Cooperation Scheme For Distributed Spectrum Sensing In Cognitive Radio Networks

Ying Dai and Jie Wu

Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122

Abstract

Spectrum sensing is an essential phase in cognitive radio networks (CRNs). It enables secondary users (SUs) to access licensed spectrum, which is temporarily not occupied by the primary users (PUs). The widely used scheme of spectrum sensing is cooperative sensing, in which an SU shares its sensing results with other SUs to improve the overall sensing performance, while maximizing its throughput. For a single SU, if its sensing results are shared early, it would have more time for data transmission, which improves the throughput. However, when multiple SUs send their sensing results early, they are more likely to send out their sensing results simultaneously over the same signaling channel. Under these conditions, conflicts would likely happen. Then, both the sensing performance and throughput would be affected. Therefore, it is important to take when-to-share into account. We model the spectrum sensing as an evolutionary game. Different from previous works, the strategy set for each player in our game model contains not only whether to share its sensing results, but also when to share. The payoff for each player is defined based on the throughput, which considers the influence of the time spent both on sensing and sharing. We prove the existence of the evolutionarily stable strategy (ESS). In addition, we propose a practical algorithm for each secondary user to converge to the ESS. We conduct experiments on our testbed consisting of 4 USRP N200s. The experimental results verify for our model, including the convergence to the ESS.

1. Introduction

Cognitive radio networks (CRNs) [1] enable *secondary users* (SUs) to utilize the licensed spectrum when *primary users* (PUs) are not using it. Spectrum sensing is the key phase in identifying the spectrum availability. The fundamental task of spectrum sensing is that: when PUs are using the licensed spectrum, each SU should be able to detect it, and should quit transmitting on the corresponding spectrum band. When PUs are not using the licensed spectrum, each SU should be able to identify the corresponding spectrum band as available. The objectives of SUs are to maximize the utilization of the available spectrum and to prevent interference with PUs.

The spectrum sensing[2] performance is usually measured by two metrics: probability of detection, which denotes the probability of a SU detecting a PU when the spectrum is occupied by the PU, and probability of false alarm, which denotes the probability of a SU falsely declaring a PU as present, when it is actually not occupied by a PU. To ensure the spectrum sensing quality, adequate sample collection is required over a period of time for analysis by SUs.

The time spent by the SU on spectrum sensing, in turn, will reduce the time spent on data transmission. The efficiency and performance of spectrum sensing for all SUs can be improved through cooperative sensing, because each SU shares its sensing results with others, and decides whether a spectrum band is available to be accessed, based on multiple users' sensing results. The merits of cooperative spectrum sensing are illustrated in Fig. 1. In the example, there are three SUs, S_1, S_2, and S_3, and two active PUs. S_1 cannot detect the existence of the $PU\ Tx$ because of the path fading. S_3 cannot detect the $PU\ Rx$. If S_3 starts to transmit, it may cause interference on the $PU\ Rx$. However, if the three perform the cooperative spectrum sensing, and S_2 shares its sensing results with the others, the interference caused to PUs will be avoided.

Many works have been done on applying game theory to spectrum sensing, which determines the relative probability of an SU participating[3, 4, 6]. Since not all SUs are willing to contribute to cooperative sensing, the strategy set of each player is usually {contribute, not contribute}. To contribute indicates that the SU needs to share its sensing results with other
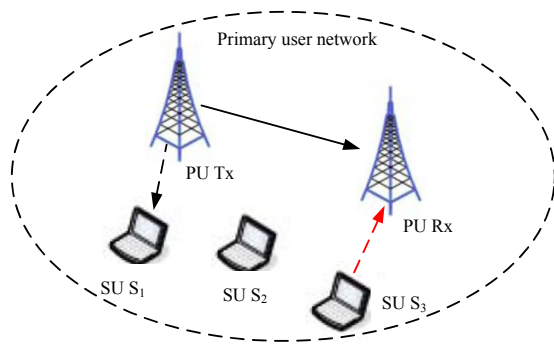
Figure 1. Merits of cooperative spectrum sensing.

SUs, or in other words, participate in the cooperative sensing. The cooperative sensing can ensure the sensing performance, and save the overall cost in spectrum sensing among SUs. However, not all SUs are willing to participate, because they will benefit from the free ride from others' sensing results. When more SUs choose not to contribute, the sensing performance will be affected. Many works have been done on the decision process of whether an SU is to contribute its sensing results or not.

However, aside from whether a SU is willing to share its sensing results or not, it is also important for each SU to decide when to share. Intuitively, SUs are willing to share their sensing results early, because this means that the time spent on this sharing phase is reduced, and more time can be used for data transmission. However, conflicts might occur because two or more SUs may send out their sensing results together, since the sensing results are usually sent through a common signaling channel. To ensure the sensing performance, the conflicted SU needs to back off for a certain amount of time, and resend its sensing results later. This would lead to an increase in the time spent on spectrum sensing, which in turn results in a decrease in the time spent on data transmission. In a distributed system, without a base station or central controller, each SU has to decide whether to share, as well as when to share, by itself. Moreover, if this is coordinated by the communication among different SUs, it would cause more overhead. So, it is better to have each SU decide when to share, itself, based on its own observation.

In this paper, we consider a CRN, in which the licensed spectrum band is divided into multiple subbands and a signaling band. Each SU is assigned a subband for data transmission, and the signalling band is for sharing sensing results. Each SU cannot only decide whether to contribute its sensing results by sharing with others, but also must decide when to send out its sensing results if it chooses to contribute. We model the process as an evolutionary game, in which each SU aims at maximizing its throughput, while assuring the sensing performance. We propose replicator dynamics for each SU, and prove

the existence of an *evolutionarily stable strategy* (ESS). We propose a distributed algorithm for each SU to evolutionarily reach the ESS.

To testify for our algorithm, we build a testbed consisting of four SUs, and one PU, using five USRP N200s. We generate a random active sequence for the PU. Each SU runs our algorithm to conduct spectrum sensing and data transmission. We collect data regarding the throughput and the spectrum sensing performance metrics to verify our model.

The main contributions of our paper are as follows:

- To the best of our knowledge, this is the first work that models the spectrum sensing as an evolutionary game, in which the strategy set of each SU (player) consists not of only whether to share its sensing results, but when to share its sensing results.

- We prove the existence of ESS in our game model, and propose a distributed algorithm for each SU to converge.

- We construct a testbed consisting of five USRP N200s, and have the SUs run our distributed algorithm. Our model is testified, in terms of the convergence to the ESS.

The remainder of our paper is organized as follows: In Section 2, related works are introduced. Section 3 describes our network model, and formulates the objective and constraints of our problem. We build the game model and propose the algorithm in Section 4. The experiment settings and results using USRP N200s regarding our model are introduced in Section 5. We discussion extensions in Section 6. We conclude our paper in Section 7.

2. Related Works

In this section, we introduce related works regarding cooperative spectrum sensing from two aspects. One is mainly the cooperative models applied on spectrum sensing. The other one is about the efforts of improving the cooperative sensing performance.

Many works are done on the cooperative model design of spectrum sensing [3–10]. [3] proposes an evolutionary game model for spectrum sensing. In the model, each user has a probability of performing sensing. The strategy set for each player is to contribute or not to contribute. The payoff of each player is defined based on the throughput, which considers the time spent on spectrum sensing. Our model is different from the model in [3] in two aspects. One is that our strategy set considers both whether to contribute, and also when to contribute. Another is that our payoff function considers the influence caused by conflicts, while sharing the sensing results. [4] studies
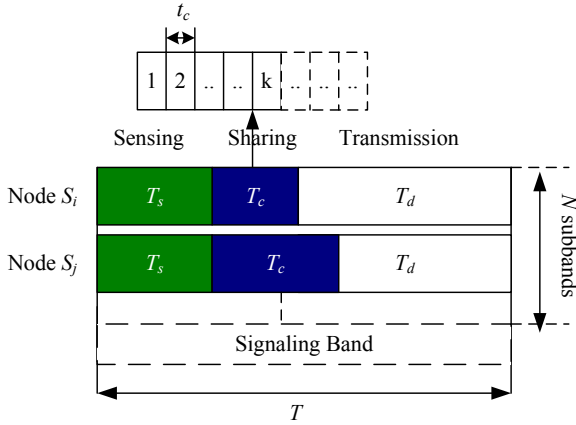
Figure 2. Example of subbands and time slot division.

spectrum sensing as a noncooperative game under the constraints of sensing performance and QoS. Their game model is decoupled into a lower-level uncoupled game, and a higher-level optimization problem. A distributed hierarchical iterative algorithm is proposed for their model. [6] has a base station for performing the scheduling among all SUs. They advocate SUs to contribute, by assigning higher priorities to be accessed. Other models, besides those using game theory, are also widely studied. [8] uses random matrix theory. Distributed rule-regulated cooperative sensing is proposed in [10]. [9] adopts cluster theory, and divides SUs into different clusters to perform cooperative sensing.

Much of the existing literature focuses on improving the spectrum sensing performance[11–15]. [11] introduces a spatial diversity technique to reduce the error probability between SUs and the data fusion center. The error in their model is mainly caused by the fading on the reporting channel. [12] aims at solving the problem when sensing samples are not sufficient for precisely detecting available channels. They apply matrix completion and joint sparsity recovery to improve the sensing performance. Cooperative compressed spectrum sensing is studied in [13]. They propose the belief propagation, based on compressed spectrum sensing for the statistical prediction of spectrum availability, and build a probabilistic graph model. A recent work in [15] has each SU equipped with multiple antennas. Each SU decides the availability of a spectrum band through combining the statistic results obtained by an improved energy detector. Our work focuses mainly on improving the sensing performance through choosing a better strategy during cooperative spectrum sensing.

3. System model & Problem Formulation

In this section, we first introduce our network model. Then, we formulate the requirements of the spectrum sensing performance, as well as the objective function for each SU.

3.1. Network Model

We consider a set of SUs, or nodes, $S = \{S_i\}$ in a CRN. Each node is assumed to know the total number of nodes N ($N = |S|$), and is able to reach another node within one hop. The privileged band is divided into N subbands. There is one signaling band. Each user is assigned one subband for data transmissions, and shares its sensing results on the signaling band, as shown in Fig. 2. We assume that each SU, or node, is equipped with two antennas. One antenna is used for spectrum sensing, sensing results sharing, and data transmission. The other antenna is used for listening to the signaling channel to overhear the sensing results shared by others, and sending back ACKs when the sensing results from others are received. The time-slotted system is used. During each time slot, the SU needs to first sense the PU's activity. Since the PU operates on the whole licensed spectrum band, its activity can be sensed by any SU. This means that each SU can choose to cooperate and share the sensing results over the signaling channel, as to ensure a high detection probability, and a low false alarm probability.

Each time slot T is divided into three parts: sensing phase T_s, sharing phase with a maximal length of T_c, and data transmission phase T_d, as shown in Fig. 2. The sensing phase is for each node to sense the channel independently. We assume that for each node, the time spent on independent sensing is static. The sharing phase is for each node to send its sensing results over the signaling channel. Suppose the minimal time required for sending the sensing results when there is no conflict is t_c. Then T_c is divided into $\lceil \frac{T_c}{t_c} \rceil$ sub slots. For a certain SU, it can choose whether to share its sensing results or not. If a node decides not to share its sensing results, its sharing time length would be 0. Also, if it chooses to cooperate with others, it needs to choose one sub slot of T_c to send the sensing results. The sensing results are confirmed to be received successfully through the ACKs. The sharing phase of a node ends as long as one ACK is received. Before that, the current SU keeps listening to the signaling channel for others' sensing results. The transmission phase is for data transmission. Therefore, as more time is spent on spectrum sharing, less time will be left for data transmission.

3.2. Objective & Constraints

The objective for each node in our model is to maximize the throughput of data transmission, while ensuring the spectrum sensing performance. Next, we will formulate the constraints regarding spectrum sensing performance, and the throughput for each node.

First, we use P_{H_0} to denote the probability that the PU does not occupy the licensed spectrum band, which means that SUs can access and transmit on their

subbands. Then $1 - P_{H_0}$ denotes the probability that the PU occupies the licensed channel, which means none of the subbands are available. We assume that each SU uses an energy detector for spectrum sensing. Suppose the PU activity is a random process with mean 0 and variance σ_s^2. Suppose the additive Gaussian white noise is a circularly symmetric complex Gaussian with mean 0 and variance σ_w^2. For a single SU S_i, the probability of detection $p_d(S_i)$ and false alarm $p_f(S_i)$ can be calculated[16]:

$$p_d(S_i) = \frac{1}{2} erfc((\frac{\lambda}{\sigma_w^2} - \lambda - 1)\sqrt{\frac{K}{2(2\lambda + 1)}}), \quad (1)$$

$$p_f(S_i) = \frac{1}{2} erfc(\sqrt{2\lambda + 1} etf^{-1}(1 - 2\bar{P}_D) + \sqrt{\frac{K}{2}}\lambda), \quad (2)$$

where $erfc()$ denotes the complementary error function; $erf^{-1}()$ denotes the inverse function of the error function; K is the number of collected samples; $\bar{P}_D$ is the given target detection probability; λ denotes the received signal-to-noise ratio (SNR) of a PU under H_1, which is equal to $h^2 \sigma_s^2 / \sigma_w^2$, and h is the gain of the channel from the PU's transmitter to the SU's receiver, which is assumed to be slow flat fading.

We assume that each SU overhears others' sensing results on the signaling channel. The channel availability is decided by both its own sensing results, and the others' collected sensing results. The data transmission starts after the sharing phase ends, and the current subband is identified as available. The fusion rule of deciding whether the channel is available or not can be different (AND, OR, Majority, etc.). Suppose we apply the OR rule here. Other fusion rules can be applied if necessary. For a SU S_i, its probability of detection and false alarm after the sensing results sharing phase would be:

$$P_d(S_i) = 1 - \prod_{k=1}^{N}(1 - p_d(S_i)A(S_i, S_k)), \quad (3)$$

$$P_f(S_i) = 1 - \prod_{k=1}^{N}(1 - p_f(S_i)A(S_i, S_k)), \quad (4)$$

where $A(S_i, S_k) = 1$ means that the sensing results of a SU S_k is received by a SU S_i before T_d starts; $A(S_i, S_k) = 0$ otherwise.

The expected throughput for a SU S_i under P_{H_0} is defined as [17]:

$$U(S_i) = P_{H_0}(1 - \frac{T_s + \delta(t_r)}{T})(1 - P_f(S_i))C_{H_0}(S_i) + (1 - P_{H_0})$$
$$(1 - \frac{T_s + \delta(t_r)}{T})(1 - P_d(S_i))C_{H_1}(S_i), \quad (5)$$

where $C_{H_0}(S_i)$ is the data rate of SU S_i under H_0, $C_{H_1}(S_i)$ is the data rate of SU S_i under H_1, and $\delta(t_r)$ denotes

the time used to send the sensing results. This is due to the fact that conflicts may occur when more than one SU sends the sensing results together. Then they would need to backoff and resend the sensing results later. We use $\delta(t_r)$ to represent the total time spent on this sensing results sharing, and obviously $0 \le \delta(t_r) \le T_c$. Since $C_{H_1}(S_i)$ is much smaller than $C_{H_0}(S_i)$, due to the interference from the PU, the second term can be omitted. Therefore, the expected payoff for S_i can be approximated as:

$$U(S_i) = P_{H_0}(1 - \frac{T_s + \delta(t_r)}{T})(1 - P_f(S_i))C_{H_0}(S_i). \quad (6)$$

The objective of each SU is to maximize its own $U(S_i)$, while satisfying the following constraints:

$$P_d(S_i) > \alpha \text{ and } P_f(S_i) < \beta, \quad (7)$$

where α and β are the required thresholds for $P_d(S_i)$ and $P_f(S_i)$.

4. Game Model

Game theory is widely used for analyzing the strategic interactions among multiple players[18, 19]. In this section, we first introduce some main concepts regarding the evolutionary game. Then we explain how to model our problem as an evolutionary game, and prove the existence of ESS. Finally, we provide the algorithm for each SU, so that it can decide its strategy.

4.1. Evolutionary Game

The key insight of evolutionary game theory is that many behaviors involve the interactions of multiple strategies of different players, and the success of any strategy depends on how it interacts with others. Therefore, the payoff of an individual strategy should be evaluated in the context of all players that it interacts with, rather than be measured in isolation. Similar to the NASH equilibrium in classic game theory, the analogous notion in evolutionary game theory is an ESS. We have the formal definition of an ESS, as follows [20]:

Definition 1. A strategy $q*$ is an ESS if and only if, for any strategy $q \ne q*$ and all $\theta > 0$,

$$U(q*, \theta q + (1 - \theta)q*) > U(q, \theta q + (1 - \theta)q*),$$

where $U(q*, \theta q + (1 - \theta)q*)$ denotes the payoff of a player who adopts $q*$, while the θ portion of the others adopt q, and the remaining portion adopt $q*$.

From the definition, we can see that the strategy is an ESS, which tends to persist once it is adopted by most players. Due to dynamics in the spectrum availability in CRNs, there is not a static stable strategy for each user conducting spectrum sensing. Therefore, we apply

the evolutionary game here to solve the problem. The strategy set of our game model is not only whether to contribute or not, but when to share the sensing results if the secondary user decides to contribute. If a node decides not to contribute but always take the "free ride", its sensing performance should be affected. Also, when to share should also be decided by the node to gain better payoff. We also consider the influence on the throughput caused by conflicts in the signaling channel during the sharing process of sensing results.

4.2. Model Construction and Analysis

The SUs are players in our game. We first give the strategy set and payoff for each player. Then, we prove the existence of the ESS.

Strategy Set. Different than traditional works, the strategy set here is no longer limited to {contribute, not contribute}, but has more considerations regarding when to share. Specifically, as introduced in Sec. 3, each user needs to pick a sub slot from T_c to send its sensing results. Since each node tries to maximize its throughput, it would be more willing to increase the time spent on data transmission, which means the time spent on the sensing results sharing phase is less. Thus, intuitively, an SU tends to send its sensing results during the early sub slots of T_c, or even may not share its sensing results, in order to have more time for data transmission. However, if more and more SUs choose not to contribute, the sensing performance constraints in Eq. 7 cannot be satisfied. If many SUs choose the sub slots in the early part of T_c, more conflicts would occur in the signaling channel, and the interfered SUs would need to resend their sensing results. Then, the T_c is delayed and T_d is reduced, which results in the decrease of throughput.

Therefore, in our model, the strategy q for each SU needs to contain not only either C (share its sensing results to contribute) or D (deny to contribute), but also when to send out its sensing results over the signaling channel. We have the following definition of the strategy set:

Definition 2. The strategy set of an SU is $\{(C, j)\}$, where $j \in \{0, 1, ..., \lceil \frac{T_c}{t_c} \rceil\}$. $j = 0$ means the SU refuses to share its sensing results. Otherwise, the SU sends its sensing results at the jth sub slot of T_c.

Payoff. The payoff is defined based on the throughput of Eq. 6. For a secondary user S_i that adopts strategy (C, j), we have

$$\delta(t_r) = jt_c + \Delta,$$

where Δ is the time spent on backing off and resending the sensing results of S_i when conflicts happen. The value of Δ depends on the strategies chosen by others.

To replace the $\delta(t_r)$ in Eq. 6, the payoff for S_i that adopts strategy (C, j) is:

$$U_{(C,j)}(S_i) = P_{H_0}(1 - \frac{T_s + jt_c + \Delta}{T})(1 - P_f(S_i))C_{H_0}(S_i).$$
(8)

Analysis. Suppose the mixed strategy adopted by user S_i is $x(S_i)$, which contains whether and when to share the information. Since the starting point of T_c is the same for all SUs, the strategy set is homogenous for all SUs. Suppose that during a time slot t, the probability of an SU S_i to adopt strategy (C, j) is: $p_{(C,j)}(S_i)$. The time evolutionary dynamic $\dot{p}_{(C,j)}(S_i)$ that determines $p_{(C,j)}(S_i)$ is:

$$\dot{p}_{(C,j)}(S_i) = [\bar{U}_{(C,j)}(S_i, -S_i) - \bar{U}_{x(S_i)}(S_i)]p_{(C,j)}(S_i), \quad (9)$$

where $\bar{U}_{(C,j)}(S_i, -S_i)$ is the average payoff for S_i playing pure strategy (C, j), and other SUs playing strategies other than S_i's strategy; $\bar{U}_{x(S_i)}(S_i)$ is the average payoff of user i using mixed strategy x_{S_i}. The intuition for these dynamics is that if S_i achieves a higher payoff using pure strategy (C, j), strategy (C, j) will be adopted more frequently. The growth rate is proportional to the excess of pure strategy (C, j) and the average payoff of the mixed strategy.

Next, we use $y_{(C,j)}$ to denote the proportion of nodes that adopt the pure strategy (C, j) at a given time t. The evolutionary dynamics $\dot{y}_{(C,j)}$ of $y_{(C,j)}$ is given by the following equation, according to the replicator dynamics:

$$\dot{y}_{(C,j)} = [\bar{U}_{(C,j)} - \bar{U}]y_{(C,j)}, \quad (10)$$

where $\bar{U}_{(C,j)}$ is the average payoff of players who use strategy (C, j), and $\bar{U}(x)$ is the average payoff of all players. The $\bar{U}_{(C,j)}$ depends on both of the populations that adopt (C, j). If more players adopt the same (C, j), then conflicts will happen, and the decrease in payoff will be shown in the replicator dynamics. In the following, we prove that starting from any y^*, the replicator dynamics converges to an ESS.

Theorem 1. There exists an ESS to our game model. Specifically, the replicator dynamics could converge to $y^*_{(C,0)} = \frac{1}{N}$, $y^*_{(C,k)} = 1/min\{N, \lceil \frac{T_c}{t_c} \rceil\}$, $k \in [1, min\{N, \lceil \frac{T_c}{t_c} \rceil\}]$ is a unique and consequent number for all users, or $y^*_{(C,0)} = \sigma, y^*_{(C,l)} = (1 - \sigma)/min\{N, \lceil \frac{T_c}{t_c} \rceil\}$, l is a unique and consequent number from 1 to $min\{N, \lceil \frac{T_c}{t_c} \rceil\}$, and σ is the probability of choosing not to share sensing results.

Proof. The first step is to prove the existence of an ESS. Since the maximal number of pure strategies for each SU is $1 + \lceil \frac{T_c}{t_c} \rceil$, the overall strategy set is closed. Since the probability of a certain SU S_i to adopt strategy $c \in \{(C, j)\}$ is $x_{c,S_i}(t)$, assume that the backoff window size is

Table 1. Payoff table of S_1 and S_2

	$(C, 0)$	$(C, 1)$	$(C, 2)$
$(C, 0)$	$0, 0$	$\Theta_1, \Theta_2 - \mathcal{K}t_c$	$\Theta_1, \Theta_2 - 2\mathcal{K}t_c$
$(C, 1)$	$\Theta_1 - \mathcal{K}t_c, \Theta_2$	$\Theta_1 - \mathcal{K}(t_c + \Delta_1), \Theta_2 - \mathcal{K}(t_c + \Delta_2)$	$\Theta_1 - \mathcal{K}t_c, \Theta_2 - 2\mathcal{K}t_c$
$(C, 2)$	$\Theta_1 - 2\mathcal{K}t_c, \Theta_2$	$\Theta_1 - 2\mathcal{K}t_c, \Theta_2 - \mathcal{K}t_c$	$\Theta_1 - \mathcal{K}(2t_c + \Delta_1), \Theta_2 - \mathcal{K}(2t_c + \Delta_2)$

doubled after each conflict for a single user during one time slot, and the initial backoff window size is t_c, then $\delta(t_c)$ is a linear function of t_c. Therefore, $\delta(t_c)$ will not affect $\frac{\partial^2 U(S_i)}{\partial S_i^2}$. From [4], we have $\frac{\partial^2 U(S_i)}{\partial S_i^2} > 0$. Therefore, an ESS exists.

Secondly, all players are treated equally. We use σ and $1 - \sigma$ to first distinguish the probability of users that do not share their results, and users that share sensing results. From [3], we know that there three cases exist:

1. $\sigma = 0$: $T_s + \delta(t_c) = 0$;

2. $\sigma = 1$: all nodes choose to share their sensing results;

3. σ is the solution to the derivation of the payoff difference among users who choose to share and not to share[3].

When case 2 happens, for any S_i that satisfies $y^*_{(C,k)} = 1/min\{N, \lceil \frac{T_c}{t_c} \rceil\}$, $k \in [1, min\{N, \lceil \frac{T_c}{t_c} \rceil\}]$ is a unique and consequential number for all users, S_i's strategy is (C, k_i). If S_i switches to another strategy (C, k_i'), there are two situations:

- No conflict happens:

$$\Delta U(S_i) = P_{H_0}(\alpha - \frac{T_c + \delta(t_c')}{T})(1 - P_f(S_i))C_{H_0}$$
$$-P_{H_0}(\alpha - \frac{T_c + \delta(t_c)}{T})(1 - P_f(S_i))C_{H_0},$$

$t_c' > t_c$ because $k \in [1, min\{N, \lceil \frac{T_c}{t_c} \rceil\}]$. Therefore, $\Delta U(S_i) < 0$, which causes a decrease in Eq. 10.

- A conflict happens between SU S_i and SU $S_{i'}$ that chooses (C, k_i'). The new t_c for both S_i and S_i' would increase because of the backoff policy, which also causes a decrease in Eq. 10.

For case 3, the value of σ is solved in [3]. The part of y^* is similar to that in case 2. □

We give an example here to describe our game model. Suppose there are two players, S_1 and S_2. There are 2 sub slots in T_c. The strategy set is $\{(C, 0), (C, 1), (C, 2)\}$. Without loss of generality, assume $C_{H_0}(S_i)$ and $P_f(S_i)$ in 8 are static. Then the payoff of the two players under H_0 can be written as $\Theta_i - \mathcal{K}(jt_c + \Delta)$, where $\mathcal{K}$ is constant. We also assume that the sensing results of a single SU cannot assure the performance requirements in Eq. 7.

The payoff table of S_1 and S_2 is shown in Table 1. There are 3 main categories regarding the different strategies picked by S_1 and S_2:

- $t_1 = t_2$: If both are equal to 0, then neither of them share their sensing results. Without 7 being satisfied, the payoff is 0. If both are equal to 1, only one of them can resend successfully by backing off 1 sub slot to send. The payoff for the player that resends successfully would be 0, since it does not receive the other's sensing results to ensure Eq. 7. If both are equal to 2, since the length of T_c has only 2 sub slots, neither of the sensing results can be shared. We can treat Δ in two cases as infinity here, which means that both payoffs are approximately 0.

- $t_1 = 0$, $t_2 = 1$ or 2 or $t_1 = 1$ or 2, $t_2 = 0$: Since we assume that the sensing results of a single SU cannot ensure the sensing performance requirements, the one that shares would have payoff 0 with an infinite Δ_i. The other one that does not share would have the maximal payoff Θ_i.

- $t_1 = 1$, $t_2 = 2$ or $t_2 = 2$, $t_1 = 1$: Both sensing results are shared without conflicts. The requirements in Eq. 7 are satisfied. The payoff equals to Θ_i minus the time slots spent on sharing.

In the first two situations, they are not ESS, because the user with payoff 0 can change its strategy in the next round to get a better payoff. In the third situation, the change of strategy by a single SU cannot get a higher payoff in the next round, because conflicts would occur, causing the payoff to be 0 in our example. If we have more users here, the second situation is also able to become an ESS. This is because, not all SUs' sensing results need to be shared to satisfy sensing performance constraints in Eq. 7.

4.3. Evolutionary Algorithm

The evolutionary dynamics for each player are in Eq. 9. To implement a distributed algorithm for each player, we need to define a practical way to calculate $\bar{U}(S_i)$. Therefore, we define a valid time window $\tilde{T}$. Only the payoff within $\tilde{T}$ will be counted to calculate the approximate values of $\bar{U}_{(C,j)}(S_i)$ and $\bar{U}(S_i)$, denoted as

Algorithm 1 Evolutionary algorithm for S_i

1: $t_0 = 0$, $\forall (C, j)$, $p_{(C,j)}(S_i) = p_0$
2: $temp = 0$
3: **while** NOT an ESS **do**
4:     $\tilde{t} = t_0$
5:     **while** $\tilde{t} < t_0 + \tilde{T}$ **do**
6:         Choose (C, j) with probability $p_{(C,j)}$
7:         Calculate $U_{(C,j)}(S_i)$ using Eq. 8
8:         $\tilde{t} = \tilde{t} + T$
9:     $t_0 = t_0 + 1$
10:     Calculate $\bar{U}_{(C,j)}(S_i)$ and $\bar{U}(S_i)$
11:     $\forall (C, j)$, update $p_{(C,j),S_i}$ using Eq. 13
12:     **if** $temp * (\bar{U}_{(C,j)}(S_i) - \bar{U}(S_i)) < 0$ **then**
13:         $\mu = \mu / 2$
14:     $temp = \bar{U}_{(C,j)}(S_i) - \bar{U}(S_i)$

$$\bar{U}_{(C,j)}(S_i) = \frac{\sum_{\tilde{t}=t_0}^{t_0+\tilde{T}} U_{(C,j)}(S_i) B_{(C,j)}(S_i)}{\sum_{\tilde{t}=t_0}^{t_0+\tilde{T}} B_{(C,j)}(S_i)}, \quad (11)$$

$$\bar{U}(S_i) = \lceil \frac{\tilde{T}}{T} \rceil \sum_{\tilde{t}=t_0}^{t_0+\tilde{T}} U(S_i, \tilde{t}), \quad (12)$$

where t_0 is the first time slot of a new time window $\tilde{T}$; $B_{(C,j)}(S_i)$ is the indicator function which is equal to 1 when S_i adopts (C, j) and is 0 otherwise; $U_{S_i}(\tilde{t})$ is the throughput of S_i during $\tilde{t}$; T denotes the default length of one time slot, as indicated before.

Therefore, the probability $p_{(C,j)}(S_i)$ of a user S_i to adopt the pure strategy (C, j) can be updated using Eq. 13. The value of the stepwise μ is not constant. To reduce the oscillation, μ would be divided by 2 if the value of the $\bar{U}_{(C,j)}(S_i) - \bar{U}(S_i)$ changes from positive to negative, or from negative to positive, during two adjacent time slots. The initial value of μ would be studied in our experiment.

$$\begin{aligned} p_{(C,j),S_i}((\tilde{t}+1)) = {} & p_{(C,j),S_i}(\tilde{t}) \\ & + \mu(\bar{U}_{(C,j)}(S_i) - \bar{U}(S_i)) p_{(C,j)}(S_i, \tilde{t}) \end{aligned} \quad (13)$$

The algorithm for each player to reach the ESS is in Algorithm 1. The player tries to converge to ESS within the loop from Step 3 to 14. In Step 4, the new starting time of calculating the average payoff is initialized. From Step 5 to 10, we calculate the average payoff only within the window size $\tilde{T}$. The starting point of the window moves forward by 1 in Step 9. At the end of $\tilde{T}$, each player uses the above equations to update the probability of choosing each strategy in Step 11. From Steps 12 and 14, the value of μ regarding to the stepwise is adjusted. Step 12 decides whether the player has passed the ESS. If it happens, the value of μ would be reduced by half, which means the stepwise is reduced. The process will end when reaching the ESS.
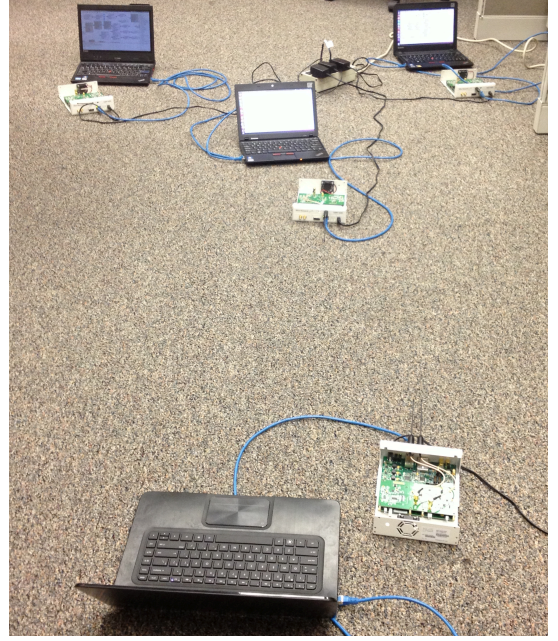
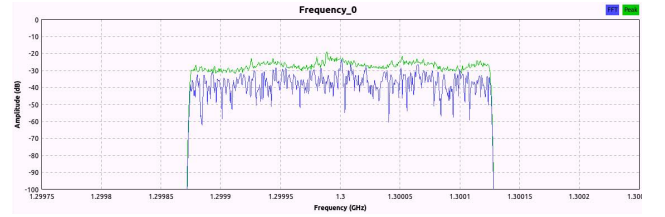

Figure 3. The positions of four USRP N200s.

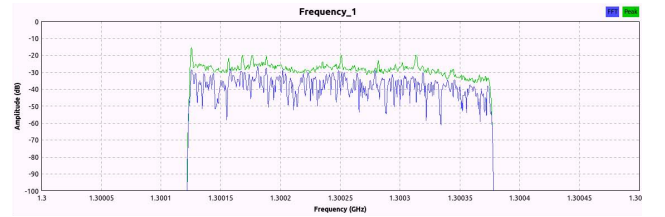**Figure 5.** Secondary user receives at 1.3GHz.

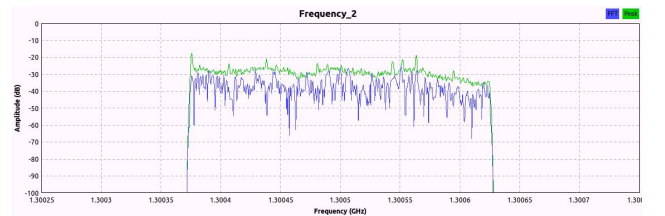**Figure 6.** Secondary user receives at 1.30025GHz.

Figure 7. Secondary user receives at 1.3005GHz.

5. Experiment

In this section, we testify for our model using our testbed of USRP N200s/Gnuradio. We first introduce
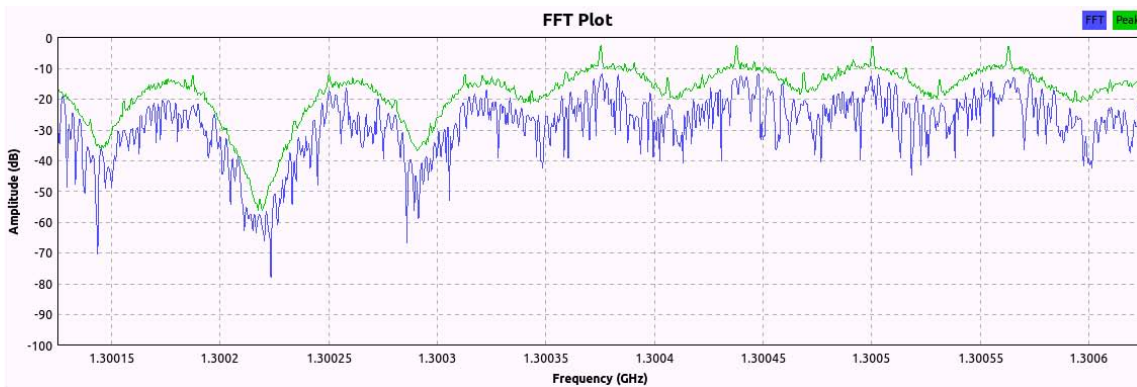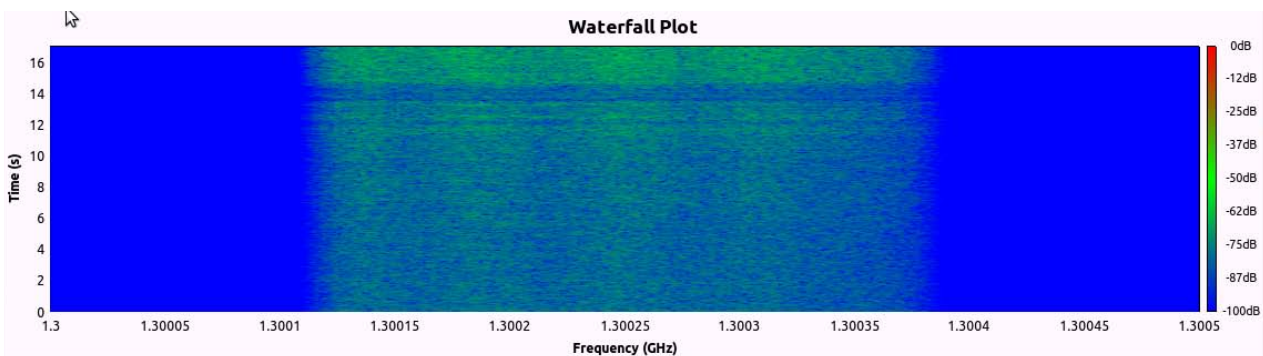
Figure 4. Primary user sends at multiple bands.

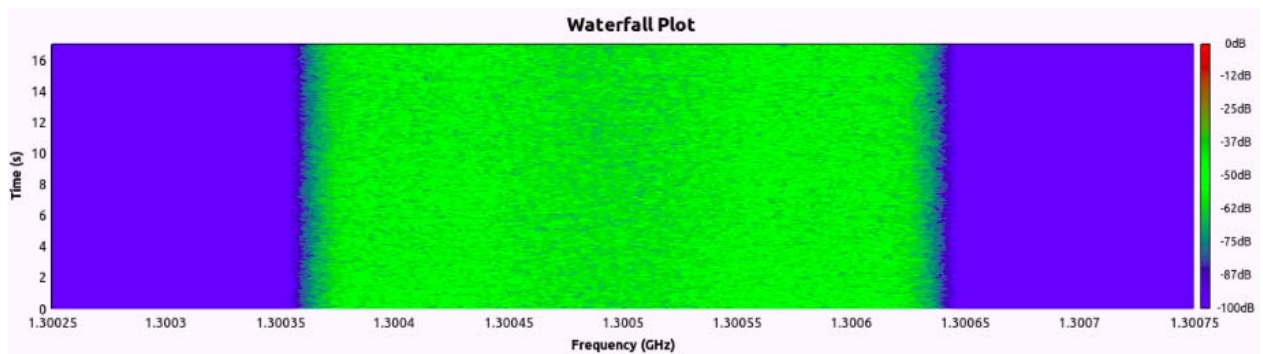**Figure 8.** Sensing results according time at 1.30025GHz.

Figure 9. Sensing results according time at 1.3005GHz.

the structure and parameter settings of our experiment. Then, we present the experimental results.

5.1. Environment Settings

Our experiment consists of four USRN N200s. Three USRPs simulate three SUs, and each works on a sub band. The remaining USRP simulates a PU. We place them at different positions. The distance between an SU and a PU ensures that the sensing results of a single SU is not sensitive enough to detect the signal from the PU. Their relative positions are shown in Fig. 3. The PU occupies multiple bands at the same time, while each SU works on a single subband. As shown in Fig. 4, the PUs occupy wide bands. The received signals on each SU have different central frequencies (1.3GHz, 1.30025GHz, 1.3005GHz), as shown in Figs. 5, 6, and 7. The green lines are the peak points.

We set the time slot length as 20s here (for better synchronization reasons). The static sensing time is set as 5s. The maximal length of T_c is 5s, which is divided into 5 sub slots. The window size for each SU to calculate the average throughput is 4 slots. The bandwidth of each SU is 50k bps. The gain at each receiver is set as 20. We generate an active sequence for the PU with P_{H_0} equal to 0. The thresholds for

probability of detection and probability of false alarm are set as 0.9 and 0.1, respectively. Our experiment works as follows:

- The PU sends out signals while in its active slots.

- SUs sense their own sub band for 4s. We set the threshold as −60 dB, as to decide if the PU is active. The sensing results are sharing on a different subband with a central frequency of 1.30075 GHz.

- After the sharing phase ends, and if it is successful, we calculate how much time remains in the current time slot. If the sharing phase does not succeed, the time left is treated as 0. The payoff is denoted by the time left for data transmission in each time slot, instead of the real throughput. This is reasonable, based on the payoff definition in Eq. 8.

5.2. Experimental Results

In this section, we first testify to the importance of sharing sensing results. Then, we evaluate the convergence based on different initial probabilities of choosing different strategies, and different values of step size.

Unreliability of single sensing. We have the PU to be active and plot the detecting results of each SU, according to the time. The results are shown in Figs. 8 and 9. Due to space limitations, we only show two SUs' receiving results here. The blue parts indicate that no signal is detected, while the green parts indicate that a signal is detected. From the two figures, we can see that the sensing results by a single SU are unstable. Here, for the SU receiving at the central frequency 1.30025GHz, the blue and green parts are mixed, although the PU is active. If this node makes a decision based on its own sensing results, it is possible that it mistakes the unavailable band for an available one, and causes interference to the PU.

Performance versus different initial probabilities. Since the maximal number of sub slots in T_c is 5, the size of the strategy set is 6 for each SU, which is (C, j) and $0 \leq j \leq 5$. We generate two different situations for the initial probabilities of choosing each strategy. One is the random choice, which means each initial probability is equal to 1/6. The second situation is that initial probabilities for 6 strategies are sorted. $(C, 0)$ has the largest initial probability to be chosen, while $(C, 5)$ has the minimal value. The results are shown in Figs. 10 and 11. We can see that under both settings, all three users converge to one pure strategy, and achieve a stable data transmission time, which indicates a stable payoff. We also testify to the sensing performance in Tables 2 and 3. The probability of detection converges to 1. The

probability of false alarm is low initially. It converges to around 0.01.

Performance versus different settings of step size. We also evaluate the influence caused by different values of step size u. We set four different values for u, and calculate the time left for data transmission for SU 1 under the random settings of initial probabilities. The results are shown in Fig. 12. We can see that all four lines converge to the same point, eventually. This is because the value of u is adjusted (reduced by half) during the process. Also, from Fig. 12, we can see that when $u = 3$ or 4, the line has oscillation instead of a continual increase when $u = 1$ or 2. Among these four settings, $u = 2$ achieves the best result.

5.3. Summary of Experimental Results

We implement a testbed consisting of four USRP N200s. One USRP node simulates the PU, and works on multiple subbands simultaneously. Three other USRP nodes simulate SUs. Each works on a subband, and keeps listening to their subband. The sensing results are shared through a common channel. We testify the convergence under different settings of initial probabilities of choosing each strategy. Each SU converges to its stable strategy. We also calculate the probabilities of detection and false alarm, which satisfy both constraints. Finally, we evaluate the influence caused by different settings of step sizes. The larger the initial value of step size, the more oscillated the performance will be. $u = 2$ achieves the best performance under our settings.

6. Extensions

In this section, we introduce two possible extensions for our model. One is about the dynamic sensing time. Another is about achieving the real throughput for our testbed.

6.1. Dynamic Sensing Time

In our current game model, each SU spends the same amount of time for spectrum sensing. This means that the average amount of sensing samples is the same for every SU during the sensing phase in one time slot. In our settings, the sensing samples for one single SU cannot ensure the performance requirements regarding the probability of detection and false alarm. Therefore, the sharing phase is necessary.

However, instead of setting the sensing time as a constant for all SUs, one possible extension is to have the sensing time be dynamic. Each SU can decide, by itself, how long the sensing time will last. This brings two main problems to our current game model. First, the strategy set would be changed. Because the starting point for the sharing phase is changed, it is
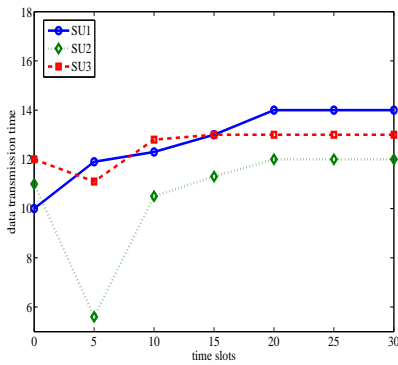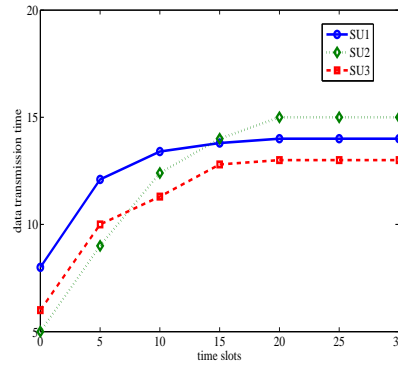
Figure 10. Random initial values.
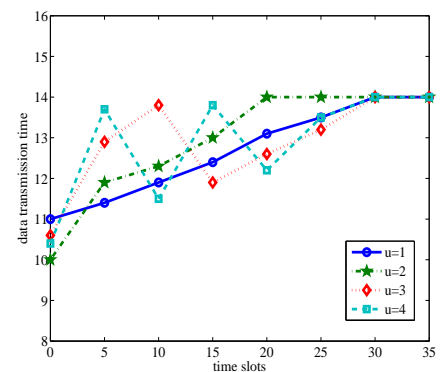

Figure 11. Sorted initial values.

Figure 12. Convergence under different u.

Table 2. Probability of Detection

	5	10	15	20	25
random	0.850	0.924	0.985	1.0	1.0
sorted	0.800	0.901	0.961	1.0	1.0

Table 3. Probability of False Alarm

	5	10	15	20	25
random	0.030	0.053	0.025	0.016	0.011
sorted	0.044	0.034	0.020	0.025	0.015

also different for different SUs. The new strategy set would have one more dimension, as to indicate the ending of the sensing phase, e.g., $\{(C, k, j)\}$, where k is the time slot at the end of the sensing phase, and j still denotes when the sensing results are sending out (with the new starting point of the sharing phase). Besides the changes in the strategy set, the payoff function is different from our existing model. T_s is no longer static, but is related to k in the chosen strategy. Δ is also changed, because the starting points of the sharing phases for different SUs are different, which means the conflicts only happen on the overlaps of the sharing phases. Based on the changes, we need to find the new evolutionary dynamics, and a practical algorithm for each SU to converge.

Testbed Extension. In our current experiment, we calculate the time left for data transmission during each time slot, to approximately represent the throughput performance. However, we can introduce more USRP nodes, and test the real throughput.

The main challenge is the coordination between the sending node and the receiving node. In our current settings, each USRP that represents a SU has its own working subband. It keeps sensing for a static time over its subband, and then shares the sensing results over the signaling channel. If we introduce more USRP nodes

to implement sessions with the current USRP nodes, a coordination scheme is needed between the senders and receivers. A pair of one sender and one receiver needs to know each other's working subband before data transmission. Moreover, when a sender wants to transmit data to a new receiver, it needs to coordinate with the new receiver again, regarding the transmission subband. One possibility is to coordinate through the common signaling channel. However, more conflicts would be brought on the signaling channel. The payoff would also be affected. Therefore, it is impractical to coordinate through the signaling channel.

The above problem can be possibly solved through both centralized schemes. We would implement another USRP serving as a centralized controller. The controller keeps listening on its own subband, the information of which is known by all others. Also, the scheduler knows all the other nodes' current working subbands. Every time a new session is created, the sender sends a request to the scheduler over the scheduler's channel. Then, the scheduler sends the information to the receiver over the receiver's subband. After the receiver switches to the sender's subband, it sends back an ACK directly to the sender, over the sender's subband. Then, the new session is created. If the sender does not receive ACK, it waits for a certain period of time, and sends the request again later.

7. Conclusion

In this paper, we consider both whether-to-share and when-to-share problems, regarding the cooperative spectrum sensing in CRNs. We build an evolutionary game model, in which each SU is treated as a player, and the payoff is the throughput. We extend the strategy set for each SU, and define the payoff based on the time left for transmission. We prove the existence of the evolutionary stable strategy (ESS). Then, a practical algorithm is proposed for each SU to converge. A constant window is defined for each SU to calculate its average throughput. In addition, we construct a

testbed using 4 USRP N200s. One simulates the PU, and the other three simulate the SUs. We evaluate the performance under different settings, regarding the initial probabilities of choosing each strategy. The performance is measured based on the length of time left in each time slot for data transmission. We also show that the probability of detection and false alarm satisfy the constraints. Finally, we study the influence of different values of step sizes on convergence to the ESS. A more in-depth study of extensions is needed, including an implementation of verification of their effectiveness.

References

[1] I. Akyildiz, W. Lee, M. Vuran, and S. Mohanty (2006) *NeXt generation/Dynamic Spectrum Access/Cognitive Radio Wireless Networks: A Survey*, Computer Networks.

[2] I. Akyildiz, B. Lo, and R. Balakrishnan (2011) *Cooperative spectrum sensing in cognitive radio networks: A survey*, Physical Communication.

[3] B. Wang, K. Liu, and T. Clancy (2010) *Evolutionary Cooperative Spectrum Sensing Game: How to Collaborate?*, IEEE Transactions on Communications.

[4] W. Yuan, H. Leung, W. Cheng, S. Cheng, and B. Chen (2012) *Participation in Repeated Cooperative Spectrum Sensing: A Game-Theoretic Perspective*, IEEE Transactions on Wireless Communications.

[5] F. Zeng, C. Li , and Z. Tian (2011) *Distributed Compressive Spectrum Sensing in Cooperative Multihop Cognitive Networks*, IEEE Journal of Selected Topics in Signal Processing.

[6] C. Sun, W. Chen, and K. Letaief (2009) *Joint Scheduling and Cooperative Sensing in Cognitive Radios: A Game Theoretic Approach*, Proc. of IEEE WCNC.

[7] S. Atapattu, C. Tellambura, and Hai Jiang (2011) *Energy Detection Based Cooperative Spectrum Sensing in Cognitive Radio Networks*, IEEE Transactions on Wireless Communications.

[8] L. Cardoso, M. Debbah, P. Bianchi, and J. Najim (2008) *Spectrum Sensing Using Random Matrix Theory*, Proc. of ISWPC.

[9] C. Sun, W. Zhang, and K. Letaief (2007) *Cluster-based Cooperative Spectrum Sensing in Cognitive Radio Systems*, Proc. of ICC.

[10] L. Cao and H. Zheng (2008) *Distributed Rule-regulated Spectrum Sharing*, IEEE Journal on Selected Areas in Communications: Special Issue on Cognitive Radio: Theory and Applications.

[11] W. Zhang and K. Letaief (2008) *Cooperative Spectrum Sensing with Transmit and Relay Diversity in Cognitive Networks*, IEEE Transactions on Wireless Communications.

[12] J. Meng, W. Yin, H. Li , E. Hossain, and Z. Han (2011) *Collaborative Spectrum Sensing from Sparse Observations in Cognitive Radio Networks*, IEEE Journal on Selected Areas in Communications.

[13] Z. Zhang, Z. Han, H. Li , D. Yang, and C. Pei (2011) *Belief Propagation Based Cooperative Compressed Spectrum Sensing in Wideband Cognitive Radio Networks*, IEEE Transactions on Wireless Communications.

[14] W. Gabran, P. Pawelczak, and D. Cabric (2011) *Multi-channel Multi-stage Spectrum Sensing: Link Layer Performance and Energy Consumption*, IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN).

[15] A. Singh, M. Bhatnagar, and R. Mallik (2012) *Cooperative Spectrum Sensing in Multiple Antenna Based Cognitive Radio Network Using an Improved Energy Detector*, IEEE Communication Letters.

[16] H. Poor (1994) *An Introduction to Signal Detection and Estimation*, 2nd ed.

[17] Y. Liang, Y. Zeng, E. Pehi, and A. Hoang (2008) *Sensing-throughput Tradeoff For Cognitive Radio Networks*, IEEE Transactions on Wireless Communications.

[18] S. Lasaulce, M. Debbah, and E. Altman (2009) *Methodologies for Analyzing Equilibria in Wireless Games*, IEEE Signal Process.

[19] B. Wang, Z. Han, and K. Liu (2009) *Distributed Relay Selection and Power Control for Multiuser Cooperative Communication Networks Using Stackelberg Game*, IEEE Transactions on Mobile Computing.

[20] J. Maynard Smith (1982) *Evolution and the Theory of Games*, Cambridege University Press.