# Enabling Fair Spectrum Sharing: Mitigating Selfish Misbehaviors in Spectrum Contention

**Kaigui Bian, Peking University**
**Jung-Min "Jerry" Park, Virginia Tech**
**Xiaojiang Du, Temple University**
**Xiaoming Li, Peking University**

**Abstract**

Cognitive radio (CR) technology enables multiple wireless networks operating in overlapping regions to opportunistically access fallow spectrum from a common pool of spectrum. This spectrum access paradigm — referred to herein as simply spectrum sharing — holds the promise of significantly greater efficiency in spectrum utilization and alleviating the spectrum shortage problem. CRs have garnered great attention from the research community, and many security and privacy problems relevant to CR networks are being studied actively at this time. However, selfish misbehaviors that can occur in the spectrum contention process have received little attention. In this article we discuss two types of selfish misbehaviors in the context of spectrum contention: selfish spectrum contention and selfish channel negotiation. These misbehaviors deteriorate the fairness and performance of spectrum sharing mechanisms in both infrastructure-based and multi-hop CR networks. We also discuss countermeasures against these threats as well as the technical challenges that must be overcome to implement such countermeasures.

Cognitive radio (CR) is an enabling technology for dynamic spectrum access (DSA) that promises to alleviate the spectrum shortage problem in wireless networks. By employing *spectrum sensing* techniques or referring to the geolocation databases, a CR (or a CR-enabled secondary user) is able to intelligently decide which licensed spectrum band is free for use without causing harmful interference to licensed (or primary) users. Besides, the spectrum sharing mechanism allows co-located CRs or CR networks to opportunistically utilize those identified fallow spectrum in a fair and efficient manner. Whereas the use of spectrum sensing and databases helps in the discovery of fallow spectrum from the perspective of licensed operation protections, the spectrum sharing mechanism aims to improve the spectrum usage for unlicensed CR networks.

Many security and privacy problems have been identified in both the sensing-based and database-driven CR networks. A single CR that employs the spectrum sensing technique may not be able to verify the authenticity of the received primary user signals or the sensing reports sent from other CRs. Thus, the spectrum sensing mechanism is subject to the primary user emulation attack, falsification of spectrum sensing results, and selfish sensing misbehaviors [1, 2]. Authentication of primary user signals and collaborative spectrum sensing techniques have been proposed to defend against these threats [3–5]. Moreover, privacy issues (e.g. unauthorized release of location information) can arise in both sensing-based and database-driven CR networks, and privacy preservation techniques have been used to eliminate these privacy concerns [6, 7].

However, security vulnerabilities in the spectrum sharing mechanism, especially those that can be exploited by selfish users, have received little attention. Performing fair spectrum sharing is a challenging task in CR networks. On the one hand, every user is selfish by nature. This implies that a selfish user tends to exploit the vulnerabilities of spectrum sharing mechanisms to gain an unfair advantage in spectrum access, if such misbehaviors cannot be easily detected. On the other hand, coexisting CRs or CR networks may be operated by competing service providers, and it is difficult to establish a centralized spectrum sharing mechanism that can monitor or regulate the behaviors of all the entities. Moreover, a CR may take a stealthy way of misbehaving, e.g. randomly switching between the norm and misbehaviors, which increases the difficulty of detection. Next, we introduce two spectrum sharing mechanisms in infrastructure-based and multi-hop CR networks.

## Spectrum Sharing Mechanisms for CR Networks

The deployment scenario of CR networks can be either an infrastructure-based architecture or a multi-hop architecture, and spectrum sharing is an essential component of both architectures.


0890-8044/13/$25.00 © 2013 IEEE

### Inter-Network Spectrum Contention

An infrastructure-based CR network architecture is employed in the IEEE 802.22 standard that specifies the air interface (both physical [PHY] and medium access control [MAC] layers) for a CR-based wireless regional area network (WRAN) [8]. An 802.22 network cell is composed of a base station (BS) and a number of 802.22 users (i.e. consumer premise equipments or CPEs). Neighboring 802.22 networks are able to coordinate with each other via the inter-BS communication method (e.g. backhaul connections).

The 802.22 standard has prescribed an on-demand frame contention (ODFC) protocol to address the spectrum sharing problem when a channel is shared by multiple co-located 802.22 networks. To avoid inter-network interference, only one network can operate on the shared channel at any time instance. In an ODFC process, multiple BSs of 802.22 networks contend for a target channel (or a time frame over the target channel) by exchanging control packets, and BSs in contention play the following game:

- Every BS randomly generates a contention number (CN) that is uniformly distributed in the range [0, $W$], where $W$ is a constant representing the contention window size. The chosen CN value of a BS will be broadcasted to neighboring competitors.
- The BS that has selected the greatest CN among all participating BSs is the winner of the contention. Other BSs (and their 802.22 networks) that fail to win will vacate the channel.

Suppose there are $k$ contending BSs, and we call such a scenario a *k-system spectrum contention process*, where each BS has an equal probability of nearly $1/k$ to win the contention. If the ODFC process is executed repeatedly by the same group of $k$ BSs, the contended spectrum (e.g. the target channel) will be shared among them fairly in the long run. A three-system spectrum contention process of ODFC is illustrated in Fig. 1.

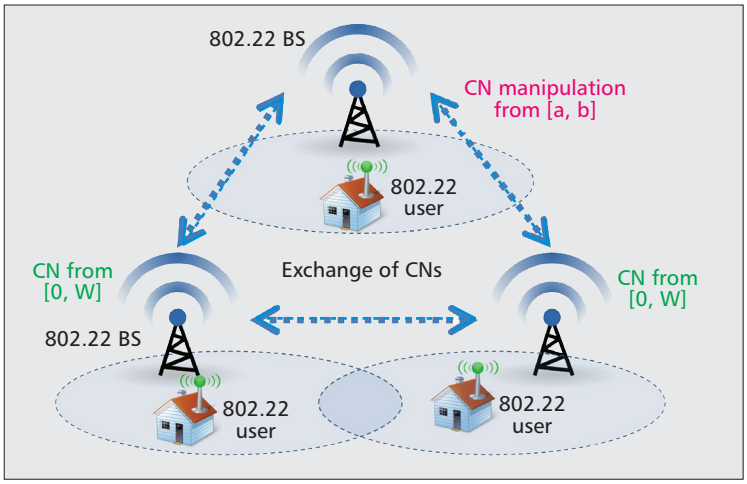


**Figure 1.** *An illustration of a three-system spectrum contention process of ODFC. Every BS is required to generate its contention number (CN) at random from the range [0, W], and then exchange the generated CN with neighbors for participating in the spectrum contention.*

### Distributed Channel Negotiation

Without a centralized controller entity, a multi-hop CR network is comprised of mobile and/or stationary devices equipped with CRs, and they interact with each other via multi-hop wireless links.

In a multi-hop network architecture, CRs contend for spectrum via a *distributed channel negotiation* process by exchanging MAC-layer control frames in a common control channel or a rendezvous channel. Since there is no access point or base station, channel negotiation is carried out in a distributed manner between each pair of neighboring nodes along a multi-hop route. Figure 2 shows an example of a channel negotiation process between a pair of neighboring nodes (nodes A and B) in a multi-hop CR network. Nodes S and D in the example represent the source and destination nodes of a multi-hop data flow. There are two possible routes from the source to the destination, where nodes A, B, and C are intermediate nodes belonging to the two routes. For the pair of nodes A and B, the upstream node A first identifies fallow spectrum bands and sends its free channel list ($FCL_A$) to one of the potential receivers (node B). Upon the reception of $FCL_A$, the receiver creates its own free channel list ($FCL_B$) and sends it back to node A. By comparing the two lists of free channels, node A is able to find a channel that is common to both sides for data packet transmission (e.g. channel 0 is selected for data transmissions between nodes A and B in Fig. 2).

Although security and privacy of CR networks is an active area of research, relevant security vulnerabilities regarding

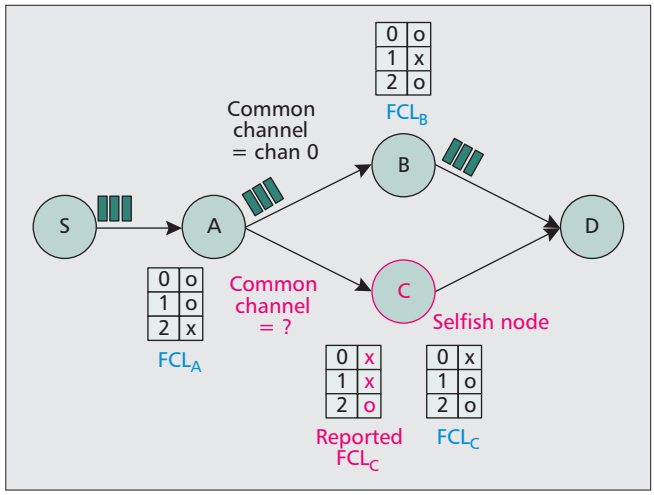


**Figure 2.** *An illustration of the channel negotiation process between two CR nodes. Suppose there are a total number of three channels, labeled as 0, 1, 2. Each table in the figure represents the free channel list of a given CR node: the circle sign in the same row of integer* i *means channel* i *is available to the node; and the cross sign means channel* i *is unavailable. For example, the table of the free channel list* $FCL_A$ *shows that channels 0 and 1 are available to node A, while channel 2 is unavailable.*

selfish misbehaviors in spectrum sharing mechanisms have yet to be studied. In the rest of this article, we describe two types of selfish misbehaviors that may wreak havoc in the above spectrum sharing mechanisms: selfish spectrum contention and selfish channel negotiation. After analyzing these threats, we also discuss potential countermeasures.

## Selfish Spectrum Contention

When multiple CR networks (e.g. 802.22 networks), operated by different service providers, coexist in overlapping regions, they will compete for spectrum using some type of a spectrum contention mechanism. Hence, there is a possibility that some of the competing networks may exploit the vulnerabilities in the contention mechanism to gain an unfair advantage over the others.

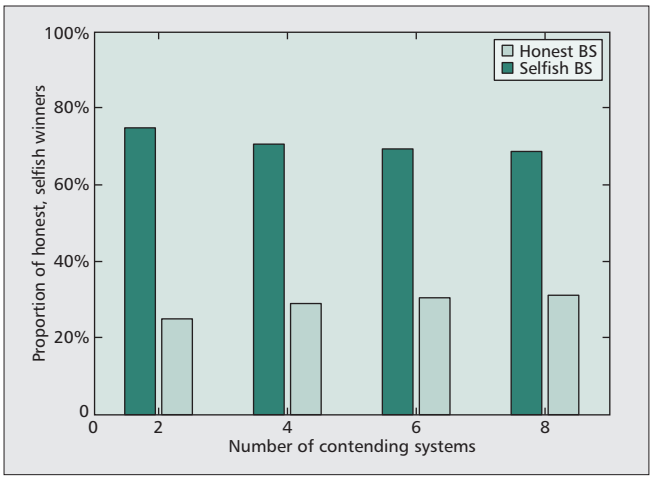


**Figure 3.** *Proportion of honest and selfish winners in a k-system spectrum contention process, where* k = *2, 4, 6, 8, half of the systems are selfish, and the other ones are honest.*

### Description of The Misbehavior

When an 802.22 network wins the spectrum contention regarding a target channel, all other secondary networks that fail to win will avoid accessing that channel until the next spectrum contention process starts. According to the 802.22 ODFC, the condition for a network to win is to generate the greatest CN value among all networks during a contention process.

*Manipulation of the Contention Number Value* — We define a *selfish* 802.22 system in spectrum contention as an 802.22 network that exploits the ODFC protocol in a covert manner to gain an unfair advantage over the other networks. Specifically, a selfish 802.22 system may use an arbitrarily large CN value without conforming to the CN selection rule defined in 802.22. We refer to such behaviors as selfish spectrum contention (SSC) misbehaviors, and the BS of a selfish 802.22 system as a *selfish BS*.

In this article, we consider the spectrum contention process between selfish and honest BSs. The selfish BS selects a CN randomly from a *modified* range to gain an unfair advantage. Specifically, it selects a CN from the range $[a, b]$, where $0 \le a \le b \le W$. The honest BS selects the CN from the range $[0, W]$. In a spectrum contention process among a selfish BS and two honest BSs (the example in Fig. 1), when $a = W/2$ and $b = W$, the selfish BS's winning probability is approximately 7/12, while each honest BS's winning probability is only 5/24, assuming the selection of CN is uniformly distributed throughout a given range.

Note that the CN manipulation from a modified range is an example strategy of the SSC misbehavior, and there may exist other strategies for conducting similar misbehaviors which are out of the scope of this article.

### Analysis of The Vulnerability

The 802.22 contention resolution rule is vulnerable to the SSC misbehavior: an honest BS infers the winner of the contention merely based on the received CN values of all competitors, and it cannot distinguish whether a CN value has been manipulated.

*Difficulty of Detecting the SSC Misbehavior* — Since the CN value is assumed to be randomly chosen, it is difficult for any entity to determine whether a received CN value is chosen by an honest BS from the normal range or by a selfish BS from a modified range. Even though the average CN value of a given BS for multiple contention processes is higher than the average CN value of other BSs, we are not 100 percent confident about whether the given BS has conducted the SSC misbehavior via CN manipulations.

*Practical Way of Conducting the SSC Misbehavior* — Due to the programmability of CRs, it is possible for a selfish CR to modify the radio software of a CR to change its parameters, including the contention window range, so as to gain priority of spectrum access over other secondaries. The potential impact of the SSC misbehavior depends on the range of CN values generated from the modified contention window.

*Impact of the SSC Misbehavior* — Due to the possibility of such selfish misbehavior, the fairness of the contention protocol is impaired. Suppose there are $k$ co-located 802.22 networks, where $x$ of them are selfish systems and the remaining $(k - x)$ of them are honest. Selfish BSs select their CN values from the modified range $[W/2, W]$, and aim to preempt the target channel with a higher winning probability compared to others. The probability that selfish systems win the channel contention is given by the following expression:

$$\left(\frac{1}{2}\right)^{(k-x)} \sum_{i=0}^{k-x} \binom{k-x}{i} \cdot \frac{x}{x+i}.$$

We carried out simulation experiments to evaluate the effect of the above SSC misbehavior. Figure 3 shows the proportion of honest and selfish winners, and we can observe that the winning probability for selfish BSs is much higher than that of honest BSs.

## Defending Against Selfish Spectrum Contention

The key to defending against selfish spectrum contention misbehaviors is to devise a prevention technique that eliminates the feasibility of manipulating CN values in the spectrum contention process. Our research findings indicate that the spectrum contention problem is equivalent to a coin-flipping game. The bit-commitment technique is useful to address the following challenges in the coin-flipping game, which are also the requirements that need to be satisfied in the design of a spectrum contention protocol:

- The winner is collectively determined by the group of all players.
- Every player has an equal probability of being chosen as the winner.
- No player can bias the result.

### Coin-Flipping and Bit-Commitment

Collective (or distributed) coin-flipping is the problem of producing a common random bit (or number) among $n$ players (where $n \ge 2$) in a distributed computing environment, such that no player can bias this bit (or number) too much [9]. A protocol that solves the coin-flipping problem requires a number of rounds of coin flips. In each round, every player generates a local coin flip and broadcasts the coin flip result. After the completion of all rounds, previous local coin flips are combined to form a global coin flip using a pre-specified *group decision* function.

The *bit-commitment* technique is used as the solution to a two-player coin-flipping game, which consists of two phases of message exchanges:

- *Commit* phase: Every player randomly selects a bit value, 0 or 1 (local coin flip result), and makes a *commitment* to

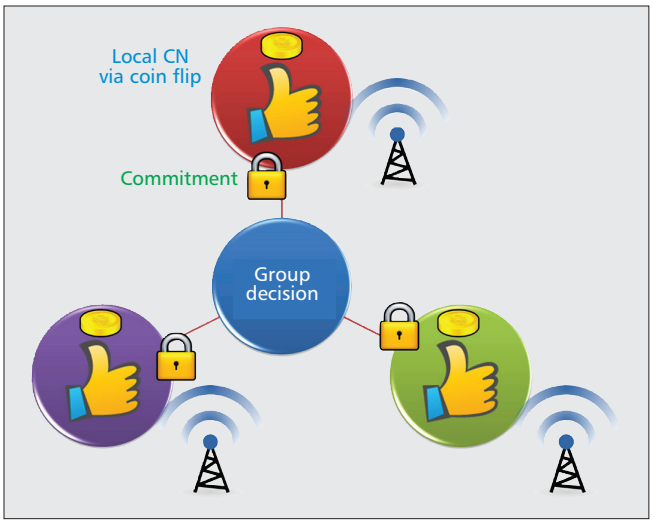


**Figure 4.** *A commitment-based spectrum contention process among three players (802.22 networks). Note that the commitment message (represented as the "lock" icon in the figure) must be broadcasted first, before the CN value (the local coin flip result) is revealed.*

its selected bit value. The committed bit value is protected in the commitment message.

- *Reveal* phase: Every player reveals its selected bit value (local coin-flip result).

Using a pre-specified decision function, every player is able to agree on a final common bit — the global coin flip result.

It is possible that a player can bias the final outcome of a collective coin-flipping game by manipulating its local coin flip result. For example, Alice and Bob start a collective coin-flipping game over the telephone. Suppose the pre-specified decision function is defined as the "exclusive-OR" of local coin flips. Alice wins if the output of the decision function is 1; otherwise Bob wins. In this game, Alice or Bob may lie to the other about her/his coin flip result in order to win the game. Therefore, two security properties must be guaranteed in a bit-commitment scheme to prevent any player from cheating in the game. Suppose ε is a negligibly small value:

- *Hiding*: Any player cannot know the other's committed value other than using a random guess, before the "reveal" phase; in other words, any player cannot guess the other's committed value with a probability greater than $(1/2 + \varepsilon)$.
- *Binding*: Any player cannot change its committed value after the "commit" phase; in other words, any player will be detected with a probability of at least $(1 - \varepsilon)$ if it changes its committed value after the "commit" phase.

The commitment message can be generated by using any one-way function, such as random number generator, one-way permutation, hash function, etc. The use of the one-way function guarantees that the bit-commitment protocol has the above two security properties. In this article, we assume the one-way function is a hash function $Hash(\cdot)$ that belongs to the class of universal hash functions, which has the following property: for a large enough security parameter $m$ and the universe $U$, the probability that two elements, $u, v \in U$, map to the same hashed value (termed as collision) is smaller than $1/m$.

## Commitment-Based Spectrum Contention

We can easily formulate a $k$-system spectrum contention process in the context of 802.22 as a *coin-flipping game* among $k$ players.

- Every system involved in the $k$-system spectrum contention is a player that is assigned a unique token (or a unique ID) $i \in \{0, 1, \ldots, (k-1)\}$.
- The player with token $i$ has a "fair" coin that randomly generates a local contention number (CN), $C_i$, that is uniformly distributed in the range $\{0, 1, \ldots, (k-1)\}$.
- The global coin $C = f(C_0, \ldots, C_i, \ldots, C_{k-1})$ indicates the winner token, where

$$f: \{0, 1, .., (k-1)\}^k \rightarrow \{0, 1, \ldots, (k-1)\}$$

is the pre-specified group decision function.

To solve this problem, we devise a *commitment-based spectrum contention* protocol that makes it difficult for any player to manipulate the contention result. Figure 4 illustrates the commitment-based spectrum contention process among three players (802.22 networks). The protocol includes the following two phases of message exchanges.

*Commitment Broadcast Phase* — Suppose system $i$ is one of the $k$ players that contend for the same channel. It selects its local CN, $C_i$, from $\{0, 1, \ldots, (k-1)\}$ at random, and constructs its commitment message to be broadcasted as follows:

$$[Hash\ ([X_i, Y_i, C_i]), Y_i],$$

where $X_i$ and $Y_i$ are two random bit-strings selected by system $i$ for this contention process.

The commitment message is composed of two parts: a hashed value $Hash\ ([X_i, Y_i, C_i])$, and a public string $Y_i$. Making string $Y_i$ public will help increase the difficulty for system $i$ to manipulate the game later by reconstructing another commitment message that contains the identical hashed value as $Hash\ ([X_i, Y_i, C_i])$ but has a different CN, $C_i'$.

*Contention Resolution (Reveal) Phase* — After receiving $(k-1)$ commitment messages from other systems, system $i$ reveals its original message by broadcasting $[X_i, Y_i, C_i]$. By collecting revealed messages from other $(k-1)$ systems, every system is able to calculate the winner token $w$ as follows.

$$w = \sum_{i=0}^{k-1} C_i \pmod{k}.$$

The system that possesses the token $w$ wins the contention.

## Robustness to CN Manipulation

In the above commitment-based spectrum contention protocol, the use of hash function in the commitment message makes it difficult for a player to manipulate the CN value in the revealed message without being detected.

Moreover, every system in a $k$-system spectrum contention process defined by the above protocol can win the channel contention with a probability of $1/k$. Let $C_0, C_1, \ldots, C_{k-1} \in \{0, 1, \ldots, (k-1)\}$ be independent random variables that represent the contention number of all contending BSs. It is easy to show that: if there exists one random variable $C_i$ that is uniformly distributed over the set $\{0, 1, \ldots, (k-1)\}$, then the random variable $w = \sum_{i=0}^{k-1} C_i \pmod{k}$ is uniformly distributed over the set $\{0, 1, \ldots, (k-1)\}$. This implies that the probability that any token is selected as the winner token is $1/k$ if there is at least one honest BS that randomly selects its CN from the set $\{0, 1, \ldots, (k-1)\}$. Since every BS possesses exactly one token, it wins the channel contention by an exact probability of $1/k$. Therefore, the manipulation of CN values by any system cannot bias the contention result under the commitment-based spectrum contention protocol.

## Selfish Channel Negotiation

In multi-hop CR networks, misbehaviors by a selfish node include a diverse range of demeanors that degrade a network's overall performance. Regarding the distributed channel negotiation protocol, a selfish node's motive is to gain an unfair advantage in terms of maximizing its channel access for its own data packet transmission, while minimizing its energy dissipation of serving others' packets.

### Description of the Misbehavior

The fairness of spectrum sharing mechanisms depends on the cooperation of neighboring CR nodes. A selfish CR may refuse to forward data packets except its own so as to save its resources such as energy and bandwidth.

In the context of distributed channel negotiation, a selfish node can readily conceal available channels from others and reserve them for its own use by providing an empty set of free channels, or a set of free channels that has non-intersection with the set of free channels given by a neighboring sender node. We use the term selfish channel negotiation (SCN) misbehaviors to refer to such a threat.

We illustrated this threat in Fig. 2. Suppose node C is selfish, and it is an intermediate node on one of the routing paths from node S to node D. Node C carries out the SCN misbehavior by providing its neighboring node (node A) a fraudulent free channel list (i.e. reported $FCL_C$ in the figure). As a result, node A cannot find a common channel from two free channel lists, $FCL_A$ and the reported $FCL_C$. Furthermore, node A cannot verify the legitimacy of this reported $FCL_C$ because it cannot overhear all transmissions within the reception range of node C.

By the SCN misbehavior, the selfish node (node C) is successful in misleading the neighbor (node A) to choose another route (via node B) to forward the neighbor's packets. Meanwhile, the selfish node can monopolize the local spectrum and save its resources.

### Analysis of the Vulnerability

The previously mentioned channel negotiation process has vulnerabilities that can be exploited to carry out SCN misbehaviors.

- First, the *spatial* variability exists in spectrum availability at different locations. As a result, the legitimacy of content in the free channel list provided by a CR cannot be easily verified by a neighboring CR at a different location.
- Second, the *temporal* variability in spectrum availability also exists, which also leaves a door open for a selfish receiver to lie during the channel negotiation process.

In other words, there is no easy way for a CR to know whether the free channel list of a neighboring CR is fraudulent or not. Deploying the geolocation database in multi-hop CR networks cannot fully address the problem. The database only provides the information regarding fallow spectrum not used by nodes of TV broadcast systems, which may not be consistent with the actual spectrum opportunities when low-power primary users (e.g. Part 74 devices in 802.22 [8]) are present.

## Defending Against Selfish Channel Negotiation

To counter the selfish channel negotiation misbehaviors effectively, we have to stimulate a CR to honestly share its available channels with other neighbors. In a channel negotiation process, sharing one's available channels with neighboring CRs may lead to its potential operations of packet forwarding for the benefit of neighbors, which incurs costs such as consumption of energy and bandwidth. In this sense, the selfish channel negotiation problem in multi-hop CR networks is equivalent to the selfish packet forwarding problem in multi-hop wireless networks.

Incentive (e.g. reputations or payments from other nodes) is the key to stimulate cooperative packet forwarding. If there is no incentive for sharing one's available channels, all CR nodes become free-riders and no packet forwarding is feasible. A CR node intends to contribute to forwarding if it can obtain reputations or payments from other nodes in compensation for its expense of forwarding a packet.

### Incentive-Based Channel Negotiation

We can borrow ideas from the existing body of research on incentive mechanisms regarding cooperative packet forwarding for the design of such an incentive scheme in the CR channel negotiation process. There are basically two approaches to motivate nodes: by denying service to misbehaving nodes on the basis of a *reputation* mechanism, or by rewarding honest nodes using a *payment* scheme.

- A well-known secure routing scheme proposed in [10] uses a "watchdog" module for reputation maintenance and a "pathrater" module for applying reputation information to routing.
- A payment system is another way to motivate cooperation [11]. If a node wants to send its own packets, it has to pay for the service; meanwhile, if the node forwards a packet for the benefit of another node, it is rewarded.

Since the SCN problem and the selfish packet forwarding problem are equivalent, we can establish an incentive mechanism for channel negotiation in multi-hop CR networks based on the idea of a credit-based payment system [11].

- A node will receive a credit over a channel from a neighbor, if it offers a free channel to forward the packets of that neighbor.
- A node has to pay a neighbor node using its credit, if it has packets to transmit over a channel where the channel access requires the consent and cooperation of the neighbor.

Similar to the system architecture adopted in [11], a central authority or a trusted third party is required to maintain the credit balance for each node. A node can report to the authority using a short message called a "receipt" indicating which packets it has forwarded. In response, the authority will assign an appropriate amount of credit to the node if its receipt is valid — i.e. the packets forwarded by the node have been successfully received by its successor node in the path of the packet traversal.

Such a scheme ensures that a cooperative CR receives credits that can be used to pay other nodes for its requirement of packet forwarding. As a result, every node will have the incentive to share the local channel access opportunities.

### Game-Theoretic Approaches

Most incentive mechanisms are simply heuristics that help establish a cooperation enforcement scheme. The game theoretic approach, on the other hand, provides a formal description of the cooperation problem, and analyzes whether the incentive mechanisms are needed for enabling cooperation [12].

The channel negotiation process between two neighboring CR nodes can be formulated as a two-player channel negotiation game. Suppose there is at least one channel that is common to both of CR nodes, and either CR node has packets to send via the other one. In this two-player game, nodes at different locations may take various strategies to achieve cooperation.

Cooperation may emerge without incentive techniques when nodes are *mutually dependent* — i.e. two nodes rely on the other to forward its own packets. In this case, there are two options for a CR to play the two-player channel negotiation game:

- To selfishly report a fraudulent free channel list.
- To honestly provide the unchanged free channel list that includes at least one channel available to both CRs.

Take the Tit-For-Tat (TFT) strategy for example. A node playing this strategy starts with cooperation, and then mimics the behavior of its opponent in the previous round. Under the TFT strategy, a selfish CR has to cooperate, because once it starts misbehaving, its opponent CR in the game will take the same misbehavior and its own packets cannot get served. Indeed, the TFT strategy creates a punishment method for enforcing the selfish nodes to cooperate.

The two-player game can also happen between a network boundary node and an inner (e.g. backbone) node in the middle of the network. In this case, the inner node is *not dependent* on the boundary node for packet forwarding, and it is not concerned about the punishment for not forwarding the boundary node's packets. An approach based on cooperative game coalitions has been proposed to solve this problem [13]. Boundary nodes can provide benefits for the inner nodes to stimulate cooperation, and the boundary nodes also receive rewards for packet-forwarding. Specifically, suppose a boundary node forms a coalition with one of its closest inner nodes, and helps forward the inner node's packets (e.g. to help reduce the transmitted power of the inner node). In this scenario, the inner node is motivated to help forward the boundary node's packets to return the favor. This approach incentivizes inner nodes to serve boundary nodes, which is similar to the incentive mechanism employed by the payment scheme mentioned above.

## Summary

In this article we identified and discussed two types of selfish misbehaviors against spectrum sharing mechanisms in CR networks: selfish spectrum contention and selfish channel negotiation. There are other types of selfish misbehaviors in CR networks, including selfish sensing, selfish routing, etc. All of these misbehaviors pose a potential threat to fair spectrum access in CR networks. However, in this article we limited our discussions to MAC-layer selfish misbehaviors against spectrum sharing mechanisms. We also presented effective countermeasures to the two previously-mentioned selfish misbehaviors.

## References


[1] R. Chen *et al.*, "Toward Secure Distributed Spectrum Sensing in Cognitive Radio Networks," *IEEE Commun. Mag.*, vol. 46, no. 4, Jan. 2008, pp. 50–55.

[2] C. Song and Q. Zhang, "Achieving Cooperative Spectrum Sensing in Wireless Cognitive Radio Networks," *ACM MC2R*, vol. 13, no. 2, Apr. 2009, pp. 14–25.

[3] Y. Liu, P. Ning, and H. Dai, "Authenticating Primary Userss' Signals in Cognitive Radio Networks via Integrated Cryptographic and Wireless Link Signatures," *Proc. IEEE Symp. Security and Privacy*, May 2010.

[4] W. Wang *et al.*, "CatchIt: Detect Malicious Users in Collaborative Spectrum Sensing," *Proc. IEEE Globecom*, Nov. 2009.

[5] O. Fatemieh *et al.*, "Using Classification to Protect the Integrity of Spectrum Measurements in White Space Networks," *Proc. NDSS*, Feb. 2011.

[6] S. Li *et al.*, "Location Privacy Preservation in Collaborative Spectrum Sensing," *Proc. IEEE INFOCOM*, Mar. 2012.

[7] Z. Gao *et al.*, "Location Privacy Leaking from Spectrum Utilization Information in Database-Driven Cognitive Radio Network," *Proc. ACM CCS*, Poster, Oct. 2012.

[8] IEEE 802.22 Working Group, http://www.ieee802.org/22/.

[9] A. Z. Broder and D. Dolev, "Flipping Coins in Many Pockets (Byzantine Agreement On Uniformly Random Values)," *Proc. Symp. Foundations of Computer Science*, Oct. 1984.

[10] S. Marti *et al.*, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," *Proc. ACM MobiCom*, Aug. 2000.

[11] S. Zhong, Y. Yang, and J. Chen, "Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad Hoc Networks," *Proc. IEEE INFOCOM*, Mar. 2003.

[12] M. Felegyhazi, J.-P. Hubaux, and L. Buttyan, "Nash Equilibria of Packet Forwarding Strategies in Wireless Ad Hoc Networks," *IEEE Trans. Mobile Computing*, vol. 5, no. 5, May 2006, pp. 463–76.

[13] Z. Han and H. V. Poor, "Coalition Games with Cooperative Transmission: A Cure for the Curse of Boundary Nodes in Selfish Packet-Forwarding Wireless Networks," *IEEE Trans. Commun.*, vol. 57 no. 1, Jan. 2009, pp. 203–13.


## Biographies


KAIGUI BIAN (bkg@pku.edu.cn) received his Ph.D. degree in computer engineering from Virginia Tech, Blacksburg, USA in 2011. He is currently an assistant professor in the Institute of Network Computing and Information Systems, School of EECS at Peking University. His research interests include mobile computing, cognitive radio networks, network security and privacy.

JUNG-MIN "JERRY" PARK [SM] (jungmin@vt.edu) received the Ph.D. degree in electrical and computer engineering from Purdue University, West Lafayette, Indiana. He is an associate professor in the Department of Electrical and Computer Engineering at Virginia Tech, where he is the Site Director of an NSF I/UCRC Center, Broadband Wireless Access & Applications Center (BWAC), and the founding Director of the research lab, Advanced Research in Information Assurance and Security (ARIAS). He is a recipient of a 2008 NSF CAREER Award, a co-recipient of the 2008 Hoeber Excellence in Research Award, and a recipient of the 1998 AT & T Leadership Award. He is a Senior Member of the ACM. His research interests include cognitive radio networks, coexistence and co-design of heterogeneous wireless systems, wireless networking, network security, and cryptographic protocols. His research has been supported by NSF, DARPA, ONR, and various industry sponsors.

XIAOJIANG DU [SM] is an associate professor at Temple University. He received his B.E. degree from Tsinghua University, China in 1996, and his Ph.D. degree from the University of Maryland, College Park in 2003, all in Electrical Engineering. His research interests are security, systems, wireless networks and computer networks. He has published over 100 journal and conference papers in these areas. He is a Life Member of ACM.

XIAOMING LI received his Ph.D. in computer science from Stevens Institute of Technology (USA) in 1986 and now is a Professor at Peking University. His research interests include Web search and mining, online social network analysis. He is an editor of Concurrency and Computation, and an editor of Networking Science.