Multiagent Jamming-Resilient Control Channel Game for Cognitive Radio Ad Hoc Networks

Brandon F. Lo and Ian F. Akyildiz
Broadband Wireless Networking Laboratory, School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
Email: {brandon.lo,ian}@ece.gatech.edu

***Abstract*—Control channel jamming is a severe security problem in wireless networks. This results from the fact that the attackers can effectively launch the denial of service attacks by jamming the control channels. Traditional approaches to combating this problem such as channel hopping sequences may not be the secure solution against intelligent attackers because the reliability of control channels in cognitive radio ad hoc networks cannot be guaranteed. In this paper, we introduce a jamming-resilient control channel (JRCC) game to model the interactions among cognitive radio users and the attacker under the impact of primary user activity. We propose the JRCC algorithm that enables user cooperation to facilitate control channel allocations and adapts to primary user activity with variable learning rates using the Win-or-Learn-Fast principle for jamming-resilience in hostile environments. It is shown that the optimal strategies converge to a Nash equilibrium or the expected rewards of the strategies converge to that of a Nash equilibrium. The results also show that the JRCC algorithm effectively combats jamming under the impact of primary user activity and sensing errors. Moreover, the control channel allocation policy can be improved by enhancing transmission and sensing capabilities. The proposed algorithm is scalable and can be applied to multiple users.**

I. Introduction

Common control channel (CCC) in cognitive radio (CR) networks [8] is the spectrum resource specifically allocated for control message exchange among CR users to facilitate network operations. In CR ad hoc networks (CRAHNs) [1] where no centralized control entity such as base station (BS) exists, CR users cooperate with each other for all spectrum management functions such as cooperative spectrum sensing [2], and thus relying even more on CCC for message exchange and normal operations. As a result, the reliability of CCC allocation is essential in CRAHNs. However, when a dedicated CCC allocated out of the licensed bands is not feasible, CCC must be dynamically allocated in licensed bands. In this case, the in-band CCC will be interrupted by primary user (PU) activity and needs to be efficiently reallocated and recovered when the existing CCC is occupied by the PU [7].

Dynamic CCC allocations in licensed bands are further complicated by jamming attacks if security issues are considered. Jamming attacks are launched by malicious users to deliberately disrupt the communications of CR users, resulting in denial of service (DoS) in CR networks. Although jamming attacks can occur in any type of channels, data or control, it is reported in [5] that jamming the broadcast channel (BCCH) of the GSM system is several order of magnitude more effective than targeting at all channels. For this reason, any intelligent attacker may prefer control jamming attack than other jamming methods due to its effectiveness of resulting in DoS. Thus, as in any wireless networks, control channel jamming is a severe security issue in CRAHNs.

This work was supported by the U.S. National Science Foundation (NSF) under Grant No. ECCS-0900930.

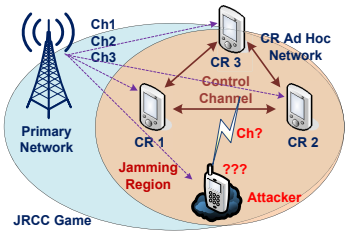


Fig. 1. Jamming-resilient control channel game.

The interactions between CR users and attackers are commonly modeled as a stochastic zero-sum game [6], [10], [11] since CR users and jammers generally have opposite goals. In these approaches, PU activities govern states of the game and state transitions, and sensing errors are generally ignored for simplicity. In [6], the Nash equilibrium strategy is obtained for the one-stage game, while the optimal attacking strategy is obtained for the multi-stage case. The latter is achieved by fixing CR user's strategy and converting the problem to the framework of the single-player partially observable Markov decision process (POMDP). [11] shows that CR users can combat jamming by increasing the number of unoccupied channels that can be observed. However, this capability is limited by PU activity and channel availability. In [10], minimax-Q learning is used by CR user to find the optimal anti-jamming channel selection policy. Although the CR user's actions consist of separate selections of control and data channels, the attacker in this work, like the one in [6] and [11], does not exclusively target at jamming control channels.

In this paper, we model the interactions among CR users, and the attacker under the impact of PU activities as a stochastic general-sum game, called jamming-resilient control channel (JRCC) game. Fig. 1 illustrates the JRCC game with the PUs, three CR users, and the attacker. The objective of the game is to find the optimal control channel allocation strategy for CR users to combat jamming attacks by using multiagent reinforcement learning (MARL). The optimal control channel allocation policy is obtained by enabling the communications among CR users to facilitate CCC allocations and the adaptation to PU activity to achieve the Nash equilibrium in the game. We demonstrate that the effectiveness of anti-jamming CCC allocations can be improved by the cooperation of CR users. By exploiting the advantages of Policy Hill-Climbing

(PHC) and the Win-or-Learn-Fast (WoLF) principle [4], our proposed JRCC algorithm effectively combats jamming under PU activity and sensing errors, and outperforms the original MARL algorithms. Our contribution can be summarized as follows:

- We model the interactions among CR users and the attacker under the impact of PU activities as a stochastic general-sum game called JRCC game with the consideration of sensing errors and limited observations of other players actions and payoffs.
- We analyze the gradient dynamics of the JRCC game by using the N-dimensional nonlinear dynamical system with the gradient ascent algorithm and show the convergence of the JRCC game.
- We propose the JRCC algorithm for CR users as optimal control channel strategy that utilizes CR user cooperation with the hill-climbing algorithm in low PU activity and exhibits resilience to jamming using variable learning rates in high PU activity.

The remainder of this paper is organized as follows: Section II discusses the system model and assumptions. Section III describes the dynamics and the proposed algorithm of the JRCC game. Section IV evaluates the performance by various test scenarios, and Section V concludes the paper.

II. System Model

The system model consists of a primary network model, a CRAHN model, a jamming attack model whose interactions are described by the JRCC game.

Primary Network Model: The primary network P consists of N_p PUs who may be active or inactive on a set of N_p licensed channels, $\mathcal{N}_p$, available for opportunistic access by CR users. Each licensed channel $i \in \mathcal{N}_p$ is occupied by one PU, P_i, whose activity follows the two-state birth-death process with the birth rate r_{b_i} and the death rate r_{d_i}. The departures and the arrivals of a PU on channel i follow a Poisson process with exponentially distributed inter-arrival time. Thus, each channel i has two states, PU active (ON) state and PU inactive (OFF) state, with transition probabilities: r_{b_i} (OFF to ON) and r_{d_i} (ON to OFF). We also assume that PU transmission is time-slotted. As a result, CR users need to periodically sense licensed channels according to the schedule of the primary network. Since the sensing operations of CR users are subject to errors, CR users need to satisfy the detection requirements in terms of probability of false alarm P_f and probability of miss detection P_m to limit the interference with PUs under a tolerable level. We also assume that the attacker needs to meet the detection requirements.

CR Ad Hoc Network Model: A group of K CR users, $\mathcal{K}$, within the jamming region of the attacker opportunistically access N_p licensed channels. Due to hardware limitations, CR users can only sense or transmit on $N_s \leq N_p$ licensed channels each time. Depending on the sensing results and channel availability, CR user $k \in \mathcal{K}$ selects a subset of channels, $\mathcal{N}_k \subseteq \mathcal{N}_p$ and $N_k = |\mathcal{N}_k| \leq N_s$, as control channels and transmits the same control messages on those selected channels. However, not all N_k channels are valid CCCs. Due to sensing errors and jamming attacks, these selected control channels may not be valid allocations for successful control transmission. In addition, a CCC must be commonly available to all CR users in the region. Thus, valid CCC allocations exist only when the selected channels are unoccupied by a PU, jamming-free, and common to CR users such that CR users can successfully exchange control messages on these channels. That is, the number of valid CCCs is $U_c = |\mathcal{U}_c| = N_c - J_c - P_c$ where N_c, J_c and P_c are the numbers of selected CCCs, jammed CCCs, and interfering CCCs due to miss detection, respectively, and $N_c = |\mathcal{N}_k \cap \mathcal{N}_l|$, $k, l \in \mathcal{K}$, $k \neq l$. We assume that all control messages are encrypted and are unable to be decrypted by the eavesdropping attacker during the period of the game. After rendezvous on these CCCs, the CR user pair can use the in-band CCCs for transmitting data or negotiating an available channel for data transmission.

Jamming Attack Model: For jamming attacks, we assume that the attacker has similar hardware capability as CR users do and can sense and jam up to $N_s \leq N_p$ licensed channels each time. According to the sensing results, the attacker selects N_j channels to jam and transmits the interference signal on those selected channels. Due to sensing errors, the attacker may select the PU-occupied channels to jam and cause the interference with PUs. Since the objective of the attacker is to disrupt CR transmission, we assume that the attacker will make efforts to avoid interfering with PUs to save its energy and avoid being exposed to PUs unless it is caused by the sensing hardware limitations. Thus, the attacker appears to PUs as a CR user. Moreover, we assume that the attacker does not behave like a PU by occupying the channels and forcing CR users to use other channels because this does not successfully jam control channels. We also assume that the attacker is unable to detect the control traffic and launch the jamming attack after the CCCs are established since such attacks require knowledge about CR users and the in-band CCCs are also used for data transmission. For these reasons, we do not consider other types of security attacks such as PU emulation attacks and node capture attacks (Byzantine failures) in our model. Assume that the attacker selects a subset of channels, $\mathcal{N}_j \subseteq \mathcal{N}_p$ and $N_j = |\mathcal{N}_j| \leq N_s$ for jamming. The number of valid jammed control channels is then $J_c = |\mathcal{J}_c| = N_j - U_j - P_j$ where U_j and P_j are the number of jammed non-CCC channels and PU-occupied channels caused by miss detection, respectively. For effective control channel jamming, $\mathcal{J}_c = \mathcal{U}_c \neq \emptyset$.

III. Jamming-Resilient Control Channel Game

In this section, we introduce the JRCC game that models the interactions among PUs, CR users, and the attacker. We analyze the game by using the gradient dynamics and then introduce the JRCC algorithm for finding the optimal control channel allocation strategy for CR users.

A. States, Actions, Transition probabilities, and Rewards

In the JRCC game, the primary network P affects the states of the game with PU activity on a set of N_p licensed channels. For a set of N_p licensed channels, there are 2^{N_p} states in the game. The state of the game at stage index n is denoted by $\mathbf{s}^n = \{s_1^n, \ldots, s_{N_p}^n\}$ where s_i^n is the state of channel i at stage index n. The state of channel s_i^n is determined by PU P_i's activity. That is, $s_i^n = 1$ if P_i occupies channel i at stage n, and $s_i^n = 0$ otherwise.

The sets of actions are denoted by $\mathcal{A}_k, k = 0, \ldots, K$ for the attacker and K CR users, respectively. The number of actions available to each player depends on the maximum number of channels that can be sensed. For sensing up to N_s channels, the number of actions is $N_{A_k} = \sum_{i=1}^{N_s} \binom{N_s}{i}$. If PU activity and jamming are not considered and all actions are equally likely, the probability of selecting m CCCs is given by

$$\Pr\{N_c = m\} = \frac{\sum_{i=m}^{N_s} \binom{N_p}{i}\binom{i}{m}\left[\sum_{j=0}^{N_{lim}} \binom{N_p - i}{j}\right]^{K-1}}{\left[\sum_{i=1}^{N_s} \binom{N_p}{i}\right]^{K}} \quad (1)$$

where $N_{lim} = \min(N_p - i, N_s - m)$ is the limitation on other CR user's remaining channel selections. The denominator is the number of all joint action combinations among K CR users. To find the probability of m selected CCCs, each CR user needs to select at least m channels. The first binomial coefficient $\binom{N_p}{i}$ in the numerator is the number of choices of one CR user selecting i out of total N_p channels. The second binomial coefficient $\binom{i}{m}$ says that which m out of the selected i channels are common to all CR users. The bracket in the numerator is the number of other CR user's choices of selecting non-CCC channels from the remaining $N_p - i$ channels not selected by the first CR user. For the attacker, the probability of selecting m channels to jam is given by $\Pr\{N_j = m\} = \binom{N_p}{m} / \left[\sum_{i=1}^{N_s} \binom{N_p}{i}\right]$. The probability of at least one successful CCC allocation is then

$$\Pr\{U_c > 0\} = \sum_{m=1}^{N_s} \Pr\{N_c = m\} \Pr\{J_c \leq m-1 | N_c = m\} \quad (2)$$

where

$$\Pr\{J_c \leq m-1 | N_c = m\} = \frac{\sum_{i=0}^{m-1} \binom{m}{i}\left[\sum_{j=0}^{N_s - i} \binom{N_p - m}{j}\right]}{\sum_{i=1}^{N_s} \binom{N_p}{i}} \quad (3)$$

The numerator in (3) is the combinations of the attacker jamming i out of m up to $m-1$ CCCs plus other $N_s - i$ non-CCC channels selected from the remaining $N_p - m$ channels.

Since the state transitions are governed by PU activity and all channels are independent, the state transition probability is given by $\Pr\{\mathbf{S}^{n+1} \mid \mathbf{S}^n\} = \prod_{i=1}^{N_p} \Pr\{s_i^{n+1} = j \mid s_i^n = k\}$, $j, k \in \{0, 1\}$ where $\Pr\{s_i^{n+1} \mid s_i^n\}$ is the probability of state transitions from state s_i^n to s_i^{n+1} on channel i depending on the PU ON/OFF status of the given state.

CR users are rewarded for the selections of un-jammed and PU-free CCCs. Thus, CR user k's immediate reward for stage n is defined as:

$$r_k^n = \begin{cases} 1/(N_c - J_c - P_c) & \text{if } U_c = N_c - J_c - P_c \geq 0, \\ 0 & \text{if } U_c = 0 \text{ or } N_k = J_c. \end{cases} \quad (4)$$

The maximum CR user's reward is unity when the selected channels are all PU-free CCCs and only one of them is not jammed. That is, $N_c - P_c = N_k$ and $N_k > J_c$. The reward of the attacker is evaluated based on whether the CCCs of CR users are all jammed. As a result, the attacker J's immediate reward for stage n is

$$r_j^n = \begin{cases} 1/(1 + (N_j - J_c)) & \text{if } U_c = 0 \text{ and } N_j > 0, \\ 0 & \text{if } U_c > 0 \text{ or } N_j = 0. \end{cases} \quad (5)$$

Although CR users and the attacker generally have the opposite goal, it can be seen from (4) and (5) that, unlike the zero-sum game, the reward of the attacker is not the negative of that of CR users in the JRCC game.

B. Gradient Dynamics Analysis

In the JRCC game, the interactions among all players can be modeled as an N-dimensional non-linear dynamical system in which the dynamics of changes are the gradient of the joint strategy in $\mathbb{R}^N$. Similar to [4], [9], we examine the dynamics of the JRCC game using the gradient ascent and show that the players' strategies or expected payoffs will converge. We focus on the dynamics of an N-player JRCC game with K CR users and one attacker ($N = K + 1$). We assume perfect sensing and full observations of PU states.

In this game, player $k \in \{0, \ldots, K\}$ chooses action $a_{k,i} \in \mathcal{A}_k$, $i = 1, \ldots, N_{A_k}$, indicating that player k selects the i-th subset of PU-free channels for CCC allocation ($k > 0$) or jamming ($k = 0$). Let $x_k = \{x_{k,i} \in [0, 1] : \sum_{i=1}^{N_{A_k}} x_{k,i} = 1\}$ be player k's action selection strategy. According to the strategy, the probability of choosing action $a_{k,i}$ is $x_{k,i}$. In each stage, player k receives reward $r_{k,j}$ for the j-th joint action $(a_0, \ldots, a_K)_j$ selected by the joint strategy $(x_0, \ldots, x_K)$. Then the expected reward R_k can be expressed as the function of the joint strategy $(x_0, \ldots, x_K)$ and rewards $r_{k,j}$, $j = 1, \ldots, \prod_{k=0}^{K} N_{A_k}$.

Since the goal of each player is to find the optimal strategy to maximize their expected rewards, the gradient ascent algorithm provides the mechanism for a player to achieve the optimal solution by iteratively adjusting its strategy with a sufficiently small step size. In the gradient ascent using variable learning rates [4], the changes in expected rewards can be expressed as iterative strategy update rules as follows:

$$x_k^{n+1} = x_k^n + \alpha^n \delta_k^n \frac{\partial R_k(x_0^n, \ldots, x_K^n)}{\partial x_k^n}, k = 0, \ldots, K \quad (6)$$

where $\delta_k^n > 0$ are the learning rates and $\alpha^n \delta_k^n$ are the step sizes for updating strategy x_k^n in stage n. $\partial R_k / \partial x_k^n$ represent the changes in player k's expected reward in response to the changes in the strategy x_k in the direction of the gradient. They are obtained by taking the partial derivatives of each player's expected reward with respect to its strategy. As a result, the dynamics of the strategy changes can be formulated as an N-dimensional constrained non-linear affine dynamical system with differential equations defined as

$$\dot{\mathbf{x}} = \mathbf{\Delta}(\mathbf{A}\mathbf{x} + \mathbf{b}(\mathbf{x}) + \mathbf{c}) \quad (7)$$

subject to the unit-hypercube constraints:

$$x_k \in [0, 1]^{N_{A_k}}, \quad k = 0, \ldots, K. \quad (8)$$

where $\mathbf{x} = [x_0 \ \ldots \ x_K]^T$, $\delta = [\delta_0 \ \ldots \ \delta_K]^T$, $\mathbf{\Delta} = \delta^T \mathbf{I}_N$, $\mathbf{A}_{N \times N}$ and $\mathbf{c}_{N \times N_{A_k}}$ are matrices whose elements are the functions of rewards $r_{k,j}$, and $\mathbf{b}(\mathbf{x})_{N \times N_{A_k}}$ contains higher-order products of $x_0, \ldots, x_K$. The constraints limit the strategies inside the unit hypercube because the strategy N-tuple are probability distributions.

The system can be linearized at a fixed point $\mathbf{x}^*$ if it has a solution $\mathbf{x}^*$ [3]. If we let $r = \|\mathbf{x} - \mathbf{x}^*\|_2$, $\mathbf{b}(\mathbf{x})/r$ approach $\mathbf{0}$ faster than r as $r \to 0$. Combined with the change of variable $\mathbf{y} = \mathbf{x} - \mathbf{x}^*$, we obtain the homogeneous linear system:

$$\dot{\mathbf{y}} = \mathbf{\Delta}\mathbf{J}\mathbf{y} \quad (9)$$

where $\mathbf{J} = \mathbf{J}_F \mid_{(x_0^*, \ldots, x_K^*)}$ and $\mathbf{J}_F$ is the Jacobian matrix of $\mathbf{X}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}(\mathbf{x}) + \mathbf{c}$. The phase portraits of the non-linear

system and its linearized system are considered qualitatively equivalent in the neighborhood of $\mathbf{x}^*$. Based on the analysis of gradient dynamics, we conclude with the following theorem.

Theorem 1 (Convergence Theorem of JRCC Game): For the N-player iterated general-sum JRCC game, if the players follow the gradient ascent algorithm with variable learning rates and a sufficiently small step size, the strategy N-tuple $(x_1, \ldots, x_N)$ will converge to a Nash equilibrium or the expected rewards of the players will converge to the expected rewards of a Nash equilibrium in the limit.

Proof: We examine the coefficient matrix $\mathbf{J}$ of the linear dynamical system (9) with the constraints (8), and show that the strategy will either converge to the fixed points of the system inside the unit hypercube or the expected rewards of the strategy will converge to that of a Nash point on the boundary of the hypercube. Since the variable learning rates in $\mathbf{\Delta}$ have no effect on the direction of the gradient, we focus on the eigenanalysis of $\mathbf{J}$ in the following two cases.

1) $\mathbf{J}$ *is singular:* In this case, the system is neutrally stable and the trajectories in the phase portrait exhibit periodic patterns and the strategy N-tuple are periodic functions of time. Since this periodicity in the strategy can be predictable and is not desired by either CR users or the attacker in the JRCC game, CR users and the attacker will enforce the system to stay away from neutrally stable states in order to make their strategies unpredictable.

2) $\mathbf{J}$ *is nonsingular:* In this case, $\mathbf{J}$ is invertible and all the eigenvalues of $\mathbf{J}$ have nonzero real part. The system has hyperbolic fixed points: the phase portraits of the nonlinear system and its linearization are qualitatively equivalent in the neighborhood of the fix points. Let n_u and n_s be the number of eigenvalues with positive or negative real part, respectively. These eigenvalues are associated with the corresponding *unstable* eigenspaces $V^u \in \mathbb{R}^{n_u}$ and *stable* eigenspaces $V^s \in \mathbb{R}^{n_s}$ of $e^{\mathbf{J}t}$, respectively. Trajectories in the phase portrait are moving away from the fixed point in V^u and approaching the fixed point in V^s as t increases. Since $n_u + n_s = N$, we have the following subcases: $n_u = 0, \ldots, N$. For $n_u = 0$, the fixed point is an attracting node and the strategy converges to this Nash point. For $n_u > 0$ and $n_u < N$, trajectories are saddle points pointing inwards with a focus in V^s and outwards along V^u. For $n_u = N$, the fixed point is an N-dimensional star node pointing outwards.

Due to the constraints (8), points on the trajectories away from the fixed point will initially reach a point on the boundary of the unit hypercube. Without loss of generality, we assume that the point is on one of the n-faces, $n \leq N$. If the projection of the gradient is zero at that point, the trajectory will stay on the point. It is a Nash point of the game since no single user can improve its payoff by changing the strategy unilaterally. If the projected gradient is nonzero, the trajectory moves toward one of the $(n-1)$-faces of the hypercube in the direction depending on the sign of the projected gradient and reaches a point on the $(n-1)$-faces. The process will stop at any point where the projected gradient is zero or continue to move toward lower dimensional faces until the trajectory reaches one of the vertices of the hypercube $(n = 1)$. Thus, $(x_1, \ldots, x_N)$ converges to a Nash equilibrium or its expected rewards converge to the expected rewards of a Nash point. ■

C. JRCC Algorithm

The gradient ascent algorithm in Section III-B requires the knowledge of rewards for all combinations of joint actions and the distributions of other players' actions available to each player. However, obtaining such knowledge in the JRCC game is infeasible. Due to the limitation of sensing capability, the actions of the players are only partially observable by other players. As a result, not all rewards can be obtained for all joint actions. More importantly, CR users and the attacker will not reveal their own action selection strategy. For these reasons, we propose the JRCC algorithm capable of selecting actions based on limited observations, updating strategy similar to gradient ascent, and obtaining the best response for each CR user individually.

The JRCC algorithm enables the cooperation between CR users with low control message overhead to facilitate CCC allocations, and adapts to PU and jamming activity by using the variable learning rates based on the win-or-learn-fast (WoLF) principle [4] in extremely hostile environment. When PU activity is low, the JRCC algorithm behaves like a rational hill-climbing algorithm that converges to a greedy strategy to maximize the payoffs. The performance is further improved by the cooperation and the exchange of a few parameters between CR users on the established CCCs since their strategies for CCC selections become similar. When PU activity is high, the available CCCs under jamming attacks are very limited, which makes the cooperation less effective. In this case, the WoLF principle can adjust the learning rates such that the players learn slowly to delay the strategy change of the opponent ("winning") or learn fast when they are outperformed by the opponent ("losing").

The JRCC algorithm is listed in Algorithm 1. In each stage, each CR user selects an action that maps to a set of selected channels as CCCs for transmission, and obtains its own reward by observing the conditions of selected CCCs. (lines 3-5). For cooperation, each CR user broadcasts the control message with the parameters recorded in previous stage, and updates its strategy with the parameters received from neighbors (lines 6-10). After the PU changes the state of the game, CR users observe the next state s' by sensing the channels, and update their Q values for current state s and action a_i (lines 11-12). By selecting the proper learning rate δ (lines 13-17), CR users update their own strategy (line 18). The value of δ is set to the maximum for greedy strategy and a variable value from the WoLF principle. The parameters $\tilde{s}$, $\tilde{a}_i$, and $\tilde{\delta}_i$ for the current greedy strategy are recorded for broadcast in the next stage (line 19). For PHC strategy updates, the probability of the best action is increased while the probabilities of other actions are evenly decreased (lines 22-30). For variable learning rates, the slow learning rate δ_w is selected for the "winning" case if the average Q value of the best action a' based on current policy π is larger than that based on average policy $\bar{\pi}$, and the fast learning rate δ_l is selected otherwise (lines 31-39).

IV. Performance Evaluation

In this section, we evaluate the performance of the proposed algorithm in the JRCC game. We show that both increasing the transmission capability of CR users and enabling the cooperation between CR users can improve the performance

Algorithm 1 : JRCC for CR User $i \in \mathcal{K}$

Initialize: α, γ, ϵ, $\delta_i \in (0,1]$, $Q(s,a) \leftarrow 0$, $\pi(s,a) \leftarrow \frac{1}{|\mathcal{A}_i|}$
for each stage n **do**
  Select $a_i \in \mathcal{A}_i$ in state s per $\pi(s)$ with w.p. $1-\epsilon$
  Transmit on channels: $\{Ch : a_i \mapsto \mathcal{N}_i\}$
  Observe U_c, J_c, P_c, P_j and calculate reward r_i
  if ($U_c > 0$ and $\exists \tilde{a}_i$) **then**
    BroadcastToNeighbors$(\tilde{s}, \tilde{a}_i, \tilde{\delta}_i)$
    ReceiveFromNeighbors$(\tilde{s}, \tilde{a}_m, \delta_m, m \in \mathcal{K}, m \neq i)$
    StrategyUpdate$(\pi(\tilde{s}, a), \tilde{a}_m, \tilde{\delta}_m)$
  end if
  Observe next state $s' \leftarrow$ *SensingChannels*$(\mathcal{N}_{s,i})$
  $Q(s,a_i) \leftarrow (1-\alpha)Q(s,a_i) + \alpha\left[r_i + \gamma \max_b Q(s',b)\right]$
  if $r_i \geq r_{th}$ **then**
    $\delta_i = \delta_{max}$
  else
    $\delta_i =$ *WoLF*$(C(s), \pi(s,a), \bar{\pi}(s,a), Q(s,a))$
  end if
  StrategyUpdate$(\pi(s,a), a' = \arg\max_b Q(s,b), \delta_i)$
  if ($U_c > 0$) **then** $\tilde{s} \leftarrow s, \tilde{a}_i \leftarrow a', \tilde{\delta}_i \leftarrow \delta_i$ **end if**
  UpdateParameters$(\alpha, \gamma, \delta_i)$, $s \leftarrow s'$
end for
procedure *StrategyUpdate*$(\pi(s,a), a', \delta)$
  $\delta_{sa} = \min\left(\pi(s,a), \frac{\delta}{|\mathcal{A}_i|-1}\right)$
  if $a \neq a'$ **then**
    $\Delta_{sa} = -\delta_{sa}$
  else
    $\Delta_{sa} = \sum_{a' \neq a} \delta_{sa'}$
  end if
  $\pi(s,a) \leftarrow \pi(s,a) + \Delta_{sa}$
end procedure
procedure *WoLF*$(C(s), \pi(s,a), \bar{\pi}(s,a), Q(s,a))$
  $C(s) \leftarrow C(s) + 1$
  $\bar{\pi}(s,a) \leftarrow \bar{\pi}(s,a) + \frac{1}{C(s)}\left(\pi(s,a) - \bar{\pi}(s,a)\right), \forall a' \in |\mathcal{A}_i|$
  if $\sum_{a'} \pi(s,a')Q(s,a') > \sum_{a'} \bar{\pi}(s,a')Q(s,a')$ **then**
    $\delta_i = \delta_{w_i}$
  else
    $\delta_i = \delta_{l_i}$
  end if
end procedure

of combating the attacker. We also show that the JRCC algorithm effectively combats jamming under the impact of PU activities and sensing errors. In the test scenarios, JRCC is compared to PHC and WoLF-PHC algorithms [4]. PHC is a greedy algorithm that improves the policy by selecting actions according to maximum Q values. WoLF-PHC is based on PHC with variable learning rates determined by the WoLF principle. In the simulation environment, we set $N = 3$, $N_p = 6$, and $N_s = 3$. For reinforcement learning parameters, we set $\alpha^n = 1/(1+n/500)$, $\delta_w^n = 1/(1+n/10)$ where n is step/stage index, $\delta_l = 4\delta_w$, $\gamma = 0.9$, $\epsilon = 0.1$, $\delta_{max} = 0.9$, and $r_{th} = 0.5$ unless otherwise specified.

A. Convergence of JRCC Game

Fig. 2 plots the expected rewards of CR users and the attacker for 10 exemplary runs when PUs are not present. The group on the top is CR users' rewards while the bottom group is the attacker's. The figure clearly shows the convergence of JRCC game for CR users and the attacker. In this case, the convergence is faster than the runs with state changes. However, the expected rewards from the runs with state changes exhibit similar convergence behavior. This shows that

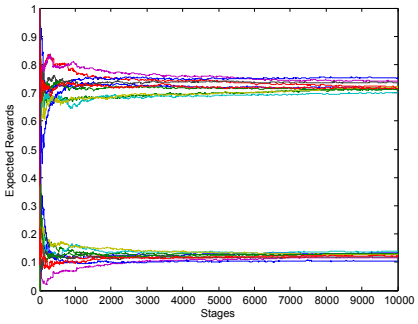


Fig. 2. Convergence of the JRCC algorithm in JRCC game.

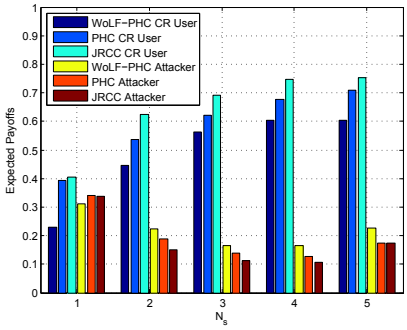


Fig. 3. Expected rewards versus transmission capability N_s.

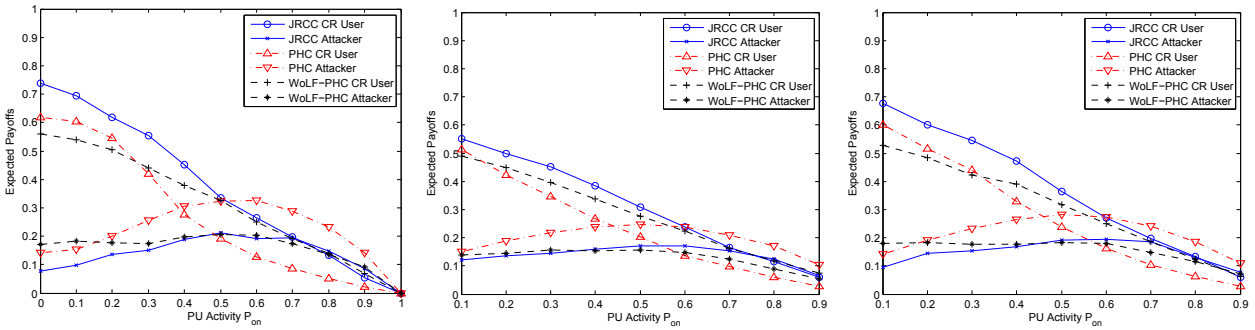
the strategy of the players converges to a Nash equilibrium or the rewards converge to the reward of a Nash point.

B. Transmission Capability

Owing to power constraints or hardware limitation, CR users and the attacker are limited to transmitting on a maximum number of channels $N_s \leq N_p$ simultaneously. To save energy, CR users may select a smaller number of channels $N_k \leq N_s$ as control channels at the higher risk of being all jammed by the attacker. Similarly, the attacker may select $N_j \leq N_s$ channels for jamming with potential loss of jamming performance. Hence, transmission capability has the effect on the performance of CCC allocation or jamming strategy of the players. For fairness, we assume that CR users and the attacker have the same transmission capability. Fig. 3 shows the expected payoffs of PHC, WoLF-PHC, and JRCC algorithms for different number of N_s given no PU activity and $N_p = 6$. As N_s increases, the expected payoffs of JRCC CR users increase monotonically. The performance gain of JRCC over PHC is mainly obtained from the cooperation of CR users. The attacker's payoffs drop as N_s increases from 1 to 3 and slightly increase as N_s increases to 5. Note that the increases in CR users' payoffs are monotonically decreasing as N_s varies from 1 to 5. This is because the attacker's transmission capability is also increased. This shows that transmitting on all channels is not necessarily the best strategy for CR users if the attacker has the same capability.

C. Impact of the PU Activity

PU activity is one of the major impacting factors of JRCC performance since the available channels for CCC allocations may be significantly reduced. Fig. 4(a) shows the expected payoffs of JRCC, PHC, and WoLF-PHC versus the probability of an active PU, P_{on}, in each channel. We assume that P_{on} is the same for all PUs and both CR users and the attacker



(a) (b) (c)

Fig. 4. Expected payoffs vs. PU activity (P_{on}) with (a) perfect sensing, (b) false alarm $P_f = 0.1$, and (c) miss detection $P_m = 0.1$.

perform perfect sensing with no sensing errors. Since there is a PU in each channel, the case of $P_{on} = 0.5$ is approximately equivalent to the case that half of the channels are occupied by PUs on the average. Hence, the expected payoffs of CR users are reduced considerably while the payoffs of the attacker increase to the maximum from no PU activity to $P_{on} = 0.5$. For high PU activity $P_{on} > 0.6$ where CCCs are less available for jamming, the payoffs of the attacker also decrease and approach zero as PU becomes mostly active on all channels. PU activity has the greatest impact on PHC and the least on WoLF-PHC in terms of decreasing rate of the expected payoffs. The proposed JRCC maintains the highest payoffs under low to medium PU activity due to CR user cooperation. For medium to high PU activity where CCCs are less available for cooperation, JRCC adopts the variable rates to combat jamming as WoLF-PHC. Hence, the performance of JRCC is comparable to that of WoLF-PHC in high PU activity cases. This scenario shows that JRCC adapts to PU activity by combining CR user cooperation and variable learning rates to maximize the payoffs for jamming-resilient CCC allocations.

D. Effects of Sensing Errors

In addition to PU activity, sensing errors such as false alarm and miss detection can have the major impacts on the JRCC performance. In the false alarm cases, CR users are mistakenly forced to allocate CCCs in the smaller subset of available channels. This increases the probability of two CR users selecting exclusive subsets of channels as CCCs. Hence, the effect of false alarms on CCC allocations can be significant even if only one CR user experiences the false alarm. Moreover, CR users may observe different states due to false alarms and thus making the cooperation less effective. As a result, false alarms, on top of existing PU activity, further reduce channel availability for CCC allocations. Fig. 4(b) shows the expected rewards versus PU activity with $P_f = 0.1$ for CR users and the attacker. As expected, CR users are greatly affected by false alarms. The cooperative gain in JRCC is also reduced compared to the perfect sensing scenario. JRCC still performs the best in low to medium PU activity cases and approaches WoLF-PHC when PU activity is high. Similarly, the attacker's performance is affected by false alarms with maximum payoffs in medium PU activity. Unlike false alarms, the effect of miss detection on CCCs requires both CR users incorrectly detecting the presence of the PU. Hence, the probability of both CR users having miss detection is much smaller and the impacts on CR users are less noticeable. Fig. 4(c) shows the expected rewards versus PU activity with $P_m = 0.1$. Compared to Figs. 4(a) and 4(b), the performance of CR users and the attacker is slightly affected.

V. Conclusions

In this paper, we tackle the control channel jamming problem in CRAHNs by modeling the interactions among CR users and the attacker under the impact of PU activities as a stochastic general-sum game called JRCC game. We analyze the gradient ascent dynamics of the game and show its convergence. We also propose the JRCC algorithm for optimal CCC allocation strategy by enabling CR user cooperation and adapting to PU activity with variable learning rates. The results demonstrate that the JRCC algorithm effectively combats jamming under the impact of primary user activity and sensing errors. The CCC allocation policy can be improved by enhancing transmission and sensing capabilities. The proposed algorithm is scalable and can be applied to multiple CR users.

References

[1] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.

[2] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication*, vol. 4, no. 1, pp. 40–62, Mar. 2011.

[3] D. K. Arrowsmith and C. M. Place, *Dynamical Systems*. London, UK: Chapman & Hall, 1992.

[4] M. Bowling and M. Veloso, "Multiagent learning using a variable learning rate," *Artificial Intelligence*, vol. 136, pp. 215–250, 2002.

[5] A. Chan, X. Liu, G. Noubir, and B. Thapa, "Broadcast control channel jamming: Resilience and identification of traitors," in *Proc. IEEE ISIT*, Jun. 2007, pp. 2496–2500.

[6] H. Li and Z. Han, "Dogfight in spectrum: Jamming and anti-jamming in multichannel cognitive radio systems," in *Proc. IEEE GLOBECOM*, Dec. 2009, pp. 1–6.

[7] B. F. Lo, I. F. Akyildiz, and A. M. Al-Dhelaan, "Efficient recovery control channel design in cognitive radio ad hoc networks," *IEEE Trans. Vehicular Technology*, vol. 59, no. 9, pp. 4513–4526, Nov. 2010.

[8] B. F. Lo, "A survey on common control channel design for cognitive radio networks," *Physical Communication*, vol. 4, no. 1, pp. 26–39, Mar. 2011.

[9] S. Singh, M. Kearns, and Y. Mansour, "Nash convergence of gradient dynamics in general-sum games," in *Proc. 16th Conf. Uncertainty in Artificial Intelligence*, 2000, pp. 541–548.

[10] B. Wang, Y. Wu, K. Liu, and T. Clancy, "An anti-jamming stochastic game for cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 877–889, Apr. 2011.

[11] Q. Zhu, H. Li, Z. Han, and T. Basandar, "A stochastic game model for jamming in multi-channel cognitive radio systems," in *Proc. IEEE ICC*, May 2010, pp. 1–6.