http://www.kkits.or.kr

International Conference on Computer Convergence Technology

ICCCT 2011

20~22 October, 2011
Seoul, Korea

KKITS
nipa
National IT Industry Promotion Agency
KOREA
TOURISM
ORGANIZATION
SEOUL METROPOLITAN
GOVERNMENT
HYUNDAIU&I

Message from the Program Chairs

We are pleased to announce an excellent technical program for the 1st International Conference on Computer Convergence Technology. The program covers a broad cross area of the related convergence topics.

This year, 335 papers were submitted from 16 countries on the four continents for consideration to the program committee. As a result, the selection process was highly competitive, and the result is a program of high-quality papers. All paper was assessed in a blind review, and each received at least four or three reviews.

The program committee meeting took place in Seoul, Korea on August, 15. At this meeting only 50 papers were accepted as full paper, resulting in a 15% acceptance rate. This means that many high quality papers could not be accepted for publication.

At the TPC meeting, and in further email discussion, five papers were nominated for the best paper award.

We would like to express our appreciation to all the contributors and authors for the 335 submissions to ICCCT2011. We thank the members of the program committee and the external reviewers for their hard work in preparing approximately 900 reviews. We thank the General co-chairs for their support. We also thank Jong-Sik Moon for his assistance with the paper submitting & management system.

We hope that you enjoy ICCCT2011 and are challenged by the program.

Chung-Huang Yang, Min Hong, Vladimir Hahanov, Jin-Sul Kim
ICCCT 2011 Program Chairs

Table of Contents

Mobile Convergence Computing & Communications

International Conference on Computer Convergence Technology (ICCCT) 2011, October 2011

Dynamic channel negotiation window based MAC protocol for cognitive radio networks

Gyanendra Prasad Joshi[1], **Seung Yeob Nam**[2], **Byung-Seo Kim**[3] **and Sung Won Kim**[4]
[1,2,4] *Department of Information and Communication Engineering, Yeungnam University*
Gyeongsan bukdo, 712-749, Korea
[3] *Department of Computer and Information Communication Engineering, HongIk University*
Chungcheongnamdo, 339-701, Korea
E-mail: [1]joshi@ynu.ac.kr, [2]synam@ynu.ac.kr, [3]jsnbs@hongik.ac.kr,[4]swon@yu.ac.kr
*Corresponding author: Sung Won Kim

Abstract

Many existing synchronized multi-channel media access control protocols for cognitive radio networks waste bandwidth and suffer from delay because of the fixed channel negotiation period allocated in the common control channel. In this paper, we improve the performance of synchronized multi-channel media access control protocol for cognitive radio networks by adjusting the channel negotiation period based on network density. We also introduce a concept that can reduce the number of channel negotiation packets exchanged and increase efficiency. 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.

***Keywords*:** Cognitive radio, cognitive MAC, dynamic channel access, common control channel.

1. Introduction

Cognitive Radio Networks (CRNs) have shown promise as a means to mitigate the spectrum scarcity problem in the near future. Since the number of users of the Industrial, Scientific and Medical (ISM) radio band is increasing daily, the spectrum scarcity problem is becoming very challenging for researchers and service providers. The main objective of CRNs 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 Media Access Control (MAC) protocol is essential to achieve this objective of CRNs. 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) nodes.

Most of the Cognitive Radio Media Access Control (CR-MAC) protocols use a Common Control Channel (CCC) for data channel negotiation. 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 [1], ECRQ-MAC [2] and ECR-MAC [3] etc. These protocols use a fixed CN window, similar to MMAC [4].

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (2011-0003322) and in part by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency (NIPA-2011- (C1090-1121-0011)).

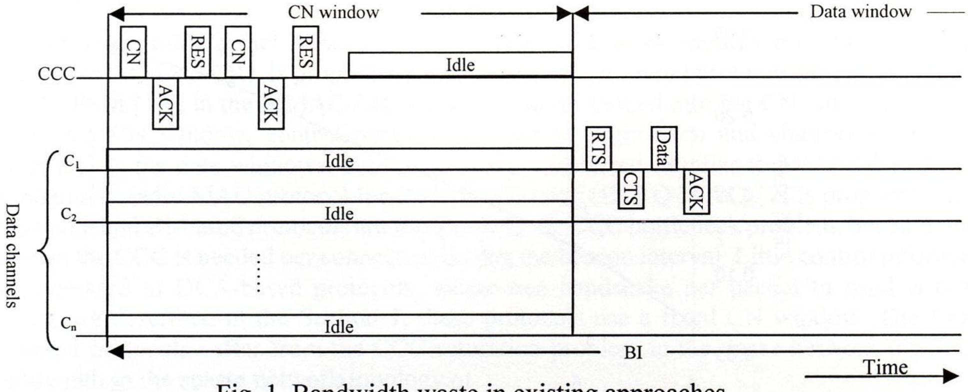

Fig. 1. Bandwidth waste in existing approaches.

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 underutilize/unutilized.

(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 Fig. 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 τ_{CIT} + n×CN in one BI, where, n is the number of channels available to be used opportunistically for secondary users, and τ_{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 high, this 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) taken from [5], shows the essentiality of CN window adjustment. Here, the expected channel access delay is calculated as

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

where, m is the maximum retry number, W_j is the contention window size, p is the packet dropping probability, I_{data} is the data window time, N_s is the number of mini-slots in one CN window, Δ is the average time to complete CN packet (CNP) exchange for channel negotiation, and $A(j,i)$ counts the number of events corresponding to a given delay (j) measured in mini-slots under the assumption that the transmission succeeds in stage i. The collision probability p is obtained by using the formulae described in Section III of [6].

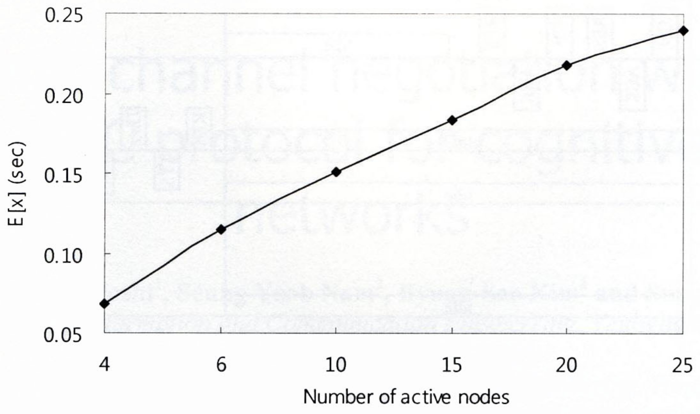

Fig. 2. Channel access delay in various node densities.

Fig. 2 shows the expected channel access delay in various node densities, when the maximum retry limit is three, Δ is 1.27 ms, I_{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 20ms. Fig. 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 80ms later.

Considering the above-mentioned shortcomings and practical issues, in this paper we propose a dynamic CN window-based MAC protocol for fully connected single hop CRNs. Since hardware costs are falling, we assume that each secondary user in our protocol is equipped with two transceivers. In ad hoc wireless networks, the number of nodes may increase or decrease arbitrarily. In the dense network topology, the small CN window will be the bottleneck, and in the sparse network topology, the large CN window decreases throughput and increases delay. The proposed approach dynamically adjusts the CN period, hence it is effective in both dense and sparse topologies. Moreover, our strategy to reduce the number of CN packets mitigates the control channel bottleneck problem, hence it increases throughput and decreases delay.

The remainder of the paper is organized as follows. Section 2 reviews existing protocols and discuss 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

There are a number of related works in literature focused on the MAC layer protocol in CRNs [13][14]. Recently, the IEEE 802.22 working group [7] published the IEEE 802-2011 standards. It aims to construct Wireless Regional Area Networks (WRANs) based on CR, reusing the spectrum allocated to the TV broadcast service. IEEE 802.22 specifies that the network should operate in a point-to-multipoint manner. The architecture of the 802.22 MAC layer is centralized and relies on the base station.

A Dynamic Channel Allocation (DCA)-based protocol called the Distributed Coordinated Spectrum Sharing MAC protocol for cognitive radio (DCSS) is proposed in [8]. Similar to DCA [9], in DCSS, Request To Send (RTS) and Clear To Send (CTS) packets are exchanged before communications. The 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, the 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 [1].

A synchronized multichannel MAC protocol, called a distributed multichannel MAC protocol for multi hop CRNs (MMAC-CR), is proposed in [1]. This is a CCC-based protocol similar to MMAC [4]. As in 802.11 PSM [10], in the MMAC-CR protocol, time is divided into the CN window and the data window. In the CN window, control packets for channel negotiation and channel reservation are transferred, and in the data windows, data packets are transferred. Similar to MMAC-CR, an energy efficient cognitive radio MAC protocol for QoS provisioning (ECRQ-MAC) [2] 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 the Section 1, these protocols use a fixed CN window. The fixed CN window-based protocols suffer from the CCC-saturation problem in the dense network topology and waste bandwidth in the sparse network topology.

The proposed adjustment of the CN window is similar to IPSM [11]. 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.

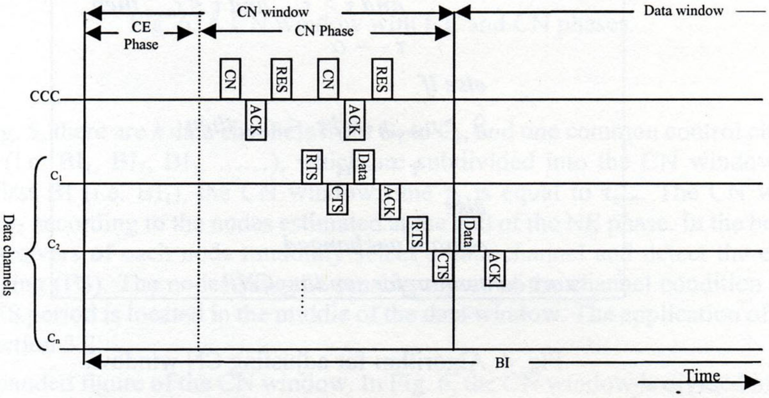

Fig. 3. Efficient bandwidth utilization in proposed protocol.

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 synchronize by sending beacon as in the Power Saving Mechanism (PSM) specified in the 802.11 Distributed Coordination Function (DCF) [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 CNP along with the available channel list to the intended receiver. After receiving the CNP, 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 expiration of the CN window to transfer data. They start sending data packets in the data channels immediately after successful CNP exchange. Fig. 3 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 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 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 [1]-[3], to reduce the number of CN packets exchanged, nodes do not send CN packets to negotiate for channels if any of them have previously negotiated with the node.

For example, nodes n_0 and n_2 have pending data packets for node n_1. Assume that node n_0 wins contention first and sends a CNP to node n_1. Now, node n_2 does not send a CNP 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 CNP to node n_1. This reduces the number of CNPs exchanged and provides more opportunities to other nodes with pending packets to send to negotiate for the channel.

```
if NE phase expired then
    estimate $\hat{n}$;
        if $\hat{n} < n_{thresh}$
                and $\tau > \tau_{min}$ and $\tau \le \tau_{max}$ then
                $\tau -= \alpha$
        else if
        $\hat{n} \ge n_{thresh}$ and $\tau_i < \tau_{max}$ then
                $\tau += \alpha$
        else
        remain unchanged
    start contention for sending CNP
```

Fig. 4. Algorithm for adjusting CN window.

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 [12], time is divided into BIs, which are subdivided into the CN window and the data window. The CN window size is adaptive and is between [τ_{min}, τ_{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 τ_{min}. If the estimated number of nodes intending to send packets ($\hat{n}$) is less than a given threshold range ($n_{thresh} = [n_{min}, n_{max}]$), the nodes decrease the CN window size according to a design parameter (α). If $\hat{n}$ is more than n_{thresh}, the nodes increase the CN window size by α. If $\hat{n}$ is within the threshold range, the nodes remain unchanged. The method for adjusting the CN window is shown in Fig. 4.

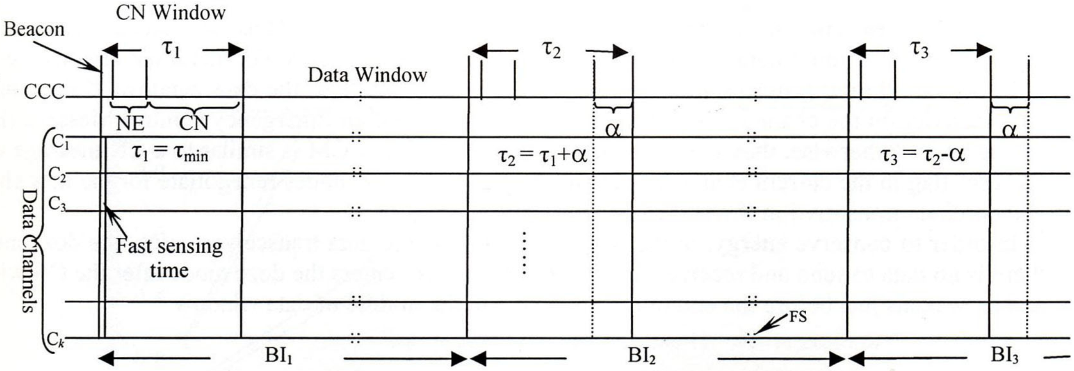

Fig. 5. CN window adjustment.

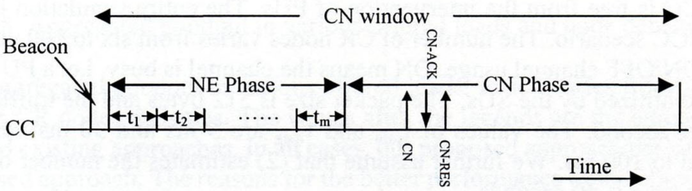

Fig. 6. A CN window with NE and CN phases.

As shown in Fig. 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, ……), which are subdivided into the CN window and the data window. In the first BI (i.e. BI_1), the CN window time τ_1 is equal to τ_{min}. The CN window size is adjusted in τ_2 and τ_3 according to the nodes estimated at the end of the NE phase. In 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 located in the middle of the data window. The application of this FS period is described in Section 3.2.

Fig. 6 is the expanded figure of the CN window. In Fig. 6, the CN window is divided into the NE and CN phases. The NE phase is subdivided into $M = \{t_1, t_2, \ldots, t_m\}$ minislots, and it is significantly smaller than the CN phase. However, t is long enough to perform carrier sensing and determine whether the slot is idle or busy. According to the 802.11 specification [10] for Direct Sequence Spread Spectrum (DSSS), the Clear Channel Assessment (CCA) for compliant hardware must be less than 15μs. To mitigate the effect of short-term fading, we set t to 20μs, which is equivalent to the default time slot in 802.11 DSSS.

In the NE phase, nodes having packets in the outgoing queue randomly select one mini-slot and transmit dummy packets. There is no information in these dummy packets that needs to be decoded. These are the same as receiving a busy-signal.

Since nodes select mini-slots 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 n 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 mini-slots observed in the NE phase, can be estimated as (2) in [12].

3.2 Energy conservation and protection of PUs

In CRNs, protection of licensed users should be the first priority. Thus, we consider that one BI is equal to the maximum tolerable interference time for the primary users. For increased effectiveness, the CR nodes hold their transmission and perform FS in the middle of the data window. 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 renegotiate for the new channel and continue transmission if channel negotiation is successful.

In order to conserve energy, in the proposed protocol, the data transceiver enters the doze mode if there is no data to send and receive. The control transceiver enters the doze mode after the CN window and reawakens just before the end of the FS period in the middle of data window.

4. Simulation Results and Evaluations

We simulate our protocol in ns-2 and compare it to the MMAC-based CR-MAC protocol. 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 six to 100 nodes. The arrival of PUs is modeled as ON/OFF channel usage. ON means the channel is busy, i.e. 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 500 packets per second. The values of τ_{min} and τ_{max} are 5 ms and 30 ms, respectively. One beacon interval is equal to 100 ms. We further assume that (2) estimates the number of nodes accurately. The total simulation time is 40 seconds.

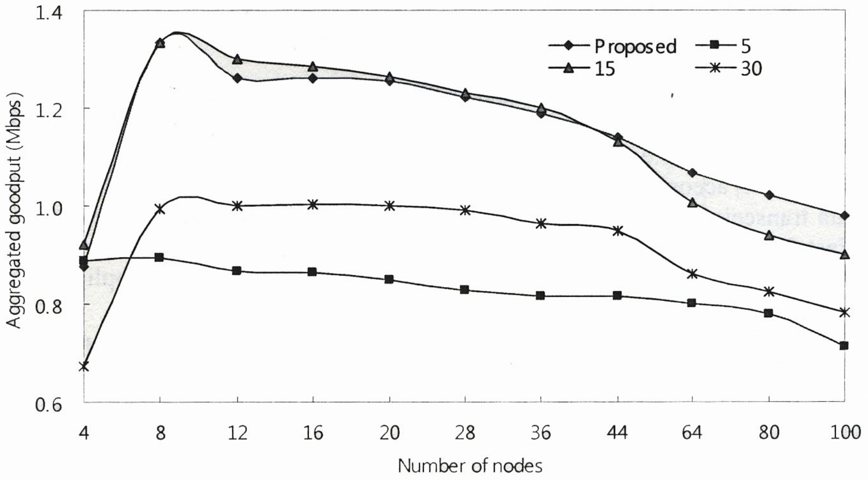

Fig. 7. Aggregated goodput with various CN window sizes and node densities.

In Fig. 7, 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 our approach we adjust the CN window as required, providing better results in various node densities.

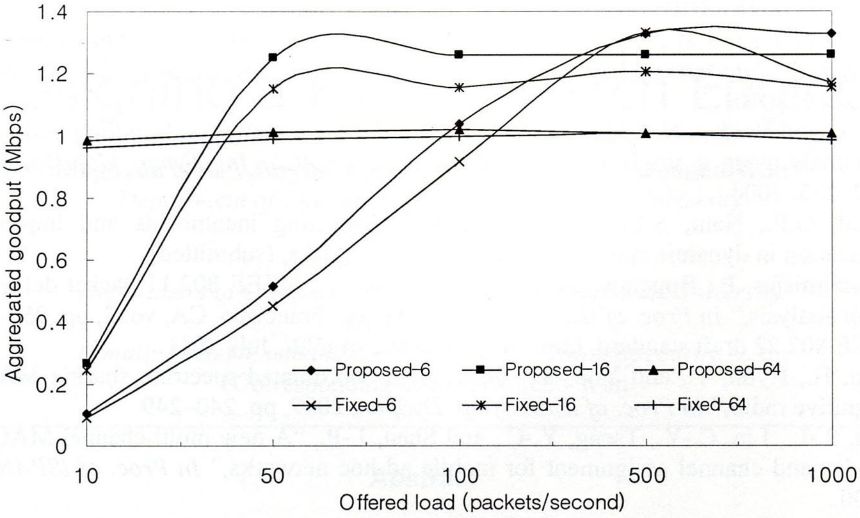

Fig. 8. Aggregated goodput in various offered loads and node densities.

Fig. 8 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. In all cases, our proposed approach performs better than the existing fixed CN based approach. The reasons for the better performance of the proposed approach are: (a) the proposed protocol adjusts the CN window and provides opportunities to more nodes to negotiate for channels. In the sparse topology, it provides more data transmission time. (b) Nodes do not have to wait until the expiration of the CN window, hence they achieve better goodput. (c) The strategy to exchange only one CN packet significantly reduces the number of CN control packets. Therefore, more nodes get the opportunity to negotiate for the data channel.

5. Conclusions and Future Work

In this paper, we presented a MAC protocol based on CN window adjustment according to node density for fully connected single hop CRNs. In ad hoc CRNs, nodes may join or leave arbitrarily. In the dense network topology, a small CN window will 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. 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 performance of our proposed protocol depends on the accuracy of the estimated number of active CR nodes. Investigation of the estimation accuracy and comparison with other similar approaches is left to our future work.

References

[1] Timmers, M., Pollin, S., Dejonghe, A., Van der Perre, L., and Catthoor, F., “A distributed multichannel MAC protocol for multihop cognitive radio networks,” *IEEE Trans. Veh. Technol.*, vo. 59, no.1, pp. 446-459, 2010.

[2] Kamruzzaman, S.M., Hamid M.A., and Abdullah-Al-Wadud, M., “An energy-efficient MAC protocol for QoS provisioning in cognitive radio ad hoc networks,” *Radioengineering*, vo. 19, no.

4, pp. 567-578, 2010.

[3] Kamruzzaman, S.M., "An energy efficient multichannel MAC protocol for cognitive radio ad hoc networks," *International Journal of Commn. Netw. and Info. Security (IJCNIS),* vo. 2, no. 2, pp. 112-119, 2010.

[4] So, J., and Vaidya, N., "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," *In Proc. of ACM Int. Symp. MobiHoc*, NY, USA, pp. 222–233, 2004.

[5] Joshi, G.P., Nam, S.Y., and Kim, S.W., "Protecting incumbents and improving spectrum utilization in dynamic spectrum access networks," 2011, (submitted).

[6] Chatzimisios, P., Boucouvalas, A.C., and Vitsas, V., "IEEE 802.11 packet delay: A finite retry limit analysis," *In Proc. of IEEE GLOBECOM*, San Francisco, CA, vo. 2, pp. 950–954, 2003.

[7] IEEE 802.22 draft standard, http://www.ieee802.org/22/, July 2011.

[8] Nan, H., Hyon, T., and Yoo, S., "Distributed coordinated spectrum sharing MAC protocol for cognitive radio," *In Proc. of IEEE Symp. DySpan*, 2007, pp. 240–249.

[9] Wu, S.-L., Lin, C.-Y., Tseng, Y.-C., and Sheu, J.-P., "A new multi-channel MAC protocol with on-demand channel assignment for mobile ad-hoc networks," *In Proc. of ISPAN*, pp. 232–237, 2000.

[10] 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] Jung, E.-S., and Vaidya, N., "Improving IEEE 802.11 power saving mechanism," *Wireless Networks (WINET)*, vo. 14, no. 3, pp. 375-391, 2008.

[12] Wang, J., Fang, Y., and Wu, D., "A power-saving multi-radio multi-channel MAC protocol for wireless local area networks," *In Proc. of 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp.1-12, 2006.

[13] Cormio, C., and Chowdhury, K.R., "A survey on MAC protocols for cognitive radio networks," *Ad Hoc Networks*, vo. 7, no. 7, pp. 1315-1329, 2009.

[14] Yucek, T., and Arslan, H., "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surveys and Tutorials*, vo. 11, no.1, pp. 116-130, 2009.