Modelling and mitigating spectrum sensing non-cooperation attack in cognitive radio network

Roshni Rajkumari* and Ningrinla Marchang

Department of Computer Science and Engineering,
North Eastern Regional Institute of Science and Technology, NERIST,
Nirjuli, Arunachal Pradesh 791109, India
Email: rajkumari.roshni@gmail.com
Email: ningrinla@yahoo.com
*Corresponding author

Abstract: Collaborative spectrum sensing (CSS) is known to improve spectrum sensing performance in Cognitive Radio Network. In CSS, secondary users participate by sharing their local sensing results. They participate in the sensing process at their own cost, i.e., they expend some amount of energy and time for sensing and sharing. But, a selfish user may refrain from collaborating in the spectrum sensing process in order to save up energy, which results in improper sensing. While this problem is widely known, we call this as the spectrum sensing non-cooperation (SSNC) attack for easy reference. In this paper, a collective action prisoner's dilemma game is used to model the SSNC attack. To handle this attack, repeated game punishment mechanisms, namely Tit-for-Tat and Grim strategies are used. In addition, modified Tit-for-Tat and modified Grim strategies are proposed to handle this attack in the presence of reporting channel error.

Keywords: CRN; cognitive radio network; CSS; collaborative spectrum sensing; spectrum sensing non-cooperation attack; game theory; fusion rules.

Reference to this paper should be made as follows: Rajkumari, R. and Marchang, N. (xxxx) 'Modelling and mitigating spectrum sensing non-cooperation attack in cognitive radio network', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Roshni Rajkumari received her BE in Information Technology from North Maharastra University, Maharastra, India in 2008, and ME in Computer Science and Engineering from Anna University, Tamil Nadu, India, in 2011. From 2011 to 2012, she worked as a Lecturer in Computer Science and Engineering Department in North Eastern Regional Institute of Science and Technology (NERIST), India. She is currently a Research Scholar in CSE Department, NERIST, India. Her research interests include wireless networks, spectrum sensing and security in cognitive radio network.

Ningrinla Marchang received her BTech degree from NERIST in 1993; the MTech degree from the Indian Institute of Technology (IIT), Delhi, India in 1995; and the PhD from NERIST in 2010, all in Computer Science and Engineering. From 1995 to 1996, she worked as a Research Engineer in the Department of Computer Science and Engineering in Indian Institute of Technology, Delhi. From 1996 to 2001, she taught in the Department of Computer Applications in Sathyabama Engineering College, Chennai, India. Since 2001, she has been a Faculty Member of NERIST where she is an Associate Professor in the Department of Computer Science and Engineering. She is currently a Team Member of an ITRA project in cognitive radio network funded by the Government. Her research interests include mobile ad hoc networks and cognitive radio networks. She is a Member of IEEE.

1 Introduction

Cognitive radio network (CRN) is an emerging technology that allows secondary users (SUs) to use the spectrum of the licensed users opportunistically when the primary users are idle. The process of scanning the activity of the primary user (PU), i.e., whether they are using the channel or not is done through spectrum sensing. It is one of the most challenging tasks in CRN.

But, spectrum sensing is often disrupted by several factors such as shadowing, fading, receiver uncertainty problem, etc. To tackle this problem, collaborative spectrum sensing (CSS) is introduced to improve spectrum sensing in fading environment where the SUs collaborate (Da Silva et al., 2007;

Meng et al., 2010). In infrastructure-based CRN, sensing reports from different SUs is combined at the fusion centre (FC) to make the final global decision, whether the PU signal is present or not. But, in infrastructure-less CRN, all the SUs act as fusion centre and receive the sensing result from its neighbouring SUs.

In a network, SUs may have a different self interests; some may act selfishly, some may behave maliciously and some may be unintentionally misbehaving. Regardless of the type of users, CSS can also be severely degraded by faulty observation of the local decision. Providing a wrong spectrum sensing decision is referred to as spectrum sensing data falsification attack (Fragkidakis et al., 2003). However, another important security threat that can disrupt CSS is the non-cooperation of the SUs. In CSS, cooperation among the SUs is essential. Each SU broadcasts the local sensing result to its neighbouring users. Spectrum sensing is energy consuming. To save up energy, a SU may not sense and thus, not contribute. It depends on other SUs to contribute their sensing decision and come to a global decision which it uses subsequently. Such type of non-cooperative SUs may be referred to as non-cooperative/selfish users and the attack caused by them is termed as spectrum sensing non-cooperation (SSNC) attack.

In this paper, a collective action prisoner's dilemma game is proposed to show the interactions between SUs in CSS. In this game, each SU has an option of either collaborating or defecting. The dominant strategy for each SU is not to collaborate. Such type of non-cooperation (SSNC) attack is mitigated by using a repeated game punishment mechanism. We use two classical punishment schemes: tit-for-tat (TFT) strategy and a Grim strategy for punishing the non-cooperating user. In this, punishment is triggered whenever a SU defects. However, in a wireless network, a SU may not receive its neighbour's sensing result due to collision. Thus, sensing result of a SU may not be received due to collision or because the SU did not send the sensing result. So, we propose a modified TFT strategy and a modified grim strategy to mitigate the SSNC attack under such a scenario.

The summary of our contribution is as follows:

- we model the SSNC attack as a collective action prisoner's dilemma game
- we present solutions to the repeated form of this game, TFT and Grim, which mitigate the SSNC attack
- we also present modified versions of the solution, modified TFT and modified grim for handling the SSNC attack in the presence of reporting channel error.

This paper is organised as follows. Section 2 gives the related work based on CSS in CRN. Section 3 presents the system model for the CRN. Section 4 presents the attack model. In Section 5, we present the game. In Section 6, the payoffs of cooperating SUs are compared using fusion rules. Section 7 presents the collective action prisoner's dilemma game and mitigation of SSNC attack. Finally, we provide the simulation result in Sections 8 and conclude in Section 9.

2 Related works

Spectrum sensing not only serves a major challenge in CRN, but securing the spectrum sensing process is also becoming an important issue. With CSS, we expect to improve the performance of our network. However, this can be realised only when all SUs cooperate with honest intention. Most of the literature with CSS assume that all SUs are honest and they always participate in the spectrum sensing process. But, in practice, a SU may not always participate in the spectrum sensing process and behave selfishly (Sun et al., 2009). Such a selfish user may cooperate in their own interest to improve their performance and try to launch the SSNC attack.

Several game theoretic solutions have been previously proposed for adhoc network (Buttyan and Hubaux, 2000; Paramasiva and Pitchai, 2013; Poongothai and Jayarajan, 2008; Wei and Liu, 2007; Zhong et al., 2003) and few game theory based solutions for non-cooperative SUs have been proposed in CRNs (Hongjoun et al., 2013; Kondareddy et al., 2011; Shui et al., 2012; Song and Zhang, 2009; Wang et al., 2010; Wei et al., 2012). Game theory serves as a proper tool that can analyse situations involving conflicting interests among different opposing users.

A carrot and stick strategy is proposed in Song and Zhang (2009) which can recover cooperation among multiple players from deviation. They have also shown that the proposed strategy can achieve mutual cooperation as well as recover from failure. Wang et al. (2010) have presented an evolutionary game model to study the interaction between selfish users in cooperative spectrum sensing. The behavioural dynamics of SUs are studied using replicator dynamics. Here, users update their strategies by exploring different actions at each time, adaptively learning during the strategic interaction and approaching the best response strategy. They have shown that the approach can achieve a higher average throughput in spectrum sensing game with more than two SUs than that of a single user sensing.

Yan and Liu (2011) stimulated cooperation among nodes using indirect reciprocity game, where user help others to accumulate good reputation. Users with high reputation has higher probability to get help from others. The key concept is *"I help you not because you helped me, but because you helped others"*. Another non-cooperative game is also presented in Wei et al. (2012), based on which a distributed algorithm is proposed to achieve the desired frequency solution outcome.

A cooperative spectrum sensing game (CSSG) is proposed in Shui et al. (2012) to study the selfish nature of SUs in cooperative sensing modelled as a Stag Hunt Game. In this game, the benefit of cooperation is proportional to the number of collaborators. Hence, whether cooperation fails or not depends on the number of SUs cooperating in the spectrum sensing process. Thus, in order to avoid cooperation failure, the authors further proposed another scheme called cooperative communication incentive scheme (CCIS) to enhance cooperation. With this scheme, a SU that suffers loss during cooperation can ask for compensation (e.g., using relay to transmit data) from a trusted authority (TA). Another work (Hongjoun et al., 2013) addressed two important issues regarding selfish users. The first is which

action to take: "whether to collaborate or not" and the second is: "which channel to sense?". For answering the first issue, the authors have used an evolutionary game model similar to Wang et al. (2010). The authors further developed an entropy based coalition formation algorithm to solve the problem of channel sensing. With this algorithm, each SU chooses the coalition (channel) that gives more information regarding the channel status. The algorithm ensures that the contributing SU autonomously collaborate and organise themselves into disjoint coalition.

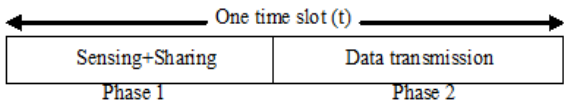
While the above solutions have been reported using different kinds of games, we opine that the Collective action prisoner's dilemma game is also a useful tool for modelling the SSNC attack. Besides, most works have not considered reporting channel error which we have addressed. Our work is for infinite interactions between players unlike in Yan and Liu (2011). In Wei et al. (2012), each SU is required to declare the frequency of sensing participation which is not a requirement in our work. While our work is based on non-cooperative game theory, cooperative game theory has been applied for modelling distributed collaborative sensing in Hongjoun et al. (2013), Kondareddy et al. (2011) and Yucek and Arslan (2011). The problem of reporting channel error is considered in both Kondareddy et al. (2011); Yucek and Arslan (2011) and solution for it is given in Yucek and Arslan (2011).

Motivated by the preceding works, we propose a game-theoretic approach to mitigate SSNC attack in the presence of reporting channel error. With our proposed scheme, nodes are compelled to cooperate because of fear of punishment due to non-cooperation.

3 System model

Consider a CRN with N SUs trying to access a single licensed channel in a network. In CSS, each of the SUs has to cooperate i.e., share its local sensing decision to reach their final destination. The lifetime of each SU is divided into fixed-period time slots and each time slot is divided into two phases as shown in Figure 1.

Figure 1 An Illustration of collaborative spectrum sensing



The first phase is the sensing and sharing phase when a SU senses the environment in t_s (t_s is the time for sensing and sharing) for the presence of incumbent transmission and reports the result. The second phase is the transmission phase, where the SU transmits its data during t_d (t_d is the time for data transmission). Each of the SUs performs a spectrum sensing process to determine whether a PU signal is present or not. We consider a scenario where prior information about the primary user is not known. Thus, an optimal detector that can be used is the energy detection spectrum sensing method (Yucek and Arslan, 2011). When a SU is sensing a licensed spectrum channel in a CR network, the received signal $r_i(t)$ at the energy detector of each SU can be given by two hypotheses, i.e., presence or absence of PU, denoted by H_1 and H_0, which can be depicted as below:

$$r_i(t) = \begin{cases} n_i(t) & \text{if } H_0 \\ h_i(t).s(t) + n_i(t) & \text{if } H_1 \end{cases}, \tag{1}$$

where, $s(t)$ is the unknown signal of the primary user which is a Gaussian process with zero mean and variance σ_x^2 . n_i(t) is the zero-mean additive white gaussian noise of the ith SU with zero mean and variance σ^2, h_i(t) is the channel gain from the primary transmitter to the ith SU.

Suppose Y_i is the sensed energy for the ith SU during time interval L with bandwidth W. Then, the distribution of Y_i is χ^2 distribution as given below:

$$Y_i \sim \begin{cases} \chi^2_{2LW} & \text{if } H_0 \\ \chi^2_{2LW}(2\gamma_i) & \text{if } H_1 \end{cases}, \tag{2}$$

where γ_i is the received signal to noise ratio of the ith SU and LW is the time bandwidth product, χ^2_{2LW} and $\chi^2_{2LW}(2\gamma_i)$ are the central and non-central chi square distribution resp., each with 2LW degree of freedom and a non-centrality parameter of $2\gamma_i$ (Urkowitz, 1967; Kostylev, 2002) for the latter one.

4 Spectrum sensing non-cooperation (SSNC) attack

This section describes the SSNC attack in CSS. In CSS, each SU are expected to participate in the spectrum sensing process. When a SU participate in the spectrum sensing process, some amount of time and energy is used for sensing and sharing. Sometimes, a SU may not be willing to participate in the sensing process. Thus, to save energy, a SU may not sense and does not participate in the spectrum sensing process. Such non-participating SU utilises the sensing decision of other SUs. Such type of non-cooperative SUs are referred to as non-cooperative/selfish users and the attack launched by them as SSNC attack. We model the interactions between SUs as a game and have adopted various game strategies for countering SSNC attack, which is described in Sections 5 and 7 respectively.

5 Game model

In this section, we describe how the interaction between SUs can be modelled as a game. In CSS, we expect the SUs in the network to work cooperatively. A SU should first submit their sensing results to a fusion centre (infrastructure-based) or broadcast to its neighbouring SUs (infrastructure-less), where the reports are aggregated and final result is sent to all. The participating SU incurs some overhead (in terms of energy in sensing and sharing the report). Being a part of a group sharing its sensing reports, a SU is aware that even if it does not share its local sensing report, there might be other truthful users doing so. Thus, a selfish user may do nothing during its sensing phase with the intention of saving up its energy. Again, a SU may choose to quit if the benefit of cooperation is less than its cost. Hence, collaboration fails in either case.

The interaction in collaborative sensing between the SUs in a time slot can be modelled as a collective action prisoner's dilemma game (Avinash et al., 2010). A collective action game is one in which a group of players (SUs) work in collaboration to achieve some common objective. The common objective for all SUs is to achieve a high accuracy of collaborative sensing decision. Each SU can adopt one of the two strategies: *cooperate or defect*.

In a CSS game, a cooperating SU will get the revenue for participating in the spectrum sensing process. At the same time, some cost for participation has to be expended (e.g., energy for sensing, etc.). On the other hand, a defecting SU gets the full revenue without incurring any cost even without participating in the spectrum sensing process.

In a scenario with N SUs where n SUs are cooperating (and N-n are defecting), the payoff of a cooperating (participating) SU is given by:

$$P(n) = B(n) - C(n), \tag{3}$$

where, ***B(n)*** is the benefit a participating SU gets and ***C(n)*** is the cost incurred by the SU for participating. The payoff of a defecting SU is given by:

$$S(n) = B(n). \tag{4}$$

A defecting SU does not incur any cost. However, it enjoys the benefit that a participating SU gets. We further expand the above two equations.

Considering the noise and channel impairments such as shadowing, fading, etc., there may be error during the local spectrum sensing process. Thus, the payoff functions of a cooperating and a defecting user are affected by both the sensing error probabilities i.e., probability of false alarm (P_{fa}) and probability of miss detection (P_{md}) of the local sensing at each participating SU. P^o_{fa} denote the overall false alarm, P^o_{md} the overall miss detection probabilities of the final collaborative sensing decision and P^o_d is the overall detection probability with P^o_d=1-P^o_{md}. The final decision is shared to all the SUs. Any of the two scenarios can happen:

A *PU is actually idle*: If PU signal is absent and if there is no false alarm, an SU can transmit in the slot. So, the payoff of the SU is $(1 - P^o_{fa})(B - C_d)$, where, B is the benefit gain in data transmission and C_d is the cost of transmission.

B *PU is actually busy*: If PU signal is present and there is miss detection an SU will think that PU signal is absent and consequently attempt to transmit causing interference. Data transmission fails. Hence, the payoff of the SU is $P^o_{md}.(-C_d)$

Hence, the payoff of a cooperating SU (by expanding equation (3)) is

$$\begin{aligned} P(n) &= (1 - P^o_{fa})(B - C_d) + P^o_{md}.(-C_d) - C_s \\ &= (1 - P^o_{fa})(B - C_d) + (1 - P^o_d).(-C_d) - C_s, \end{aligned} \tag{5}$$

where,

$$B(n) = (1 - P^o_{fa})(B - C_d) + P^o_{md}.(-C_d)$$

$C(n) = C_s$ and C_s is the cost of sensing and sharing.

The final sensing decision is shared to all the SUs whether they cooperate or not. Thus, a defecting SU only enjoys the benefit without incurring the cost of sensing and sharing its local sensing result. Hence, expanding equation (4), the payoff of a non-cooperating SU is

$$\begin{aligned} S(n) &= (1 - P^o_{fa})(B - C_d) + P^o_{md}.(-C_d) \\ &= (1 - P^o_{fa})(B - C_d) + (1 - P^o_d).(-C_d). \end{aligned} \tag{6}$$

6 Payoff analysis using different fusion rules

This section investigates the payoff function of a SU using the three different decision rules, i.e., OR, K2 and Majority. In equations (5) and (6), $P(n)$ and $S(n)$ depend upon the values of P^o_{fa}, and P^o_d, which in turn depend on the decision rule being used.

6.1 OR rule

In OR rule, if any of the local decision is '1' (busy), then the final decision is busy, else it is '0' (free). Assuming that the individual statistics (Δ_i) of each of the SUs are quantised to 1 bit with Δ_i = 0, 1, 1 gives the presence of signal and 0 gives the absence of signal. The cooperative rule gives the result H_1 if $\sum_i^N \Delta_i \geq 1$. The probability of detection (P_d) and probability of false alarm (P_{fa}) at each local detector is given by Althunibat et al. (2012); Shen et al. (2008)):

$$P_d = Q\left(\frac{\lambda - (\sigma_s^2 + \sigma_x^2)}{(\sigma_s^2 + \sigma_x^2)/\sqrt{S}}\right) \tag{7}$$

$$P_{fa} = Q\left(\frac{\lambda - \sigma_x^2)}{(\sigma_x^2)/\sqrt{S}}\right), \tag{8}$$

where

$$Q(a) = \frac{1}{\sqrt{2\pi}} \int_a^\infty \exp\left(\frac{-t^2}{2}\right).\mathrm{d}t.$$

Here, S is the number of samples of each of the local detectors. According to the OR rule, the overall probability of detection is defined as the probability that at least one SU gives the local sensing result '1' when the channel is actually occupied. Hence, the overall detection probability is given by Wei et al. (2008):

$$P^o_d = \sum_{i=\varepsilon}^{N} \binom{N}{i} (P_d)^i (1 - P_d)^{N-i}, \tag{9}$$

where $\epsilon = 1$.

The overall false alarm is defined as the probability that at least one of the SU give the sensing result '1' when the channel is free (Wei et al., 2008).

$$P^o_{fa} = \sum_{i=\varepsilon}^{N} \binom{N}{i} (P_{fa})^i (1 - P_{fa})^{N-i}, \tag{10}$$

where $\epsilon = 1$.

6.2 K2 rule

The rule states that when at least two out of the SUs report the presence of the primary user, the final decision is primary user present. The cooperative rule gives the result H_1 if $\sum_i^N \Delta_i$. The overall detection probability with K2 rule is defined as the probability that at least two users report a local decision '1' when the channel is busy. Thus, the overall detection probability is given by equation (9) where $\epsilon = 2$.

The overall false alarm probability is defined as the as the probability that at least two of the SUs give the sensing result '1' when the channel is free. Thus, the overall false alarm probability is given by equation (10) where $\epsilon = 2$.

6.3 Majority rule

The majority rule decides the presence of a signal when at least half of the SUs in the network report '1' i.e., channel busy. Thus, the cooperative rule gives the result H_1 if $\sum_i^N \Delta_i \geq N/2$. According to majority rule, the overall probability of detection is defined as the probability that at least half of the SUs give sensing result '1' when the channel is actually busy. Thus, the overall detection probability is given by equation (9) where $\epsilon = N/2$.

The overall probability of false alarm is defined as the probability that at least half of the SUs give sensing result '1' when the channel is actually free. Thus, the overall false alarm probability is given by equation (10) where $\epsilon = N/2$.

7 Collective action prisoner's dilemma game

This section describes the mechanism to counter the SSNC attack. Assume that *n* out of *N* SUs are cooperating. The choice of action of each SU depends on what the other remaining $(N-1)$ SUs are doing. Even if a SU defects, it still enjoys the benefit resulting out of the cooperation of n SUs. So, a shirking SU gets a payoff of *S(n)*. When the shirking SU starts cooperating, the number of cooperating SUs becomes $(n+1)$ and thus, the SU gets a payoff of $P(n+1)$. A collective action prisoner's dilemma game is one in which the dominant strategy for each SU is to defect whereas it is more beneficial for each SU to cooperate. Two conditions for a game to be a Prisoner's Dilemma game are (Avinash et al., 2010):

A $P(n+1) < S(n)$

B $P(N) > S(0)$.

We find that under all the fusion rules; OR, K2 and Majority, our game is a Prisoner's Dilemma Game. Please refer to Figure 2. Assuming 32 SUs ($N = 32$) trying to access the channel, the figure illustrates the payoffs of a cooperating SU, $P(n)$ and non-cooperating SU, $S(n)$ from equations (5) and (6), when S = 10, σ_s^2 = –9 dB, C_d = 0.81, C_s = 0.6. The relationship between B, C_d and C_s will be explained later in Section 8. The payoff of a non-cooperating SU is found to be always larger than that of a cooperating SU for all fusion rules. This implies that no matter how many number of SUs participate in the game, each SU's payoff is higher if the SU shirks than when it participates. Thus, $P(n+1) < S(n)$. Hence, our game satisfies the first condition.

Next, we check whether our game satisfies condition B. If n=0, it depicts the worse case scenario when the overall probability of detection, $P_d^o = 0$ and probability of false alarm, $P_{fa}^o = 1$. So, from equation (6),

$$S(0) = -C_d. \tag{11}$$

Therefore, $P(N) > S(0)$ for all rules, which satisfies the second condition for the Prisoner's Dilemma. Thus, the CSS game is indeed a Prisoner's Dilemma game, where the SUs tend to defect.

Thus, the best way to discourage such kind of non-cooperating behaviour in the Prisoner's Dilemma situation, is to repeat the game (Avinash et al., 2010). Interestingly, spectrum sensing goes on during the whole lifetime of a SU. Thus, the game is repeated at each slot during its lifetime. If the game were to be a one-shot game, then all SUs may choose to defect. But, if the game is repeated, all SUs would be careful enough about what the consequence of their defection in the present game would be in the remaining games. The other cooperating SUs may choose to punish the defecting SU in the remaining game if the SU defected in the present game. Thus, to sustain cooperation, two classical punishment strategies for repeated game can be considered (Avinash et al., 2010).

- *Tit-for-tat (TFT) strategy*: Cooperate with the rival if it cooperated during the most recent game and defect if the rival defects.
- *Grim strategy*: Cooperate till your rival cooperates, but once defection occurs, punish by defecting forever.

7.1 Classical repeated game strategies

In TFT strategy, if a SU deviates at a slot, the remaining SUs stop cooperating from the next slot as a punishment. When the deviating SU comes back to cooperation, the remaining SUs cooperate again. By deviating, we mean that the SU does not sense and also does not send its sensing result. In Grim strategy, if a SU deviates at a slot, then the remaining SUs deviate forever. In both the strategies, punishment is triggered when a SU defects. But, punishment is more severe in Grim strategy as remaining SUs defect forever once an SU defects.

All SUs try to defect in both the games as they get higher payoff when they shirk. But, both in TFT and Grim strategies, a SU can be made to cooperate always, if the one-time gain from defecting is less than the present value of the infinite sum of per-period loss from perpetual defecting (Avinash et al., 2010), i.e.,

$$S(n) - P(n) < \frac{P(n) - S(0)}{r}, \tag{12}$$

where, $S(n)$ is the payoff of the defector, $P(n)$ is the payoff of a cooperating SU, $S(0)$ is the payoff when all SU defects and r is the rate of return. Equation (12) gives the condition under which a SU will always cooperate. Here, $r = \frac{1-\delta}{\delta}$ where δ is the mean session length and denotes how long a SU plans to participate in the game.

Substituting the values of $P(n)$, $S(n)$ from equations (5) and (6), and $S(0)$ from equation (11) into equation (12), the condition under which a SU will always cooperate is given by:

$$B < \frac{\delta.C_d(P_{fa}^o + P_d^o - 1) - C_s}{\delta.(P_{fa}^o - 1)}. \quad (13)$$

7.2 Modified repeated game strategies

In a wireless scenario, there is collision. We now consider the two repeated game strategies when there is collision of packets. A SU participates in the spectrum sensing process, but due to collision, its sensing result may not reach the remaining SUs. So, the remaining SUs conclude that the SU has deviated and thus punishment is triggered. The SU has expended some amount of energy, i.e., cost of sensing. But, due to collision, it may not receive any benefit for its cooperation.

We propose the modified TFT and modified grim strategy which can handle collision as well as take care of the SSNC attack. *Not receiving the sensing result from one SU in one slot may not mean that it is not cooperating*. Thus, to check the behaviour of a SU, whether it is cooperating or not, a sliding window size of W slots is used. When a sensing report is not received from a SU, it is checked whether at least T reports were received during the past W slots. If it is so, then the SU is considered to be cooperating. If not, it is considered as defecting. Here, W and T are predefined thresholds.

Punishment is very heavy in modified grim strategy as compared to modified TFT, as SUs defect forever once deviation is found. But, in both cases, we attempt to trigger punishment when actual deviation is found and not when collision occurs. To capture such a scenario, proper setting of the values of T and W is required. If P_c is the probability of collision in a single slot and collisions occur independently in the slots, the distribution is a binomial distribution. Thus, the expected number of collisions in W slots is $W.P_c$. Therefore, the threshold, T, can be set as a value greater than the expected number of collision i.e., $T > W.P_c$. With these modified strategies, the incentive that the cooperating SUs get is sustained.

The proposed *Modified TFT* technique is given in *Algorithm Modified TFT*. *Modified Grim* is similar to *Modified TFT* except for a few changes in the algorithm which are stated later. Let A_i^t and $payoff_i^t$ denote the action and payoff of SU i at slot t respectively. $A_i^t \in \{C, D\}$ where C and D denote *cooperate* and *defect* respectively. Let $R_{ij}^{W(t-1)}$ denote the number of sensing reports received by SU i from SU j in the past W slots (i.e., in the slots $t-W, \ldots, t-2, t-1$). Let F_i represent a flag of SU i which takes on the value (P-punishment, or NP-not punishment) to denote whether the last defection by SU i was a punishment meted out as a result of defection by another SU j. Under such event, F_i is set to P and M_i is set to j where M_i denotes the SU that SU i punished last.

Modified grim strategy is similar to Modified TFT. The only difference is that the statements under case (ii) in Modified TFT will be replaced by $A_i^{t+1} = D$ to convert it to modified grim algorithm. The algorithm is not shown to save space.

Algorithm Modified TFT.

1. At the initial slot, each SU cooperates. $A_i^1 = C$; $F_i = NP$ for $i = \{1, .., N\}$.

2. At each time slot t, each SU i updates its action as:

case a: $(A_i^t == C)$ [*if i is cooperating in slot t*]
- if sensing report is not received from any SU j
  - if $R_{ij}^{W(t-1)} < T$
    - **i.** $A_i^{t+1} = D$
    - **ii.** $F_i = P$; $M_i = j$
  - endif
- else [*sensing reports received from other SUs*]
  - if i is an attacker
    - $A_i^{t+1} = D$ or C [*depending on the attack frequency policy*]
  - else $A_i^{t+1} = C$ [*cooperate because others are cooperating*]
  - endif
- endif
- calculate $payoff_i^t$

case b: $(A_i^t == D)$ [*if i is defecting in slot t*]
- **case i:** $(F_i == NP)$ [*defection is not a punishment, i is an attacker*]
  - calculate $payoff_i^t$
  - if $payoff_i^t < payoff_i^{t-1}$
    - $A_i^{t+1} = C$
  - else $A_i^{t+1} = D$
  - endif
- **case ii:** $(F_i == P)$
  - if $A_s^t = C$ where $s = M_i$ [*punished SU is cooperating*]
    - $A_i^{t+1} = C$; $F_i = NP$; [*reset F_i*]
  - else $A_i^{t+1} = D$ endif

8 Numerical simulation

In this section, we present results of simulation. The numerical simulations are performed using C programming language. First, we show the payoff comparison of cooperating SUs with different fusion rules, namely, OR, K2 and Majority rules. We assume 32 SUs($N = 32$) trying to access the channel of the primary network. Each SU uses a local energy detector with threshold λ where $\lambda = \sigma_x^2 + 0.1\sigma_s^2$ (empirical value of the experimental analysis in Althunibat et al. (2012). Here, σ_s^2 is kept within the range [–9 dB to –2 dB], while σ_x^2 = –10 dB. The value of B is normalised to 1. Since, C_d and C_s are the costs of transmission of data and sensing respectively, their values are kept lower than B, which is the benefit gained due to data transmission. Moreover, it is assumed that C_d is higher than C_s. We have set P_{md}=0.09 and P_{fa}=0.10 which are within the acceptable limit as defined by FCC (Carlos et al., 2007; Kang et al., 2013).

Figure 2 shows the payoff comparison of OR, K2 and Majority rules. In the figure, the payoff of a cooperating SU under Majority rule (denoted by the label $P(n)\_Maj$) increases as the number of cooperating SUs increases. Other labels have similar meanings. But, for OR and K2 rules,

there is a gradual decrease in the payoff of a cooperating SU even though the number of cooperating SUs increases. Thus, when OR and K2 rules are used, increasing the number of cooperating SUs does not increase the incentive of a cooperating SU.

Figure 2 Payoff of a cooperating and non-cooperating SU when S=10, $\sigma_s^2 = -9$ dB, $C_d = 0.81$, $C_s = 0.6$, $N = 32$ (see online version for colours)

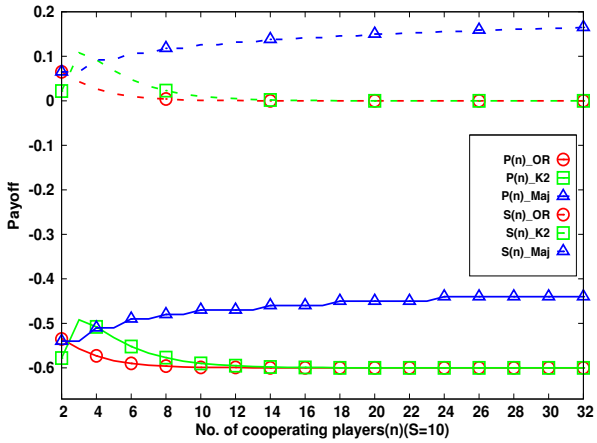


Figure 3 shows the average utility comparison of the proposed game strategy with an existing scheme (Hongjoun et al., 2013) called as utility based spectrum sensing (UBSS) with varying C_s (Cost of sensing). In the existing algorithm, the action of an SU: whether it will cooperate or defect depends on the average utility of the past t slots. The probability of a SU_i choosing action for the next time slot can be computed by Hongjoun et al. (2013):

$$P_{su_i}(e,(t+1)) = P_{su_i}(e,t) + \eta_{su_i}\left[\bar{U}_{su_i}(e) - \bar{U}_{su_i}\right]p_{su_i}(e,t),$$

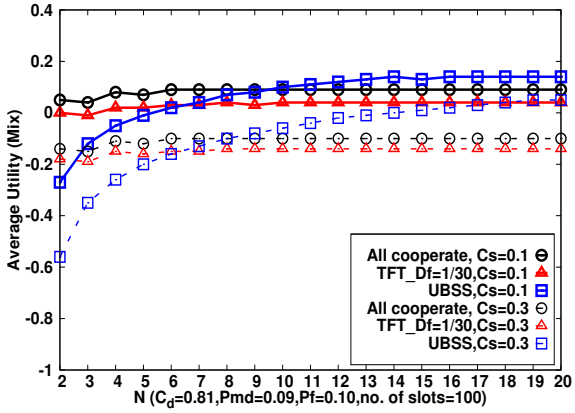
where, η_{su_i} is the stepsize adjustment determined by SU_i, $\bar{U}_{su_i}(e)$ is the average utility for the action $e \in C, D$ for the past t slots and $\bar{U}_{su_i}$ is the average utility of the mixed actions (both cooperate and defect). Here, η=0.06. We compare UBSS with the TFT strategy. While the approaches of these two strategies are very different, for the sake of comparison, we assume that a malicious SU (attacker) defects with a frequency of D_f. $D_f = 1/10$ means that the attacker defects after at every 10^{th} slot. We assume that majority fusion rule is used and the number of attackers is 1. The no. of slots, l=100. Equations (5) and (6) are used for calculating the utility (payoff) for both UBSS and TFT game strategy. The average utility (Mix) is calculated as given below:

$$\bar{U}_{su_i} = \frac{\sum_{i=1}^{N}\sum_{j=1}^{l} U_{\text{action}=C\|D}}{N \times l}, \quad (14)$$

where, N is the number of SUs, $U_{\text{action}=C\|D}$ is the utility for either cooperating or defecting and l is the number of slots.

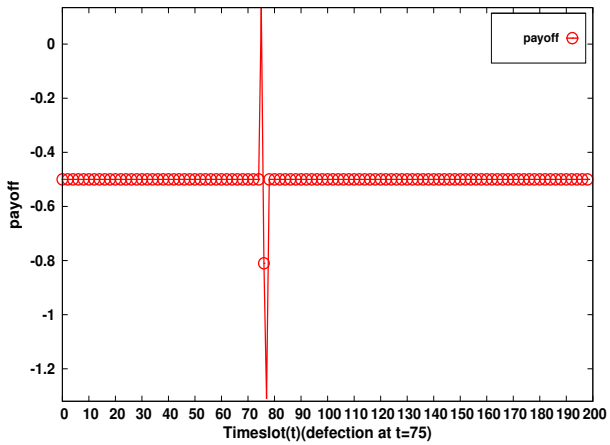
The comparison is given in Figure 3. The plot *All cooperate* denotes one when all the SUs cooperate and none defects. We observe that as the cost of sensing C_s reduces, the average utility increases for all plots. Moreover, we observe that the average utility for TFT is better than that of UBSS until $N = 7$ for both $C_s = 0.1$ and $C_s = 0.3$. But, for higher values of N, UBSS performs better. One interesting observation is that the utility of UBSS surpasses the condition when all SUs cooperate for N higher than some value. This shows that when the value of N is large, it is overall more profitable when some SUs defect (and consequently save up energy). Hence, we are able to infer that redundancy exists in collaborative sensing (especially majority fusion rule). It would be of interest to find the optimal operating point which is out of the scope of this paper. The approach we adopt in our work is to discourage SUs from defecting. Hence, the average utility will not be more than when all cooperate. On the other hand, in UBSS the SUs start defecting from the beginning with a probablity of 0.5. That could be the reason why average utility is less for smaller values of N.

Figure 3 Average utility with varying C_s (see online version for colours)



Figures 4 and 5 illustrate the working of the repeated game strategies TFT and Grim respectively. The payoff of a particular SU for TFT strategy when $N = 20$ is shown in Figure 4. The decision rule used is Majority rule and equation (5) is used for calculating the payoff. Initially, all SUs cooperate.

Figure 4 Illustration of TFT strategy without collision when $N = 20$ (see online version for colours)



Assume that the SU deviates at time slot $t = 75$ because of which its payoff jumps from about –0.52 to 0.1 since all the

other SUs are still cooperating. However, all the remaining SUs come to know about the defection of the SU. So, as a punishment, the remaining SUs starts deviating at $t = 76$. Consequently, the SU's payoff is seen to drop at $t = 76$ to about –0.8. When it observes that its payoff has reduced, the shirking SU decides to come back to cooperation at $t = 77$, while the remaining SUs are still not cooperating. Therefore, a sudden drop in the payoff is noticed at $t = 77$. This is because this SU is the only one expending energy to cooperate while others do not. Seeing that the defecting SU has come back to cooperation, the remaining SUs come back to cooperation from $t = 78$ and the payoff is –0.52 once again.

Figure 5 Illustration of grim strategy without collision when $N = 20$ (see online version for colours)

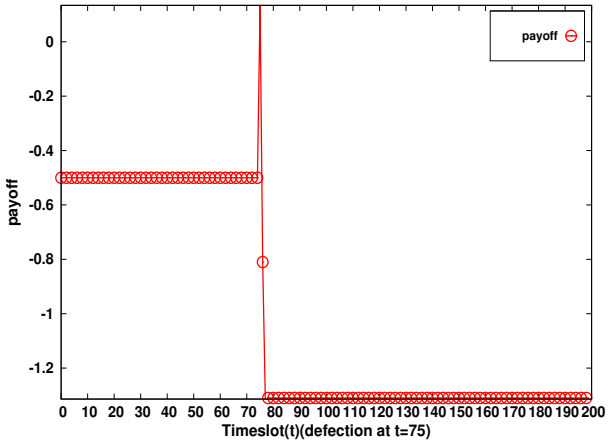


Figure 5 shows the payoff of a SU for Grim strategy when $N = 20$. Assume that the SU deviates at $t = 75$ and gets a higher payoff. But, the remaining SUs start deviating at $t = 76$. Thus, its payoff drops at $t = 76$. At $t = 77$, the shirking SU decides to come back to cooperation, while the remaining SUs are still defecting. The remaining SUs defects forever since they have encountered the SU's deviation at $t = 75$. So, the payoff of the SU drops suddenly at $t = 77$, as it is the only SU to cooperate while the remaining SUs defect forever.

Figure 6 shows the payoff of a SU for TFT strategy with collision when $N = 20$ with $P_c = 0.05$. Initially, all SUs cooperate. Then, the SU deviates at t=75 and comes back to cooperation at $t = 77$ as in Figure 4. From Figure 6, it is seen that collision occurs in slot nos. 48, 73, 105, 117, 122 and 143. Because of collision, the remaining SUs assume the SU has deviated and so they all start deviating as punishment. Consequently, a drop in the payoff is noticed in the corresponding next time slot after collision has occurred i.e., in slot nos. 49, 74, 106, 118, 123 and 144. Similarly, Figure 7 shows the payoff of a SU for Grim strategy with collision when N=20 with P_c=0.05. Here, collision occurs in slot nos. 48, 73, 105, 117, 122 and 143. But, remaining SUs will defect forever from the next time slot (slot no. 49), after detecting the first sensing report collision at slot no. 48.

From Figures 6 and 7, it is seen that random error in a network can cause collision and thus affect the spectrum sensing process in a network. This in turn affect the payoff earned by an individual SU as seen in Figures 6 and 7.

Figure 6 TFT strategy with collision when $N = 20$ (see online version for colours)

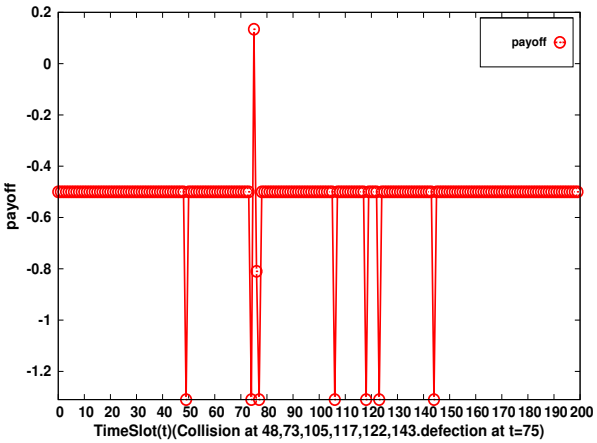
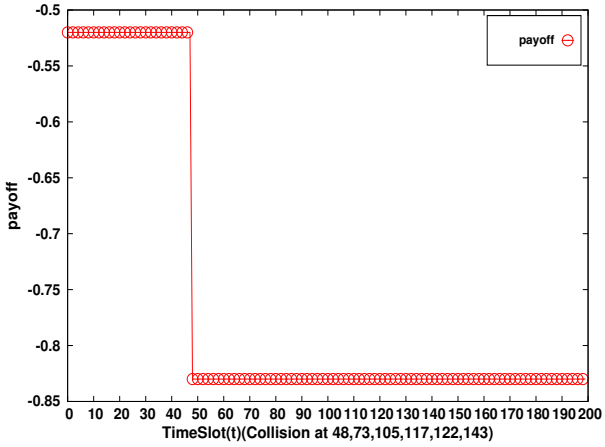
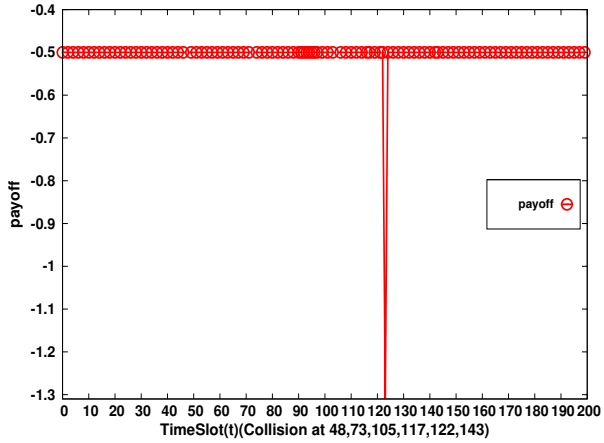


Figure 7 Grim strategy with collision when $N = 20$ (see online version for colours)



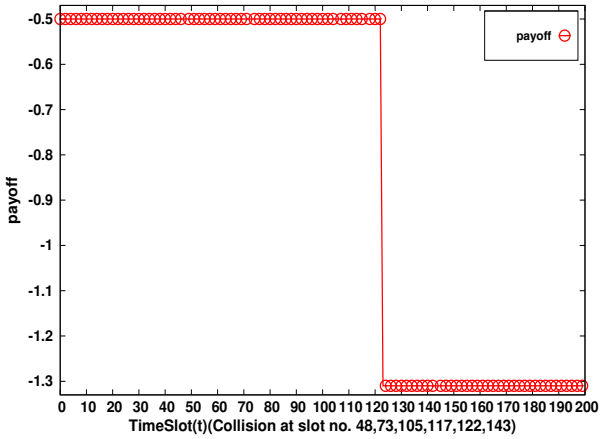
From Figures 6 and 7, it is seen that whenever a collision occurs, punishment is triggered since it is assumed there is a deviation. This shortcoming is overcome with modified TFT and modified grim strategy. Figure 8 shows the payoff of a SU for modified TFT strategy when $N = 20, T = 2$ and $W = 10$. Punishment is triggered only when a SU is suspected to have defected and not every time when collision occurs. If at least T reports were received during the past W slots, then SU is considered to be cooperating, otherwise defecting. In the figure, with $P_c = 0.05$, collision occurs in slot nos. 48, 73, 105, 117, 122 and 143. The drop in the payoff of the SU is at slot no. 123, meaning that the SU is detected by the other SUs as deviating at slot no. 122. However, this is a false alarm since it did not actually deviate at slot 122, but there was a collision at 122. In TFT (refer Figure 6), punishment is triggered whenever a collision occurs and consequently the payoff reduces. But, in modified TFT (refer Figure 8), punishment is triggered only when the SU is suspected to be defecting. Hence, all collisions are not considered as defections. Thus, modified TFT avoids unnecessary punishment triggering in the presence of collision. However, depending on the values of T and W, some collisions may be detected as deviation as in slot no. 122 of Figure 8. Moreover, some defections may pass off being undetected.

Figure 8 Modified TFT strategy with collision when $N = 20$ (see online version for colours)



In Figure 9, the payoff for a particular SU is shown for modified grim strategy. Collision occurs in slot nos. 48, 73, 105, 117, 122, 143 and deviation is detected at slot no. 122 which is actually a collision. In Figure 7 (Grim strategy), the first sensing report collision occurs at slot no. 48 and thus, the remaining SUs defect forever from the next time slot i.e., at 49. Hence, the SU gets a higher payoff till slot no. 48 only. But, in Figure 9, other SUs cooperate till slot no.122, because of which it gets a higher payoff for a longer period of time i.e., till slot no. 122. But, as soon as deviation is found, the other SUs defect forever which is seen by the sudden drop in the payoff at slot no. 123. Thus, with modified TFT and modified grim, a SU receive higher payoffs for a longer period of time than with classical TFT and grim.

Figure 9 Modified grim strategy with collision when $N = 20$ (see online version for colours)



9 Conclusion

A Collective Action Prisoner's Dilemma Game is presented to model the interaction between secondary users in CSS. The solutions of this game TFT and Grim strategies are used to mitigate the spectrum sensing non-cooperative (SSNC) attack. Moreover, we modified these solutions and presented the Modified TFT and modified grim strategies. With the proposed strategies, the adverse effect of reporting channel error in the network is greatly reduced, while mitigating the SSNC attack.

Acknowledgement

This work is partially supported by Information Technology Research Academy (ITRA), Government of India under, ITRA-Mobile grant [ITRA/15(63)/Mobile/MBSSCRN/02/2015].

References

Althunibat, S., Palacios, R. and Granelli, F. (2012) 'Energy-efficient spectrum sensing in cognitive radio networks by coordinated reduction of the sensing users', *IEEE International Conference on Communications (ICC)*, 10–15 June, Ottawa, pp.1399–1404.

Avinash, K.D., Skeath, S. and David, H.R. (2010) *Games of Strategy*, 3rd ed., W.W. Norton & Company, New York.

Buttyan, L. and Hubaux, J. (2000) 'Enforcing service availability in mobile adhoc WANs', *First Annual Workshop on Mobile Adhoc and Networking and Computing, MohiHOC*, Boston, MA, pp.87–96.

Carlos, C., Ghosh, M., Cavalcanti, D. and Challapali, K. (2007) 'Spectrum sensing for dynamic access of TV bands', *IEEE 2nd International Conference on Cognitive Radio Oriented Wireless Networks and Communications*, August, Orlando, FA, USA, pp.225–233.

Da Silva, C., Choi, B. and Kim, K. (2007) 'Distributed spectrum sensing for cognitive radio Systems', *Information Theory and Applications Workshop*, 29 January–2 February, La Jolla, CA, pp.120–123.

Fragkidakis, A.G., Tragos, E.Z. and Askoxylakis, I.G. (2003) 'A survey on security threats and detection techniques in cognitive radio networks', *IEEE Communications Surveys and Tutorials*, Vol. 15, No. 1, First Quarter, pp.428–445.

Hongjoun, L., Xiuzhen, C., Keqiu, L., Xiaoshuang, X. and Tao, J. (2013) 'Utility-based cooperative spectrum sensing scheduling in cognitive radio networks', *IEEE Proceedings INFOCOM*, 14–19 April, Turin, pp.165–169,.

Kang, X., Liang, Y.C., Garg, H.K. and Zhang, L.(2008) 'Sensing-based spectrum sensing in cognitive radio networks', *IEEE Telecommunications Conference GLOBECOM*, New Orleons, pp.1–5.

Kondareddy, Y. and Agrawal, P. (2011) 'Enforcing cooperative spectrum sensing in cognitive radio networks', *IEEE Global Telecommunications Conference (GLOBECOM)*, 5–9 December, Houston, TX, USA, pp.1–6.

Kostylev, V.I. (2002) 'Energy detection of signal with random amplitude', *Proceeding of IEEE International Conference on Communications*, April, New York, NY, pp.1606–1610.

Meng, J., Yin, W., Li, H., Houssain, E. and Han, Z. (2010) 'Collaborative spectrum Sensing from sparse observations using matrix completion for cognitive radio networks', *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, March, Dallas, TZ, pp.3114–3117.

Paramasiva, B. and Pitchai, K.M. (2013) 'Modeling intrusion detection in mobile adhoc networks as a non-cooperative game', *International Conference on Pattern Recognition, Informatics and Mobile Engineering(PRIME)*, 21–22 February, Salem, pp.300–306.

Poongothai, T. and Jayarajan, K. (2008) 'A non-cooperative game approach for intrusion detection in mobile adhoc networks', *International Conference on Computing, Communication and Networking*, St. Thomas, 18–20 December, pp.1–4.

Shen, J., Lui, S., Zhang, R. and Lui, Y. (2008). 'Software versus hard cooperative energy detection under low SNR', *Third International Conference on Communications and Networking*, 25–27 August, Hangzhou, pp.128–131.

Shuai, L., Haojin, Z., Bo, Y., Cailian C., Xinping, G. and Xiaodang, L. (2012) 'Towards a game theoritical modeling of rational collaborative spectrum sensing in cognitive radio networks', *IEEE International Conference on Communication (ICC)*, 10–15 June, Ottawa, pp. 88-92.

Song, C. and Zhang, Q. (2009) 'Achieving cooperative spectrum sensing in wireless cognitive radio networks', *ACM SIGMOBILE Mobile Computing And Communications Review*, Vol. 13, No. 2, pp.14–25.

Sun, C., Wei C. and Ben, L. (2009) 'Joint scheduling and cooperative sensing in cognitive radios: a game theoritic approach', *Wireless Communications and Networking Conference IEEE, WCNC*, April, Budapest, pp.1–5.

Urkowitz, H. (1967) 'Energy detection of Unknown deterministic signals', *Proceedings of IEEE*, Vol. 55, No. 4, pp.523–531.

Wang B., Liu K.J. and Clancy T.C. (2010) 'Evolutionary cooperative spectrum sensing game: how to collaborate', *IEEE transactions on Communication*, Vol. 58, No. 3, March, pp.890–900.

Wei, Y. and Liu, K.J.R. (2007) 'Game theoritic analysis of cooperation stimulation and security in autonomous mobile adhoc networks', *IEEE transaction on Mobile Computing*, Vol. 6, No. 5, pp.507–521.

Wei, Z., Mallik, R.K. and Khaled, L. (2008) 'Cooperative spectrum sensing optimization in cognitive radio networks', *IEEE International Conference on Communications*, 19–23 May, Beijing, pp.3411–3415.

Wei, Y., Leung, H., Wenqing, C., Siyue, C. and Bokan C. (2012) 'Participation in repeated cooperative spectrum sensing: a game theoritic perspective', *IEEE transactions on Wireless Communications*, Vol. 11, No. 3, pp.1000–1011.

Yan, C. and Liu, K.J.R. (2011) 'Indirect reciprocity game modeling for cooperation stimulation in cognitive radio network', *IEEE Transaction on Communications*, Vol. 59, No. 1, pp.159–168.

Yucek, T. and Arslan, H. (2011) 'A survey of spectrum algorithms for cognitive radio applications', *IEEE Communication Surveys and Tutorials*, Vol. 11, No. 1, pp.116–130.

Zhong, S., Chen, J. and Yang, Y.R. (2003) 'Sprite: a simple, cheat-proof, credit-based system for mobile adhoc networks', *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, INFOCOM*, 30 March, San Francisco, California, USA, pp.1987–1997.