

## Research Article

# Adaptive Window Size-Based Medium Access Control Protocol for Cognitive Radio Wireless Sensor Networks

Gyanendra Prasad Joshi,<sup>1</sup> Seung Yeob Nam,<sup>1</sup> Sung Won Kim,<sup>1</sup> and Byung-Seo Kim<sup>2</sup>

<sup>1</sup>Department of Information and Communication Engineering, Yeungnam University, 280 Daehak-ro, Gyeongsan-si, Gyeongbuk-do 38541, Republic of Korea

<sup>2</sup>Department of Computer and Information Communication Engineering, Hongik University, Chungcheongnam-do 339-701, Republic of Korea

Correspondence should be addressed to Sung Won Kim; [swon@yu.ac.kr](mailto:swon@yu.ac.kr)

Received 2 September 2015; Revised 20 December 2015; Accepted 6 January 2016

Academic Editor: Alberto J. Palma

Copyright © 2016 Gyanendra Prasad Joshi 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.

Many existing medium access control protocols for cognitive radio wireless sensor networks waste bandwidth and suffer from delay because of the fixed channel negotiation period in the common control channel. In this paper, we propose a medium access control protocol for cognitive radio wireless sensor networks that adjusts the channel negotiation period based on network density. We simulate the proposed approach and compare it to the performance of the fixed channel negotiation period based approaches. The results show that the proposed approach efficiently utilizes white spaces and increases the aggregated goodput.

## 1. Introduction

Because the number of wireless sensor networks that basically work in the Industrial, Scientific, and Medical (ISM) radio band is increasing, the spectrum scarcity problem is becoming very challenging for researchers and service providers. Cognitive radio wireless sensor networks (CRWSNs) [1] have shown promise as a means to mitigate the spectrum scarcity problem in the near future. The main objective of CRWSNs is to utilize the underutilized/unutilized licensed bands without affecting the incumbent license holders of those bands, also called the Primary Users (PU). An efficient Medium Access Control (MAC) protocol is essential to achieve this objective of CRWSNs. There have been several works that aimed to utilize the underutilized/unutilized licensed channels opportunistically by unlicensed users, also called Secondary Users (SU) or Cognitive Radio (CR) wireless sensor nodes.

Incumbents have the first priority to utilize the licensed bands. Therefore, most of the Cognitive Radio Medium Access Control (CR-MAC) protocols use a Common Control Channel (CCC) for data channel negotiation to protect their rights and to mitigate the interference with the incumbents. Many existing CCC-based protocols divide time into Beacon

Intervals (BIs) and further divide the BIs into the Channel Negotiation (CN) window, also called the ad hoc traffic indication message (ATIM) window, and the data transfer period, also called the data window. Some examples of those protocols are MMAC-CR [2], ECRQ-MAC [3], P-MAC [4], and so forth. These protocols use a fixed CN window, similar to MMAC [5].

In the CN window, nodes send control packets for channel negotiation and reservation. In the data window, nodes send actual data packets. In dense network environments, the existing “fixed CN window” based protocols suffer from CCC saturation, also called the CCC bottleneck problem. This is associated with the following problems:

- (a) *Channel Utilization Limitation.* In general, the CN window size is around one-fourth of the data window size. In these approaches, CN messages are only sent in the CN window. Hence, the CN window can become overcrowded and cannot negotiate for all available channels, when the number of communicating pairs exceeds the available time slots in the CN window. This may lead to some of the data channels becoming underutilized/unutilized.

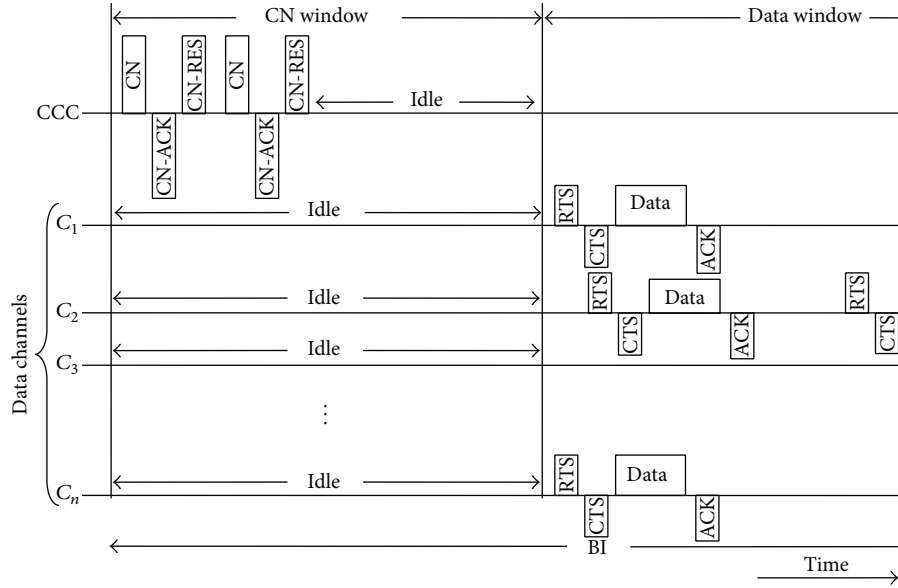


FIGURE 1: Bandwidth waste in existing approaches.

(b) *Bandwidth Waste in Channel Negotiation.* If the CN window is too large, then the bandwidth of all the data channels is wasted because, in the CN window, nodes do not send or receive data packets. This problem is represented in Figure 1, which shows that there are only a few nodes contending for channels. After the channel negotiation, CCC is idle. Even worse, all the data channels are idle for the entire CN window, which is approximately  $\tau_{\text{CIT}} + n \times \text{CN}$  in one BI, where  $n$  is the number of channels available to be used opportunistically for secondary users and  $\tau_{\text{CIT}}$  is the control channel idle time. Furthermore, allocating more periods for the CN window will decrease the data window, which undeniably decreases the networks throughput.

(c) *Long Channel Access Delay.* If the CN window is too short, the contention loser nodes have to wait until the next BI. In the worst case, if the number of nodes is very large, waiting delay may be several BIs. Therefore, they might have to wait for a long time to access idle channels. Since the data window is much larger than the CN window, waiting for more than one beacon interval is costlier in terms of delay and bandwidth utilization.

The expected channel access delay (1) shows the essentiality of CN window adjustment. Here, the expected channel access delay is calculated as

$$E[X] = \Delta \sum_{j=0}^m \left( \frac{W_j + 1}{2} \right) \frac{p^j - p^{m+1}}{1 - p^{m+1}} + \sum_{i=0}^m \frac{p^i (1-p)}{1 - p^{m+1}} \cdot \sum_{j=i+1}^{W_0+\dots+W_i} \frac{I_{\text{data}}}{N_s} \cdot A(j, i) \cdot \left( \prod_{l=0}^i \frac{1}{W_l} \right) \sum_{k=0}^{N_s-1} \left\lfloor \frac{k+j}{N_s} \right\rfloor, \quad (1)$$

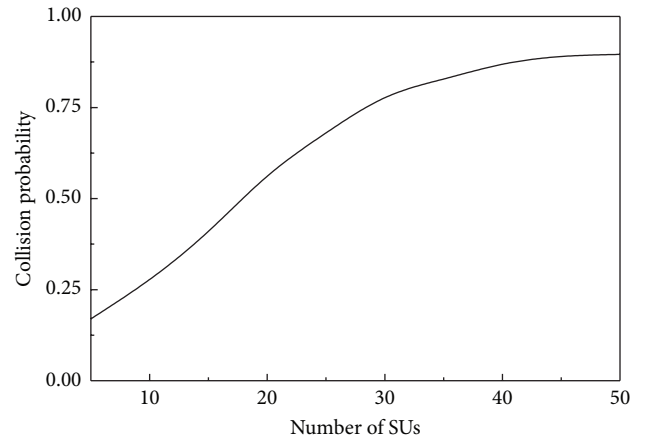


FIGURE 2: Channel access delay in various node densities.

where  $m$  is the maximum retry number,  $W_j$  is the contention window size,  $p$  is the collision probability,  $I_{\text{data}}$  is the data window time,  $N_s$  is the number of minislots in one CN window,  $\Delta$  is the average time to complete a CN packet exchange for channel negotiation, and  $A(j, i)$  counts the number of events corresponding to a given delay ( $j$ ) measured in minislots under the assumption that the transmission succeeds in stage  $i$ . The collision probability  $p$  is obtained by using the formula described in Section III of [6]. A more detailed description about (1) can be found in [7].

Figure 2 shows the expected channel access delay in various node densities, where the maximum retry limit is three,  $\Delta$  is 1.27 ms,  $I_{\text{data}}$  is 80 ms, and the minimum CW size is 16. In general the CN window size is one-fourth of the data window. Hence, the CN window is about 20 ms. Figure 2 shows that when the number of nodes increases, the channel access delay increases. When the number of nodes exceeds 15, nodes have to wait for channel access to the next BI, which is 80 ms later.

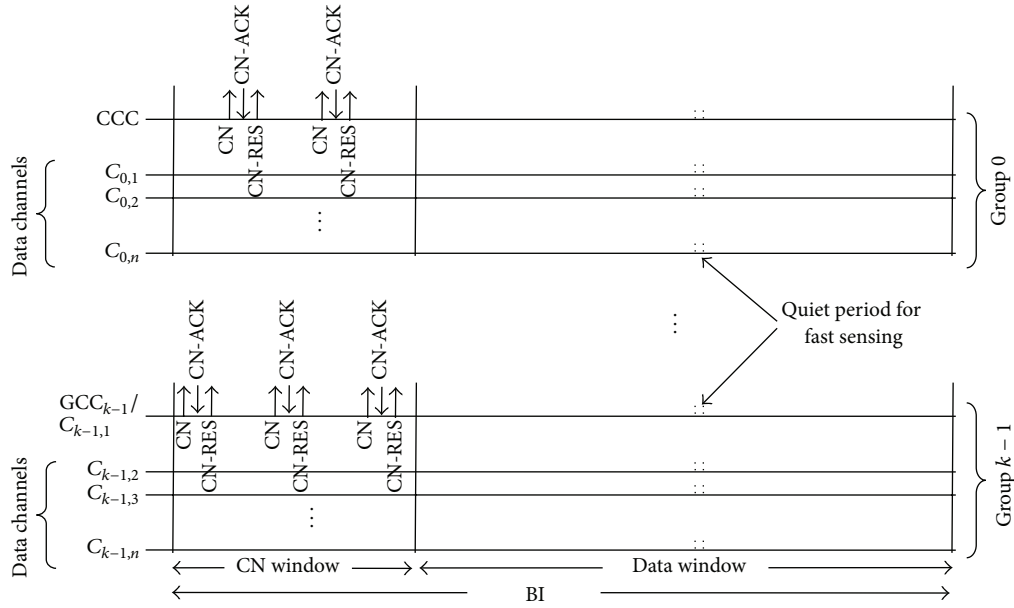


FIGURE 3: Channel grouping in the proposed approach.

The number of nodes may increase or decrease arbitrarily in ad hoc wireless networks. The small CN window is a bottleneck in a dense network environment and the large CN window decreases the throughput and increases the delay in a sparse network environment. Considering all these shortcomings and the above-mentioned issues, we propose an adaptive CN window-based MAC protocol for fully connected single-hop CRWSNs. The proposed protocol dynamically adjusts the CN period and hence mitigates the channel utilization limitation, waste of bandwidth in channel negotiation period, and long channel access delay. The proposed protocol is effective in both dense and sparse CRWSNs topologies. Moreover, this protocol adopts a strategy that reduces the number of CN packets mitigating the control channel bottleneck problem. Hence, it increases the throughput and decreases the delay. Since hardware costs are falling, we assume that each secondary user in our protocol is equipped with two transceivers.

The remainder of the paper is organized as follows. Section 2 reviews existing protocols and discusses the drawbacks. Section 3 describes the proposed MAC layer protocol. The corresponding simulation results and evaluations are presented in Section 4. The conclusions of the paper are summarized in the final section.

## 2. Related Work

A Dynamic Channel Allocation- (DCA-) based protocol called the Distributed Coordinated Spectrum Sharing (DCSS) MAC protocol for cognitive radio is proposed in [8]. Similar to DCA [9], Request To Send (RTS) and Clear To Send (CTS) packets are exchanged before communications in DCSS. RTS and CTS include the available data channel list. The time slot mechanism in DCSS is used to detect incumbents. However, DCA fully relies on CCC, so they may

incur control channel starvation. In DCA-based protocols, CCC can become a bottleneck if too much control information is sent over this channel. All nodes need to contend for access to the control channel, and the data channels remain underutilized [2].

A synchronized multichannel MAC protocol, called a distributed multichannel MAC protocol for multihop CRNs (MMAC-CR), is proposed in [2]. This is a CCC-based protocol similar to MMAC [5]. As in IEEE 802.11 PSM [10], time is divided into the CN window and the data window in the MMAC-CR protocol. Control packets for channel negotiation and channel reservation are transferred in the CN window, and data packets are transferred in the data windows. Similar to MMAC-CR, an energy efficient cognitive radio MAC protocol for QoS provisioning (ECRQ-MAC) [3] is proposed. All these synchronization and BI-based protocols are less prone to the CCC bottleneck problem because only one handshake on the CCC is needed per connection during the beacon interval. Little control information is required compared to DCA-based protocols, where one handshake per packet to send is required. However, as we described in Section 1, these protocols use a fixed CN window. The fixed CN window-based protocols suffer from the CCC saturation problem in a dense network topology and waste bandwidth in a sparse network topology.

A MAC protocol to mitigate the CCC bottleneck problem in dense cognitive radio wireless sensor networks is proposed in [11]. This is a group-based approach, where channel negotiation control packets are distributed into the multiple data channels. In this protocol, all channels are divided into  $k$  groups. Each SU has a unique ID and the default group of a node is decided by dividing a unique ID by the number of groups. Except the first group, every group has one default group control channel (GCC). The first channel of the group is assigned as the GCC of the group. In Figure 3, CCC is the

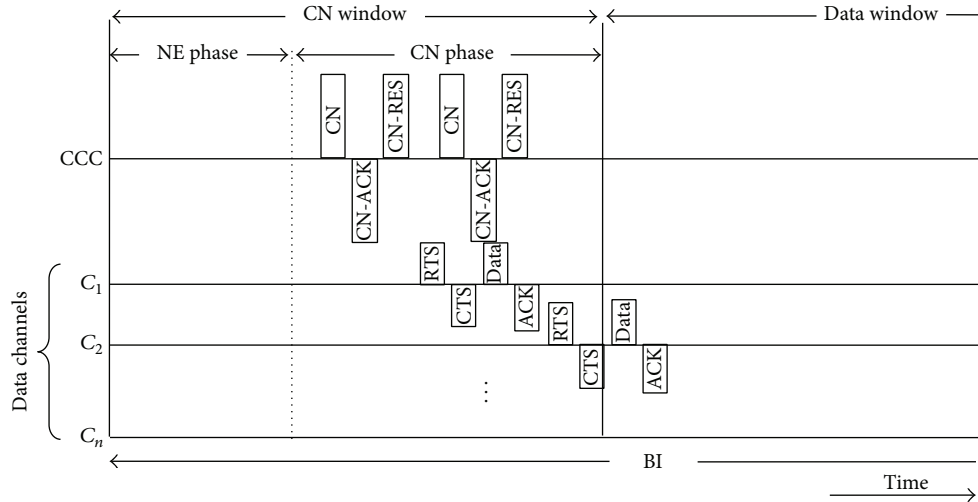


FIGURE 4: Efficient bandwidth utilization in proposed protocol.

GCC for Group 0, and  $GCC_{k-1}$  is a GCC for Group  $k-1$ . The GCC of each group is the normal data channel and is used for data communication in the data window. However, CCC is not used for data communication and is reserved to send emergency messages, in case the PU arrives at the channel. This CN message exchange is performed in CCC for Group 0 and in GCCs for the rest of the groups.

The entire sender nodes know the group which the receiver belongs to by its ID. If the receiver node is in the same group as the sender node, the sender starts sending the data packet in the data window after successfully exchanging CN packets. If it is in the different group with the sender, then the sender tunes its control transceiver into the GCC of the receiver's group. The data transceiver switches to the negotiated channel and starts data transmission in the data window if it wins contention. It switches back to its default group after completion of the data transmission. If the control transceiver tunes into the receiver's GCC for contention, the data transceiver starts listening to its own GCC for the CN window time to avoid missing packets destined to it.

Although the protocol to mitigate the CCC bottleneck problem by Joshi and Kim [11] significantly mitigates CCC bottleneck problem in dense CRNs, it needs more sophisticated and intelligent device, which can decide how many  $k$  should be there and which channel belongs to which group by monitoring the number of primary channels available to use opportunistically for SUs. On the contrary, CR nodes do not need to have such sophisticated cognition capabilities in the proposed protocol.

The basic idea of window size adaptation in the proposed protocol is similar to IPSM [12]. In IPSM, at the end of the ATIM window, each node measures how long the channel was continuously idle, which is called the current Channel Idle Time (CIT). If the CIT is greater than a given threshold, the channel is idle long enough to assume that no node intends to transmit an ATIM frame. Thus, nodes do not increase their ATIM window size. However, nodes take a random backoff from 0 to Contention Window (CW).

When the backoff reaches 0, nodes transmit an ATIM packet and they double the CW size if they do not receive an acknowledgement. Because of the random backoff, even if there are few nodes contending for channel access, the CIT may not be less than the threshold. Therefore, measuring the CIT at the end of the ATIM window is not always advisable.

### 3. Proposed Protocol

In the proposed protocol, we estimate the network density and adjust the CN window accordingly. At the beginning of the BI, nodes randomly contend to transmit the synchronization beacon as in the Power Saving Mechanism (PSM) specified in the IEEE 802.11 Distributed Coordination Function [10]. In the first stage of the CN window, nodes estimate the number of active pairs and adjust the CN window. In the second stage of the CN window, nodes having packets in the outgoing queue contend for channel access. The contention winner node sends a CN packet along with the available channel list to the intended receiver. After receiving the CN packet, the receiver node checks its channel status table, selects a common channel, and sends an acknowledgement (CN-ACK) with the selected common channel. Finally, the sender sends a reservation confirmation message (CN-RES) with the selected channel. Neighbor nodes update their channel status table after overhearing the CN-RES.

All the secondary users are equipped with two transceivers; thus they do not need to wait until the expiration of the CN window to transfer data. They start sending data packets in the data channels immediately after successful CN packet exchange. Figure 4 shows how our approach achieves more effective bandwidth utilization. After successful channel negotiation in the CN phase, nodes exchange RTS and CTS packets and transmit data packets according to IEEE 802.11 DCF. Whenever a primary user reclaims the channel currently used by a secondary user, the latter stops sending immediately and buffers packets. The node sends an

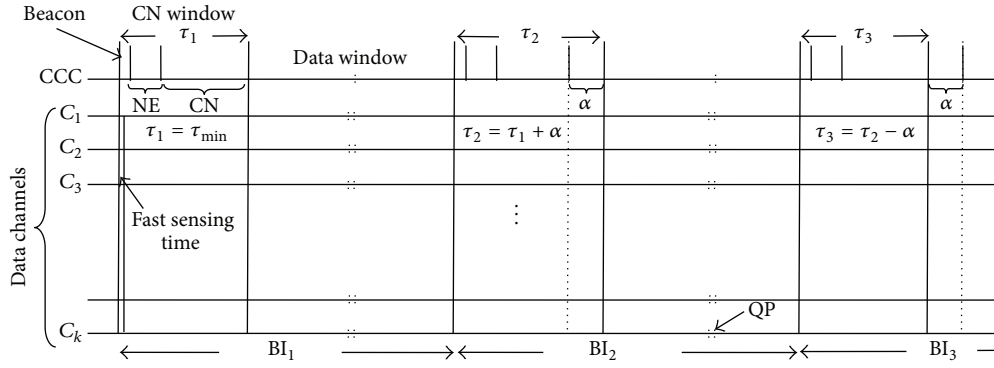


FIGURE 5: CN window adjustment.

emergency message to the CCC to inform the receiver and neighbor that the primary user is active in the channel.

Reducing the number of CN packets certainly mitigates the control channel bottleneck problem. Therefore, unlike some existing fixed CN window-based CR-MAC protocols [2–4] if any neighbour node has previously negotiated with the same receiver node, the node does not send CN packet to negotiate for channels to reduce the number of CN packets exchanged.

For example, nodes  $n_0$  and  $n_2$  have pending data packets for node  $n_1$ . Assume that node  $n_0$  wins contention and sends a CN packet to node  $n_1$ . Now, node  $n_2$  does not send a CN packet to node  $n_1$  because while node  $n_0$  was negotiating with node  $n_1$ , node  $n_2$  overhears the channel negotiation control packets. Hence, node  $n_2$  knows which channels are available for node  $n_1$ . It also knows that  $n_1$  will remain in the wakeup state for the entire current BI. Therefore, node  $n_2$  does not need to send a CN packet to node  $n_1$ . This reduces the number of CN packets exchanged and provides more opportunities to other nodes with pending packets to send to negotiate for the channel.

**3.1. Protocol Descriptions.** We assume that a cognitive radio is an intelligent device that can make observations, orient itself, and make decisions. Each CR device is equipped with two transceivers, the control transceiver and the data transceiver. The control transceiver is permanently hooked into the CCC and the data transceiver is dynamic and capable of channel hopping and working in different frequency bands.

Similar to [13], time is divided into BIs, which are subdivided into the CN window and the data window. The CN window size is adaptive and is within  $[\tau_{\min}, \tau_{\max}]$ . The CN window has two phases, the Nodes Estimation (NE) phase and the Channel Negotiation (CN) phase. In the NE phase, nodes estimate the number of nodes that have packets in the outgoing queue. Initially, every node has the same CN window size of  $\tau_{\min}$ . If the estimated number of nodes intending to send packets ( $\hat{n}$ ) is less than a given threshold range ( $n_{\text{thresh}}$ ), the nodes decrease the CN window size according to a design parameter ( $\alpha$ ). As discussed in Section 1, the average time to complete a CN packet exchange for channel negotiation ( $\Delta$ ) is 1.27 ms. Therefore,  $\alpha$  is a multiple of  $\Delta$ . If  $\hat{n}$  is more than  $n_{\text{thresh}}$ , the nodes increase the CN window size

```

if NE phase expired then
  estimate  $\hat{n}$ ;
  if  $\hat{n} < n_{\text{thresh}}$ 
    and  $\tau > \tau_{\min}$  and  $\tau \leq \tau_{\max}$  then
       $\tau - = \alpha$ 
  else if
     $\hat{n} \geq n_{\text{thresh}}$  and  $\tau_i < \tau_{\max}$  then
       $\tau + = \alpha$ 
  else
    remain unchanged
  start contention for sending CN packet

```

ALGORITHM 1: Algorithm for adjusting CN window.

by  $\alpha$ . If  $\hat{n}$  is within the threshold range, the CN window size is unchanged. The method for adjusting the CN window size is shown in Algorithm 1.

As shown in Figure 5, there are  $k$  data channels from  $C_1$  to  $C_k$  and one common control channel. Time is divided into BIs (i.e.,  $BI_1, BI_2, BI_3$ , etc.), which are subdivided into the CN window and the data window. In the first BI (i.e.,  $BI_1$ ), the CN window time  $\tau_1$  is equal to  $\tau_{\min}$ . The CN window size is adjusted in  $\tau_2$  and  $\tau_3$  according to the number of nodes estimated at the end of the NE phase. At the beginning of the BI, the data transceivers of each node randomly select a data channel and detect the energy level to perform Fast Sensing (FS). The nodes save the sensing report of the channel condition in the channel status table. The FS period is also located in the middle of the data window. The application of this FS period is described in Section 3.2.

Figure 6 is the expanded figure of the CN window. In Figure 6, the CN window is divided into NE and CN phases. NE phase is subdivided into  $M = \{t_i \mid i = 1, 2, \dots, m\}$  minislots, and it is significantly smaller than the CN phase. However,  $t_i$  is long enough to perform carrier sensing and determine whether the slot is idle or busy. According to the IEEE 802.11 specification [10] for Direct Sequence Spread Spectrum (DSSS), the Clear Channel Assessment (CCA) for compliant hardware must be less than  $15 \mu\text{s}$ . To mitigate the effect of short-term fading, we set  $t_i$  to  $20 \mu\text{s}$ , which is equivalent to the default time slot in IEEE 802.11 DSSS.

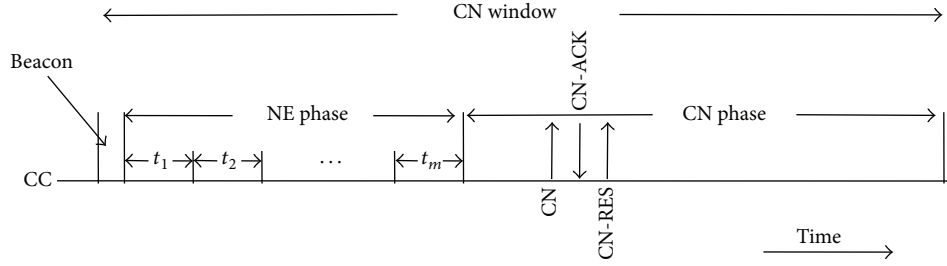


FIGURE 6: A CN window with NE and CN phases.

In the NE phase, nodes having packets in the outgoing queue randomly select one minislot and transmit dummy packets. There is no information in these dummy packets that needs to be decoded. These are the same as those receiving a busy signal. Since nodes select minislots randomly, one or more nodes can send a busy signal in the same slot. Therefore, the number of communication pairs can exceed the number of busy signals. Even though collision occurs at the receivers end, receivers can still know that the slot is busy.

Assuming that the variation in number of active nodes is relatively gradual, it can be estimated from the statistical distributions of the number of busy slots in the current and past beacon intervals. The number of communication pairs with pending packets for transmission, given the number of busy minislots observed in the NE phase, can be estimated as in (4).

Let  $N$  and  $B$  be the random variables representing the number of wireless nodes and the number of busy minislots in the current NE phase, respectively. We first derive the relation between the number of the wireless nodes,  $N = n$ , and the average number of busy minislots in the current NE phase; that is,  $E[B | N = n] = f(n)$ . We estimate the number of nodes based on the relation between  $E[B | N = n]$  and  $n$ . We assume that the number of minislots in the NE phase  $M$  is selected sufficiently large such that  $M$  is usually not smaller than the number of nodes  $n$ ; that is,  $n \leq M$ . The average number of busy slots is estimated using exponential moving average of (5) and we attempt to estimate the number of wireless nodes,  $n$ , again by reversing the relation of  $E[B | N = n] = f(n)$ , that is, by using (4).

In order to facilitate the mathematical derivation, we consider a little modified situation. Let us consider the case where the NE phase is performed in a centralized manner by some center as follows. Each wireless node notifies the center of its own decision on the specific minislot number for the busy signal one by one. The center collects the selected minislot information from all the nodes within the coverage and advertises the aggregated information back to every node again. Then, the number of busy slots does not change compared with the original case where each node broadcasts the busy signal on its own selected minislot, and each wireless node does not feel any difference in terms of the number of busy slots compared with the original case.

Thus, we derive a relation on the probability  $\Pr(B = j | N = i)$  assuming that each node notifies its own selected

minislot in sequence. Then, we can derive the following set of initial condition and recursive equations:

$$\Pr(B = 1 | N = 1) = 1,$$

$$\Pr(B = 1 | N = i) = \Pr(B = 1 | N = i - 1) \cdot \frac{1}{M},$$

$$2 \leq i \leq M,$$

$$\Pr(B = j | N = i) = \Pr(B = j | N = i - 1) \cdot \frac{j}{M}$$

$$+ \Pr(B = j - 1 | N = i - 1)$$

$$\cdot \frac{M - j + 1}{M},$$

$$2 \leq i \leq M, 2 \leq j < i,$$

$$\Pr(B = i | N = i) = \Pr(B = i - 1 | N = i - 1)$$

$$\cdot \frac{M - i + 1}{M}, \quad 2 \leq i \leq M.$$

Table 1 shows the structure of the probabilities  $\Pr(B = j | N = i)$ 's obtained by (2). It is not easy to derive a closed form relation for  $\Pr(B = j | N = i)$ . The value of each element in Table 1 can be easily obtained from the recursive relation (2) if the value of  $M$  is given. If the value of the probability  $\Pr(B = j | N = i)$  is obtained by (2), then the value of  $E[B | N = n]$  can also be obtained from

$$E[B | N = n] = \sum_{j=1}^n j \times \Pr(B = j | N = n). \quad (3)$$

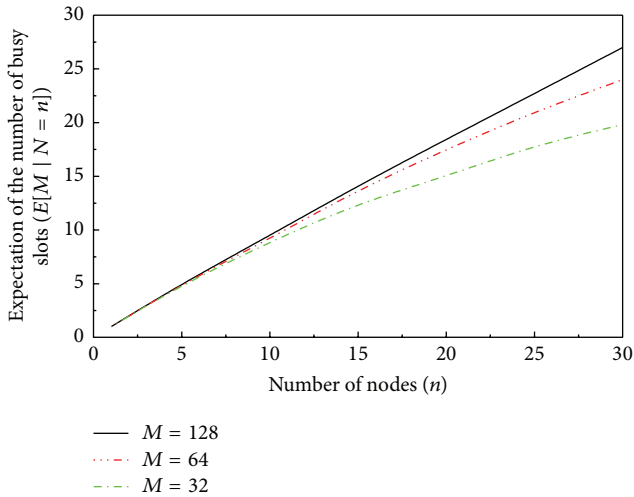
Figure 7 shows the graph of  $f(n) = E[B | N = n]$  for various values of  $M$ , that is, 32, 64, and 128. Since  $f(n)$  is a monotonically increasing function and we assume that  $n$  does not change abruptly, if we can measure the average number of busy slots  $\bar{B}$ , then the corresponding number of nodes can be estimated by

$$\hat{n} = f^{-1}(\bar{B}), \quad (4)$$

where the inverse function  $f^{-1}(\cdot)$  can be evaluated numerically using (2) and (3) and the graph in Figure 7. We can also infer that the accuracy of (4) can be improved by using a large

TABLE 1: A table summarizing the probability  $\Pr(B = j | N = i)$  for various values of  $i$  and  $j$ .

$N$	$B$				
	1	2	3	...	$M$
1	1	0	0		0
2	$\frac{1}{M}$	$1 - \frac{1}{M}$	0		0
3	$\frac{1}{M^2}$	$\frac{3}{M} - \frac{3}{M^2}$	$1 - \frac{3}{M} + \frac{2}{M^2}$		0
...					...
$M$	$\frac{1}{M^{M-1}}$			...	$\left(1 - \frac{1}{M}\right) \dots \left(1 - \frac{M-1}{M}\right)$

FIGURE 7: Expected number of busy slots, that is,  $E[B | N = n]$ , obtained for various values of  $M$ .

value of  $M$  at the cost of increased NE phase duration since the slope is steeper for larger values of  $M$ .

If  $B_k$  denotes the number of measured slots at the  $k$ th beacon interval, then the average number of busy slots  $\bar{B}$  can be estimated by using exponential moving average:

$$\hat{B}_k = \gamma \hat{B}_{k-1} + (1 - \gamma) B_k, \quad (5)$$

where the weight  $\gamma$  is set to 0.9 in the current version.

**3.2. Energy Conservation and Protection of PUs.** We consider that one BI is equal to the maximum tolerable interference time for the primary users because licensed users' right to access the channel should be the first priority in CRWSNs. To increase effectiveness, CR nodes hold their transmission and buffer packets and perform fast sensing at the quiet period in the middle of the data window as shown in Figure 3. If they detect a PU's activity on the channel, they stop transmission and send an Emergency Control Message (ECM) on the CCC. Otherwise, they continue their transmission. The ECM is similar to a CN message with a "no use" flag in the current channel. After sending an ECM, CR nodes contend for channel access and renegotiate for the new channel and continue transmission if channel negotiation is successful.

In the proposed protocol, the data transceiver enters the doze mode if there is no data to send or receive in order to conserve energy. The control transceiver enters the doze mode after the CN window and reawakens just before the end of the quiet period in the middle of data window as shown in Figure 8. This is possible because one BI is equal to the maximum tolerable interference time for the PUs. Assuming that control transceiver of a cognitive radio wireless sensor (CRWS) node can be turned off completely in the doze mode, the doze and wakeup cycles in the proposed protocol conserve around 40% of the total energy consumed by the control transceiver.

#### 4. Simulation Results and Evaluations

We simulate the proposed protocol in ns-2 and compare it to the MMAC-based CR-MAC protocol that has fixed CN window. We assume that CCC is free from the intervention of PUs. The entire simulation is done in four channels including the CCC scenario. The number of CR nodes varies from four to 100 nodes. The arrival of PUs is modeled as ON/OFF channel usage. ON means the channel is busy; that is, a PU is active on the channel and it cannot be utilized by the SUs. The packet size is 512 bytes and the traffic rate ranges from 10 to 1000 packets per second. A beacon interval is equal to 100 ms and a CN window size for the fixed CN window-based protocol is 20 ms. The total simulation time is 40 seconds.

In Figure 9, we compare the fixed CN window-based approach and the proposed dynamic CN window-based approach. The numbers after the legends are the size of the CN windows in milliseconds. The figure shows that, in the fixed CN window-based approach, the goodput decreases if the CN window sizes are very small (5) or large (30). The reason is that if the CN window is too small, it can exchange fewer CN packets, resulting in lower goodput. When the CN window is too large, more CN packets are exchanged, but there is only a short time to transfer the data packets and the channel bandwidth is wasted.

Conversely, in the proposed approach, the CN window is adjusted as required, providing better results in various CRWS node densities. As shown in the figure, the proposed protocol performs similar to the fixed CN window-based approach with average CN window size of 15 up to 44 nodes. When the number of nodes is more than 44, the goodput of the proposed dynamic window size-based protocol is better

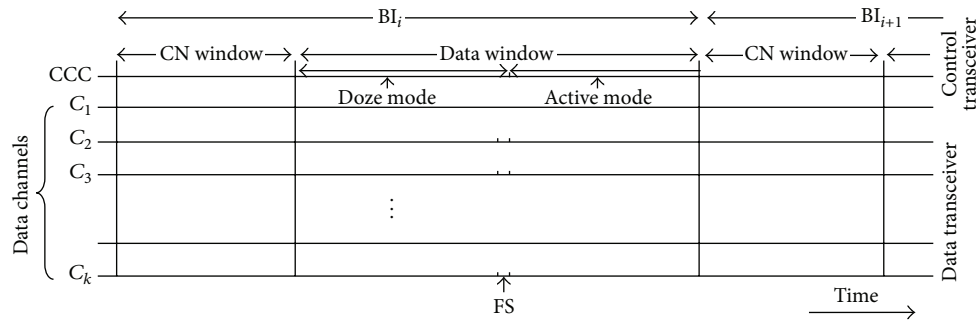


FIGURE 8: Energy conservation and PUs' protection.

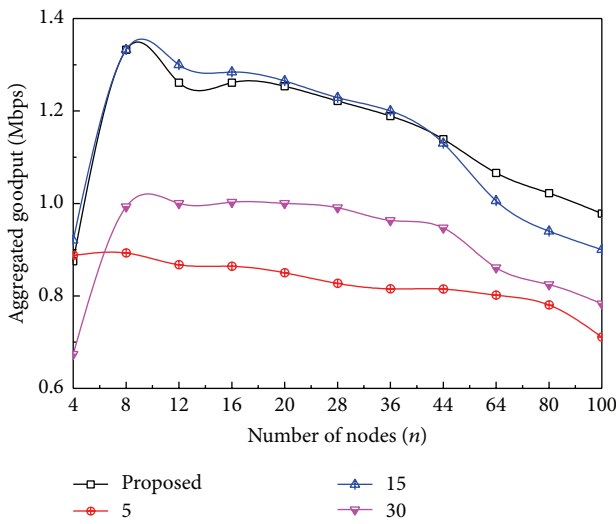


FIGURE 9: Aggregated goodput with various CN window sizes and node densities.

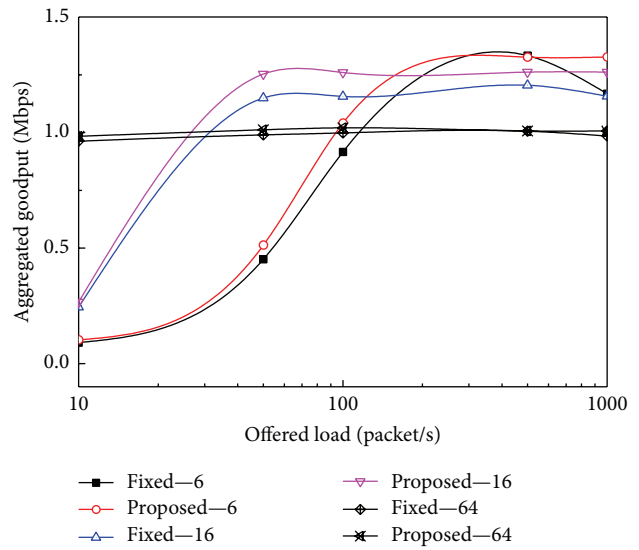


FIGURE 10: Aggregated goodput in various offered loads and node densities.

than others. Thus, the proposed method outperforms other protocols drastically when the CRWSN is dense.

Figure 10 shows the aggregated goodput in various offered loads, which varies from 10 to 1000 packets per second in various CR nodes densities. The values after the legends are the number of CR nodes in both the proposed and existing approaches. The fixed CN-based protocol has 20 ms of CN window. Comparing the aggregated goodput results of fixed CN-based and dynamic CN-based 6 nodes and 16 nodes, high goodput can be yielded with the dynamic CN-based protocol. The fixed CN-based protocol wastes bandwidth and cannot provide the best output as compared to the proposed dynamic CN-based protocol. In 64-node case, the aggregated goodput is similar because of the channel saturation problem that is ordinary problem of the wireless networks. In all cases, our proposed approach performs better than the existing fixed CN-based approach.

Figure 11 shows the average packet delay in six nodes and different offered loads. The values after the legends are the CN window sizes. In all the cases, the proposed protocol has lower delay. The Following are the reasons for the better

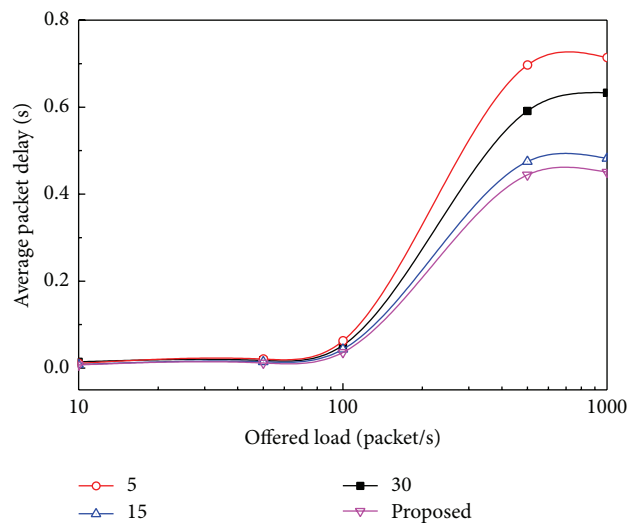


FIGURE 11: Average packet delay in six nodes and different offered loads.



performance in the proposed approach. (a) In the proposed protocol, SUs do not have to wait until the expiration of the CN window; hence it achieves better goodput and lower delay. (b) The proposed protocol adjusts the CN window and provides opportunities to more nodes to negotiate for channels in the dense topology. In the sparse topology, it provides more time for data transmission. (c) The strategy to exchange only one CN packet for the same receiver reduces the numbers of CN control packets significantly. Therefore, more nodes get the opportunity to negotiate for the data channel. Hence, it increases network goodput and decreases delay.

## 5. Conclusions

We presented a MAC protocol for cognitive radio wireless sensor networks based on CN window adjustment according to node density for fully connected single-hop CRWSNs. In ad hoc CRWSNs, nodes may join or leave arbitrarily. In the dense network topology, a small CN window can be a bottleneck, and in the sparse network topology, a large CN window decreases network performance and increases delay. Our proposed adaptive CN window-based approach performs better than the fixed CN window-based schemes in highly dense and highly sparse topologies. The strategies used to turn off transceivers conserve energy. It also protects licensed users by performing sensing in the middle of the data window and sending an ECM in case of signal detection of licensed users. The strategy to reduce the number of CN packets carried out in the proposed protocol mitigates the common control channel bottleneck problem; hence it increases throughput and decreases delay.

## Conflict of Interests

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

## Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01058751).

## References

- [1] G. P. Joshi, S. Y. Nam, and S. W. Kim, "Cognitive radio wireless sensor networks: applications, challenges and research trends," *Sensors*, vol. 13, no. 9, pp. 11196–11228, 2013.
- [2] M. Timmers, S. Pollin, A. Dejonghe, L. Van der Perre, and F. Catthoor, "A distributed multichannel MAC protocol for multihop cognitive radio networks," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 1, pp. 446–459, 2010.
- [3] S. M. Kamruzzaman, M. A. Hamid, and M. Abdullah-Al-Wadud, "An energy-efficient MAC protocol for QoS provisioning in cognitive radio ad hoc networks," *Radioengineering*, vol. 19, no. 4, pp. 567–578, 2010.
- [4] G. P. Joshi, S. Y. Nam, and S. W. Kim, "Decentralized predictive MAC protocol for Ad Hoc cognitive radio networks," *Wireless Personal Communications*, vol. 74, no. 2, pp. 803–821, 2014.
- [5] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MoBiHoc '04)*, pp. 222–233, ACM, Tokyo, Japan, May 2004.
- [6] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, "IEEE 802.11 packet delay—a finite retry limit analysis," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '03)*, vol. 2, pp. 950–954, San Francisco, Calif, USA, December 2003.
- [7] G. P. Joshi, S. Y. Nam, and S. W. Kim, "An analysis of channel access delay in synchronized MAC protocol for cognitive radio networks," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 5, pp. 485–489, 2014.
- [8] H. Nan, T.-I. Hyon, and S.-J. Yoo, "Distributed coordinated spectrum sharing MAC protocol for cognitive radio," in *Proceedings of the 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN '07)*, pp. 240–249, IEEE, Dublin, Ireland, April 2007.
- [9] S. L. Wu, C. Y. Lin, Y. C. Tseng, and J. P. Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *Proceedings of the IEEE International Symposium on Parallel Architectures, Algorithms and Networks (I-SPAN '00)*, pp. 232–237, Dallas, Tex, USA, December 2000.
- [10] IEEE 802.11, "IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems—Local and Metropolitan Area Networks—Specific Requirements Part II: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specs., TG 802.11," 2003.
- [11] G. P. Joshi and S. W. Kim, "Mitigating the control channel bottleneck problem in dense cognitive radio networks," *International Journal of Physical Sciences*, vol. 6, no. 20, pp. 4832–4837, 2011.
- [12] E.-S. Jung and N. H. Vaidya, "Improving IEEE 802.11 power saving mechanism," *Wireless Networks*, vol. 14, no. 3, pp. 375–391, 2008.
- [13] J. Wang, Y. Fang, and D. O. Wu, "A power-saving multi-radio multi-channel MAC protocol for wireless local area networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–12, IEEE, Barcelona, Spain, April 2006.



**Hindawi**

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

