

## Research Article

# A Fully Distributed Resource Allocation Mechanism for CRNs without Using a Common Control Channel

**Adil Mahmud, Youngdoo Lee, and Insoo Koo**

*School of Electrical and Computer Engineering, University of Ulsan, Ulsan 680-749, Republic of Korea*

Correspondence should be addressed to Insoo Koo; [iskoo@ulsan.ac.kr](mailto:iskoo@ulsan.ac.kr)

Received 18 June 2014; Accepted 11 October 2014

Academic Editor: Jong-Hyuk Park

Copyright © 2015 Adil Mahmud et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Spatial and temporal variations in the available spectrum for cognitive devices add additional complexity in designing a medium access strategy for cognitive radio (CR) networks. To ease this complication, a common control channel (CCC) is often used as a common platform to exchange control messages, so the CR users can make decisions concerning resource allocation. Much of the literature in this area opts for forming groups and negotiating a CCC from the available channels rather than considering a dedicated common control channel. However, if a primary user appears on the CCC, all the CR users have to vacate that channel and negotiate another CCC, and thus overhead continues to grow with an agile primary network. We propose a medium access control protocol that can work in the absence of a CCC and reduce the possible overhead to a greater extent. In our proposed protocol, CR users take advantage of similar spectrum availability in their neighborhood for resource utilization. We also propose a contention-based spectrum allocation mechanism that works in a distributed manner over different available channels. Simulation results show that this approach can reduce broadcast overhead significantly while maintaining connectivity success similar to its counterparts.

## 1. Introduction

Immense market demand for wireless communications technologies, including video streaming and other applications, precipitates spectrum scarcity. But it has been observed that the static nature of users limits the ability to dynamically use the whole spectrum at any given time. Statistics show that, in some cases, spectrum usage even falls below 5–10% [1]. In such situations, new users are deprived of spectrum resources, which is absolutely unreasonable.

The ideas of cognitive radio (CR) and dynamic spectrum access bring a promising opportunity to solve the spectrum scarcity problem [2]. A CR device, also called a secondary user (SU) or simply a node, has a cognitive capability with which it obtains information about the networks surrounding it and can opportunistically access unused spectrum without interfering with licensed users, also known as primary users (PUs) [2, 3]. This opportunistic spectrum utilization is commonly known as spectrum sharing. Spectrum sharing among SUs is a major issue to focus on in a CR network (CRN).

There are many solutions to medium access control (MAC) for traditional sensor and ad hoc networks. Some of those are centralized protocols [4–8] and some are decentralized [9]. But none of those protocols are directly applicable to a CRN because of different spectrum etiquette. Nevertheless, the advancement in MAC mechanisms for a CRN is worth mentioning. A number of proposals with a variety of unique ideas have already emerged so far for CRNs. Those proposals comprise both centralized [10, 11] and decentralized approaches [12, 13], which may be further categorized on the basis of common control channel (CCC) usage. Though the centralized network approach using a dedicated common control channel (DCCC) simplifies the task of spectrum management, assignment, and utilization, it has been proven impractical in a CRN.

Keeping the above-mentioned constraints in mind, several research articles have been proposed, primarily aimed at reducing coordination overhead while maintaining spectrum management efficiency at the level of centralized solutions. One group of such proposals considers choosing a CCC from

among the available channels in a neighborhood in designing distributed MAC protocols for CRN [12, 14]. Despite the remarkable success over their centralized counterparts, as well as over the protocols using a DCCC, the protocols that consider a CCC or coordination channel have drawbacks in CR application systems. For example, a CCC can be a viable center of attack for an intruder. Moreover, when a PU appears on the CCC, all the nodes using that channel as a CCC must leave it and negotiate for establishing a new CCC. As a result, this approach is vulnerable and, therefore, inappropriate in a CRN when PU activity is high. To solve the problems mentioned above, some researchers opted to design MAC protocols without using any sort of CCC. For example, Timalisina et al. [15] proposed a protocol that minimizes wait time and delay but uses two radios. As a result, their solution increases hardware cost.

In addition, a distributed broadcast protocol for a multi-hop CR ad hoc network was proposed by Song and Xie [16] in which a channel rendezvous technique employing broadcasts instead of using a CCC was adopted.

In this paper, we propose a resource allocation mechanism that does not require a CCC. For this work, we were motivated by the fact that neighboring CR users generally experience similar spectrum availability. Zhao et al. [14] also showed the degree of correlation in spectrum availability among neighbors in their experiments. In their research, a coordinated scheme was developed where CR nodes in a small neighborhood select the most commonly available channel among them as a coordination channel, and the transmission pairs utilize that channel to negotiate data transmission channels. However, a coordinated scheme still requires additional control overhead to establish a coordination channel. Instead, in this paper, we propose a unique contention-based method, followed by each CR user, to reduce the extra overhead caused by the coordination channel selection process. Another advantage of our approach is that contention for occupying a channel is distributed over the available channels in the vicinity. However, in such a unique contention-based method, the prime issue will be how to allocate resources (i.e., channels) to users. Therefore, we also propose a channel acquisition scheme to address this issue where the basic concept of carrier sense multiple access (CSMA) is adopted as a tool.

The rest of the paper is arranged as follows. In Section 2, we present assumptions and the system model. The proposed protocol is described in detail in Section 3. Simulation results are given in Section 4. Finally the paper is concluded in Section 5.

## 2. Assumptions and the System Model

In this section, we first mention the assumptions we make in designing the protocol, and, then, we describe the system model.

*2.1. Assumptions.* In this paper, we refer to the licensed network as the primary network and the unlicensed network as the secondary network. We assume there are  $N$  uncorrelated channels in the primary network. Let the set of channels

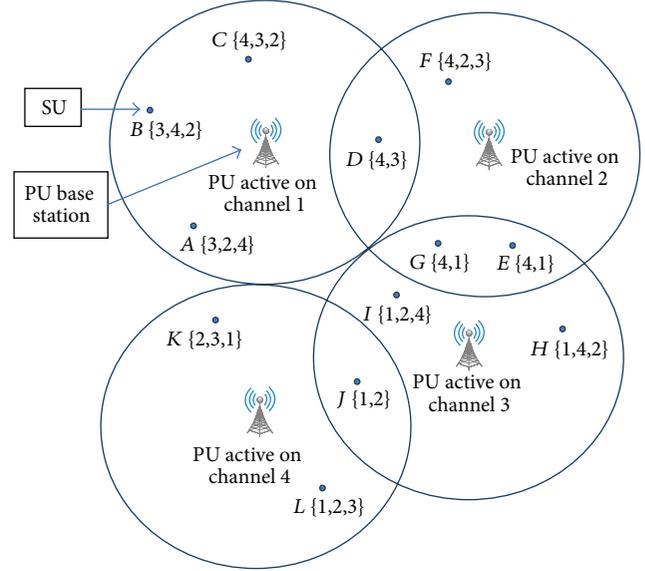


FIGURE 1: An example of CR network topology.

for the primary network be  $C$  ( $C = \{1, 2, \dots, N\}$ ). Each SU is equipped with a single radio having cognition capability and holding predefined characteristics such as transmission range, transmit power, and modulation. We also assume that SUs can sense the channels with high reliability. How they sense channels with high reliability is beyond the scope of this work. In our proposed model, each of the SUs can sense and utilize a number of channels available at its position. No two SUs within transmission range of each other can use the same channel. The SUs can be static or quasistatic, which means channel availability does not change frequently unless PUs abruptly occupy the channels.

*2.2. System Model.* Figure 1 shows a simple exemplary CR network suitable for explaining the proposed scheme. There are four primary network base stations serving their users (PUs) on different channels (channels 1–4). At the same time, several SUs, denoted as  $A, B, C$ , and so forth, reside inside the primary network. According to the regulations of the opportunistic spectrum access policy, no SU can use a channel such that the SU interferes with any of the PUs.

In this paper, SUs operate in overlay mode, which means that an SU will use a channel if there is no PU activity on that particular channel. The available channels for SUs are indicated within the curly brackets beside each of the nodes in Figure 1. For example, the available channel set for SU  $A$  is denoted as  $\{3, 2, 4\}$ . Generally, the available channel set of an SU is expressed as follows:

$$A_i = \{c \mid c \in C \text{ and } d_{i,c} > t_b\}, \quad (1)$$

where  $A_i$  is the available channel set of node  $i$ ,  $d_{i,c}$  is the distance of node  $i$  from a PU or a primary network base station that is currently using channel  $c$ , and  $t_b$  is the radial distance covered by a primary base station. For convenience, the coverage area of a node or a base station is assumed to be

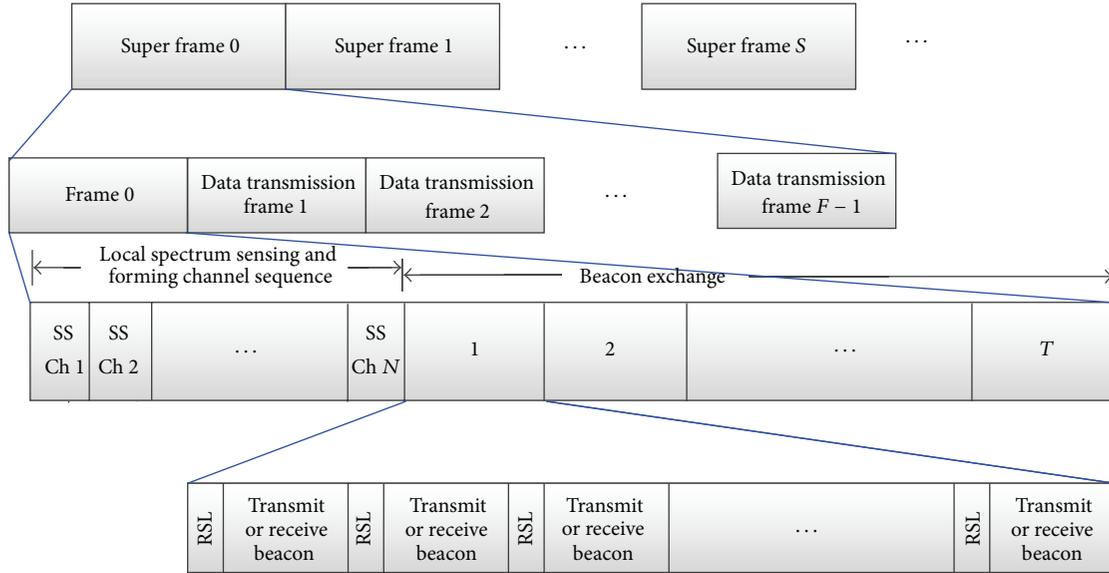


FIGURE 2: The frame structure of the proposed scheme where each super frame is composed of frame 0 (used for spectrum sensing and discovering neighbors) and frames 1 to  $F - 1$  (used for data transfer).

circular. We further assume that two SUs can communicate only if they are within transmission range of each other. We denote the transmission range of an SU by the symbol  $t_s$ , which is the radius of the circular area it can cover. Moreover, we consider  $t_b > t_s$ , which is a common assumption in the literature.

### 3. General Frame Structure

In the proposed framework, time is divided into super frames; each super frame consists of  $F$  frames. This sort of structure is used in the IEEE 802.22 wireless regional area network (WRAN) MAC framework [12]. Though the exterior construction is similar to that of the WRAN structure, we organize the frames in a different way to fit the proposed protocol as follows: frame 0 is used for discovering the available channels of neighbors. For this, the first part of frame 0 is used for spectrum sensing and the rest of it is used for beacon exchange. A number of papers consider node information or beacons to be exchanged among all the neighbors but do not give any explanation about the way to exchange this information [14, 17, 18]. Therefore, in Section 3.2, we propose a beacon exchange method. The other frames of a super frame, that is, frames 1 to  $F - 1$ , are called *data transmission frames* in this paper. These frames are used for transmitting data. Because nodes should be allocated conflict-free channels to communicate with fairness, we further divide these frames into a number of parts to achieve these goals. We will explain data transmission frames in detail in Section 3.3.

**3.1. Spectrum Sensing.** At the beginning of frame 0, each node sequentially senses all  $N$  channels to check channel availability. For decision-making, the nodes can use any detection method, such as energy detection, which is what we propose

in this work. The spectrum decision is obtained according to the following well-known equation:

$$x(t) = \begin{cases} n(t), & H_0, \\ hs(t) + n(t), & H_1, \end{cases} \quad (2)$$

where  $h$  denotes gain in the channel between PU and SU,  $s(t)$  denotes the signal transmitted by the PU, and  $n(t)$  denotes additive noise for the SU.

After sensing all channels, a node creates a channel sequence for an available channel set based on the received normalized signal energy according to the following equation:

$$E_c^i = \frac{(\sum_{j=1}^{N_s} |x_c(j)|^2) / N_s}{\max \{|x_c(j)|^2\}}, \quad (3)$$

where  $E_c^i$  is the normalized energy received on the  $c$ th channel by the  $i$ th node,  $x_c(j)$  is the  $j$ th sample of the received PU signal on the  $c$ th channel, and  $N_s$  is the number of samples received during each sensing interval. The lower the received energy of a channel, the higher its position in the channel sequence. If required, a node hops from one channel to another following this channel sequence. Hence, we can also define the channel sequence as the hopping sequence of a node. Next, nodes need to know their neighboring nodes' information, specifically their channel sequences. This is done by beacon exchange, which is described next.

**3.2. Beacon Exchange.** Nodes are required to discover their neighbors as well as to know each other's channel sequences. To do this, the second part of frame 0 is divided into  $T$  slots, as shown in Figure 2. During the first slot, a node tunes itself to its first channel, which is also called the *best channel* in this

paper, according to its channel sequence. Then in the second slot, it moves to the second channel according to its channel sequence and so on. In each of the slots, a node resides on a particular channel and broadcast beacon, which contains its ID, channel sequence, its neighboring nodes' IDs, and their channel sequences.

Two nodes become direct neighbors (DNs) of each other if they exchange beacons. A neighbor, for which node ID and channel sequence are obtained from one of the DN's, is categorized as a neighbor-of-a-neighbor (NoN) node. A node keeps separate records for DN's and NoN's. An NoN becomes a DN if the nodes directly exchange beacons at some stage.

In a particular slot, in order to avoid collision among the nodes on a channel, each node randomly selects a *random short listening* (RSL) time and listens on the channel if a node has already started beaconing. The node with the shortest RSL time broadcasts its beacon, while the other nodes on the channel receive it. After a node successfully broadcasts its beacon in a slot, it goes into a listening state and waits to receive other nodes' beacons during the rest of the time in that slot. The remaining nodes on that channel follow the same mechanism mentioned above. However, if the number of available channels of a node is less than  $T$ , it repeats broadcasting beacons on its available channels in the order of its channel sequence. Since the number of slots should be limited to meet the frame size constraint, it is important to formulate an efficient rendezvous method so that most, if not all, of the nodes within transmission range of a node can exchange beacons. However, in this study, we assume that all the neighbors within transmission range are discovered, either as a DN or as an NoN.

To illustrate the beacon exchange process, we take an example where there are three channels,  $c_1$ ,  $c_2$ , and  $c_3$ , and five nodes within transmission range of each other:  $n_1\{c_1, c_2\}$ ,  $n_2\{c_1, c_3\}$ ,  $n_3\{c_1\}$ ,  $n_4\{c_2, c_3\}$ , and  $n_5\{c_1, c_2\}$ . Here, channel numbers enclosed in curly brackets after each node represent the channel sequence of a node. During the first subslot, we have nodes  $n_1$ ,  $n_2$ ,  $n_3$ , and  $n_5$  on channel  $c_1$  whereas node  $n_4$  is on channel  $c_2$ . Node  $n_4$  is alone on channel  $c_2$ , so it broadcasts its beacon after waiting for its RSL time but receives no neighbor beacon. On the other hand, all four nodes on channel  $c_1$  wait for their respective RSL times. Let  $n_1$  be the winner in the contention to transmit first, which means it has the shortest RSL time. So  $n_1$  broadcasts its beacon while other nodes receive it. After broadcasting its beacon, a node goes into a listening state to receive the beacons of other nodes. The other nodes ( $n_2, n_3, n_5$ ) exchange their beacons in the same way. During the second subslot, nodes  $n_1$  and  $n_5$  move to channel  $c_2$ , nodes  $n_2$  and  $n_4$  move to channel  $c_3$ , and node  $n_3$  keeps repeating its broadcast on channel  $c_1$ . It is important to note here that node  $n_4$  can make a direct neighborhood only with node  $n_2$ . But other nodes will learn about  $n_4$  through  $n_2$ . With the help of our proposed data communications mechanism described in Section 3.3, the other nodes will be able to communicate with node  $n_4$  using  $n_2$  as a relay. In Section 3.3, we address the process of accessing channels and the overall data communications mechanism among nodes.

**3.3. Data Communication.** Henceforth, we call a node that has data to send the *sender* and a node that is supposed to receive data the *receiver*. As mentioned earlier, frame 1 to frame  $F-1$  of a super frame will be used for data transmission, so we call these frames data transmission frames. A data transmission frame is divided into three segments. Moreover, the way of transmitting in a data transmission frame differs based on neighborhood status between sender and receiver (i.e., DN or NoN) and, similarly, on channel availability. In Section 3.3.1, we discuss the transmission frame structure, and, in Section 3.3.2, we discuss transmission cases.

**3.3.1. Transmission Frame Structure.** A transmission frame is divided into three parts: contention, transmission, and reporting. Figure 3(a) depicts such a frame structure. In the following parts (1), (2), and (3), we discuss details for each part.

(1) *Contention.* The contention period is divided into  $M$  slots. At the start of each slot, nodes perform fast sensing (FS), a technique similar to the one used in a WRAN [6]. With this approach, a node can recheck in a particular frame for the activity of PU on the channel where it resides. This provides extra protection against interference with primary users. FS plays a vital role in a primary network where PUs are very active.

We assume that not all nodes have data to send in a particular frame. There are some nodes that have data to send, while some nodes do not. Among the nodes that have no data to send, some are receiving nodes, whereas the other nodes may be used as relay nodes. A node that has no data to send always prefers to tune itself to the best channel among its available channels in any contention time slot. Here, an available channel means that the channel of concern is occupied by neither PU nor SU in that time slot. So, if a channel is occupied by a transmission pair, all the nodes having no data to send move to the next channel in their respective channel sequences. On the other hand, a sender follows its intended receiver's available channel sequence. If a node finds there is no active PU on a particular channel, the node waits for the RSL time to avoid collision with other contenders.

If the sender wins contention to use the channel (i.e., the sender has the shortest RSL time), it sends a transmission request to send (TRTS) message to the receiver, while the receiver responds with transmission clear to send (TCTS), forming a transmission pair. At this stage, the transmission pair starts data transmission without waiting for the actual transmission time to begin while the other contenders on that channel jump to next channels according to their respective receivers' channel lists. Such a scenario is depicted in Figure 3(b). The benefit of starting transmission right after the occupation of a channel is twofold. First, the transmission pair gets more time for data transmission, which increases total throughput. Second, if there is no activity until the actual transmission time starts, other nodes might wrongfully determine that the channel is vacant if they somehow do not receive, or miss, the TRTS-TCTS messages.

It is also possible that a sender is unable to occupy a common channel between itself and its intended receiver because

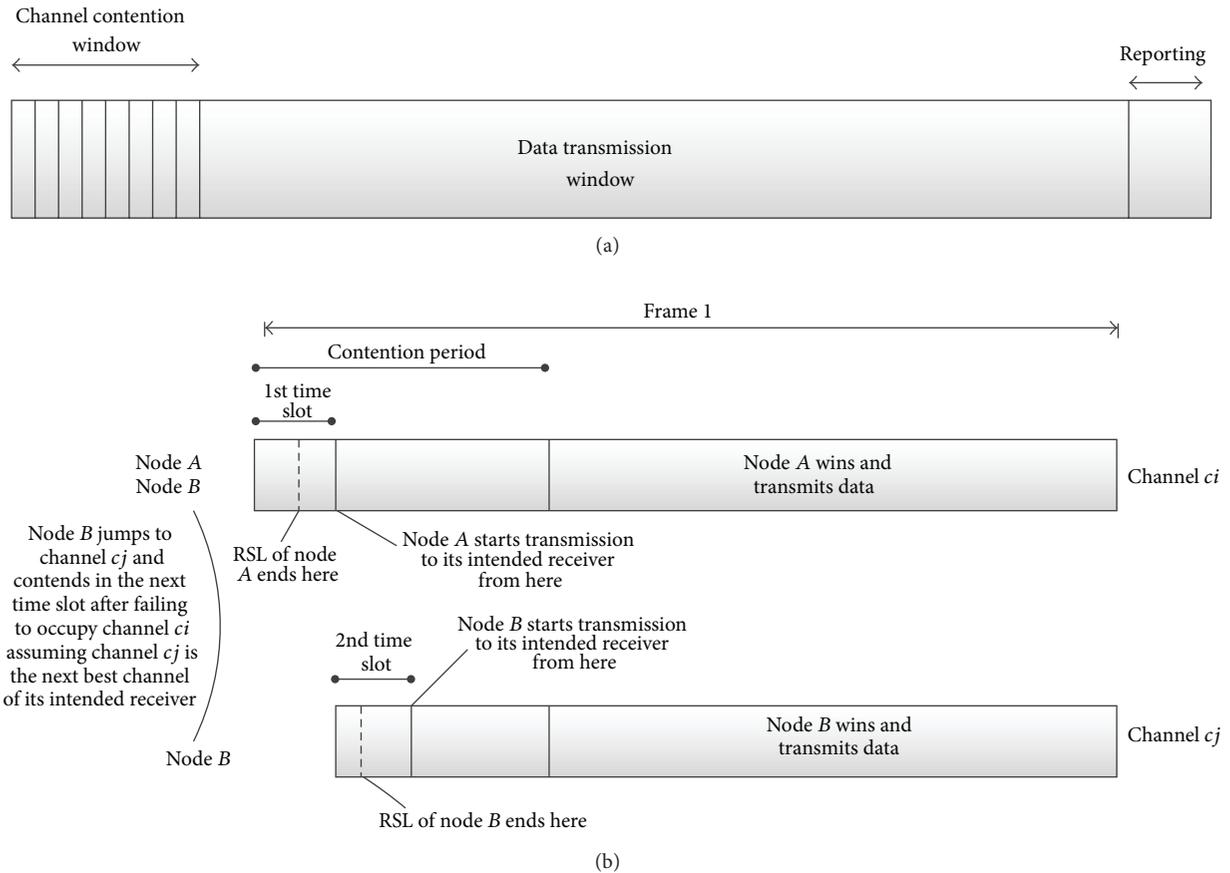


FIGURE 3: (a) Structure of data transmission frame and (b) the proposed contention-based channel occupation method.

of repeated failure during the contention period. We call such a node an *unsuccessful sender*. The unsuccessful sender returns to its intended receiver’s best channel after the end of the contention period.

(2) *Transmission*. The data transmission slot starts just after the designated contention period ends, although the winning transmission pairs commence communications before the start of the transmission slot, as described in Section 3.3.1(1). The start of the transmission frame signifies that the contention period ends, and, henceforth, no node is able to contend for any channel for transmission until the end of the frame. Senders that could not win a channel, or could not find a relay node, will quietly wait on their intended receivers’ best channels until the end of the transmission period.

(3) *Reporting*. During the reporting slot, prospective senders for the next frame will tune themselves to their receiver’s best channel. An unsuccessful sender will broadcast an “unsuccessful” message to notify nodes about its failure in the current frame so that it gets priority during the contention period of the next frame. This reporting mechanism will help reduce delays. However, the mechanism for giving priority to a node over others is not discussed in this study and is left for future research.

3.3.2. *Transmission Cases*. As we mentioned before, there are three possible transmission cases, based on channel availability and neighborhood status between sender and receiver. These cases are discussed in the next three parts (1), (2), and (3). A flow chart of all the cases is depicted in Figure 4.

(1) *Case 1: Receiver Is a DN, and Sender Has Receiver’s Best Channel*. When a node has data to send to one of its DNs and it has the best channel for that neighbor, the sender first moves to the best channel of that receiver during the first time slot of the contention period. Because there may be several nodes that want to send data in the same time slot and on the same channel, the sender waits for the RSL time to avoid collision and to check whether any other node has already occupied the channel. If no other node occupies the channel within the elapsed RSL time, the node sends TRTS and waits for TCTS from its intended receiver. Once the receiver acknowledges TCTS, communication begins between them. Overhearing these TRTS-TCTS messages, the other nodes leave that channel. However, if the sender fails to win during contention, following the channel sequence of its intended receiver, it switches itself to the next channel, if that channel is available to it. If the channel is not available, the sender moves to the best channel of its receiver and waits until the end of the transmission period. When the reporting time starts,

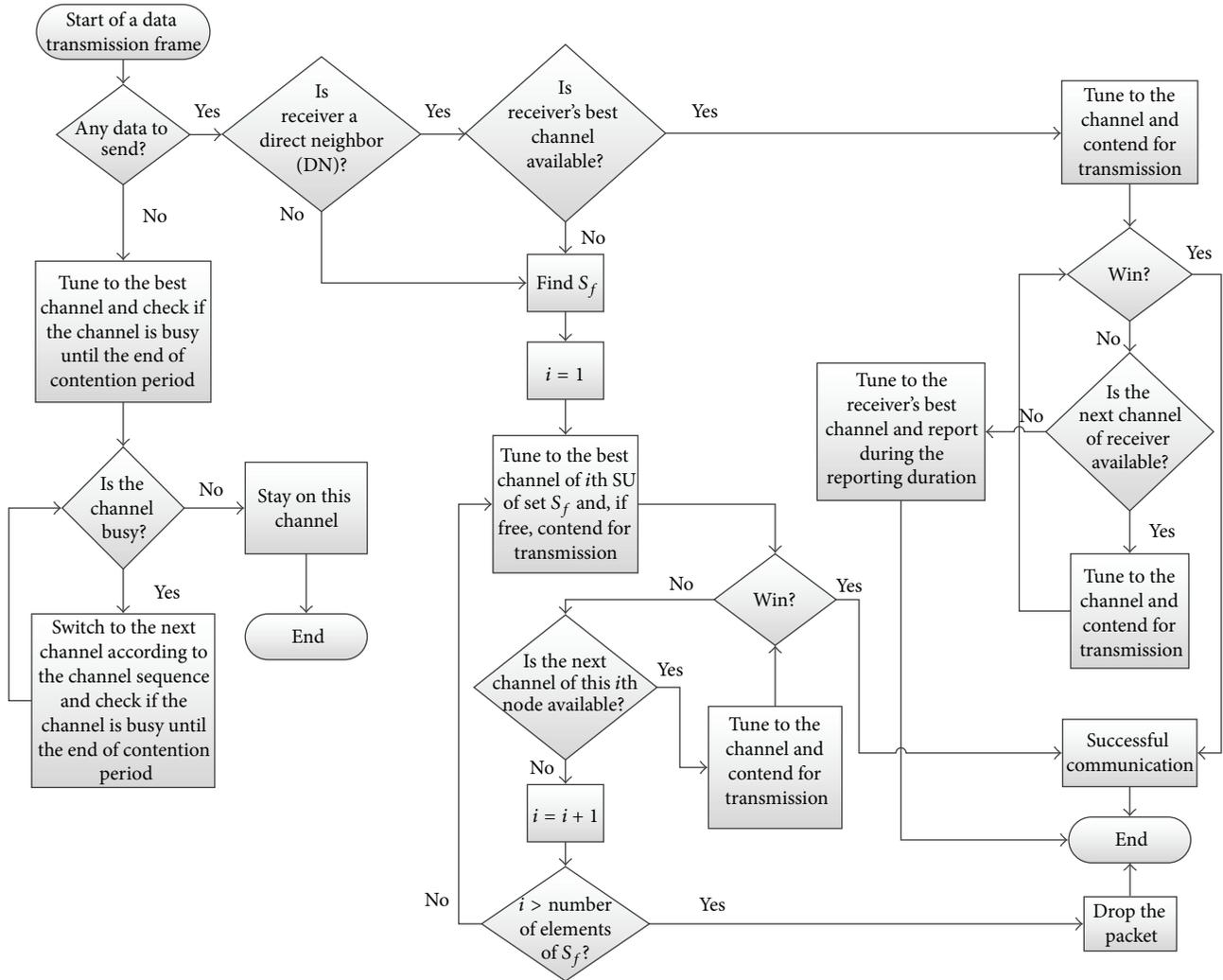


FIGURE 4: Flow chart of data transmission mechanism in the proposed scheme.

the sender broadcasts an unsuccessful message to get priority in the next frame, as mentioned in Section 3.3.1(3).

(2) *Case 2: Receiver Is a DN, but Sender Does Not Have Receiver's Best Channel.* This situation can occur when a node does not have the best channel of a neighbor but meets with it on any other channel during the beacon exchange period. Since the sender does not have the best channel of the receiver, it cannot directly communicate with it. In this situation, the sender relies on any DN that has the best channel of the receiver. Therefore, the sender first finds a set of neighbors,  $S_r$ , with the best channel of the receiver in their channel sequences.  $S_r$  is defined as follows:

$$S_r = \{k \mid \text{The node } k \text{ has the best channel of receiver in its channel list}\}. \quad (4)$$

Next, the sender will find another set of neighbors,  $S_f$ , such that  $S_f \subset S_r$ , and the best channels of all of the neighbors

in  $S_f$  are available on the sender's available channel list. Formally, the definition of  $S_f$  is as follows:

$$S_f = \{m \mid \text{The node } m \text{ is a neighbor of the receiver and sender has the best channel of node } m\}. \quad (5)$$

Finally, the sender sends data to one of the neighbors in  $S_f$  according to the method described in Section 3.3.2(1). The neighbor, also called the relay node, then sends the data to the intended receiver in the next frame, which also follows the method described in Section 3.3.2(1).

(3) *Case 3: Receiver Is One of the NoNs.* A neighbor of a neighbor can be within transmission range of a sender, or it can be outside the range. But this information is not known to the sender. In such a situation, the sender first moves to the best channel of the intended receiver and proceeds with a mechanism similar to that described in Section 3.3.2(1). If the sender does not receive TCTS from the receiver,

TABLE 1: Nodes and their channels at the beginning of a frame.

Channel number	Nodes on the channel
$c1$	$n1, n2, n3, n4, n5$
$c2$	$n6, n7, n8, n9$
$c3$	$n10, n11, n12$
$c4$	$n13, n14, n15, n16$

TABLE 2: Sender-receiver pairs, along with their channel sequences.

Sender	Receiver
$n3 \{c2, c1, c4\}$	$n1 \{c1, c4\}$
$n4 \{c4, c1\}$	$n14 \{c4, c1\}$
$n7 \{c1, c4, c2\}$	$n15 \{c4, c2, c1\}$
$n12 \{c3, c4, c2\}$	$n8 \{c2, c3\}$
$n16 \{c3, c1, c4\}$	$n2 \{c1, c3\}$

it assumes that the receiver is outside of its transmission range. In that case, the sender proceeds with the method described in Section 3.3.2(2).

(4) *An Illustrative Example.* To illustrate the mechanism, let us consider an example where 16 nodes (numbered as  $n1$ – $n16$ ) look for opportunities to communicate within a primary network consisting of four channels ( $c1$ ,  $c2$ ,  $c3$ , and  $c4$ ). Table 1 shows which nodes are residing on which channels at the beginning of a data transmission frame. In Table 2, we consider some sender-receiver pairs. The channel sequences for each of the sender and receiver nodes are provided within curly brackets. According to the protocol, nodes record these channel sequences from the beacon exchange step described in Section 3.2.

According to the mechanism, the nodes that do not have data to send initially stay tuned to their best channels according to their channel sequences. Consequently, receivers  $n1$  and  $n2$  should be on channel  $c1$ , receiver  $n8$  should be on channel  $c2$ , and receivers  $n14$  and  $n15$  should be on channel  $c4$ . At the beginning of a transmission frame, senders  $n3$  and  $n16$  move to channel  $c1$ , senders  $n4$  and  $n7$  move to channel  $c4$ , and sender  $n12$  moves to channel  $c2$ .

During the contention period, let us assume that senders  $n3$  and  $n4$  win use of channels  $c1$  and  $c4$ , respectively, whereas node  $n12$  is the only sender on channel  $c2$ . Soon after winning contention, the sender-receiver pairs exchange TRTS-TCTS messages and transmit their data. Meanwhile, node  $n2$  moves to channel  $c3$ , which is still free (i.e., not yet occupied by an SU). Because a sender knows the next channel on which it should search for the receiver, node  $n16$  moves to channel  $c3$  and communicates with the receiver. On the other hand, node  $n15$  (and, consequently, node  $n7$ , because it wants to send data to node  $n15$ ) moves to channel  $c2$  and then to channel  $c1$  but finds no channel free. In this case, node  $n7$  returns to channel  $c4$  (as it is the best channel of its intended receiver) and broadcasts an unsuccessful message during the reporting time slot.

## 4. Performance Analysis

In this section, we demonstrate and compare the performance of our proposed scheme. To measure performance, we specifically focus on the number of broadcast packets and the connectivity among the nodes. Reduction of broadcast traffic is the main goal of this study, and to achieve this goal, we propose a protocol that does not rely on a common control channel. But without a common control channel, it is difficult to establish connectivity among the nodes and, thus, the resource allocation task becomes complicated. We compare our proposed scheme with the one proposed by Zhao et al. [14] on the basis of the above-mentioned performance metrics, that is, number of broadcast packets and connectivity among nodes. The reason behind choosing the protocol proposed by Zhao et al. [14] is that this protocol is also designed based on spectrum homogeneity in the neighborhood.

4.1. *Simulation Setup.* We use MATLAB to simulate performance and randomly deploy four primary network base stations in a 100-by-100 area. Each base station serves the PUs on one of the four channels, and each channel is considered busy all the time, because we assume that primary users are always communicating with their nearest base stations. So, an SU can only use those channels on which no PU is active inside its sensing range. Within the coverage area of the base stations, we deploy 100 secondary users, but the number of SUs we consider for measuring performance is chosen at random. More specifically, each time we run the simulation, we randomly consider a group of nodes that are at most two hops away from each other. Because the positions of the SUs change each time, their available channel sets (as well as their channel sequences) also change. We argue that, with these sorts of settings, the results offer better credibility. The values for  $t_b$  and  $t_s$  are set to 30 and 20, respectively.

4.2. *Comparison of the Number of Broadcast Packets.* As mentioned before, we compare our proposed protocol with the one proposed by Zhao et al. [14]. Both protocols use the beacon broadcast to discover neighbors and exchange available channel sets. But, unlike the proposal by Zhao et al. [14], we avoid forming coordination channels. Rather, with the available information of the nodes, we propose a contention-based resource allocation mechanism that greatly reduces the number of broadcast packets. This is evident from Figure 5, where we can see that the number of broadcast packets is always higher under the Zhao et al. protocol [14], compared to our proposed scheme.

4.3. *Comparison of Connectivity.* Connectivity is a major issue for a resource allocation mechanism, especially in the absence of any sort of coordination channel or CCC. A CCC or coordination channel provides a common platform for all the nodes to exchange information and negotiate a channel. Since no CCC or coordination channel is considered here, it is important to observe the nodes' rates of success when connecting with each other. To show connectivity performance, we considered two cases: (i) connectivity with nodes within

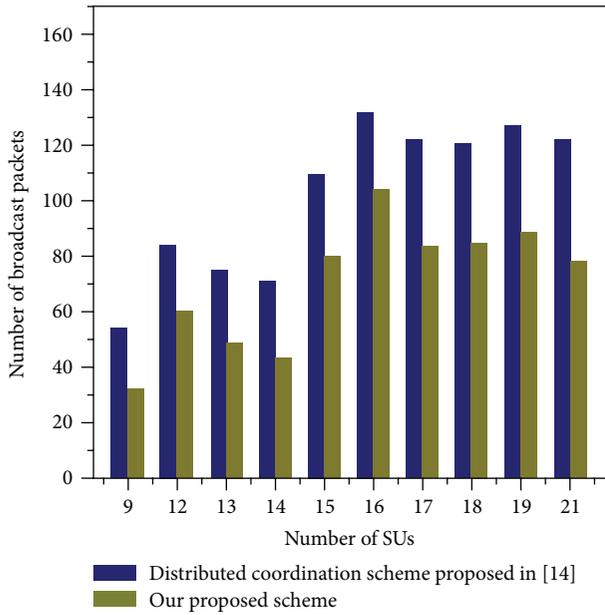


FIGURE 5: Number of broadcast packets when four PU channels are considered, and the number of SUs is chosen at random.

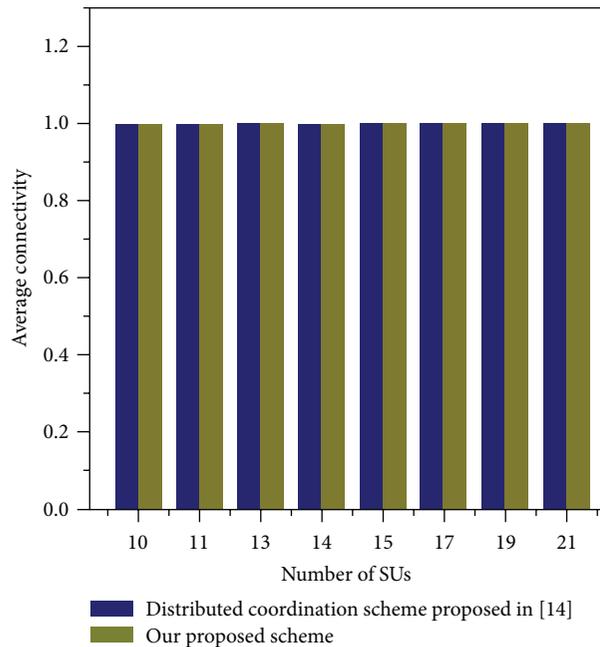


FIGURE 6: Average connectivity, according to the number of SUs, when four PU channels are considered. All receivers are within transmission range of senders. The number of SUs randomly changes between 10 and 21.

transmission range and (ii) connectivity when nodes are two hops away. The simulation result for Case (i) is shown in Figure 6. We can see that the performance is exactly equal under both protocols. Moreover, connectivity is 100% in both cases, which means that, with the current settings, any node

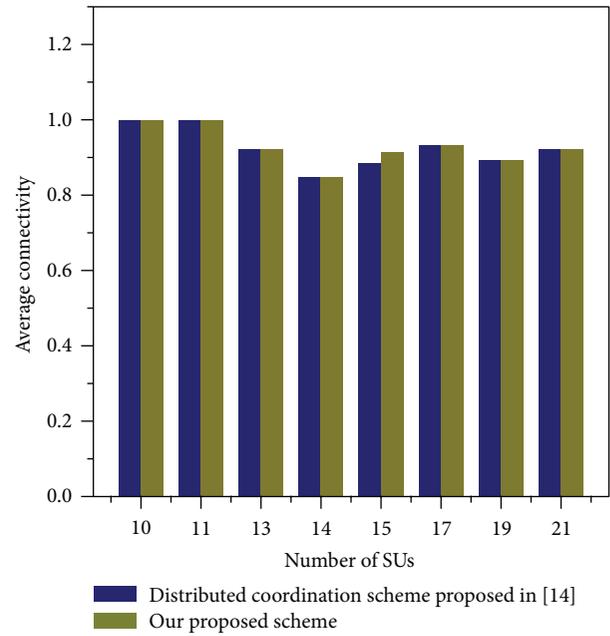


FIGURE 7: Average connectivity, according to the number of SUs for a multihop (two-hop) scenario when four PU channels are considered. The number of SUs randomly changes between 10 and 21.

can connect to any other node within its transmission range. The reason is that nodes within transmission range have very similar spectrum availability. In Figure 7, we can see the average connectivity in Case (ii). It is clear that connectivity is almost the same for both schemes, except for some minimal differences. This signifies that, with two-hop nodes, there can be some packets lost due to differences in spectrum availability.

## 5. Conclusion

A fully distributed mechanism for spectrum sharing and access is proposed in this paper. While a common strategy is to choose a CCC, of any form, to solve these problems, we propose a strategy that does not need a CCC. Rather, it works in a distributed fashion over available channels. This removes the need for extra coordination overhead caused by negotiation for a CCC. In a CRN, where channel availability differs temporally and spatially and possibly quite frequently, our proposed protocol prevents the network from being overwhelmed by coordination overhead. In simulations, we compared broadcast performance with a protocol that uses a coordination channel [14]. Results show that our proposed method reduces broadcasts significantly while keeping connectivity in the network similar to that of the reference work.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

## Acknowledgments

This work was supported by the KRF funded by the MEST (NRF-2013R1A2A2A05004535) and the Ministry of Education (2013R1A1A2063779).

## References

- [1] M. McHenry, "Spectrum white space measurements," in *Proceedings of the New America Foundation Broadband Forum*, June 2003.
- [2] J. Mitola III and G. Q. Maguire Jr., "Cognitive radio: making software radios more personal," *IEEE Personal Communications*, vol. 6, no. 4, pp. 13–18, 1999.
- [3] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005.
- [4] M. Yoon, Y.-K. Kim, and J.-W. Chang, "An energy-efficient routing protocol using message success rate in wireless sensor networks," *Journal of Convergence*, vol. 4, no. 1, pp. 15–22, 2013.
- [5] A. Kusdaryono, K. O. Lee, and Y. Lee, "A clustering protocol with mode selection for wireless sensor network," *Journal of Information Processing System*, vol. 7, no. 1, pp. 29–42, 2011.
- [6] Y. Yang, F. Zhen, T.-S. Lee, and M.-S. Park, "TASL: a traffic-adapted sleep/listening MAC protocol for wireless sensor network," *Journal of Information Processing Systems*, vol. 2, no. 1, pp. 39–43, 2006.
- [7] S. Kumar, M. Sarkar, S. Gurajala, and J. D. Matyjas, "MMMP: a MAC protocol to ensure QoS for multimedia traffic over multi-hop ad hoc networks," *Journal of Information Processing Systems*, vol. 4, no. 2, pp. 41–52, 2008.
- [8] A. Sinha and D. K. Lobiyal, "Performance evaluation of data aggregation for cluster-based wireless sensor network," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, pp. 1–17, 2013.
- [9] S. P. Algur and N. P. Kumar, "Novel user centric, game theory based bandwidth allocation mechanism in WiMAX," *Human-Centric Computing and Information Sciences*, vol. 3, no. 1, article 20, 2013.
- [10] C. Cordeiro, K. Challapali, D. Birru, and N. Sai Shankar, "IEEE 802.22: the first worldwide wireless standard based on cognitive radios," in *Proceeding of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 328–337, Baltimore, Md, USA, November 2005.
- [11] S.-Y. Lien, C.-C. Tseng, and K.-C. Chen, "Carrier sensing based multiple access protocols for cognitive radio networks," in *Proceedings of the IEEE International Conference on Communications (ICC '08)*, pp. 3208–3214, Beijing, China, May 2008.
- [12] Y. Lee and I. Koo, "A distributed MAC protocol using virtual control channels for CRSNs," *Wireless Personal Communications*, vol. 71, no. 2, pp. 1021–1048, 2013.
- [13] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: a hardware-constrained cognitive MAC for efficient spectrum management," *IEEE Journal on Selected Areas in Communications*, vol. 26, no. 1, pp. 106–117, 2008.
- [14] J. Zhao, H. Zheng, and G.-H. Yang, "Distributed coordination in dynamic spectrum allocation networks," in *Proceedings of the 1st IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '05)*, pp. 259–268, Baltimore, Md, USA, November 2005.
- [15] S. K. Timalisina, S. Moh, I. Chung, and M. Kang, "A concurrent access MAC protocol for cognitive radio ad hoc networks without common control channel," *Eurasip Journal on Advances in Signal Processing*, vol. 2013, no. 1, article 69, 2013.
- [16] Y. Song and J. Xie, "A distributed broadcast protocol in multi-hop cognitive radio ad hoc networks without a common control channel," in *Proceedings of the IEEE Conference on Computer Communications (INFOCOM '12)*, pp. 2273–2281, Orlando, Fla, USA, March 2012.
- [17] J. Liu and S.-H. Chung, "An efficient load balancing scheme for multi-gateways in Wireless Mesh Networks," *Journal of Information Processing Systems*, vol. 9, no. 3, pp. 365–378, 2013.
- [18] H.-R. Lee, K.-Y. Chung, and K.-S. Jhang, "A study of wireless sensor network routing protocols for maintenance access hatch condition surveillance," *Journal of Information Processing Systems*, vol. 9, no. 2, pp. 237–246, 2013.



# Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

