Q-Learning for Non-Cooperative Channel Access Game of Cognitive Radio Networks

He Jiang*, Haibo He*, Lingjia Liu†, and Yang Yi†
*Department of Electrical, Computer, and Biomedical Engineering
University of Rhode Island
Kingston, RI 02881, USA
Email: {hjiang, he}@ele.uri.edu
†Bradley Department of Electrical and Computer Engineering
Virginia Tech, Blacksbur, VA, 24060, USA
Email: {ljliu, cindy_ yangyi}@vt.edu

***Abstract*—This paper investigates the channel access problem of cognitive radio networks. In the cognitive radio network, communication channels are assigned to primary users with priority while secondary users are able to detect the spectrum holes and switch among the channels for data transmission opportunities. The channel access problem of this kind of system can be formulated as a non-cooperative game. However, in prior works, the secondary users are usually assumed to be able to switch to any channel instantaneously, which is not possible in reality because the channel switching will incur transmission delays. In this paper, we formulate the channel access problem as a non-cooperative game where each channel can be used by only one user at a time. Moreover, considering the transmission delays, we limit the channel switching distance of the secondary users to a certain scope. In this case, the optimal channel access policy of each secondary user will depend on the long-term behaviors of primary users as well as the actions of other secondary users. For this non-cooperative game, we propose a multiagent Q-learning algorithm which requires neither the prior knowledge of channel dynamics nor the negotiations among players. Simulation examples are provided to demonstrate the effectiveness of the algorithm.**

I. Introduction

The radio spectrum is necessary for data transmission of wireless devices. Traditionally, static spectrum access policies are used to allocate the spectrum to different users, where the assigned pieces of spectrum can only be accessed by the corresponding licensed users. In recent years, as the population of wireless devices increasing dramatically, available radio spectrum has almost been fully allocated in several countries [1]. However, it is shown that a lot of assigned radio spectrum bands are often in an idle status [2], which is a great waste. Obviously, exploiting the idle assigned spectrum will alleviate the spectrum shortage problem. The cognitive radio network (CRN) [3] with opportunistic spectrum access policy has been proposed as a promising solution to improve the spectrum utilization efficiency.

In a CRN, the wireless devices are divided into two categories, i.e., primary users and secondary users, which are also referred to as licensed users and cognitive users, respectively. Communication channels are assigned to primary users with priority such that they can use the channels any time they need. The use permissions of the communication channels are also given to secondary users while the channels are not occupied by primary users, which is known as the opportunistic spectrum access. The spectrum utilization efficiency can be largely improved if secondary users can make use of the idle spectrum bands appropriately. To adapt to the opportunistic spectrum access, cognitive users should be able to detect spectrum holes and share the idle spectrum bands with other cognitive users. In this paper, we focus on the spectrum access problem while spectrum sensing ability of the secondary user is assumed to be perfect to simplify the analysis. However, our method can be conveniently integrated with traditional channel detection technique, such as energy detector and matched-filtering [4].

Opportunistic spectrum access can be deemed as an optimization problem which involves a group of secondary users. To solve this kind of problem, game theory is usually employed and solutions depend on the properties of players and the optimization objectives. For the secondary users with cooperative behaviors or the common objective, cooperative game theory can be applied [5]–[7]. However, secondary users are not always cooperative in reality as the wireless devices are usually independent. In this case, some previous works formulate the channel access problems as non-cooperative games, in which each secondary user has its own optimization objective and compete with its peer users for the opportunistic spectrum bands [8], [9]. Generally, a channel access game is dynamical since the spectrum opportunities are not constant and the peer secondary users may access different channels from time to time. As a result, the optimal channel access policy of a secondary user depends on the behaviors of both the primary users and its peers. However, it can be very difficult for a secondary user to get knowledge of the behaviors of the primary users or the policies of its peers. Thus, manually designing an optimal channel selection policy

978-1-5090-6014-6/18/$31.00 ©2018 IEEE

is not appropriate.

To deal with this problem, reinforcement learning can be applied. Reinforcement learning [10] is a machine learning algorithm which enables an agent to learn an optimal policy adaptively without environment knowledge. Reinforcement learning has already been applied to many areas such as optimal control [11], [12], finance investment [13], smart grid [14], [15], and social behaviors [16], [17], just to name a few. Reinforcement learning has also been employed to solve the spectrum access problems in several prior works [18]–[20]. In [18], the non-cooperative channel selection problem of two users and two channels are solved with Q-learning. The Q-learning algorithm is extended to the case of multiple users and multiple channels in [20]. The cooperative spectrum access problem is solved in [19] where the distributed learning algorithms are proposed to maximize the total throughput of the cognitive system. In these works, the communication systems are time slotted and the users could switch to any channel at every time slot. In this way, the users only need to consider the current channel state, and select channel based on the immediate reward. However, switching from one channel to another will incur a transmission delay which is proportional to the channel distance in reality [21], [22]. Therefore, it is necessary to develop a new method for the channel access problem with considering the channel switching ability.

In this paper, we formulate the opportunistic channel access problem as a non-cooperative game. In the game, primary users will use the communication channels dynamically. The secondary users will access the idle channel with probable collisions with others. For an idle channel, if more than one secondary user decides to use it, there will be a collision and none of them will complete the transmission. Each secondary user needs to switch among the opportunistic channels to maximize its data transmission over the whole game. Considering the transmission delays, the channel switching of each secondary user is limited to a certain distance. Under these settings, the secondary users need to make decisions based on a long-term consideration instead of being attracted by the immediate payoff. We have two main contributions in this paper. First, we formulate the channel access problem as a non-cooperative game and limit the channel switching ability of the player, which has not been studied in existing works. Second, a multiagent Q-learning algorithm is designed to solve the channel access game. The prior knowledge of the channel dynamics and the negotiations among the players are not required by the algorithm.

The rest of this paper is organized as follows. In Section II, the system model studied in this paper is introduced and the corresponding channel access game is formulated. Section III presents an overview of the Q-learning algorithm and proposes a multiagent Q-learning solution to the channel access game. Simulation experiments are provided to demonstrate the effectiveness of the method in Section IV. Finally, a brief conclusion is given in Section V.

II. System model and Problem formulation

In this section, we first describe the system model of the cognitive radio network. Then, the non-cooperative channel access game is formulated.

A. System model

Consider a cognitive radio network containing N secondary users and M communication channels of the same bandwidth and different center frequencies. These M channels compose a sequence $\{C_i\}_{i=1}^{M}$ where C_i is the channel of the i-th lowest center frequency. Additionally, we define the distance between C_i and C_j as $d_{ij} = \parallel i - j \parallel$. The communication system is discretized into time slots with equal length. At each time slot, the channels may be occupied by primary users. The state of a channel is denoted by '1' if it is occupied by a primary user; otherwise, '0'.

The secondary users are allowed to access the channels which are in '0'. The secondary users are located close to each other and will be influenced by the same group of primary users. We assume the data transmission of the secondary users follows the collision mechanism used in [18]–[20], where if more than one secondary user tries to transmit data through a same idle channel at one time slot, none of them will succeed. Therefore, to transmit data as much as possible, the secondary users should be able to detect the channel state while avoid colliding with other secondary users. As channel detection technique is out of the research scope of this paper, we assume that the secondary users can detect channel states perfectly to simplify the analysis. The objective of this paper is to design a learning algorithm that enables secondary users to find the optimal channel selection policy through which the data transmission can be maximized.

B. Non-cooperative Channel Allocation Game

For the communication system described above, we design the following non-cooperative channel access game. The players in the game are the secondary users and the i-th secondary user is denoted by u_i, $i \in \{1, \ldots, N\}$. The total length of the game is T. At time slot t, $1 \leq t \leq T$, the state of the channels is described by the state vector

$$\overline{s}^t = \begin{bmatrix} s_1^t \\ s_2^t \\ \vdots \\ s_M^t \end{bmatrix}, \tag{1}$$

where $s_i^t \in \{0, 1\}$ is the state of C_i. During the game, the transition of $\overline{s}^t$ is a Markov process [23]. Here, we assume that the transition matrix of $\overline{s}^t$ is unknown to $\forall u_i$. Each u_i can detect $\overline{s}^t$ perfectly, however, the secondary users can only implement channel detection every p time slots since

the channel detection is an energy consuming process [24]. Based on the channel detection results, u_i can select a channel to transmit data for the following p time slots. The timing structure of the game is illustrated in Fig. 1. Due to the

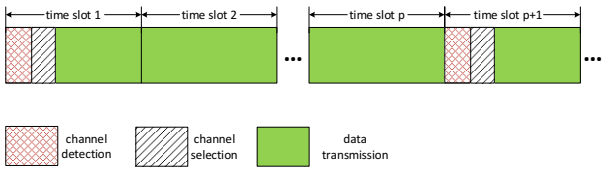


Fig. 1. Timing structure of the channel access game. The players can perform the channel detection and channel selection every p time slots.

hardware limitation, switching from one channel to another will induce a transmission delay which is proportional to the channel distance [21], [22]. Therefore, given a player currently accessing C_i, we set it can only switch to the channel that is in $\{C_j| \parallel i-j \parallel \leq 1\}$. If u_i transmits data through channel j at time slot t, it will receive a reward about whether its transmission is successful or not, which is

$$r_{ij}^t = \begin{cases} 1, & \text{if the transmision success;} \\ 0, & \text{if the transmision fail.} \end{cases} \tag{2}$$

In one game, the total reward of user i is $\sum_{t=0}^{T} r_{ij}^t$, which is the objective to maximize. It should be noted that in the time slots where the secondary users perform channel detection and selection, the data transmission length is shorter than that of the normal time slot so that it is more reasonable to set the success reward in these time slots less than 1. However, we neglect this difference since the channel detection and selection are not frequently performed.

It can be observed that the channel access game is non-cooperative since the reward of each player just concerns about its own data transmission. Additionally, since the channel access of one secondary user will influence others' data transmission, the optimal policy of a player depends on the actions of all its peers. To deal with this problem, previous works usually introduce the negotiation process [25], [26]. However, the negotiation is time-consuming and inefficient so that it should be avoided in a realistic communication system. To search the optimal policy without negotiations, we set that during the channel detection process, each player can get the information of other players' current channel selections, which can be achieved by the broadcast technique introduced in [27], [28]. Thus, if a player learns the policy of others, it can predict their future behaviors based on their current channel selections. In the following section, we propose a multiagent Q-learning algorithm that enables the secondary users to maximize the data transmission.

III. Multiagent Q-learning for the channel access game

In this section, we first introduce the framework of Q-learning for optimization problems. Then, we extend the Q-learning to solve the non-cooperative channel access game.

A. Q-learning

Q-learning is a reinforcement learning algorithm that enables an agent to learn the optimal policy of a Markov decision process (MDP) from the interactions with the environment. Generally, the agent-environment interaction in a reinforcement learning process can be illustrated by Fig.2. At time step

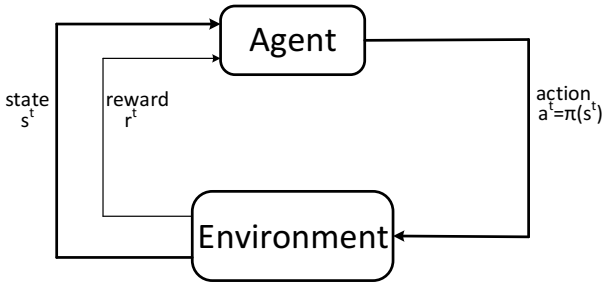


Fig. 2. The general agent-environment interaction process [10].

t, the agent will observe the environment sate s^t and take an action a^t correspondingly, which can be described by

$$a^t = \pi(s^t) \tag{3}$$

where $\pi(\cdot)$, also known as the policy, is a mapping from the environment state to the action space. Then, the environment transits to the next state with probability of $P(s^{t+1}|s^t, a)$ and returns a feedback reward $r^{t+1} = r(s^t, a^t, s^{t+1})$ to the agent. Under policy π and state s, the expected discounted cumulative rewards over an infinite time horizon is defined as

$$V_\pi(s) = E[\sum_{i=0}^{\infty} \gamma^i r^{t+i+1})|s^t = s]. \tag{4}$$

In reinforcement learning, the objective of an agent is to find an optimal policy to maximize the cumulative rewards. Let the action value, or the Q value, be defined as

$$Q_\pi(s,a) = E_\pi[\sum_{i=0}^{\infty} \gamma^i r^{t+i+1}|s^t = s, a^t = a] \tag{5}$$

which represents the expected discounted rewards of taking action a under state s. Given all action values of a certain state, the optimal policy should be selecting the action with the highest expected returns. Therefore, the optimal policy can be expressed by

$$\pi^*(s) = \arg\max_a \{Q_{\pi^*}(s,a)\}. \tag{6}$$

Additionally, the expected cumulative rewards of the optimal policy under state s is

$$V_{\pi^*}(s) = \arg\max_a Q_{\pi^*}(s,a). \tag{7}$$

Furthermore, we can obtain the relation of Q_{π^*} and V_{π^*} which is

$$Q_{\pi^*}(s,a) = \sum_{s'} P(s'|s,a)r(s,a,s') + \sum_{s'} P(s'|s,a)V_{\pi^*}(s'). \tag{8}$$

Q-learning is an off policy algorithm that aims to find the optimal policy by approximating Q_{π^*} recursively. In Q-learning, the agent take actions randomly and keep updating the Q value of each action with

$$Q(s^t,a^t) \leftarrow (1-\alpha)Q(s^t,a^t) + \alpha(r^{t+1} + \gamma \max_a Q(s^{t+1},a)) \tag{9}$$

where α is the learning rate. As demonstrated in [29], Q_π will converge to Q_{π^*} with probability 1.

B. Multiagent Q-learning Scheme

Since the channel access game we formulated is a MDP, it is reasonable to solve it with Q-learning. As described in Section II, each secondary user will implement the channel selection every p time steps so that the total number of the actions taken by u_i in a game of length T is

$$K = \lfloor \frac{T}{p} \rfloor + 1 \tag{10}$$

where $\lfloor \cdot \rfloor$ is the floor function. Let the kth action of u_i be represented by a_i^k, $k \in \{1,\ 2,\ \cdots,\ K\}$. It can be observed that action index and the time slot index are not consistent. To make the expression clear, we use the following expression

$$k_t = (k-1)p + 1 \tag{11}$$

where k_t is the time slot index when the kth action is performed. Since an action of u_i is selecting a channel to transmit data, we define the action space of the player by the channel index, i.e., $a_i^k \in \{1,\ 2,\ \cdots, M\}$. According to the game setting, the kth channel option of u_i is related to the channel it selects at $k-1$, which can be described by

$$a_i^k \in \begin{cases} \{a_i^{k-1}-1,\ a_i^{k-1},\ a_i^{k-1}+1\}, & 1 < a_i^{k-1} < M; \\ \{a_i^{k-1},\ a_i^{k-1}+1\}, & a_i^{k-1} = 1; \\ \{a_i^{k-1}-1,\ a_i^{k-1}\}, & a_i^{k-1} = M. \end{cases} \tag{12}$$

Define $i-$ as the set that contains all players other than i. Additionally, let a_{i-} be the collection of actions taken by the users in $i-$. Based on the system property, the reward of u_i at time slot t is determined by the channel state and the actions of all players together, which indicates the action value of u_i should be in the form of $Q_i(\overline{s}^{k_t}, a_{i-}^k, a_i^k)$. Moreover, it can be inferred that the optimal policy of u_i should be

$$\pi_i^*(\overline{s}^{k_t}, a_{i-}^k) = \arg\max_a \{Q_i(\overline{s}^{k_t}, a_{i-}^k, a)\}. \tag{13}$$

Obviously, the optimal action of u_i depends on other players' actions. If all players make their decisions based on others' actions, it will incur an iterative process. However, as communications among the players are not allowed in the game, it is not possible to get the value of $Q_i(\overline{s}^{k_t}, a_{i-}^k, a_i^k)$. An alternative approach is considering the peer users as a part of the environment. In this case, for u_i, the game state can be deemed as

$$\hat{s}_i^{k_t} \equiv (\overline{s}^{k_t}, a_{i-}^{k-1}). \tag{14}$$

As a result, u_i only needs to learn the dynamics of $\hat{s}_i^{k_t}$. Accordingly, the Q value of u_i under policy π_i can be written as $Q_{\pi_i}(\hat{s}_i^{k_t}, a_i^k)$. Similar as (8), the optimal policy should fulfill

$$\begin{aligned} Q_{\pi_i^*}(\hat{s}_i^{k_t}, a_i^k) =& E[\sum_{t=k_t}^{k_t+p-1} r_i^t] + \\ & \sum_{\hat{s}_i^{(k+1)_t}} P(\hat{s}_i^{(k+1)_t} | \hat{s}_i^{k_t})) V_{\pi_i^*}(\hat{s}_i^{(k+1)_t}). \end{aligned} \tag{15}$$

Here, the discounted factor is set to $\gamma = 1$. Base on (15), we design a multiagent Q-learning algorithm for the non-cooperative channel access game which is shown in Table I.

TABLE I
PSEUDO CODE OF MULTIAGENT Q-LEARNING FOR CHANNEL ACCESS GAME

Initialization:
(1) ϵ, α, p, $\overline{m}$, T, K;
(2) $Q_i^k(\hat{s}_i, a_i) = 0$ and $R_i^{k+1} = 0$ for $\forall i \in \{1,\ \cdots\ N\}$, $\forall k \in \{1,\ \cdots\ K+1\}$;
(3) $m = 1$;
while $m \leq \overline{m}$ **do**
  reset $\overline{s}(1)$ and a_i^0 for $\forall u_i$; $t = 1$ and $k = 1$;
  for $t \leq T$ **do**
    if $(t-1)\%p == 0$
      choose action from $Q_i^k(\hat{s}_i^{k_t}, a_i)$ with ϵ-greedy policy;
      if $k \geq 2$
        update Q value with:
        $Q_i^{k-1}(\hat{s}_i^{(k-1)_t}, a_i^{k-1})$
        = $(1-\alpha)Q_i^{k-1}(\hat{s}_i^{(k-1)_t}, a_i^{k-1}) +$
        $\alpha(R_i^k + \max Q_i^k(\hat{s}_i^{(k-1)_t}, a_i^k))$
      end if
      $k = k+1$;
    end if
    $R_i^{k+1} = R_i^{k+1} + r_i^t$;
    $t = t+1$;
  end for
  $m = m+1$;
end while

In the algorithm, m is the trial index and $\overline{m}$ is the maximum trial number; '$\%$' is the modulo operation; and ϵ-greedy policy is applied, where the action will be randomly selected with the probability of ϵ. a_i^0 represents the channel position of u_i at the beginning of the game, which is random set. We can see that the Q value is related to index k because the channel access game is of finite length rather than infinite length.

IV. Simulations and Analysis

In this section, we apply the multiagent Q-learning algorithm to three simulated non-cooperative channel access games. The simulation settings and the simulation results are discussed in detail.

A. Simulation Settings

In our simulation, the cognitive radio network consists six channels with four different states. Specifically, these states are

$$\begin{cases} \overline{s}_1 = [1\ 0\ 1\ 0\ 0\ 0]^{\mathrm{T}} \\ \overline{s}_2 = [0\ 1\ 0\ 1\ 1\ 0]^{\mathrm{T}} \\ \overline{s}_3 = [1\ 0\ 0\ 1\ 0\ 1]^{\mathrm{T}} \\ \overline{s}_4 = [0\ 0\ 1\ 0\ 1\ 1]^{\mathrm{T}} \end{cases}$$

where '1' means the channel is occupied by the primary user and '0' denotes the channel is idle. During the game, the transition of the four states is a Markov process with the transition matrix

$$M = \begin{bmatrix} 0.8506 & 0.0906 & 0.0408 & 0.0180 \\ 0.0037 & 0.9267 & 0.0502 & 0.0194 \\ 0.0564 & 0.0235 & 0.8496 & 0.0705 \\ 0.1065 & 0.0728 & 0.0221 & 0.7986 \end{bmatrix}.$$

The length of the game is set to $T = 200$ and the channel selection interval is set to $p = 10$. Through Monte Carlo experiments, we can obtain the expected idle time slots quantity of each channel in one game which is shown in Fig.3. It can be observed that channel 3 is the idlest one with being idle for 132 time slots on average; channel 4 is the busiest one; and the average expected value of all channels is 106. Moreover, we can conclude that for the game where there is only one secondary user with fixed channel access policy, on average, if the user always choose channel 3, it will receive the total rewards of 132; if the user choose channel 4, it will get the total rewards of 67; and if the user selects channel randomly, the total rewards will be 106. We will show that, with the training of the Q-learning algorithm, the secondary user will learn a much better dynamic channel selection policy.

B. Simulation Results

First, we study the situation where there is only one player. In this case, this secondary user only needs to consider the dynamics of the primary users and switch its transmission channel adaptively. The simulation results are shown in Fig. 4. As we can see, the secondary user improves its performance gradually. Finally, the learned policy enables the user to get

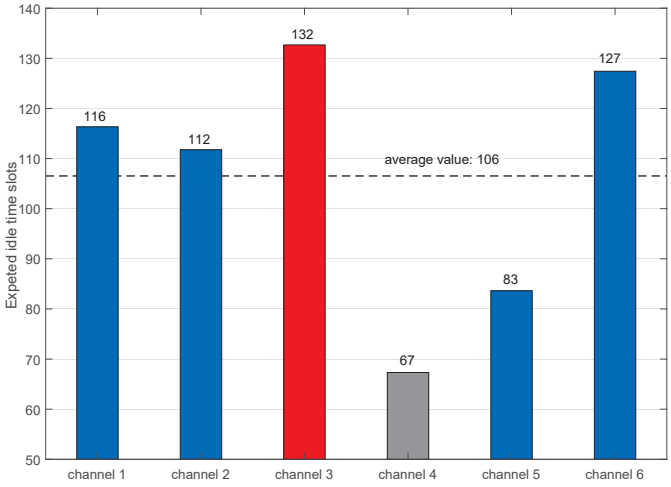


Fig. 3. The expected amount of the idle time slots of each channel in one game. The value will also be the expected total rewards of statically selecting the corresponding channel in an one-player game.

the average total rewards over 160. If the player applied fixed channel access policy, and keep accessing the idlest channel, which is channel 3, the expected total rewards will be 132. Therefore, with the training of the Q-learning algorithm, the expected success transmission rate is enhanced from 66% to 81%. Then, we simulate the channel access games of multiple

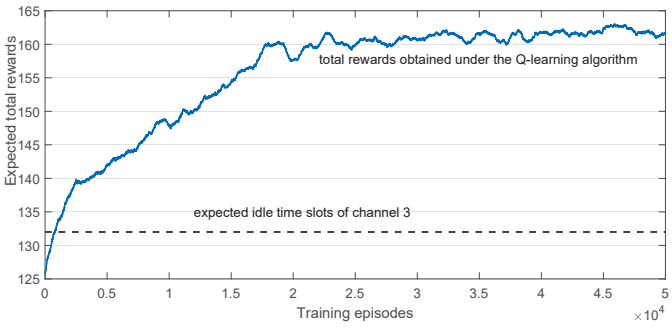
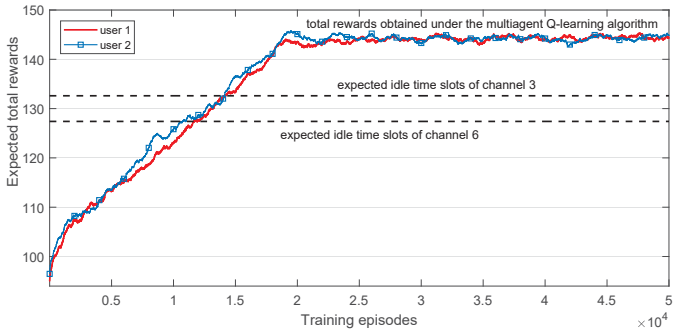


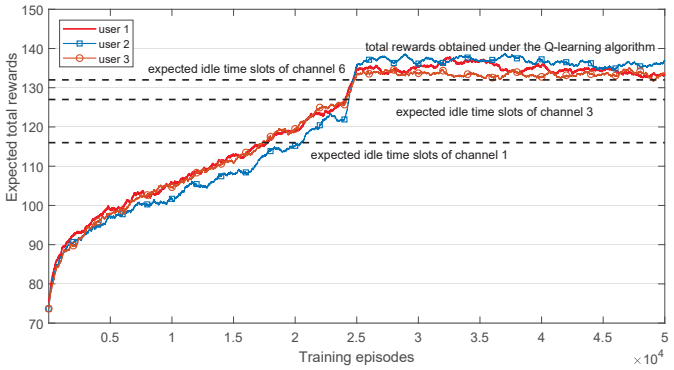
Fig. 4. The total rewards estimation of the user under the Q-learning algorithm in a one-player game. Trained by the Q-learning algorithm, the user's performance improves gradually and the total rewards converges to the value around 160. If the user applies the static channel access policy, on average, the high expected total rewards will be 132 which is the dashed line in the figure.

players. As the secondary users may collide with each other in the game, besides the dynamic of the primary users' behaviors, the players also need to learn the policy of its peers. We apply the multiagent Q-learning algorithm to the game of two-player and three-player, respectively. The simulation results are shown in Fig. 5.

Fig. 5 (a) shows the results of the two-player game. It can



(a)



(b)

Fig. 5. The expected total reward estimations of the multiple-player game under the multiagent Q-learning algorithm. (a) In the two-player game, the users' performances improve gradually and the total rewards converge to the value around 145. (b) In the three-player game, the expected total rewards increase to the values over 132. In the figure, the dash line denotes the expected total rewards of the fixed channel access policies.

be observed that the multiagent Q-learning algorithm improves the performances of the players. With the learned channel access policies, both of the users will get an expected total rewards around 145, which is higher than the expected total rewards of selecting channel 3 and channel 6 statically. In the three-player game, if the three players access the idlest three channels, which are channel 6, channel 3, and channel 1, they will obtain the expected total rewards around 132, 127, and 116, respectively. From Fig. 5 (b), we can see that the performances of all players get improved through the training, and the expected total rewards of any player is higher than the highest value of the fixed channel access policy.

Comparing simulation results of the three games, we can find that the expected total rewards of the multiple-player games are lower than that of the one-player game, and the total rewards decrease with the player population increasing. Obviously, this decreasing is due to the collision among the secondary users. However, in all the three games, the trained dynamic channel access policies perform better than the fixed channel access policies.

V. Conclusions and Future Works

This paper studies the channel access problem of the cognitive radio network. The channel access problem is formulated as a non-cooperative game of multiple players. The players can detect the spectrum opportunities and select a channel to transmit data. During the game, the players take actions with a certain interval and negotiations are not allowed. Considering the transmission delay, each player can only switch to the channels which are close to its current selection. To improve the performance of the players, we design a multiagent Q-learning algorithm in which each player deems its peers as a part of the environment. Simulation results demonstrate the effectiveness of the algorithm.

In this paper, we use Q tables to implement the proposed algorithm, which work effectively in our numerical experiments. However, the size of the Q table increases exponentially with the secondary users' quantity. In the future, we will employ artificial neural network as the approximator of the Q-table [30] to solve this problem.

Acknowledgment

This work was supported by National Science Foundation under grant ECCS 1731672 and ECCS 1811497.

References

[1] Y.-C. Liang, K.-C. Chen, G. Y. Li, and P. Mahonen, "Cognitive radio networking and communications: An overview," *IEEE transactions on vehicular technology*, vol. 60, no. 7, pp. 3386–3407, 2011.

[2] C. Cordeiro, K. Challapali *et al.*, "Spectrum agile radios: utilization and sensing architectures," in *New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on*. IEEE, 2005, pp. 160–169.

[3] J. Mitola, "Cognitive radio—an integrated agent architecture for software defined radio," 2000.

[4] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE communications surveys & tutorials*, vol. 11, no. 1, pp. 116–130, 2009.

[5] J. E. Suris, L. A. DaSilva, Z. Han, and A. B. MacKenzie, "Cooperative game theory for distributed spectrum sharing," in *Communications, 2007. ICC'07. IEEE International Conference on*. Ieee, 2007, pp. 5282–5287.

[6] C.-G. Yang, J.-D. Li, and Z. Tian, "Optimal power control for cognitive radio networks under coupled interference constraints: A cooperative game-theoretic perspective," *IEEE transactions on vehicular technology*, vol. 59, no. 4, pp. 1696–1706, 2010.

[7] Z. Han, Z. Ji, and K. R. Liu, "Fair multiuser channel allocation for ofdma networks using nash bargaining solutions and coalitions," *IEEE Transactions on Communications*, vol. 53, no. 8, pp. 1366–1376, 2005.

[8] D. Niyato and E. Hossain, "Competitive spectrum sharing in cognitive radio networks: a dynamic game approach," *IEEE Transactions on wireless communications*, vol. 7, no. 7, 2008.

[9] B. Wang, Y. Wu, and K. R. Liu, "Game theory for cognitive radio networks: An overview," *Computer networks*, vol. 54, no. 14, pp. 2537–2561, 2010.

[10] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[11] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, "Optimal and autonomous control using reinforcement learning: A survey," *IEEE Transactions on Neural Networks and Learning Systems*, 2017.

[12] Z. Ni, H. He, D. Zhao, X. Xu, and D. V. Prokhorov, "Grdhp: A general utility function representation for dual heuristic dynamic programming," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 3, pp. 614–627, 2015.
[13] A. Weissensteiner, "A q-learning approach to derive optimal consumption and investment strategies," *IEEE transactions on neural networks*, vol. 20, no. 8, pp. 1234–1243, 2009.
[14] J. Yan, H. He, X. Zhong, and Y. Tang, "Q-learning-based vulnerability analysis of smart grid against sequential topology attacks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 200–210, 2017.
[15] H. He and J. Yan, "Cyber-physical attacks and defences in the smart grid: a survey," *IET Cyber-Physical Systems: Theory & Applications*, vol. 1, no. 1, pp. 13–27, 2016.
[16] C. Yu, M. Zhang, F. Ren, and G. Tan, "Emotional multiagent reinforcement learning in spatial social dilemmas," *IEEE transactions on neural networks and learning systems*, vol. 26, no. 12, pp. 3083–3096, 2015.
[17] V. Vassiliades, A. Cleanthous, and C. Christodoulou, "Multiagent reinforcement learning: Spiking and nonspiking agents in the iterated prisoner's dilemma," *IEEE transactions on neural networks*, vol. 22, no. 4, pp. 639–653, 2011.
[18] H. Li, "Multi-agent q-learning of channel selection in multi-user cognitive radio systems: A two by two case," in *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on*. IEEE, 2009, pp. 1893–1898.
[19] A. Anandkumar, N. Michael, and A. Tang, "Opportunistic spectrum access with multiple users: Learning under competition," in *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.
[20] H. Li, "Multi-agent q-learning for competitive spectrum access in cognitive radio systems," in *Networking Technologies for Software Defined Radio (SDR) Networks, 2010 Fifth IEEE Workshop on*. IEEE, 2010, pp. 1–6.
[21] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Networks*, vol. 9, no. 3, pp. 228–248, 2011.
[22] S. Bayhan and F. Alagoz, "Scheduling in centralized cognitive radio networks for energy efficiency," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 2, pp. 582–595, 2013.
[23] S. Geirhofer, L. Tong, and B. M. Sadler, "Cognitive medium access: constraining interference based on experimental models," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, 2008.
[24] W.-Y. Lee and I. F. Akyildiz, "Optimal spectrum sensing framework for cognitive radio networks," *IEEE Transactions on wireless communications*, vol. 7, no. 10, 2008.
[25] Y. Xu, J. Wang, Q. Wu, A. Anpalagan, and Y.-D. Yao, "Opportunistic spectrum access in unknown dynamic environment: A game-theoretic stochastic learning solution," *IEEE transactions on wireless communications*, vol. 11, no. 4, pp. 1380–1391, 2012.
[26] D. Niyato, E. Hossain, and Z. Han, "Dynamics of multiple-seller and multiple-buyer spectrum trading in cognitive radio networks: A game-theoretic modeling approach," *IEEE Transactions on Mobile Computing*, vol. 8, no. 8, pp. 1009–1022, 2009.
[27] P. Mertikopoulos and A. L. Moustakas, "Correlated anarchy in overlapping wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 7, 2008.
[28] M. Azarafrooz and R. Chandramouli, "Distributed learning in secondary spectrum sharing graphical game," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*. IEEE, 2011, pp. 1–5.
[29] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.
[30] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.