IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received October 6, 2020, accepted October 24, 2020, date of publication November 5, 2020, date of current version November 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3036027

An Intelligent Anti-Jamming Scheme for Cognitive Radio Based on Deep Reinforcement Learning

JIANLIANG XU[1], **HUAXUN LOU**[1], **WEIFENG ZHANG**[2], **AND GAOLI SANG**[2]
[1]Science and Technology on Communication Information Security Control Laboratory, Zhejiang 314000, China
[2]College of Mathematics, Physics and Information Engineering, Jiaxing University, Zhejiang 314001, China

Corresponding author: Weifeng Zhang (zhangweifeng@zjxu.edu.cn)

This work was supported by the Natural Science Foundation of Zhejiang Province under Grant LQ18F020007.

ABSTRACT Cognitive radio network is an intelligent wireless communication system which can adjust its transmission parameters according to the environment thanks to its learning ability. It is a feasible and promising direction to solve the spectrum scarcity issue and has become a research focus in communication community. However, cognitive radio network is vulnerable to jamming attack, resulting in serious degradation of spectrum utilization. In this article, we view the anti-jamming task of cognitive radio as a Markov decision process and propose an intelligent anti-jamming scheme based on deep reinforcement learning. We aim to learn a policy for users to maximize their rate of successful transmission. Specifically, we design Double Deep Q Network (Double DQN) to model the confrontation between the cognitive radio network and the jammer. The Q network is implemented using Transformer encoder to effectively estimate action-values from raw spectrum data. The simulation results indicate that our approach can effectively defend against several kinds of jamming attacks.

INDEX TERMS Ant-jamming communication, cognitive radio, deep reinforcement learning.

I. INTRODUCTION

Cognitive radio (CR) is a new form of wireless communication whose transceiver can detect available communication channels intelligently [1]. By optimizing usage of available radio-frequency (RF) spectrum, cognitive radio can relieve contradiction between limited RF spectrum resource and growing demand for spectrum. In cognitive radio network, users are capable to sense the available portion of the spectrum, and use the idle channel for communication. Hence, cognitive radio has attracted extensive attention and become a research focus.

However, cognitive radio network is vulnerable to security attacks since its openness and broadcast nature [2], among which channel jamming attack is the most common one that can severely degrade network performance. The jammer can interfere communication channels by injecting continuous jamming signals or short jamming pulses to deteriorate the signal to noise ratio (SNR) in these channels. As a result, the throughput capacity of ongoing transmission declines, or even the transmission is interrupted. Traditional radios usually use spread spectrum techniques, such as frequency hopping or direct-sequence spread spectrum [3] to mitigate jamming attacks. However, smart jammer can track and interfere the hopping frequencies and these anti-jamming schemes cannot be directly used by cognitive radios. Recently, game theory was extensively studied to address anti-jamming task and achieved impressive result [4]–[6]. However, these approaches need prior knowledge such as the jamming pattern, which is unpractical in actual usage scenario.

Therefore, in this article we aim to mitigate channel jamming attack in cognitive radio network and develop an anti-jamming scheme based on deep reinforcement learning techniques. As shown in Fig. 1, the cognitive radio network is composed of a transmitter-receiver pair and the channels of their communication link is attacked by a jammer. In this anti-jamming communication game, the transmitter-receiver pair tries to setup communication link in appropriate channels to avoid or mitigate the jamming attack and maximize its throughput capacity, whereas the jammer aims to estimate the channels of the ongoing communication between transmitter and receiver to interrupt the network communication. Thus, the anti-jamming communication decisions of the transmitter-receiver pair in such a dynamic game is a typical Markov decision process (MDP), and the optimal

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

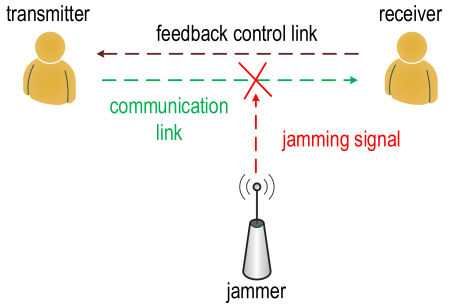


FIGURE 1. An illustration of cognitive radio network composed of transmitter and receiver under channel attack from jammer.

communication policy can be learned by reinforcement learning (RL) techniques, even the transmitter-receiver pair has no prior knowledge of the jamming pattern and communication channel model.

RL based anti-jamming policy learning has been studied recently. For instance, literature [7] also modeled the anti-jamming game as a MDP and proposed policy iteration method to solve this task. With the rapid development of deep learning [8], deep reinforcement learning (DRL) which can better sensing the environment and learning policy by combining deep neural networks and reinforcement learning has been proposed to tackle video games [9] and robot control [10]. Inspired by the success of DRL, Liu *et al.* [11] proposed a deep anti-jamming Q-network to estimate the Q-values of communication actions by directly inputting the spectrum waterfalls [12] into a convolutional neural network (CNN) [13]. Compared with these existing works, our anti-jamming scheme adopts deep reinforcement learning, specifically Double DQN [14], in which a Transformer Encoder [15] is used as the Q-network to effectively model the action-value function. The Transformer Encoder takes the raw spectrum data as input and outputs the action-value of each communication action. This Transformer Encoder style Q network is more flexible and powerful than CNN style Q network, since it can estimate action-values more accurately with arbitrary spectrum vectors. To evaluate the effectiveness of our proposed anti-jamming scheme, simulations are performed. We apply our anti-jamming approach to cope with three typical kinds of channel jamming attacks, including sweep jamming, random jamming, and sensing-based jamming. Our main contributions can be summarized as follows.

- An intelligent anti-jamming communication scheme is proposed based on deep reinforcement learning. This is a model-free approach, which means that the jamming patterns and channel models are not needed as a prior.
- A Transformer Encoder style Q-network is designed to map the state space to action space. Specifically, the raw spectrum data is defined as a state to describe the features of the jammer and channels without any information loss. Our simulation results demonstrate that algorithm with this Transformer Encoder style Q network is more effective to defend against jamming attacks than that with traditional CNN style Q-network.

The remaining of this article is organized as follows: First, we make a brief review of recently published works strongly associated with our study in Section II. Then we build the system model in Section III. Our anti-jamming scheme is then introduced in section IV. We give the detailed design of our proposed intelligent anti-jamming scheme based on Double DQN. Furthermore, the structure of the Q-network implemented with Transformer Encoder is also illustrated in this section. The simulation settings and results are given in Section V. We conclude our work in the end.

II. RELATED WORK

Wireless communication is vulnerable to security attacks since its openness and broadcast nature. The signal-to-interference-plus-noise ratio (SINR) at the receiver end can be decreased by the jammer who injects noise or recorded signal into the channels to disrupt the ongoing communications. Thus, anti-jamming ability is essential for radios. Traditional anti-jamming approaches such as frequency hopping spread spectrum (FHSS) and direct sequence spread spectrum (DSSS) [3] have fixed transmission patterns and are vulnerable to smart jammers powered by machine learning techniques [16], [17].

Jamming and anti-jamming between jammers and cognitive radios can be considered as a game process. With the rapid development of cognitive radio which is equipped with sensing, learning, and decision abilities, game theories have been studied to mitigate jamming attacks in wireless communication. Wu *et al.* [18] proposed a power allocation strategy based on Colonel Blotto anti-jamming game to withstand jamming attacks. Similarly, other game theories such as Stackelberg game theory were also tried to achieve anti-jamming defense in wireless networks in [19], [20]. To select appropriate frequency channel, the stochastic game has been investigated to find the optimal control and data channels to achieve maximum throughput under jamming attacks [21]. In spite of the successful application of game theories in anti-jamming task, these approaches need prior knowledge such as the jamming pattern, which is unpractical in actual usage scenario.

Recently reinforcement learning techniques have been applied to help the communication agent achieve an optimal policy via continuous interaction with environment and jammers without prior knowledge of the jammers. A novel channel access strategy to cope with channel jamming based on Q learning has been proposed in [22]. Literature [23] designed an interference-aware routing protocol and proposed a cooperation framework based on reinforcement learning to defend the network against jamming attacks. Since traditional Q learning is inefficient and hard to converge when the state space or action space is large, deep neural networks are adopted by reinforcement learning to achieve deep reinforcement learning which can take the spectrum waterfall as input and outputs channel selection actions [12].

Bi *et al.* [24] designed a multi-user anti-jamming strategy based on deep Q learning to achieve global optimization for multi-user system. A sequential deep reinforcement learning algorithm is studied in [25] to confront with multiple jammers. [26] proposed a fast DQN-based anti-jamming mobile communication scheme to cope with jamming attacks. DQN is also used to achieve optimal strategy to cope with unmanned aerial vehicle jamming attack [27]. Benefited from the powerful learning ability of deep neural networks, the above anti-jamming methods achieved superior performance than traditional approaches. Our work is inspired by these previous works and our proposed anti-jamming scheme is also based on deep reinforcement learning. Being different from existing works, instead of CNN, we use a Transformer Encoder style neural network.

III. SYSTEM MODEL

As shown in Fig. 1, without loss of generality, the cognitive radio network considered in this article consists of one transmitter, one receiver, and a jammer. We divide the continuous time into discrete time slots and we assume that both the transmitter and jammer share the same time slot. This operation simplifies the analysis. At each time slot t, the transmitter selects one channel $f_{U,t}$ from a predefined frequency set $\mathbf{f} = \{f_1, f_2, \ldots, f_{N_C}\}$ of the communication band to transmit data packet to the receiver with power $P_{U,t}$, while the jammer also selects an arbitrary channel $f_{J,t}$ of the same band, trying to disrupt this transmission with power $P_{J,t}$. Following [11], [28], we assume the bandwidth of jamming signal (b_J) is equal to the bandwidth of communication signal (b_U), denoted as $b_J = b_U$.

Based on the above setting, the SINR (Signal to Interference plus Noise Ratio) at the receiver can be calculated using the following formula:

$$SINR = \frac{P_{U,t}h_U}{n + P_{J,t}h_J I(f_{J,t} = f_{U,t})} \tag{1}$$

where $I(x)$ is an indicator function whose value is 1 if x is true, otherwise 0. n is the noise power, h_U is the channel gain from transmitter to receiver while h_J is the channel gain between jammer and receiver. Following existing work [11], we assume that the transmitter and jammer make their communication and jamming decisions at the beginning of each time slot. As shown in Fig. 2, the blue block and yellow block are respectively selected by the transmitter and jammer. If the transmitter transmits data in a channel which is also selected by the jammer (see the red block), the SINR of received signal deteriorates seriously and the transmission fails when the SINR is under the demodulation threshold. Since the transmitter cannot be able to know the jamming channel selected by the jammer in the current slot, it has to select communication channel based on its previous interactions with environment.

Following [11], we also use the raw spectrum data as environment state description. To be specific, the transmitter continuously senses the channel frequencies and stores the sensed results. The sensed spectrum vector of time slot t is denoted as $\mathbf{P}_t = [p_{t,1}, p_{t,2}, \ldots, p_{t,N_S}]$, where N_S is the number of sample points over the whole bandwidth and [,] denotes concatenation operation. To sufficiently sense and analyze the channel information, we record the historical spectrum data in the recent M time slot, denoted as:

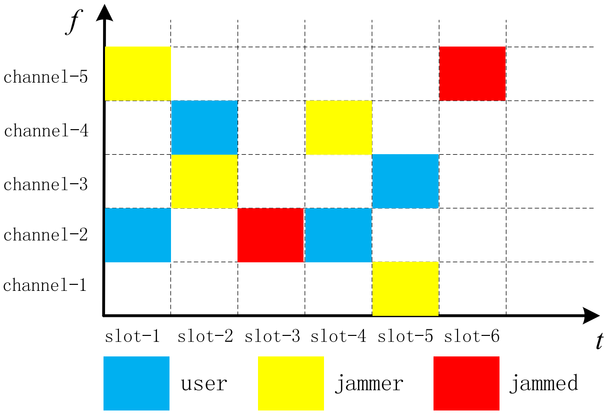


FIGURE 2. Time-frequency schematic diagram of user and jammer.

$$s_t = \begin{bmatrix} \mathbf{P}_{t-1} \\ \mathbf{P}_{t-2} \\ \cdots \\ \mathbf{P}_{t-M} \end{bmatrix} \tag{2}$$

This two-dimensional historical spectrum data contains rich spectrum information until time t and has been proven that it is a better choice to describe the channel status than traditional estimated channel parameters [11]. In the dynamic anti-jamming game, we use the above s_t as the environment state. The transmitter makes action decision a_t based on s_t and receives immediate reward r_t. In our setting, the action is a combined selection of channel and power level, e.g., $a_t = \{f_{U,t}, P_{U,t}\}$ represents the action on time slot t. Since the transmitter-receiver pair is expected to achieve successful transmission with low cost of channel changing and energy cost, the reward is designed as follows:

$$r_t = r_{SINR}(a_t) - c(a_t) - C_p P_{U,t} \tag{3}$$

The reward is composed of three terms, and they are all scalars without units. The first term $r_{SINR}(a_t)$ is the reward of successful transmission. The transmission is considered successful if the SINR of the received signal (denoted as $SINR_t$) exceeds demodulation threshold $SINR_{threshold}$. Then the transmitter gets a reward r_m, otherwise the transmission is failed and the reward is zero. Hence, $r_{SINR}(a_t)$ is formally defined as:

$$r_{SINR}(a_t) = \begin{cases} r_m, & SINR_t \geq SINR_{threshold}, \\ 0, & SINR_t < SINR_{threshold}. \end{cases} \tag{4}$$

The SINR of the received signal, which is transmitted by the control link, is the basis for calculating reward. Thus, we assume that the control link is jamming-resistant, and this assumption is widely used and can be found in many literatures [12], [28], [29].

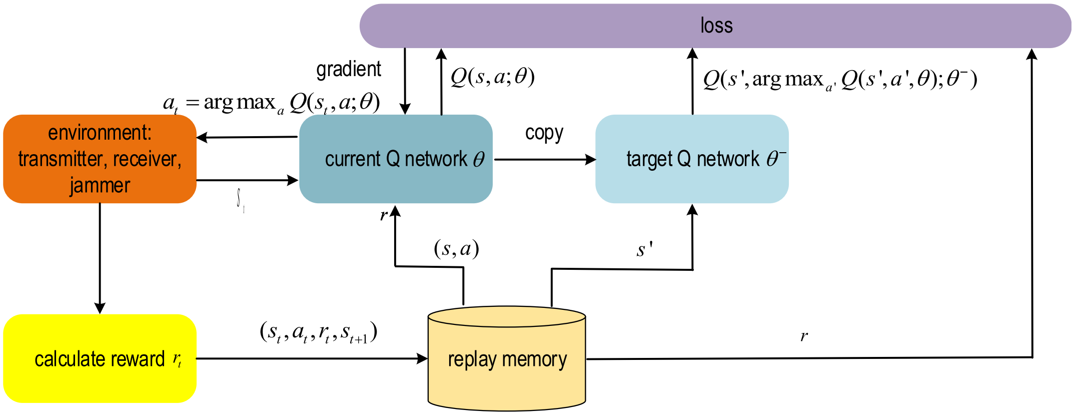

loss
gradient
$a_t = \arg\max_a Q(s_t, a; \theta)$
$Q(s, a; \theta)$
$Q(s', \arg\max_{a'} Q(s', a', \theta); \theta^-)$
environment: transmitter, receiver, jammer
current Q network θ
copy
target Q network θ^-
s_t
r
(s, a)
s'
calculate reward r_t
(s_t, a_t, r_t, s_{t+1})
replay memory
r

FIGURE 3. Diagram of the proposed anti-jamming algorithm based on deep reinforcement learning.

TABLE 1. Simulation Parameters.

Parameter name	Value
N_C: Number of channels	5
b_U: Bandwidth of communication signal	2.4MHz
b_J: Bandwidth of jamming signal	2.4MHz
f: Center frequency of channels	{53, 57, 61, 65, 68}MHz
P_U: Transmit power level	{25 - 45}dBm
M: Number of spectrum vectors	400
N_S: length of spectrum vectors	200
$SINR_{threshold}$: Demodulation threshold	10dB
r_m: successful communication reward	1
C_c: Cost of switching channel	0.2
C_p: Cost of unit transmit power	0.005
T: Number of time slots	1000
γ: Discount factor	0.6
ϵ_{min}: Minimum exploring rate	0.05
ϵ_{max}: Maximum exploring rate	0.9
d: Attenuation cycle of exploring rate	5
N_T: Target Q network update cycle	10
B: Batch size	32

The second term $-c(a_t)$ is added since the transmitter-receiver pair is expected to communicate on fixed channel using stable power since communication cost is need for changing channels. Hence we define the cost of switching channels. If the transmitter takes a different action at time slot t ($a_t \neq a_{t-1}$), it will be penalized. The formal definition is as follows:

$$c(a_t) = \begin{cases} 0, & a_t = a_{t-1}, \\ C_c, & a_t \neq a_{t-1}. \end{cases} \tag{5}$$

The third term $-C_p P_{U,t}$ denotes the cost of the transmit power, where C_p is the cost of the unit transmit power. It is obvious that higher transmit power results in higher probability of successful transmission. If there is no constraint on transmit power, the best policy for transmitter is always selecting the highest transmit power. However, in our system, the transmitter-receiver pair is expected to achieve successful transmission with as lower power consumption as possible. Thus, we add this term to the reward. The values of all the above hyper-parameters can be found in Table 1.

IV. INTELLIGENT ANTI-JAMMING SCHEME

A. DOUBLE DQN

The anti-jamming communication decision of the transmitter-receiver pair in a dynamic environment is a typical Markov Decision Process (MDP) and has been studied using value-based or policy-based reinforcement learning algorithms. However, traditional Q-learning algorithm is unable to handle the game in our work described in Sec. III, since the raw historical spectrum data, which is viewed as environment state, is infinite. We propose a novel anti-jamming scheme based on Double DQN [14]. Fig. 3 illustrates the framework of our approach. At each time slot t, the transmitter interacts with the environment by selecting action a_t according to the sensed state s_t. After executing the action, the transmitter receives an immediate reward r_t and observes the next state s_{t+1}. According to equation 3, higher reward means higher probability of successful transmission. Therefore, the transmitter aims to select anti-jamming actions to maximize cumulative reward $R_t = \sum_{i=0}^{\infty} \gamma^i r_{t+i+1}$ where γ is the discount factor. Double DQN algorithm used in this work achieves this goal by finding the optimal action-value function $Q^*(s, a)$:

$$Q^*(s, a) = max_\pi \mathbb{E}[R_t | s_t = s, a_t = a, \pi] \tag{6}$$

where $\mathbb{E}[]$ is to calculate expectation. π is a policy mapping sequences to actions. Given the environment, Q function outputs the action-values for all actions. Actions with higher action-values have more cumulative rewards and are selected with higher probability. Thus, at each time slot, the transmitter can select the action with highest action-value to effectively mitigate jamming attacks.

Accurately modeling the Q function is the key of our approach. We use the interactions to train a deep Q network which is detailed in section IV-B as function approximator to estimate the action-values, denoted as $Q(s, a; \theta) \approx Q^*(s, a)$. In general terms, the probability of taking an anti-jamming action should be a function of the environment. According to the universal approximation theorem for neural networks, the Q network composed of deep neural network can approximate the function at any accuracy when enough previous interactions are provided as training samples.

To stabilize the optimization of approximation, two deep neural networks are used in Double DQN: one called current Q network, namely $Q(s, a; \theta)$, for action selection and another called target Q network, namely $Q(s, a; \theta^-)$, for evaluating the target action-value. The idea of using two Q networks is to decouple the selection from the evaluation, alleviating overestimated value problem [30]. It is worth to note that these two Q networks have the same structure. The weights of target Q network are periodically copied from current Q network. The current Q network is trained by minimizing the following loss function which calculates the mean squared error between current action-value and target action-value:

$$\mathcal{L}(\theta) = \frac{1}{B}\sum_{i=1}^{B}(y_i - Q(s_i, a_i; \theta))^2 \tag{7}$$

where B denotes the batch size and y_i is the target action-value estimated by target Q network using greedy strategy:

$$y_i = r_i + \gamma Q(s_i', argmax_{a'} Q(s_i', a'; \theta); \theta^-) \tag{8}$$

where γ is the discount factor. The gradient of the loss function with respect to the learnable weights can be calculated as follows:

$$\nabla_\theta \mathcal{L}(\theta) = \frac{1}{B}\sum_{i=1}^{B}[(y_i - Q(s_i, a_i; \theta))\nabla_\theta Q(s_i, a_i; \theta)] \tag{9}$$

Finally, gradient decent algorithm can be adopted to update the weights θ of current Q network. The details of the algorithm for intelligent anti-jamming scheme based on deep reinforcement learning are given in Algorithm 1.

B. Q-NETWORK BASED ON TRANSFORMER ENCODER

As discussed in Sec. III, we use the historical spectrum data in recent time slots as state, denoted as $s_t \in \mathbb{R}^{N_S \times M}$. The Q network is used to estimate action-values based on this state, which plays the key role of Double DQN algorithm. In this article, we design a modified Transformer Encoder to implement the Q network. Existing works [12], [28], [29] use Convolutional Neural Network (CNN) as Q network, which mines the correlation between historical spectrum vectors in an implicit manner and is hard to explore the relations between all the spectrum vectors due to the limited receptive field of CNN structure. Our Transformer Encoder style Q network is not subject to this restriction, since it is capable to exploit the correlation between all the historical spectrum vectors from multiple perspectives by using multi-head attention mechanism. Fig. 4 shows the detailed structure of our proposed Q network, which is composed of three modules including a Transformer Encoder and two classifiers composed of fully connected layers.

Transformer composed of encoders and decoders is originally proposed to address machine translation task [15]. Now it has become a popular module in many computer vision tasks such as Visual Question Answering [31]. In this article, we adopt Transformer Encoder to extract features from historical spectrum vectors. As shown in Fig. 4, the Transformer Encoder contains a multi-head self-attention sub-layer and a feed-forward sub-layer. At time slot t, the transmitter senses the environment and store the historical spectrum vectors in recent M time slots, denoted as $s_t = [\mathbf{P}_{t-1}; \ldots; \mathbf{P}_{t-M}] \in \mathbb{R}^{N_S \times M}$. Then the state is updated by incorporating the correlations between these spectrum vectors by multi-head self-attention as follows:

Algorithm 1 Intelligent Anti-Jamming Scheme Based on Double DQN

```
Input: Number of iteration $T$, action set $\mathcal{A}$, discount factor $\gamma$, exploring rate range $[\epsilon_{min}, \epsilon_{max}]$, attenuation cycle of exploring rate $d$, weights of current Q network $\theta$ and target Q network $\theta^-$, batch size $B$, target update cycle $N_T$
randomize $\theta$ and copy it to target Q network $\theta^- = \theta$, empty replay memory $\mathcal{D}$
for $t = 1$ to $T$ do
    sensing the spectrum and record the historical spectrum data $s_t$ as environment state
    calculate exploring rate $\epsilon = \epsilon_{min} + (\epsilon_{max} - \epsilon_{min})e^{-t/d}$
    generate random number $\varepsilon$, and select action using $\epsilon$-greedy algorithm:
        $a_t = \begin{cases} random, , & \varepsilon < \epsilon \\ argmax_a Q(s, a; \theta), & \varepsilon \geq \epsilon. \end{cases}$   (10)
    select channel $f_{U,t}$ and transmit power $P_{U,t}$ according to $a_t$ and launch communication
    sensing the new spectrum $s_{t+1}$ and calculate reward $r_t$ according to Eq. 3
    store the trajectory $(s_t, a_t, r_t, s_{t+1})$ into replay memory $\mathcal{D}$
    if $|\mathcal{D}| \geq B$ then
        randomly select a batch of samples $(s_i, a_i, r_i, s_i')$, $i = 1, \ldots, B$ from $\mathcal{D}$
        calculate loss for the batch according to Eq. 7
        update current Q network's weights $\theta$ using gradient Eq. 9
    end if
    if $t \% N_T = 0$ then
        update target Q network's weights: $\theta^- = \theta$
    end if
end for
```

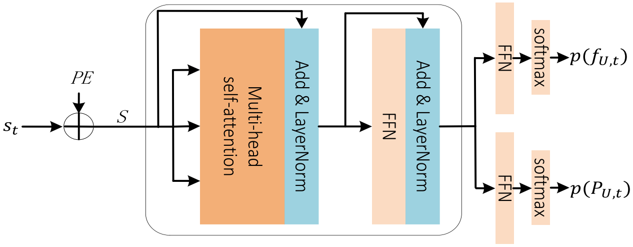


FIGURE 4. The structure of our proposed Transformer Encoder style Q network. This network takes historical spectrum vectors as input and estimates the action-value $Q(s_t, a_t; \theta) = p(f_{U,t}) \times p(P_{U,t})$.

$$S_{sa} = \mathbf{W}_O[head_1, head_2, \ldots, head_H] \tag{11}$$

$$head_h = (\mathbf{W}_V^h S)\mathbf{a}, h = 1, \ldots, H \tag{12}$$

where $\mathbf{W}_O \in \mathbb{R}^{N_S \times N_S}$, $\mathbf{W}_V^h \in \mathbb{R}^{\frac{N_S}{H} \times N_S}$ are learnable matrix and H is the number of attention heads. $S = s_t + PE$ is the element-wise sum of the spectrum data and their position embedding $PE = [pe_{t-1}; \ldots; pe_{t-M}]$. Following [15], the position embedding of the $(t-i)^{th}$ spectrum vector is as follows:

$$pe_{t-i,j} = \begin{cases} sin((t-i)/10000^{j/\frac{N_S}{H}}), & j\%2 = 0, \\ cos((t-i)/10000^{(j-1)/\frac{N_S}{H}}), & j\%2 = 1. \end{cases} \tag{13}$$

The usage of position embedding enables our model to utilize the order of the sequence. $\mathbf{a} \in \mathbb{R}^{M \times M}$ in Eq. 12 depicts dependence between each spectrum vector and can be calculated by using row-wise softmax on the dot-product of query $Q = \mathbf{W}_Q^h S$ and key $K = \mathbf{W}_K^h S$ as follows:

$$\mathbf{a} = softmax(\frac{Q^T K}{\sqrt{N_S/H}}) \tag{14}$$

where $\mathbf{W}_Q^h$ and $\mathbf{W}_K^h$ are also learnable matrix. These updated spectrum vectors are further passed through a Feed-Forward Network (FFN) composed of two fully connected layers:

$$S_{ec} = LN[FFN(LN(S + S_{sa})) + LN(S + S_{sa})] \tag{15}$$

where

$$FFN(x) = \mathbf{W}_2(ReLU(\mathbf{W}_1 x + b_1)) + b_2 \tag{16}$$

$\mathbf{W}_1$ and $\mathbf{W}_2$ are learnable projection matrix while b_1 and b_2 are bias terms. Residual connection [32] and Layer Normalization (LN) [33] are also applied to facilitate optimization.

By using Transformer Encoder, the updated spectrum feature S_{ec} contains rich information. Now we can estimate action-value based on S_{ec} using simple feed-forward networks. In our anti-jamming setting, at each time slot t, the action has two dimensions, one for selecting channel $f_{U,t}$ and another for determining transmit power level $P_{U,t}$. Thus, the action-value can be denoted as $Q(s_t, a_t; \theta) = p(f_{U,t}) \times p(P_{U,t})$ and can be calculated as follows:

$$p(f_{U,t}) = softmax(FFN_1(S_{ec})) \tag{17}$$

$$p(P_{U,t}) = softmax(FFN_2(S_{ec})) \tag{18}$$

where FFN_1 and FFN_2 are two feed-forward networks with different weights.

V. NUMERICAL SIMULATION

A. SIMULATION SETTINGS

To verify the effectiveness of our proposed anti-jamming scheme, we conduct extensive simulations. In our simulation, the transmitter-receiver pair and the jammer combat with each other. Following existing works [11], [25], they combat with each other in a frequency band of 20MHz. Specifically, we uniformly select 5 ($N_C = 5$) center frequencies ($\mathbf{f} = \{53, 57, 61, 65, 68\}$MHz) as candidate frequency channels. The frequency resolution of spectrum sensing is 100kHz and the transmitter senses the full band every 1 ms and store the spectrum vectors in recent 400 ms. Thus, $M = 400$, $N_S = 200$. One time slot is defined as 5ms. At the beginning of each time slot, the transmitter selects one frequency channel to sends data packet and the jammer injects jamming signal into one channel. At the end of each time slot, the receiver sends the SINR to transmitter through control link. The demodulation threshold is $10dB$ ($SINR_{threshold} = 10dB$), which is same as [11]. The transmit power level are chose from $25dBm - 45dBm$ with $0.5dBm$ step. We implement our algorithm using PyTorch on a machine with Intel i5 CPU, 16GB RAM, and NVIDIA Geforce 1070 GPU. More detailed simulation parameters are given in Table 1.

Without loss of generality, one malicious jammer is considered in our anti-jamming game. This jammer sends jamming signal on the selected channel to disrupt the ongoing communication of the transmitter-receiver pair by heavily deteriorating the SINR at the receiver. In this article, we consider the following three popular kinds of jammers similar to [34] and [11].

- *Random jammer* which randomly jams a channel in each time slot. The jamming frequency is randomly changed with the step of 0.5 MHz.
- *Sweep jammer* which jams all the communication band with 0.8 GHz/s sweeping speed.
- *Sensing-based jammer* that continuously observes the probability that the communication signal appears at each frequency point, and chooses the one with largest probability as jamming channel.

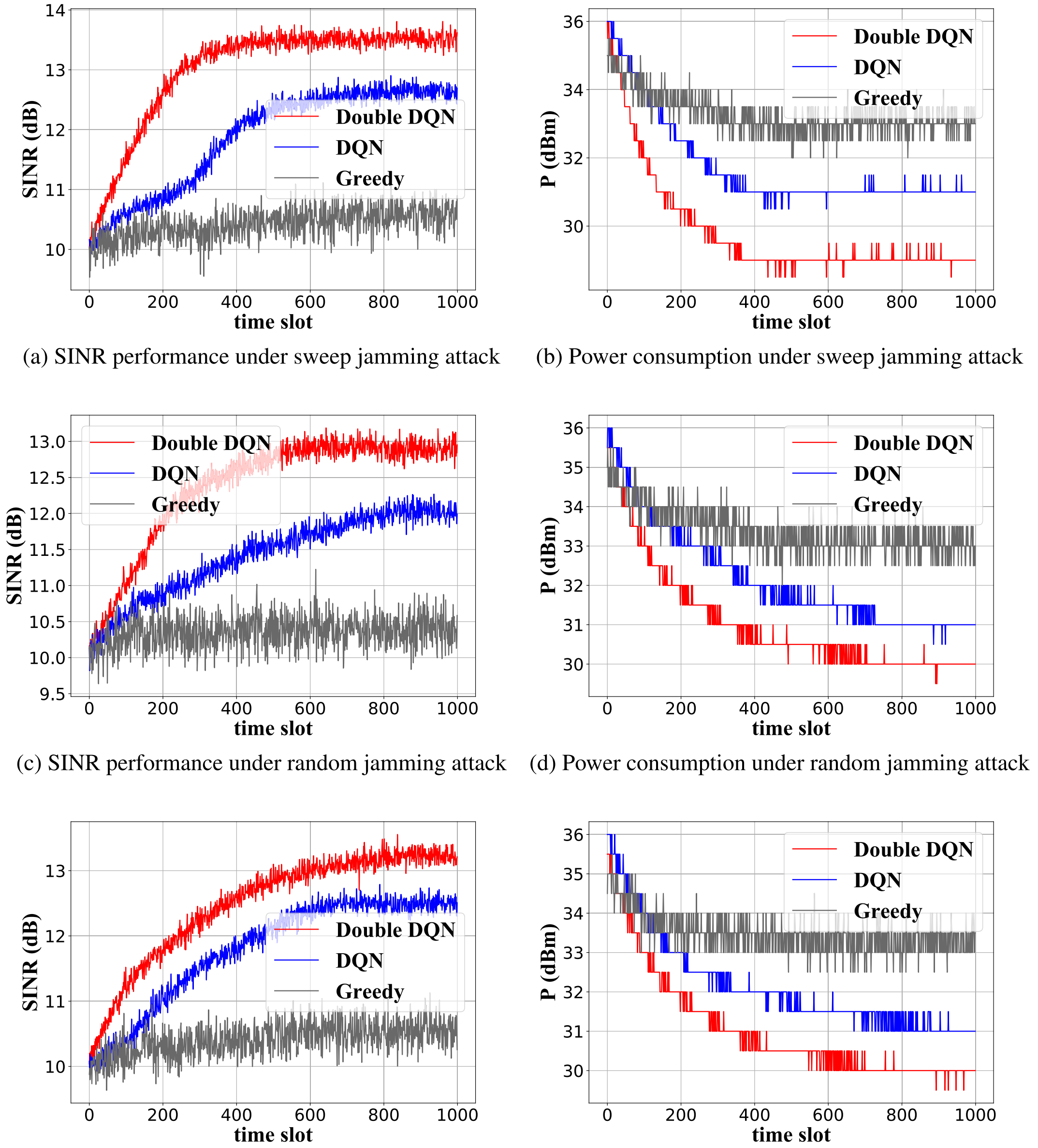
B. SIMULATION RESULTS

1) DOUBLE DQN VS OTHER DEEP REINFORCEMENT LEARNING

As we have illustrated in Sec. IV, the core of our intelligent anti-jamming scheme is the *Double DQN*. To prove the effectiveness of our scheme, we compare our method with the following popular approaches:

- We implement another popular value-based reinforcement learning algorithms *DQN*, which is similar to *Double DQN* except that only one Q network is adopted to simultaneously select action and evaluate target action-value. In *DQN*, the target action-value y_i is estimated as follows:
$$y_i = r_i + \gamma max_{a'} Q(s_i', a'; \theta) \tag{19}$$
For fairly comparison, we also use Transformer Encoder as shown in Fig. 4 to implement this Q network.
- To prove the effectiveness of applying reinforcement learning technique, we also try a *Greedy* method, in which the transmitter updates the score of each action according to the average reward it ever received and selects the action with highest average reward at each time slot.

Fig. 5 shows the experimental results. We show the SINR performance and power consumption of different methods under different jamming attacks. From these results, we can arrive the following points:



(a) SINR performance under sweep jamming attack (b) Power consumption under sweep jamming attack

(c) SINR performance under random jamming attack (d) Power consumption under random jamming attack

(e) SINR performance under sensing-based jamming attack (f) Power consumption under sensing-based jamming attack

FIGURE 5. Performance comparison between different deep reinforcement learning under different kinds of jamming attacks.

a: OUR METHOD CONVERGES TO HIGHER SINR WITH LOWER POWER CONSUMPTION THAN OTHER APPROACHES

For example, the comparison in Fig. 5(a) demonstrates that our intelligent anti-jamming scheme converges to high SINR (13.5 dB) after about 350 time slots under the sweep jamming attack, while the method based on *DQN* needs 500 time slots to arrive its convergence (SINR=12.6 dB). Fig. 5(b) shows that our method can achieve lower power consumption than other approaches. Consistent results are yielded under other jamming attacks.

b: DEEP REINFORCEMENT LEARNING BASED ANTI-JAMMING METHODS ARE SUPERIOR TO GREEDY METHOD

By comparing the results of deep reinforcement learning based methods and greedy method, we can see that both of Double DQN based anti-jamming scheme and DQN based anti-jamming scheme achieve higher SINR performance and lower power consumption than greedy method, under all the three kinds of jamming attacks. In addition, performance fluctuation of greedy method is more evident.

c: OUR METHOD COPES WITH SENSING-BASED JAMMING ATTACK SUCCESSFULLY

As shown in Fig. 5(e) and (f), our intelligent anti-jamming scheme can perfectly dodge the sensing-based jamming attack. The reason of this phenomenon can be explained as follows. Our method selects each channel with almost the same probability, resulting that the sense-based jammer cannot recognize the communication pattern of our transmitter-receiver pair. To prove the above explanation,

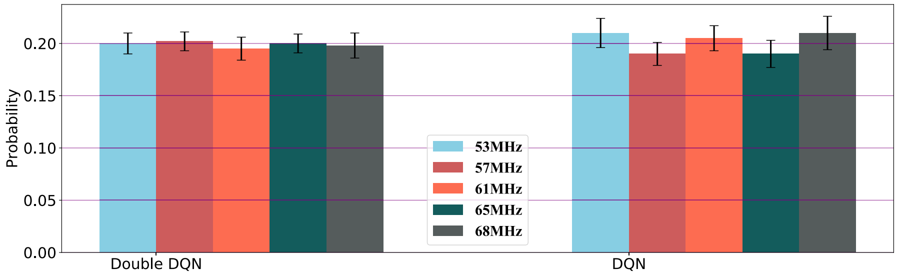


FIGURE 6. Probabilities of selecting each channel.

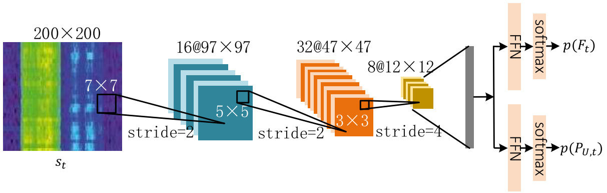


FIGURE 7. The structure of CNN style Q network.

we give the statistical probabilities of selecting each channel in Fig. 6. We can see that the transmitter using our anti-jamming scheme selects each channel equally with about 20% probability after convergence, which is much more stable than that using DQN based method.

In our simulation test, the floating-point operations per second (FLOPS) of our algorithm is 0.45 × 10^9, and the transmitter takes on average 0.8ms to update the network parameters and make the communication decision on our machine.

2) TRANSFORMER ENCODER STYLE VS CNN STYLE

As we have discussed in Sec. IV-B, we use Transformer Encoder to implement the Q network in our algorithm. These Transformer Encoder style Q networks play key role to the good performance. To prove our point, we conduct an experiment in this section. In this experiment, following [11], we design a Convolutional Neural Network (CNN) to replace the Transformer Encoder.

As shown in Fig. 7, the CNN is composed of three convolutional layers and two classifiers. We also use the spectrum vectors s_t as input. The first convolution layer has 16 kernels and the kernel size and stride are set to be 7 and 2 respectively. Thus, the output of first convolution layer is a tensor composed of 16 feature maps whose size are 97 × 97. Then these feature maps are further passed through the second and third convolution layers, whose hyper-parameters are given in Fig. 7. Finally, the classifiers estimate action-values including $p(f_{U,t})$ and $p(P_{U,t})$ based on the output of the last convolution layer.

From the experimental result shown in Fig. 8, we can see that the Transformer Encoder style Q network works better than CNN style Q network. To be specific, the Transformer Encoder Q network converges to higher SINR (12.9 dB) than CNN style Q network (12.3 dB), since that Transformer Encoder style Q network has the ability to exploit the relations between all spectrum vectors by self-attention operation.

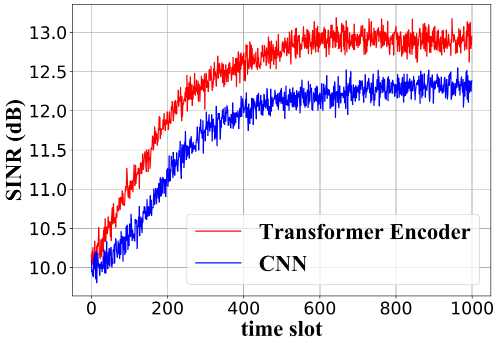


FIGURE 8. Performance comparison between different deep reinforcement learning under random jamming attacks.

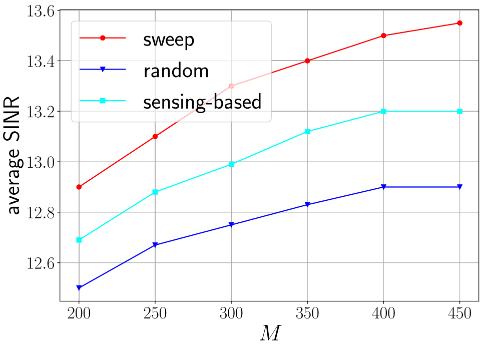


FIGURE 9. The impact of *M* (Number of spectrum vectors).

3) IMPACTS OF HYPER-PARAMETERS

In this section, we emphatically analyze several important hyper-parameters of our approach, including M (number of spectrum vectors), C_c (cost of switching channel), and C_p (cost of unit transmit power).

We first analyze the impact of M which denotes the number of spectrum vectors. As shown in Fig. 9, under all the three kinds of jamming attacks, the performance of our anti-jamming scheme based on Transformer Encoder style Q network obtains steady increase with the increasement of spectrum vectors until $M = 400$. The reason for this phenomenon is that more historical spectrum vectors provide the Q network with more sufficient information to make correct decisions. Hence, the hyper-parameter M is set to 400 in all the other experiments.

C_c, the cost of switching channel, is another important hyper-parameter in our approach. Thus, we evaluate its impact on the SINR performance here. We set C_c to 0, 0.1,

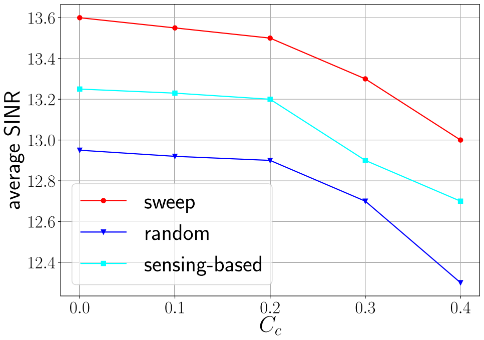


FIGURE 10. The impact of C_c (Cost of switching channel).

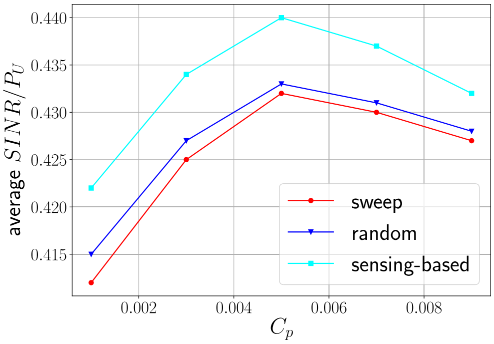


FIGURE 11. The impact of C_p (Cost of unit transmit power).

0.2, 0.3, 0.4 respectively and record the average SINR of the received signal. The result is shown in Fig. 10. We can see that the average SINR remains stable when C_c increases from 0 to 0.2, and then declines rapidly after 0.2. Hence, we finally set $C_c = 0.2$ to achieve a tradeoff between performance and cost of switching channel.

As discussed in section III, we add a term $-C_pP_{U,t}$ to our reward definition, guiding our algorithm to cope with jamming attacks with low power consumption. Hence, we evaluate the impact of C_p which denotes the cost of unit transmit power. We set C_p to 0.001, 0.003, 0.005, 0.007, 0.009 respectively and record the average performance ratio calculated as $\frac{SINR}{P_U}$. As shown in Fig. 11, small C_P encourages the transmitter to select high transmit power to achieve successful transmission, but resulting in low performance ratio, while big C_P enforces the transmitter to excessively concern about power consumption, also leading to low performance ratio. $C_P = 0.005$ results in best tradeoff between performance and power cost.

VI. CONCLUSION

In this article, we propose an intelligent anti-jamming scheme based on deep reinforcement learning. Specifically, we adopt Double DQN to model the confrontation between the cognitive radio network and the jammer. Different from existing work which use CNN style Q network, we design Transformer Encoder style Q network to effectively estimate action-value from raw spectrum data. To evaluate our proposed anti-jamming scheme, we conduct extensive experiments. Simulation results indicate that our approach can effectively defend against several kinds of jamming attacks, including sweep jamming, random jamming, and sensing-based jamming.

REFERENCES

[1] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[2] A. G. Fragkiadakis, E. Z. Tragos, and I. G. Askoxylakis, "A survey on security threats and detection techniques in cognitive radio networks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 1, pp. 428–445, 1st Quart., 2013.

[3] D. Torrieri, *Principles of Spread-Spectrum Communication Systems*. Cham, Switzerland: Springer, 2018.

[4] Y. Xu, G. Ren, J. Chen, and, "A one-leader multi-follower bayesian-stacklberg game for anti-jamming transmission in uav communication networks," *IEEE Access*, vol. 6, pp. 21697–21709, 2018.

[5] H. Noori and S. Sadeghi Vilni, "Jamming and anti-jamming in interference channels: A stochastic game approach," *IET Commun.*, vol. 14, no. 4, pp. 682–692, Mar. 2020.

[6] I. K. Ahmed and A. O. Fapojuwo, "Stackelberg equilibria of an anti-jamming game in cooperative cognitive radio networks," *IEEE Trans. Cognit. Commun. Netw.*, vol. 4, no. 1, pp. 121–134, Mar. 2018.

[7] C. Chen, M. Song, C. Xin, and J. Backens, "A game-theoretical anti-jamming scheme for cognitive radio networks," *IEEE Netw.*, vol. 27, no. 3, pp. 22–27, May 2013.

[8] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[9] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. A. Riedmiller, "Playing atari with deep reinforcement learning," *CoRR*, vol. abs/1312.5602, pp. 1–8, Dec. 2013.

[10] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[11] X. Liu, Y. Xu, L. Jia, Q. Wu, and A. Anpalagan, "Anti-jamming communications using spectrum waterfall: A deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 22, no. 5, pp. 101–198, Dec. 2018.

[12] W. Chen and X. Wen, "Perceptual spectrum waterfall of pattern shape recognition algorithm," in *Proc. 18th Int. Conf. Adv. Commun. Technol. (ICACT)*, Jan. 2016, pp. 382–389.

[13] Z. Li, W. Yang, S. Peng, and F. Liu, "A survey of convolutional neural networks: Analysis, applications, and prospects," 2020, *arXiv:2004.02806*. [Online]. Available: http://arxiv.org/abs/2004.02806

[14] H. van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *AAAI Conf. Artif. Intell.*, 2016, pp. 2094–2100.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, and, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.

[16] S. Amuru, C. Tekin, M. van der Schaar, and R. M. Buehrer, "Jamming bandit," *IEEE Trans. Wireless Commun.*, vol. 15, no. 4, pp. 2792–2808, 2016.

[17] T. Erpek, Y. E. Sagduyu, and Y. Shi, "Deep learning for launching and mitigating wireless jamming attacks," *IEEE Trans. Cognit. Commun. Netw.*, vol. 5, no. 1, pp. 2–14, Mar. 2019.

[18] Y. Wu, B. Wang, K. J. R. Liu, and T. C. Clancy, "Anti-jamming games in multi-channel cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 1, pp. 4–15, Jan. 2012.

[19] L. Jia, Y. Xu, Y. Sun, S. Feng, and A. Anpalagan, "Stackelberg game approaches for anti-jamming defence in wireless networks," *IEEE Wireless Commun.*, vol. 25, no. 6, pp. 120–128, Dec. 2018.

[20] L. Jia, F. Yao, Y. Sun, Y. Xu, S. Feng, and A. Anpalagan, "A hierarchical learning solution for anti-jamming stackelberg game with discrete power strategies," *IEEE Wireless Commun. Lett.*, vol. 6, no. 6, pp. 818–821, Dec. 2017.

[21] B. Wang, Y. Wu, K. J. R. Liu, and T. C. Clancy, "An anti-jamming stochastic game for cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 877–889, Apr. 2011.

[22] Y. Gwon, S. Dastangoo, C. Fossa, and H. T. Kung, "Competing mobile network game: Embracing antijamming and jamming strategies with reinforcement learning," in *Proc. IEEE Conf. Commun. Netw. Secur. (CNS)*, Oct. 2013, pp. 28–36.

[23] N. Abuzainab, T. Erpek, K. Davaslioglu, Y. E. Sagduyu, Y. Shi, S. J. Mackey, M. Patel, F. Panettieri, M. A. Qureshi, V. Isler, and A. Yener, "QoS and jamming-aware wireless networking using deep reinforcement learning," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Nov. 2019, pp. 610–615.

[24] Y. Bi, Y. Wu, and C. Hua, "Deep reinforcement learning based multi-user anti-jamming strategy," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.

[25] S. Liu, Y. Xu, X. Chen, and, "Pattern-aware intelligent anti-jamming communication: A sequential deep reinforcement learning approach," *IEEE Access*, vol. 7, pp. 169204–169216, 2020.

[26] L. Xiao, D. Jiang, D. Xu, H. Zhu, Y. Zhang, and H. V. Poor, "Two-dimensional antijamming mobile communication based on reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9499–9512, Oct. 2018.

[27] N. Gao, Z. Qin, X. Jing, Q. Ni, and S. Jin, "Anti-intelligent UAV jamming strategy via deep Q-Networks," *IEEE Trans. Commun.*, vol. 68, no. 1, pp. 569–581, Jan. 2020.

[28] L. Xiao, G. Han, D. Jiang, H. Zhu, Y. Zhang, and V. Poor, "Twodimensional anti-jamming mobile communication based on reinforcement learning," *CoRR*, vol. abs/1712.06793, pp. 1–15, Oct. 2017.

[29] X. Wang, J. Wang, Y. Xu, J. Chen, L. Jia, X. Liu, and Y. Yang, "Dynamic spectrum anti-jamming communications: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 58, no. 2, pp. 79–85, Feb. 2020.

[30] H. van Hasselt, "Double q-learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, pp. 2613–2621.

[31] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *Proc. ACM SIGKDD Conf. Knowl. Discovery Data Mining*, 2019, pp. 1–8.

[32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[33] J. Lei Ba, J. Ryan Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*. [Online]. Available: http://arxiv.org/abs/1607.06450

[34] M. Wilhelm, I. Martinovic, J. B. Schmitt, and V. Lenders, "Short paper: Reactive jamming in wireless networks: How realistic is the threat?" in *Proc. 4th ACM Conf. Wireless Netw. Secur.*, 2011, pp. 47–52.

JIANLIANG XU received the B.S. degree in physics from Zhejiang University, Hangzhou, China, in 1992, and the M.S. degree in communication and information system from Xidian University, Xi'an, China, in 1999. He is currently a Senior Researcher with the Science and Technology on Communication Information Security Control Laboratory, Jiaxing, China. He holds seven patents. His research interests include software-defined radio, spectrum management, and deep learning-based radio signal processing.

HUAXUN LOU was born in Zhejiang, China, in 1981. He received the B.S. degree in computer science and technology from the Zhejiang University of Science and Technology, in 2006, and the M.S. degree in computer science from Hangzhou Dianzi University, in 2009. From 2009 to 2010, he was a Researcher in electric engineering with Zhejiang University. He is currently an Associate Researcher with the Science and Technology on Communication Security Control Laboratory, Jiaxing, China. He holds ten patents. His research interests include computer science and technology, cognitive radio, spectrum management, and deep learning-based radio signal processing.

WEIFENG ZHANG received the B.S. degree in electronic information engineering from the Beijing University of Technology, in 2009, the M.S. degree in pattern recognition from Beihang University, in 2012, and the Ph.D. degree in computer science from Hangzhou Dianzi University, in 2019. From 2012 to 2018, he was a Senior Engineer with the Science and Technology on Communication Information Security Control Laboratory. He is currently an Associate Professor with Jiaxing University. His work has been published in prestigious journals such as the IEEE Transactions on Multimedia, the IEEE Transactions on Image Processing, *Pattern Recognition*, and *Information Fusion*. His research interests include machine learning and its application on multimodal intelligence and communication.

GAOLI SANG received the Ph.D. degree in computer science and technology from Sichuan University, in 2016. She is currently a Full Lecturer with the Department of Mathematics and Information Engineering, Jiaxing University, Jiaxing, China. Her current research interests include pattern recognition and image processing.