized in Table 2. After that, we investigate the coexistence of our proposed access scheme with other protocols mentioned above.

4.1. Simulation Setting

(1) *Hyperparameter Settings.* As shown in Figure 4, the number of neurons in LSTM layer and the fully connected layer are 128. The activation functions used for the neurons are ReLU functions [34], and the Adam algorithm [35] is used to conduct a stochastic gradient descent for the update of θ. The state history length l is 16. And the learning rate α of the DRL node is set to 0.001. The experience-replay pool size is set to1000, and the experience-replay pool is updated in a first-input-first-output manner: the oldest experience will be replaced by the new experience in it. The minibatch size M_E is set to 64. More detailed parameter settings are listed in Table 2

(2) *Performance Metric.* The main performance metric we consider in the paper is "throughput," which is

```
Initialize $s_0$,$\alpha$,$\gamma$,$\varepsilon_{\max}$,$\varepsilon_{\min}$,$\varsigma$,$K$,$l$,$N$.
Initialize experience pool $\mathscr{D}$ and mini-batches $M_{\mathrm{E}}$.
Initialize the parameter of the estimation network as $\theta$.
Initialize the parameter of the target network $\theta^- = \theta$.
For episode $i = 0, 1, \cdots, I$do
  For time-slot $t = 0, 1, \cdots, T$do
    Input $s_t$ into the estimation network and output the $\{Q^\pi(s_t, a\,;\theta)|a \in \mathscr{A}\}$;
    Select the action $a_t$ using the adaptive $\varepsilon - greedy$policy algorithm
        And update $\varepsilon$ according to the equation (19);
    Execute action $a_t$ and generate the observation $o_t$ and $r_{t+1}$;
    Compute $s_{t+1}$ from $s_t$,$a_t$;
    Store($s_t, a_t, r_{t+1}, s_{t+1}$) into the experience-replay pool $\mathscr{D}$.
    If$t \geq M_{\mathrm{E}}$then
      Randomly generate an index subset $\Theta$;
        Sample $M_{\mathrm{E}}\{(s_j, a_j, r_{j+1}, s_{j+1})\}_{j\in\Theta}$ from $\mathscr{D}$;
        For each sample $(s_j, a_j, r_{j+1}, s_{j+1})$ in $M_{\mathrm{E}}$do
          Compute the target_$Q = r_{j+1} + \gamma \max_{a_{j+1}\in\mathscr{A}} Q^\pi(s_{j+1}, a_{j+1}\,;\theta^-)$ and obtain $a_{j+1}$.
        End for
        Calculate the loss function according to the equation (16) and update $\theta$ according to the equation (17);
        Minimize the loss function with learning rate $\alpha$.
    End if
    Every $K$ time slots: Update $\theta^-$ by setting $\theta^- = \theta$.
  End for
End for
```

Algorithm 1: Training process of Dueling DRQN.

Table 2: Parameter settings.

Parameter	Values
Episodes	20
The number of time slots in one episode	5500
State history length (l)	16
Experience-replay pool size	1000
Experience-replay minibatch size	64
Discount factor γ	0.9
Learning rate α	0.001
The maximal exploration probability $\varepsilon_{\max}$	0.8
The minimal exploration probability $\varepsilon_{\min}$	0.001
The decay factor ς	0.001
Target network update frequency K	100

defined as the probability of successful transmission for each episode, i.e., $T_\tau = \sum_{\tau=t-L+1}^{t} r_\tau / L$,where L is the number of time slots in one episode

(3) *Baselines*. To evaluate our proposed access policy, we consider the following baselines in this paper

   (i) *DQN Access Policy*. The node accesses one channel in each time slot using the standard DQN [19]

   (ii) *Whittle Index Policy*. The multichannel access strategy proposed in [9]

   (iii) *Random Access Policy*. The node randomly accesses one channel in each time slot

   (iv) *Optimal Access Policy*. The node is the model-aware node, and it has all the prior knowledge about the heterogeneous networks and the MAC mechanisms of the coexisting nodes, so it can access the channels using the optimal scheme

4.2. Simple Heterogeneous Network Scenarios. In this section, we investigate the performance of the proposed algorithm in four different simple heterogeneous network scenarios. Specifically, we set the number of channels is four, i.e., $N = 4$. Every scenario includes one authorized node and three other nodes which operate the same MAC protocol (there are only two hopping nodes in case II, and the details will be described in the following). The descriptions of four scenarios and related settings are summarized as follows:

(1) *Case I*. Coexistence with authorized node and TDMA node: in this case, we consider the coexistence of the DRL node with other protocol-based nodes including one authorized node and three TDMA nodes. The authorized node occupies channel 1 and transmits at all times. Three TDMA nodes occupy the rest three channels, respectively. The TDMA frame Y is set to 10, and the transmission slots X assigned to each TDMA node are slot 8, 5, and 2. It is noted that each TDMA node occupies

the according channel for consecutive X time slots and idle for the rest $Y-X$ time slots in a frame

(2) *Case II*. Coexistence with authorized node and hopping nodes: in this case, the DRL node coexists with one authorized node and two hopping nodes. The authorized node occupies channel 1 and transmits at all times. Two hopping nodes occupy the rest three channels, and the dynamic transmission pattern of the hopping nodes is $C2C3 \longrightarrow C3C4 \longrightarrow C4C2 \longrightarrow C2C3$

(3) *Case III*. Coexistence with authorized node and q-ALOHA nodes: in this case, the DRL node coexists with one authorized node and three q-ALOHA nodes. The authorized node occupies channel one and transmits at all times. Three q-ALOHA nodes occupy the rest three channels, and the transmission probability of each q-ALOHA node in different channels is [0.6, 0.9, 0.3]

(4) *Case IV*. Coexistence with authorized node and two-state Markov nodes: in this case, the DRL node coexists with one authorized node and three two-state Markov nodes. The authorized node occupies channel 1 and transmits at all times. The three two-state Markov nodes occupy the rest channels, and the transmission transition probability of these nodes on the n-th channel follows a two-state Markov chain

$$\mathscr{P}_n = \begin{pmatrix} p_{00}^n & p_{01}^n \\ p_{10}^n & p_{11}^n \end{pmatrix}, \tag{22}$$

where $p_{ij} = \Pr(\text{state}_j | \text{state}_i)$, and $\Pr(\cdot)$ denotes the transition probability. state_i represents the channel state of last time slot, and state_j represents the channel state of current time slot.

Figure 5(a) shows the experiment result of the coexistence with authorized node and TDMA nodes. As we can see, both the Dueling DRQN and DQN can quickly learn the variation characteristics of channel state and capture the idle channel without the transmission schedule of the TDMA nodes. Compared to the whittle index policy and random access policy, the throughput growths of the Dueling DRQN are more than 30% and 40%, respectively. In the whittle index policy, the node estimates the state transition probability matrices of all channels and selects one channel to access. However, because the frame length and the transmission time slots of the TDMA nodes are stationary, the state transition model of the TDMA channel cannot be captured by the two-state Markov transition model in the whittle index policy. This is why although the whittle index policy has the better performance than the random access policy, it is unable to make full utilization of the idle channels in this scenario. In addition, since all time slots are aligned and the transmission schedule of the TDMA nodes is fixed, we can calculate that the highest throughput achieved by the optimal access policy is $1 - \min(2/10, 5/10, 8/10) = 0.8$ in each episode.

Figure 5(b) presents the result of the coexistence with authorized node and hopping nodes. Even though the Dueling DRQN and the DQN achieve the approximate throughput, the Dueling DRQN has a faster convergence speed and smaller variance than the DQN during the convergence stage. For one thing, the idle slots are sparsely distributed in an episode, and there is only one idle channel in each time slot. In addition, the channel with idle slot switches according to the round-robin scheduling [14], which increases the difficulty of learning and convergence of DQN. For another thing, the Dueling DRQN can aggregate and make full utilization of historical data to predict the idle channel in the next time slot. Therefore, it is more efficient than the DQN. The Dueling DRQN outperforms the whittle index policy and the random access policy with the throughput improvement of 60% and 65%, respectively. In this scenario, the whittle index policy also performs poorly since the state transition process of all channels does not follow a two-state Markov transition model. Besides, the sparsity of underutilized channels further degrades the whittle index policy. Since there is always an idle channel in each time slot, so the optimal throughput in theory is one.

Figure 5(c) gives the result of the coexistence with authorized node and q-ALOHA nodes, and Figure 5(d) gives the result of the coexistence with authorized node and two-Markov nodes. Compared with case I and case II, the spectrum states in case III and case IV are more stochastic. In addition, the channel states have the temporal correlation in case IV. As we can see, the Dueling DRQN can reach convergence during $4 \sim 5$ episodes, but the DQN reaches the convergence during $8 \sim 10$ episodes. The reason is that the LSTM layer not only has the memory ability but also has the ability to infer the temporal correlation of channel states. However, the neural networks in standard DQN are fully connected layers that do not have the ability to memorize and ratiocinate. Therefore, the learning process of the Dueling DRQN is quicker than the DQN. In addition, compared to the whittle index policy and the random access policy, the throughput improvement of the Dueling DRQN is more than 30% and 45% in case III, respectively. However, in case IV, the two-state Markov transition model in the whittle index can capture the behavior of the two-Markov nodes absolutely. Thus, it can make full utilization of the underutilized channels and achieve the near-optimal performance. It is worth noting that the Dueling DRQN can also obtain a performance similar to the whittle index policy after a period of learning. However, the whittle index policy relies heavily on the prior system information, while our scheme is model-free. Due to the stochastics of channel states, there may not be any available channel in one time slot. Thus, we count the idle channels in all time slots and get the optimal throughputs in case III and case IV are 0.9 and 0.93, respectively.

4.3. Complex Heterogeneous Network Scenarios. For further analysis, we study the performance of our scheme in two more complicated heterogeneous network scenarios.

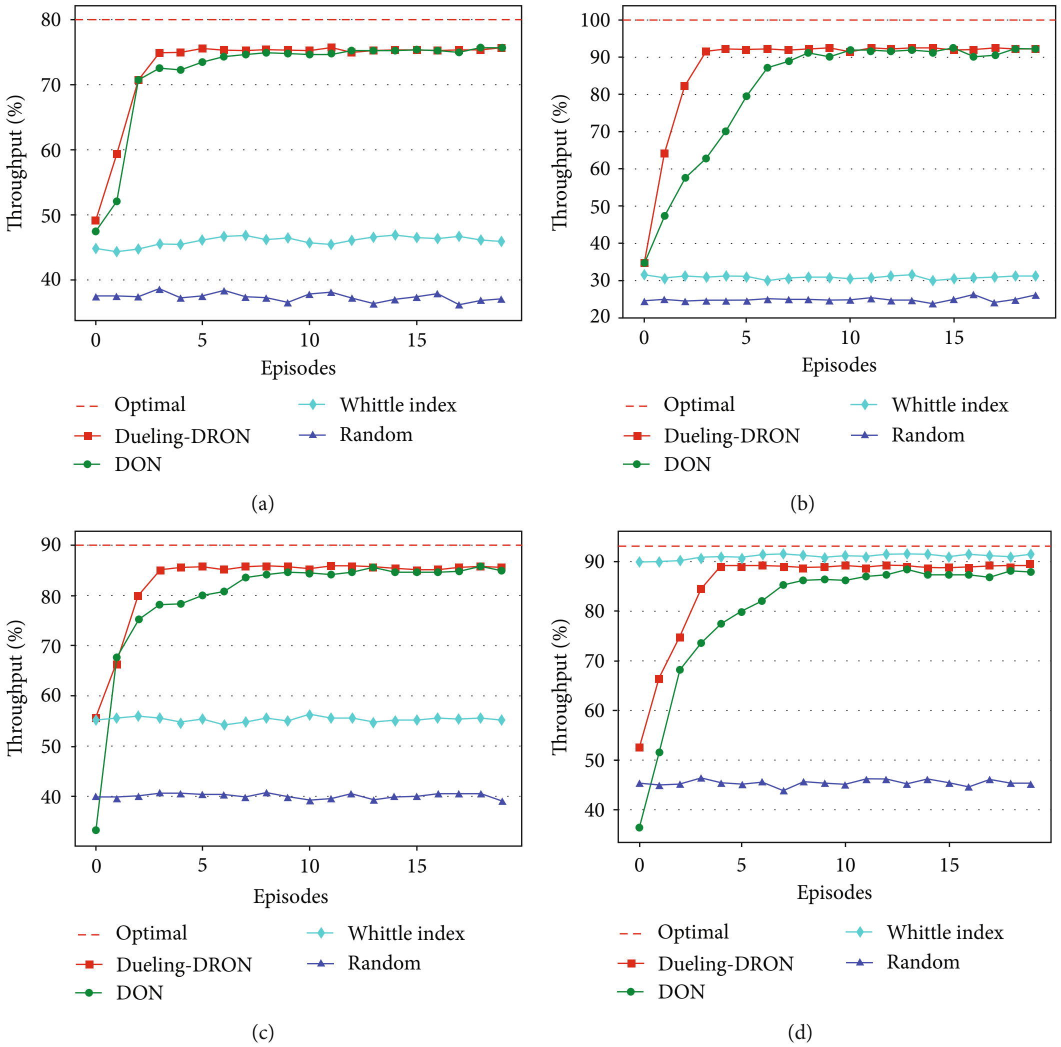

Figure 5: The throughput of the DRL node using different spectrum access policies in different cases. (a) Case I. (b) Case II. (c) Case III. (d) Case IV.

Compared with the simple network scenarios, the complex network scenarios contain multiple heterogeneous terminals from different networks, and these terminals coexist each other with different MAC protocols. The network scale becomes lager, and the spectrum states are more complex. The two scenarios are described as follows.

(1) *Complex Scenario I*. The number of channels is set to 16, i.e., $N = 16$. The heterogeneous wireless networks include two authorized nodes, two TDMA nodes, three hopping nodes, and eight q-ALOHA nodes. Two authorized nodes occupy channel 1 and channel 16, respectively. They transmit packets all the time. Two TDMA nodes occupy channel 6 and channel 15, respectively. One TDMA node transmits 15 time slots within a frame of 16 time slots, and the other node transmits 14 time slots within a frame of 16 time slots. Three hopping nodes dynamically occupy channels 2, 3, 4, and 5 with the patterns: $C2C3C4 \longrightarrow C3C4C5 \longrightarrow C4C5C2 \longrightarrow C5C2C3 \longrightarrow C2C3C4$. Eight q-ALOHA nodes occupy the rest channels 7, 8, 9, 10, 11, 12, 13, and 14; and the transmission probability of each q-ALOHA node in different channels is [0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]

(2) *Complex Scenario II*. The number of channels is set to 16, i.e., $N = 16$. The network environment is the same as the complex scenario I basically. The only difference is the two-Markov nodes replace the q-ALOHA nodes and occupy the channels 7, 8, 9, 10, 11, 12, 13, and 14. The transmission transition probability of these nodes on the n-th channel follows a two-state Markov chain

The experimental results are shown in Figure 6. We can see that both Dueling DRQN and DQN can converge after certain episodes. However, the expanding in the size of observation space and action space increases the difficulty

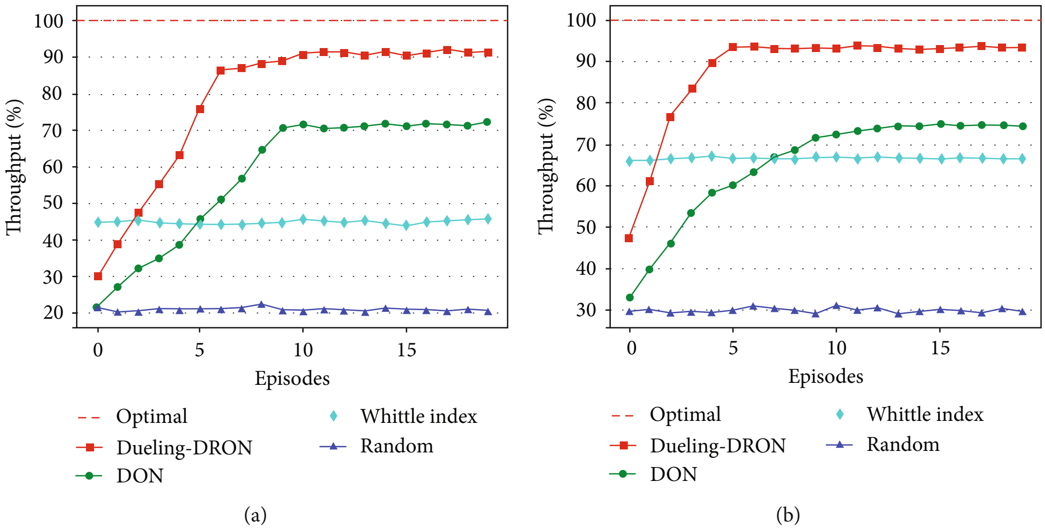

Figure 6: The throughput of the DRL node using different spectrum access policies. (a) Complex scenario I. (b) Complex scenario II.

of learning. The DQN needs to spend more episodes to explore different possible actions and learn the variation characteristics of the spectrum environment, but Dueling DRQN can still learn the spectrum characteristics quickly and obtain a better throughput than DQN. In addition, by comparing the performance of Dueling DRQN and DQN in both simple scenarios and complex scenarios, we can find that DQN obtains a better performance in the simple scenario but fails in the complex scenario. Unlike the DQN, the Dueling DRQN can keep robust in different scenarios due to the ability to infer and predict the complex temporal correlation of spectrum states. Moreover, we count the channel access frequency of different access strategies during the whole simulation process in two scenarios, and the results are shown in Figure 7. Since the relevance of the channel state is hidden deeply and changes over time, the strategy of DQN is ambiguous. It is very difficult for the DRL node to make an effective decision, especially in a partially observed environment. However, for the Dueling DRQN, by capturing the related information from historical observations and inferring the relevance of channel states, it focuses to access some channels (channel 7 and 8) as many as possible because these two channels have more idle time slots to share with the DRL node than other channels. Thus, the strategy of Dueling DRQN is explicit. In other words, the DQN looks more aggressive but the Dueling DRQN looks more comprehensive. The access strategy range of the whittle index policy is mainly between channel 7 and channel 14 both in two scenarios. However, it performs worse in the scenario I than in the scenario II because these channels in the scenario I do not follow the two-state Markov transition models.

4.4. Time-Varying Network Scenarios. In previous evaluations, the network parameters are fixed in a particular scenario. In this section, we further study the adaptability of the Dueling DRQN policy in a time-varying environment, in which the number of channels and the transmission schedule of nodes will be changed. As shown in Figure 8, we first consider that the number of channels is six, i.e., $N = 6$. The network includes two authorized nodes and four TDMA nodes. Two authorized nodes occupy channel 1 and channel 2, respectively. They transmit packets all the time. Four TDMA nodes occupy the rest channels and transmit $X = 10, 12, 13, 14$ time slots within a frame of $Y = 16$ time slots. In the second state, the transmission schedule of these TDMA nodes will change. Specifically, the order of time slots used by the TDMA nodes in a frame is changed. The detailed changes for the transmission schedule of TDMA nodes are displayed in Table 3. In the third state, the number of channels is increased to eight. Two TDMA nodes depart the network (originally occupy channel 5 and channel 6) and three hopping nodes join the network. Three hopping nodes dynamically occupy channels 5, 6, 7, and 8 with the patterns: $C5C6C7 \longrightarrow C6C7C8 \longrightarrow C7C8C5 \longrightarrow C8C5C6 \longrightarrow C5C6C7$. In the fourth state, the number of channels is increased to 12. Four q-ALOHA nodes join the network, which occupy the new channels in the network, and the transmission probability of each q-ALOHA node on the new channels is [0.4, 0.5, 0.8, 0.9]. In the fifth state, the number of channels is increased to 16. Four two-Markov nodes join the network and occupy the new channels in the network. The transmission transition probability of these nodes on the new channel follows a two-Markov chain. It is noted that the performance metric is reset when the environment changes. We can see that the Dueling DRQN can autonomously adjust to these changes in the environment and relearn the optimal access policy during $4 \sim 6$ episodes. It also demonstrates that the proposed scheme can follow the change in the transmission schedule of nodes without the prior knowledge of the MACs of other nodes. In addition, the optimal throughputs that the DRL node can achieve in different subenvironments are different, and the gap between the proposed algorithm and the optimal access scheme is getting larger as the network environment changes. The reason is that the increasing in the number of nodes and channels will lead to an exponential increase in the space of state. The expansion of state space

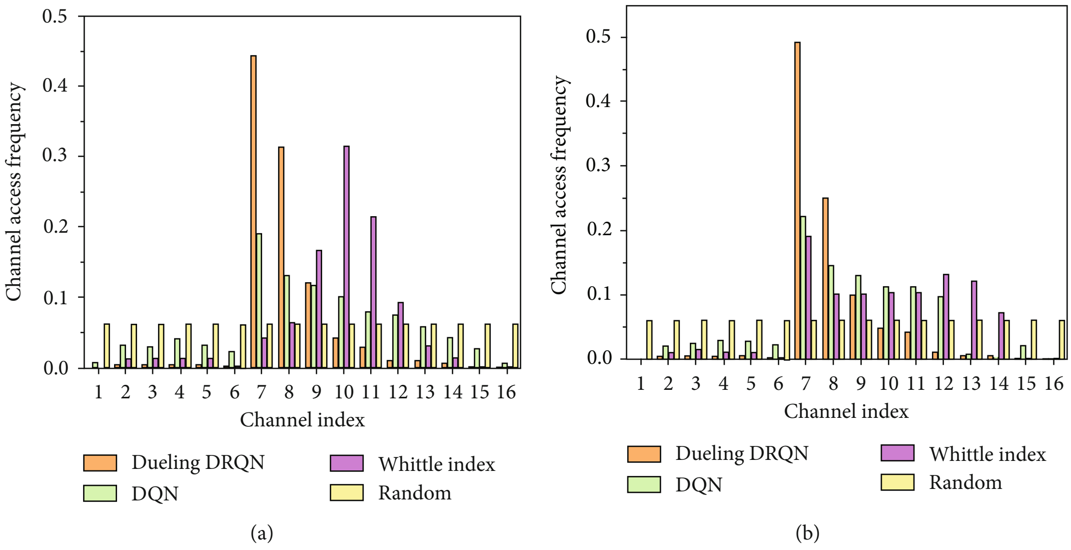

Figure 7: The statistics for channel access frequency of DRL node in different complex heterogeneous scenarios. (a) Complex scenario I. (b) Complex scenario II.

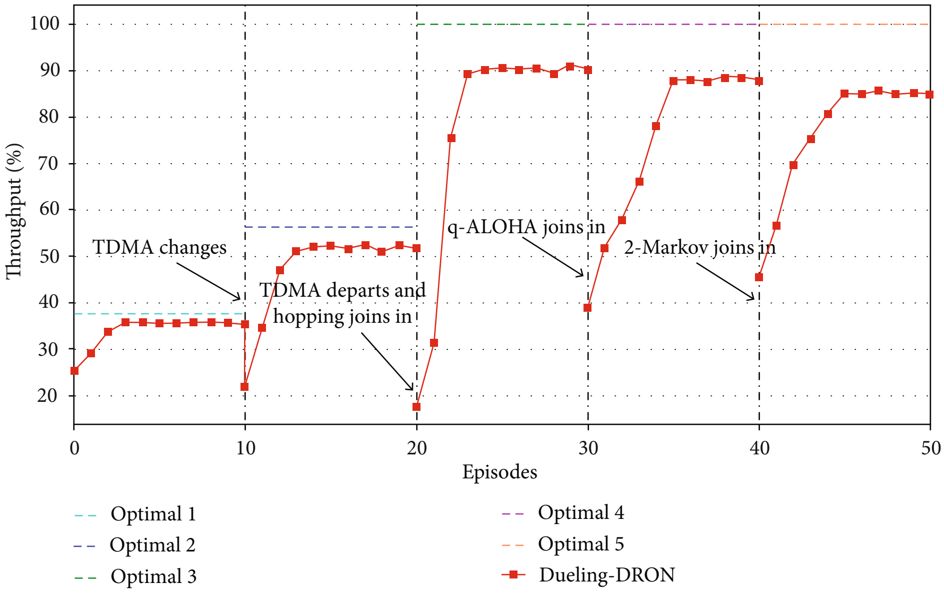

Figure 8: The throughput of the DRL node using Dueling DRQN access policy in the time-varying environment.

Table 3: Changes in transmission schedule of TDMA nodes.

TDMA node 1	Use 1, 2, 3, 7, 8, 9, 10, 13, 14, 15 slots in a frame
TDMA node 2	Use 1, 2, 3, 4, 6, 7, 8, 9, 12, 13, 14, 15 slots in a frame
TDMA node 3	Use 1, 2, 3, 4, 5, 6, 9, 10, 11, 12, 13, 14, 15 slots in a frame
TDMA node 4	Use 1, 2, 3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15 slots in a frame

will greatly increase the complexity of learning the optimal access strategies. In addition, the temporal feature of spectrum states will be more difficult to learn and capture by the agent as the network environment changes. Last but not least, the collision between the DRL node and other heterogeneous nodes may surge with the increase of the number of nodes and channels, which means the transmission failure of the DRL node. Statistically, the optimal throughputs for different subenvironments are 0.375, 0.5625, 1, 1, and 1, respectively.

4.5. Computational Complexity Analysis. In the last section, we give the specific computational complexity analysis of the proposed Dueling DRQN algorithm. The complexity of LSTM is closely related with the "cell" which is the basic computing component of LSTM. According to [36], the complexity of an LSTM at each time step can be written as $C = 4 \times {n_c}^2 + 4 \times n_i \times n_c + n_c \times n_{i+1} + 3 \times n_c$, where n_c is the number of memory cells, i.e., the number of memory blocks. n_i is the number of neural units in the last layer, and n_{i+1} is the number of neural units in the following layer. The rest of the neural networks in Figure 4 is fully connected, and the computational complexity of these neural networks can be represented by the number of multiplications in them. The number of multiplications through the Dueling DRQN with M layers (except LSTM layer) is $D = U \cdot m_1 + \sum_{j=1}^{M-2} m_j \cdot m_{j+1} + L \cdot m_{M-1}$, where U is the size of the input layer, m_j is the number of neural units in the j-th layer, and L is the size of the output layer. Thus, the complexity of our proposed Dueling DRQN is given by $O(C + D)$ at each time step.

5. Conclusion

In this paper, we investigated the problem of the dynamic multichannel access in heterogeneous wireless networks and proposed a DRL framework based on the combination of DQN, LSTM, and Dueling architecture, namely, Dueling DRQN. In the Dueling DRQN framework, the DQN is used to explore the idle spectrum resources, the LSTM is employed to make full use of historical observations and infer the temporal features of the heterogeneous networks, and the Dueling architecture is introduced to overcome the observability problem about dynamic environment in neural networks. We examined our scheme in various heterogeneous scenarios and compared it with other baseline channel access schemes such as the DQN access policy, the whittle index policy, and the random access policy. Simulation results showed that the Dueling DRQN can achieve a better throughput improvement than these baseline schemes and keep robust in complex heterogeneous network scenarios. Besides, through simulating our Dueling DRQN in the time-varying scenario, we found that our proposed scheme can also adapt well to the time-varying environment without any artificial adjustment or prior knowledge about the environment. For the future work, we will investigate the model-free dynamic multichannel access methods for multiple intelligent nodes without extra message interaction in the heterogeneous wireless networks.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there is no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant nos. 61931020, 62001483, and 62171449.

References

[1] H. Yang, "Application and development of mobile communication technology," in *2021 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 893–896, Harbin City, China, 2021.

[2] Z. Ning, S. Sun, X. Wang et al., "Intelligent resource allocation in mobile blockchain for privacy and security transactions: a deep reinforcement learning based approach," *Science China (Information Sciences)*, vol. 64, no. 6, pp. 172–187, 2021.

[3] Y. Liu, Q. Sun, Y. Tang, Y. Li, W. Jiang, and J. Wu, "Virtual reality system for industrial training," in *2020 International Conference on Virtual Reality and Visualization (ICVRV)*, pp. 338-339, Vigo, Spain, 2020.

[4] R. H. Tehrani, S. Vahid, D. Triantafyllopoulou, H. Lee, and K. Moessner, "Licensed spectrum sharing schemes for mobile operators: a survey and outlook," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2591–2623, 2016.

[5] Z. Ning, Y. Yang, X. Wang et al., "Dynamic computation offloading and server deployment for UAV-enabled multiaccess edge computing," *IEEE Transactions on Mobile Computing*, p. 1, 2021.

[6] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[7] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *International conference on machine learning*, pp. 1–9, New York, NY, USA, 2016.

[8] K. Wang, L. Chen, Q. Liu, and K. Al Agha, "On optimality of myopic sensing policy with imperfect sensing in multichannel opportunistic access," *IEEE Transactions on Communications*, vol. 61, no. 9, pp. 3854–3862, 2013.

[9] K. Liu and Q. Zhao, "Indexability of restless bandit problems and optimality of whittle index for dynamic multichannel access," *IEEE Transactions on Information Theory*, vol. 56, no. 11, pp. 5547–5567, 2010.

[10] H. Zhou, B. Liu, F. Hou et al., "Database-assisted dynamic spectrum access with QoS guarantees: a double-phase auction approach," *China Communications*, vol. 12, no. 1, pp. 66–77, 2015.

[11] F. Li, K. -Y. Lam, L. Meng, H. Luo, and L. Wang, "Trading-based dynamic spectrum access and allocation in cognitive internet of things," *IEEE Access*, vol. 7, pp. 125952–125959, 2019.

[12] S. Barrachina-Muñoz, A. Chiumento, and B. Bellalta, "Multi-armed bandits for spectrum allocation in multi-agent channel bonding WLANs," *IEEE Access*, vol. 9, pp. 133472–133490, 2021.

[13] Y. Zhang, Q. Zhang, B. Cao, and P. Chen, "Model free dynamic sensing order selection for imperfect sensing multichannel cognitive radio networks: a Q-learning approach," in *2014 IEEE International Conference on Communication Systems*, pp. 364–368, Macau, China, 2014.

[14] C. Dhahri and T. Ohtsuki, "Q-learning cell selection for femtocell networks: single- and multi-user case," in *2012 IEEE Global Communications Conference (GLOBECOM)*, pp. 4975–4980, Anaheim, CA, USA, 2012.

[15] K. Malon, J. Łopatka, and P. Skokowski, "Q-learning based radio channels utility evaluation algorithm for the local dynamic spectrum management in mobile ad-hoc networks," in *2020 Baltic URSI Symposium (URSI)*, pp. 28–32, Warsaw, Poland, 2020.

[16] N. Morozs, D. Grace, and T. Clarke, "Distributed Q-learning based dynamic spectrum access in high capacity density cognitive cellular systems using secondary LTE spectrum sharing," in *2014 International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pp. 462–467, Sydney, NSW, Australia, 2014.

[17] Z. Ning, S. Sun, X. Wang et al., "Blockchain-enabled intelligent transportation systems: a distributed crowdsensing framework," *IEEE Transactions on Mobile Computing*, p. 1, 2021.

[18] X. Wang, Z. Ning, S. Guo, M. Wen, L. Guo, and V. Poor, "Dynamic UAV deployment for differentiated services: a multi-agent imitation learning based approach," *IEEE Transactions on Mobile Computing*, p. 1, 2021.

[19] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, "Deep reinforcement learning for dynamic multichannel access in wireless networks," *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 2, pp. 257–265, 2018.

[20] C. Zhong, Z. Lu, M. C. Gursoy, and S. Velipasalar, "A deep actor-critic reinforcement learning framework for dynamic multichannel access," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 4, pp. 1125–1139, 2019.

[21] S. B. Janiar and V. Pourahmadi, "Deep-reinforcement learning for fair distributed dynamic spectrum access in wireless networks," in *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–4, Las Vegas, NV, USA, 2021.

[22] P. Yang, L. Li, J. Yin et al., "Dynamic spectrum access in cognitive radio networks using deep reinforcement learning and evolutionary game," in *2018 IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 405–409, Beijing, China, 2018.

[23] M. J. Liston and K. R. Dandekar, "Entropy based exploration in cognitive radio networks using deep reinforcement learning for dynamic spectrum access," in *2021 IEEE 21st Annual Wireless and Microwave Technology Conference (WAMICON)*, pp. 1–5, Sand Key, FL, USA, 2021.

[24] P. Chen, S. Guo, and Y. Gao, "Deep reinforcement learning with bidirectional recurrent neural networks for dynamic spectrum access," in *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, pp. 1–5, Norman, OK, USA, 2021.

[25] E. F. Badran, A. A. Bashir, A. I. Zaki, and W. K. Badawi, "Orthogonal codes-based dynamic spectrum access in cognitive radio networks," *China Communications*, vol. 16, no. 12, pp. 34–46, 2019.

[26] Y. Yu, T. Wang, and S. C. Liew, "Deep-reinforcement learning multiple access for heterogeneous wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1277–1290, 2019.

[27] Y. Yu, S. C. Liew, and T. Wang, "Non-uniform time-step deep Q-network for carrier-sense multiple access in heterogeneous wireless networks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 9, pp. 2848–2861, 2021.

[28] H. Yang, J. Zhao, K. -Y. Lam, Z. Xiong, Q. Wu, and L. Xiao, "Distributed deep reinforcement learning based spectrum and power allocation for heterogeneous networks," *IEEE Transactions on Wireless Communications*, vol. 21, no. 9, pp. 6935–6948, 2022.

[29] N. Zhao, Y. -C. Liang, D. Niyato, Y. Pei, M. Wu, and Y. Jiang, "Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 11, pp. 5141–5152, 2019.

[30] X. Wang, Z. Ning, S. Guo, M. Wen, and V. Poor, "Minimizing the age-of-critical-information: an imitation learning-based scheduling approach under partial observations," *IEEE Transactions on Mobile Computing*, vol. 21, no. 9, pp. 3225–3238, 2022.

[31] V. Mnih, K. Kavukcuoglu, D. Silver et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[32] J. Xiao, J. Li, and X. Wang, "Network reconfiguration of the shipboard power system based on logistic function particle swarm optimization," in *2008 7th World Congress on Intelligent Control and Automation*, pp. 5366–5370, Chongqing, 2008.

[33] R. A. Callejas-Molina, V. M. Jimenez-Fernandez, and H. Vazquez-Leal, "Digital architecture to implement a piecewise-linear approximation for the hyperbolic tangent function," in *2015 International Conference on Computing Systems and Telematics (ICCSAT)*, pp. 1–4, Xalapa, Mexico, 2015.

[34] K. Tachibana and K. Otsuka, "Wind prediction performance of complex neural network with ReLU activation function," in *2018 57th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*, pp. 1029–1034, Nara, Japan, 2018.

[35] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[36] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *15th Annual Conference of the International Speech Communication Association*, pp. 1–5, 2014.