TWO-DIMENSIONAL ANTI-JAMMING COMMUNICATION BASED ON DEEP REINFORCEMENT LEARNING

Guoan Han[*], *Liang Xiao*[*], *H. Vincent Poor*[†]

[*] Dept. Communication Engineering, Xiamen University, 361000, China. Email: lxiao@xmu.edu.cn
[†] Dept. Electrical Engineering, Princeton University, Princeton, NJ, USA

ABSTRACT

In this paper, a two-dimensional anti-jamming communication scheme for cognitive radio networks is developed, in which a secondary user (SU) exploits both spread spectrum and user mobility to address jamming attacks, while not interfering with primary users. By applying a deep Q-network algorithm, this scheme determines whether to recommend that the SU leave an area of heavy jamming and chooses a frequency hopping pattern to defeat smart jammers. Without knowing the jamming model and the radio channel model, the SU derives an optimal anti-jamming communication policy using Q-learning in a proposed dynamic game, and applies a deep convolution neural network to accelerate the learning speed with a large number of frequency channels. The proposed scheme can increase the signal-to-interference-plus-noise ratio and improve the utility of the SU against cooperative jamming, compared with a Q-learning-only based benchmark system.

***Index Terms*—** Cognitive radio networks, jamming, deep reinforcement learning, game theory, deep Q-networks

1. INTRODUCTION

In cognitive radio networks (CRNs), secondary users (SUs) have to avoid interfering with the communications of primary users (PUs) and should counteract jammers that inject jamming signals to interrupt the ongoing transmissions of SUs for the purpose of denial of service (DoS). Spread spectrum techniques, such as frequency hopping (FH) and direct-sequence spread spectrum have been used for decades as anti-jamming techniques in wireless communications. However, by applying smart radio devices such as universal software radio peripherals, jammers can cooperate to block most frequency channels of the CRN and thus interrupt the transmissions of FH-based SUs in the area. In addition, spectrum sensing and eavesdropping on the control channel that broadcasts the FH pattern further increases the jamming strength against FH-based CRN [1].

User mobility can improve the signal-to-interference-plus-noise ratio (SINR) of the SU signals, if the user simply leaves the area of heavy jamming in which most frequency channels are blocked by smart jammers. Therefor, we consider a two-dimensional (2-D) anti-jamming communication system that applies both frequency hopping and user mobility to address smart jammers. However, the communication system has to make a tradeoff between the signal SINR and the security cost as it requires a much higher cost for an SU to change its geographical location compared with frequency hopping.

Game theory has been widely applied to address jamming in wireless communications [2–7]. For instance, an anti-jamming power control Stackelberg game presented in [3] formulates the interactions among a jammer, a relay node and a source node that choose their power allocation strategies in sequence without interfering with PUs. The prospect-theory based dynamic jamming game in [5] investigates the impact of the subjective decision making process of a jammer. Communication against reactive jamming is formulated in [6] as a Stackelberg game. Further, the Bayesian anti-jamming communication game in [7] studies jammers with unknown types of intelligence.

Reinforcement learning techniques such as Q-learning can learn an optimal policy via trials in Markov decision processes. For example, a Q-learning based power control strategy developed in [3] effects a tradeoff between the cost of defense and communication efficiency without being aware of the jamming model. The Q-learning based channel allocation procedure proposed in [8] provides an optimal channel accessing strategy in the multi-channel dynamic anti-jamming game. The on-policy synchronous Q-learning based channel allocation in [9] proactively avoids the jammed channels in the CRN. However, the Q-learning algorithm suffers from slow learning speeds if the state space and the action set are large, thus yielding anti-jamming performance degradation.

In this paper, we investigate a frequency-spatial anti-jamming communication game and propose a 2-D anti-jamming system based on a deep Q-network (DQN) algorithm, a deep reinforcement learning technique recently

This research was supported in part by National Natural Science Foundation of China (61671396, 61271242), in part by the U.S. National Science Foundation under Grants CNS-1456793 and CMMI-1435778, and in part by China Computer Federation Venustech Hongyan Research Initiative (2016-010).

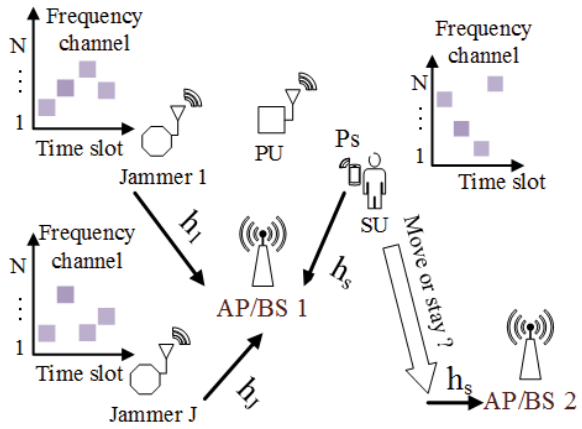


Fig. 1. Network model.

developed by Google DeepMind [10]. By exploiting a deep convolutional neural network (CNN), the DQN-based anti-jamming system can address the curse of high-dimensionality of Q-learning and accelerate the learning speed. In particular, in our approach the SU uses a DQN to choose a frequency channel and determine whether to leave the area of heavy jamming without being aware of the jamming model and the radio channel model in a dynamic game formulation based on the history against jamming attacks.

The rest of this paper is organized as follows: We present the anti-jamming communication game in Section 2 and propose a 2-D DQN-based anti-jamming communication system in Section 3. We provide simulation results in Section 4 and conclude in Section 5.

2. ANTI-JAMMING COMMUNICATION GAME

We consider the transmission of an SU against J cooperative jammers that are randomly located in the CRN with frequency hopping over N frequency channels, as shown in Fig. 1. At time slot k, the SU chooses an action, denoted by $x^k \in \{0, \cdots, N\}$ to determine whether to leave the geographical area, and which channel to use to send signals with a given power P_s. For example, the user in Fig. 1 stays in the area and sends the signals to access point (AP)1 if $x^k > 0$; otherwise, the user moves to another area and connects to AP2 if $x^k = 0$. Let h_s denote the radio channel power gain from the SU to the serving AP or Base Station (BS), and C_m be the extra cost of user mobility compared with frequency hopping. For simplicity, we assume that each AP or BS can receive signals from all N channels, although our scheme can be extended to other cases as well.

Jammer j chooses channel $y_j^k \in \{1, \cdots, N\}$ to send jamming signals with a given power P_j. The channel power gain from the jammer to the current serving AP or BS is denoted by h_j, $j = 1, \cdots J$. Both the SU and the J jammers have to avoid interfering with PUs, whose presence is denoted by λ, which equals 1 if a PU is accessing Channel x^k in the current area and equals zero otherwise.

For simplicity, we assume that these J jammers cannot interfere with the new AP if the SU moves to a new area and the channel power gain to the new AP is still h_s in the simulations. Based on the SINR and the transmission cost, the utility of the SU (or jammer) at time slot k in the zero-sum game, denoted by u_s^k (or u_j^k), is defined as

$$u_s^k = -u_j^k = \frac{P_s h_s \lambda}{\sigma + \Sigma_{j=1}^{J} P_j h_j f(x^k = y_j^k)} - C_m f(x^k = 0), \tag{1}$$

where σ is the receiver noise power, and $f(\xi)$ is an indicator function that equals 1 if ξ is true, and 0 otherwise. The first term on the right-hand-side of (1) is the SINR of the signal.

3. DQN-BASED ANTI-JAMMING COMMUNICATION

In the dynamic anti-jamming game, an SU can apply Q-learning to derive an optimal policy to determine whether to leave the area and choose a channel, without being aware of the jamming model and the radio channel model. The action of the SU is selected based on the system state at time k denoted by $\mathbf{s}^k$, which represents the state of the radio environment including the PUs, the jammers and the serving BS/AP. More specifically, the system state $\mathbf{s}^k$ consists of the presence of PUs and the SINR of the signal at time $k-1$, i.e., $\mathbf{s}^k = [\lambda^{k-1}, \text{SINR}^{k-1}]$.

Table 1. CNN parameters

Layer	Conv 1	Conv 2	FC 1	FC 2
Input	6×6	$4 \times 4 \times 20$	360	180
Filter size	3×3	2×2	/	/
Stride	1	1	/	/
No. of filters	20	40	180	$N+1$
Activation	ReLU	ReLU	ReLU	ReLU
Output	$4 \times 4 \times 20$	$3 \times 3 \times 40$	180	$N+1$

As illustrated in Fig. 2, the DQN-based communication system uses a deep convolutional neural network to accelerate the learning speed of Q-learning as a large number (N) of frequency channels are involved. The DQN algorithm updates a quality-function or Q-function for each state-action pair, which is the expected discounted long-term reward for state $\mathbf{s}$ and action x, i.e.,

$$Q(\mathbf{s}, x) = \mathbb{E}_{\mathbf{s}'}\left[u_s + \gamma \max_{x'} Q(\mathbf{s}', x') | \mathbf{s}, x\right], \tag{2}$$

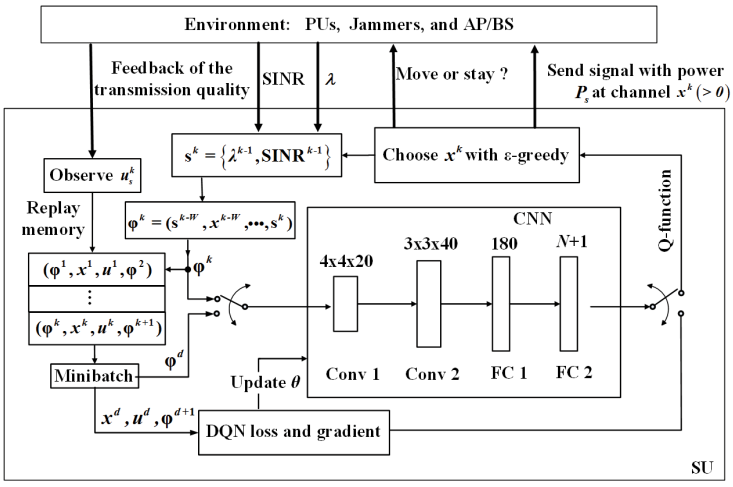


Fig. 2. DQN-based anti-jamming communication system.

where $\mathbf{s}'$ is the next state if the SU takes action x at state $\mathbf{s}$, and γ is the discount factor that represents the uncertainty of the SU about the future reward.

The CNN is used as a nonlinear function approximator to estimate the value of the Q-function for each action, because the number of feasible values of $\mathbf{s}^k$ is too large to quickly derive the optimal policy. The CNN consists of two convolutional (Conv) layers and two fully connected (FC) layers. The first Conv layer includes 20 filters each with size 3×3 and stride 1, and the second Conv layer has 40 filters each with size 2×2 and stride 1, as shown in Table 1. Both Conv layers use the rectified linear unit (ReLU) as the activation function. The first FC layer involves 180 rectified linear units, while the second FC layer has $N + 1$ units for the action set. The filter weights of the four layers in the CNN at time k are denoted by θ^k.

Let $\boldsymbol{\varphi}^k$ denote the state sequence at time k, which consists of the current system state and the previous W system state-action pairs, i.e., $\boldsymbol{\varphi}^k = (\mathbf{s}^{k-W}, x^{k-W}, \cdots, x^{k-1}, \mathbf{s}^k)$. The state sequence is then reshaped into a 6×6 matrix as the input to the CNN to estimate $Q(\boldsymbol{\varphi}^k, x|\theta^k)$, $\forall 0 \le x \le N$. The CNN parameters θ^k are updated at each time slot based on experience replay.

The experience observed by the SU is denoted by $\mathbf{e}^k = (\boldsymbol{\varphi}^k, x^k, u_s^k, \boldsymbol{\varphi}^{k+1})$, and the memory pool at time k is given by $\mathcal{D} = \{\mathbf{e}^1, \cdots, \mathbf{e}^k\}$. The experience replay chooses an experience $\mathbf{e}^d$ from memory pool $\mathcal{D}$ at random, with $1 \le d \le k$ to update θ^k according to a stochastic gradient descent (SGD) algorithm. The mean-squared error of the target optimal Q-function value is minimized with minibatch updates, and the loss function is chosen by [10] as

$$L(\theta^k) = \mathbb{E}_{\boldsymbol{\varphi}^k, x, u_s, \boldsymbol{\varphi}^{k+1}} \left[\left(R - Q\left(\boldsymbol{\varphi}^k, x; \theta^k\right) \right)^2 \right], \quad (3)$$

where R is the target optimal Q-function, which is given by

$$R = u_s + \gamma \max_{x'} Q\left(\boldsymbol{\varphi}^{k+1}, x'; \theta^{k-1}\right). \quad (4)$$

The gradient of the loss function with respect to the

Algorithm 1. 2-D anti-jamming communication system based on DQN

Initialize θ, γ, P_s, λ^0, SINR^0, $\mathbf{s}^0 = [\lambda^0, \mathrm{SINR}^0]$, W, B, $\mathcal{D} = \emptyset$
For $k = 1, 2, \cdots$
  If $k \le W$
    Choose Channel $x^k \in \{0, 1 \cdots N\}$ at random
  Else
    Obtain the output $Q(\boldsymbol{\varphi}^k, x|\theta^k)$ from the CNN with input $\boldsymbol{\varphi}^k$ and weights θ^k
    Choose x^k via the ϵ-greedy algorithm
  End if
  If $x^k = 0$
    Recommend that the SU leave the area
  Else
    Use Channel x^k to send signals with power P_s
  End if
  Observe SINR^k and λ^k
  Obtain u_s^k and $\mathbf{s}^{k+1} = [\lambda^k, \mathrm{SINR}^k]$
  $\boldsymbol{\varphi}^{k+1} = \left(\mathbf{s}^{k-W+1}, x^{k-W+1}, \cdots, x^k, \mathbf{s}^{k+1}\right)$
  Store the new experience $\left\{\boldsymbol{\varphi}^k, x^k, u_s^k, \boldsymbol{\varphi}^{k+1}\right\}$ in $\mathcal{D}$
  For $d = 1, 2, \cdots, B$
    Select $\left(\boldsymbol{\varphi}^d, x^d, u_s^d, \boldsymbol{\varphi}^{d+1}\right)$ from $\mathcal{D}$ at random
    Calculate R via (4)
  End for
  Update θ^k via (5)
End for

weights θ^k is given by

$$\nabla_{\theta^k} L(\theta^k) = \mathbb{E}_{\boldsymbol{\varphi}^k, x, u_s, \boldsymbol{\varphi}^{k+1}} \left[R \nabla_{\theta^k} Q\left(\boldsymbol{\varphi}^k, x; \theta^k\right) \right] - \mathbb{E}_{\boldsymbol{\varphi}^k, x} \left[Q\left(\boldsymbol{\varphi}^k, x; \theta^k\right) \nabla_{\theta^k} Q\left(\boldsymbol{\varphi}^k, x; \theta^k\right) \right]. \quad (5)$$

This process repeats B times at each time slot and θ^k is updated according to the B randomly selected experiences.

The action x^k is chosen according to the Q-function and $\mathbf{s}^k$. According to the ϵ-greedy algorithm, the optimal action $x^* = \arg\max_{x'} Q(\boldsymbol{\varphi}^k, x')$ is chosen with a high probability $1-\epsilon$, and another action is selected with a low probability ϵ/N to avoid staying in the local maximum. If x^k is 0, it is suggested that the SU leave the area. Otherwise, the SU transmits on Channel x^k. Next, the SU receives the SINR information as feedback from the serving AP or BS, and receives the utility u_s^k. According to the next state sequence $\boldsymbol{\varphi}^{k+1}$, the SU stores the new experience $\left\{\boldsymbol{\varphi}^k, x^k, u_s^k, \boldsymbol{\varphi}^{k+1}\right\}$ in the memory pool $\mathcal{D}$. The DQN-based anti-jamming system is summarized in Algorithm 1.

Algorithm 2. Q-learning based system

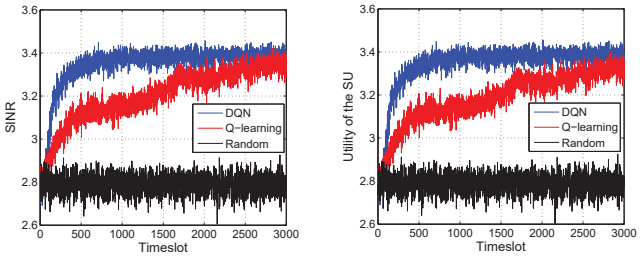
Initialize γ, α, P_s, λ^0, SINR^0, $\mathbf{s}^0 = [\lambda^0, \mathrm{SINR}^0]$,
$Q(\mathbf{s}, x) = 0$, $V(\mathbf{s}) = 0$, $\forall \mathbf{s}, x$
For $k = 1, 2, \cdots$
  Choose x^k via ϵ-greedy
  If x^k = 0
    Recommend that the SU leave the area
  Else
    Use Channel x^k to send signals with power P_s
  End if
  Obtain SINR^k, u_s^k and λ^k
  $\mathbf{s}^{k+1} = [\lambda^k, \mathrm{SINR}^k]$
  Update $Q(\mathbf{s}^k, x^k)$ via (6)
  Update $V(\mathbf{s}^k)$ via (7)
End for

As a benchmark, we propose a Q-learning based 2-D anti-jamming communication system as shown in Algorithm 2, in which the Q-function is estimated according to iterative Bellman equation as follows:

$$Q\left(\mathbf{s}^k, x^k\right) \leftarrow \alpha\left(u_s^k\left(\mathbf{s}^k, x^k\right) + \gamma V\left(\mathbf{s}^{k+1}\right)\right) + (1-\alpha) Q\left(\mathbf{s}^k, x^k\right) \quad (6)$$

$$V\left(\mathbf{s}^k\right) \leftarrow \max_x Q\left(\mathbf{s}^k, x\right), \quad (7)$$

where $V(\mathbf{s}^k)$ is the value function of $\mathbf{s}^k$, and α is the learning rate. The convergence rate of the Q-learning algorithm depends on the size of the state space, which increases with N, and thus the system has to address the curse of dimensionality of Q-learning.



(a) SINR (b) Utility of the SU

Fig. 3. Performance of the DQN-based anti-jamming communication scheme.

4. SIMULATION RESULTS

Simulations have been performed to evaluate the performance of our proposed anti-jamming communication scheme. The primary user randomly chooses a channel out of the $N = 128$ channels during the dynamic game, with $\sigma = 1$, $\epsilon = 0.1$, $C_m = 0.2$, $h_s \in (0,1)$, $h_j \in (0,1)$, and $P_s = P_j = 5$, against $J = 2$ jammers. One jammer sweeps the N channels and the other applies the ϵ-greedy algorithm to choose the jamming channel based on the last transmission channel of the SU.

As shown in Fig. 3, the DQN-based anti-jamming communication system outperforms the Q-learning based system, which in turn exceeds the FH system that randomly chooses a channel, with a higher SINR, lower cost of defense and higher utility. More specifically, the DQN-based system has a faster convergence rate than the Q-learning algorithm, and achieves a higher SINR. For instance, the SINR of the SU's signal increases from 2.78 at the beginning of the game to 3.4 at time slot 1000, while the SINR of the Q-learning based strategy is only 3.16 at that time. Therefore, the utility of the SU increases from 2.73 at the beginning to 3.39 at time slot 1000, which is 8.3% higher than that of the Q-learning strategy, and it does so with a faster convergence rate.

5. CONCLUSIONS

In this paper, we have formulated a dynamic anti-jamming communication game for CRNs, which exploits both spread spectrum and user mobility to improve the SINR of the signals against cooperative smart jammers. A DQN-based communication system is proposed for an SU to achieve the optimal frequency hopping policy and decide whether to leave the area of heavy jamming, without being aware of the jamming model and the radio channel model. By applying the deep CNN technique, our proposed anti-jamming system outperforms the Q-learning strategy with a faster convergence rate, higher SINR, lower cost of defense and higher utility of the SU.

6. REFERENCES

[1] W. Trappe, "The challenges facing physical layer security," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 16–20, Jun. 2015.

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

[3] L. Xiao, Y. Li, J. Liu, and Y. Zhao, "Power control with reinforcement learning in cooperative cognitive radio networks against jamming," *Journal of Supercomputing*, pp. 3237–3257, Apr. 2015.

[4] M. Labib, S. Ha, W. Saad, and J. H. Reed, "A Colonel Blotto game for anti-jamming in the internet of things," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, San Diego, CA, Dec. 2015.

[5] L. Xiao, J. Liu, Q. Li, N. B. Mandayam, and H. V. Poor, "User-centric view of jamming games in cognitive radio networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2578–2590, Dec. 2015.

[6] X. Tang, P. Ren, Y. Wang, Q. Du, and L. Sun, "Securing wireless transmission against reactive jamming: A Stackelberg game framework," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, San Diego, CA, Dec. 2015.

[7] A. Garnaev, Y. Liu, and W. Trappe, "Anti-jamming strategy versus a low-power jamming attack when intelligence of adversary's attack type is unknown," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 1, pp. 49–56, Mar. 2016.

[8] 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 Conference on Communications and Network Security (CNS)*, pp. 28–36, Washington, DC, Oct. 2013.

[9] F. Slimeni, B. Scheers, Z. Chtourou, and V. L. Nir, "Jamming mitigation in cognitive radio networks using a modified Q-learning algorithm," in *Proc. IEEE International Conference on Military Communications and Information Systems (ICMCIS)*, pp. 1–7, Cracow, Poland, May. 2015.

[10] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, and G. Ostrovski, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Jan. 2015.