Received April 15, 2019, accepted May 16, 2019, date of publication May 20, 2019, date of current version June 3, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2917906

Cluster-Based Distributed MAC Protocol for Multichannel Cognitive Radio Ad Hoc Networks

CHIEN-MIN WU, MENG-SIN WU, YOU-JIE YANG, AND CHENG-YUE SIE
Department of Computer Science and Information Engineering, Nanhua University, Chiayi 62248, Taiwan

Corresponding author: Chien-Min Wu (cmwu@nhu.edu.tw)

This work was supported by the Ministry of Science and Technology, Taiwan, under Grant MOST 106-2221-E-343-001.

ABSTRACT This paper proposes a cluster-based distributed medium access control (CDMAC) protocol for multichannel and multihop mobile cognitive radio ad hoc networks (CRAHNs). Secondary users (SUs) of the CRAHNs sense the spectrum to obtain licensed channels opportunistically from primary users (PUs) according to the channel-sensing model. The CRAHN is divided into clusters, with cluster heads determined according to the degree of importance of the nodes. The control and data channels are dynamically selected based on the success probability and stability of the idle PU channels. The traffic adaptively determines the contention window size, and a dynamic contention window mechanism is proposed to reduce the MAC contention delay. Furthermore, this mechanism achieves high throughput for a given contention window size. We also develop a Markov chain model to characterize the performance of our proposed CDMAC protocol for a saturated network. The Markov chain is also characterized by initial-state probabilities and a state-transition probability matrix. We perform extensive simulations to verify that the proposed CDMAC protocol can achieve higher performance in the mobile CRAHNs than other approaches.

INDEX TERMS Channel-sensing model, cluster, cognitive radio ad hoc networks, medium access control protocol, Markov chain, multichannel.

I. INTRODUCTION

The Federal Communications Commission has estimated that activity in the fixed spectrum ranges from 15% to 85%, which leads to low utilization [1]. A serious challenge is that the spectrum is a limited resource and difficult to use efficiently. Static spectrum allocation policies result in underutilization and spectrum shortages [2]. Furthermore, the rapid development of wireless communication technology will create difficulty in meeting spectrum demand. To solve this problems, Akyildiz *et al.* [1] proposed a method for accessing the spectrum in an opportunistic manner for wireless communications. This opportunistic access scheme is called a cognitive radio network (CRN). In CRNs, secondary users (SUs) can opportunistically utilize the spectrum of primary users (PUs) when it is idle.

In cognitive radio ad hoc networks (CRAHNs), the spectrum can be divided into several channels. A single-channel in a CRAHN has a limited maximum throughput, high propagation delay, and poor quality of service (QoS). By using multiple channels, CRAHNs can overcome these disadvantages. If one mobile SU can use a multichannel system, throughput can be increased faster than with single-channel system, and multichannel systems have less propagation delay per channel than a single-channel system. Furthermore, collision probabilities are reduced if multiple channels are applied. Finally, QoS is achieved more easily using multichannel system [3].

Although medium access control (MAC) protocols have been developed for single-channel [4] and multichannel systems for distributed ad hoc networks [5], [6], they are not directly applicable to the CRNs. This lack of compatibility occurs because the set of available channels for communication is always changing due to dynamic primary activity and because the set of available channels for each node differs depending on its spatial location [7].

For a multichannel MAC protocol, channel assignment and medium access are two important issues in a CRAHN. Channel assignment decides which channel can be used by which node. Medium access resolves contention and collision problems among nodes using a particular channel [3].

In multichannel CRAHNs, channels are unreliable owing to collisions between SUs and PUs. Therefore, MAC protocols are critical for avoiding collisions among SUs and

The associate editor coordinating the review of this manuscript and approving it for publication was Mahdi Zareei.

between SUs and PUs. A standard that has been widely accepted, based on the single-channel model, is IEEE 802.11 [8]. When using the IEEE 802.11 MAC protocol in CRAHNs, increasing the number of SUs decreases system performance, as it increases contention, which results in collisions between SUs.

Multichannel MAC protocols can be used to overcome contention and collision problems among SUs in CRAHNs. As technology advances, empowering a mobile node to access multiple channels has become feasible. Thus, we define a multichannel MAC protocol to have this capability.

A cluster structure can reduce the overhead of control channels and overcome the issue of mobility of SUs [9]. In CDMAC, time is divided into many intervals. All SU nodes are divided into several clusters, each of which cluster has a cluster head, some members, and some gateways. The cluster head is determined by the degree of importance of each node, which is determined in turn by the number of one-hop neighbors and switches to the PU channels.

In CDMAC, each cluster head maintains a channel status recording table. This table records the success probability history of the PU channel borrowed by the cluster head. The sensing results show that the channel with the highest priority is dynamically selected as the control channel. Higher sensing success probabilities have higher priority and lower sensing success probabilities have lower priority. In addition, the remaining idle channels can take as the usable data channels.

Due to the characteristics of CRAHNs hardware constraints, PU interference problems, collisions among SUs, and collisions between the SU and PU—the primary motivation underlying almost all previous MAC protocols is throughput awareness. However, efficient energy saving and delay awareness QoS MAC protocols are needed for emergency and public safety applications, vehicular communications, and novel applications of CRAHNs.

In this paper, we propose a CDMAC protocol to solve the aforementioned issues. The corresponding control channel can be dynamically selected, as the cluster head is determined according to the degree of importance of each node. The dynamic contention window frame length in CDMAC is adjusted based on real system traffic to reduce the MAC delay and energy consumption. Idle time slots in the contention window in CDMAC are deleted to reduce MAC delay and save energy.

The main contributions of the proposed CDMAC protocol are as follows:

1) It applies a "one node contract" concept based on a one-hop cluster to select the cluster head and control channel.
2) A dynamic control channel (DCC) is used to solve the bottleneck problem that occurs in saturated traffic.
3) A dynamic contention window is used to decrease contention-based MAC delays and reduce energy consumption.
4) A Markov chain model for CDMAC can be used to characterize performance for a saturated CRAHN.

The remainder of this paper is organized as follows: existing MAC protocols for CRAHNs are introduced in Section II. The system model is introduced in Section III. The proposed CDMAC protocol is introduced in Section IV, and the detailed principles of and steps in CDMAC are introduced in Section V. Throughput analysis using a Markov chain model for a saturated network is discussed in Section VI. Performance evaluation is discussed in Section VII, and Section VIII presents our conclusions.

II. EXISTING MAC PROTOCOLS IN CRAHNS

In [7], Debroy *et al.* proposed a distributed collision-free MAC (CFMAC) protocol for mobile CRAHNs under a dedicated control channel and fixed contention window. However, the dedicated control channels and fixed contention windows limited the system's throughput, increased energy consumption, and increased MAC contention delay.

In [10], Azarfar *et al.* proposed a method of multichannel MAC contention delay analysis in CRNs. This buffering MAC protocol showed outperformed a switching MAC protocol, as the exchange of control messages in the switching MAC protocol is more frequent than in the buffering MAC protocol.

In [11], Yadav and Misra proposed an optimal control channel assignment mechanism using k-hop clustering (k-CCCP). k-CCCP can react quickly to PU dynamics and reduce re-clustering. The handover sequence for the control channel is determined by the ranking of the PU channel. In addition, the authors proposed a scheme for finding the optimal value of k for clustering and minimizing the influence of PU dynamics.

In [12], Tang *et al.* proposed a cluster-based link recovery mechanism (CLR) for CRAHNs. CLR can recover a failed link and retransmit data until the route link is recovered. However, the control channel in CLR is fixed. In addition, CLR cannot support topology changes and suddenly active PU channels.

A MAC protocol for cluster-based CRAHNs is proposed in [13]. This method's cluster formation is based on idle PU channels and geographical location. Each SU node records the occupation history of idle PU channels to support neighbor discovery and cluster formation. This re-clustering scheme was also presented for mobile SU nodes.

In [14], Liu *et al.* proposed an opportunistic cluster-based control channel allocation for CRNs in which the dynamic control channel is determined based on changes in time and space.

In [15], Chen *et al.* proposed a cluster-based MAC protocol for CRNs called CogMesh, which is based on a mesh network. CogMesh provides neighbor discovery, cluster formation, and cluster maintenance mechanisms when the topology and PU state change. The cluster head and master control channel are determined by the maximal degree, that is, the number of neighbors. However, CogMesh has no mechanism for re-clustering due to node mobility.

In [16], Zareei*et al.* proposed a novel cross-layer mobility-aware (CMCS) MAC protocol for cognitive radio sensor networks (CRSNs). In CMCS, a spectrum and mobility-aware cluster formation and maintenance scheme is also proposed to overcome PU dynamics and SU mobility. CMCS integrated channel sensing into the physical layer and channel assignment into the MAC layer in CRSNs.

In [17], Sultana *et al.* proposed a traffic-adaptive synchronized cluster-based MAC (TAMAC) protocol for CRAHNs. Cluster formation in TAMAC is based on a cluster creation process [13]. In TAMAC, the MAC contention period has three phases−request-to-send (RTS), clear-to-send (CTS), and acknowledgment (ACK)−similar to the IEEE 802.11 DCF mechanism. The number of data exchange time slots equals the number of licensed PU channels. Each node exchanges control frames and transmits data with other SUs in the available licensed PU channels. Then, SUs in TAMAC can use licensed PU channels efficiently. However, TAMAC cannot support topology variations due to node mobility. In addition, the data exchange period cannot adaptively adjust to actual traffic.

In [18], Mansoor *et al.* proposed an efficient cluster model for CRAHNs using graph theory. The clustering scheme takes the spectrum of usable variations and defines it as a maximum edge biclique problem.

In [19], Mansoor *et al.* proposed a spectrum aware cross-layer (RARE) MAC protocol for CRAHNs. In RARE, a maximum edge biclique problem is also considered to divide the CRAHN into clusters. RARE also proposed cluster formation and control channel structure protocols to maintain a cluster-based CRAHN. A control channel is dynamically selected according to the channel-hopping sequence. For MAC, RARE employs a contention-free scheme. Each SU preassigned a mini-slot and used that slot to transmit data. However, when each SU occupies a mini-slot, the slot will be wasted if one potential node has no any packets to transmit. Therefore, fixed and preassigned mini-slots for data transmission will increase the end-to-end delay and decrease the system throughput.

Each cluster has a secondary cluster head to overcome re-clustering due to topology changes. The cluster head is determined by the cluster head determination factor (CHDF), which is calculated from the number of neighbors and idle PU channels [18], [19].

From the above studies, we know that many current problems remain in existing MAC protocols for CRAHNs. In addition, several novel applications, such as emergency and public safety applications, encourage the promotion and development of mobile CRAHNs.

To solve these remaining issues, we designed a CDMAC protocol to improve energy efficiency, reduce MAC delay, and increase system throughput. CDMAC can use idle PU channels efficiently by applying dynamic control channels and contention windows. The PU channel is opportunistically idled, thus allowing SUs to use the idle spectrum. The PU does not experience performance degradation as the spectrum is used opportunistically by SUs. The following sections will introduce the network model for our proposed CDMAC in multichannel CRAHNs.

III. SYSTEM MODEL

A. NETWORK ENVIRONMENT

In this section, we discuss the network environment for multichannel mobile CRAHNs. CDMAC sets the maximum size of the contention window to the tolerance range of the PUs in order to overcome interference to PUs [7]. The time structure in CDMAC is similar to that in the IEEE 802.11 PSM scheme.

In the proposed CDMAC protocol, CRAHNs are divided into several clusters, in which cluster head is determined according to the number of switches to idle PU channels and the number of one-hop neighbors. A dynamic control channel MAC protocol is designed to explore the multichannel CRAHN's performance. The selection of the control channel is determined by the probability of successful acquisition of the idle PU channels. When the collision probability of this control channel reaches a critical value, the new control channel will be dynamically reselected from the idle PU channels.

In [20], energy detection does not require a priori knowledge of the primary signal. Therefore, energy detection can simply measure the received signal in an observation period. Energy detection compares the received signal with a predetermined threshold to decide whether a signal is present. Energy detection is also easy to implement and commonly used as a spectrum sensing scheme for SUs to sense the status of PUs. Therefore, SUs in CDMAC are based on energy detectors, to sense licensed PU channel availability.

There are n PU channels in a CRAHN. These channels have the same bandwidth and no channels overlap. Therefore, there is one dynamic control channel and up to $(n-1)$ data channels in the proposed CDMAC system. The available licensed channels are known in advance by all SUs. Each SU is equipped with two half-duplex transceivers. One is used for the control channel and another is used for data transmission. The number of time slots in the sensing window is the number of channels of the PU. This control channel is not a fixed channel, but is dynamically selected according to the stability and usage of the idle PU channels.

B. CLUSTER FORMATION

When SU_i completes the PU sensing, it denotes the idled PU channels (IPCs) as IPC_{SU_i}. Fig. 1 shows the connectivity graph of a CRAHN with the IPC associated with each SU denoted in brackets [18], [19]. Here, SU_i senses IPCs and creates $IPC_{SU_i} = [1, 2, 3, 4, 6, 7, 8]$. Each SU in the CRAHN has its own IPC. In addition, SU_i shares IPC_{SU_i} and the one-hop neighbors list N_{SU_i} with one-hop neighbors.

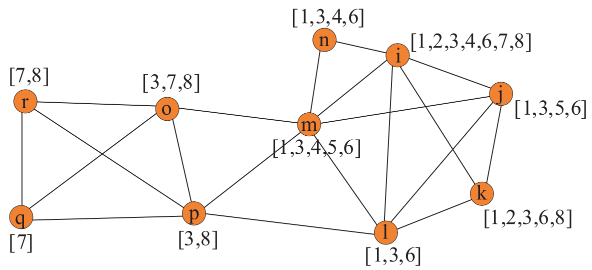

FIGURE 1. **Connectivity graph of a CRAHN with the idled PU channels in the brackets.**

Initially, SU_i creates an undirected bipartite graph based on IPC_{SU_i} and N_{SU_i}. A simple graph G is called bipartite if its vertex set V can be partitioned into two disjoint sets X and Y such that every edge in the graph connects a vertex in X and a vertex in Y. When this condition holds, we call the pair (X, Y) a bipartite of the vertex set V of G. For SU_i, $X_i = SU_i \bigcup N_{SU_i}$ and $Y_i = IPC_{SU_i}$. An edge (x, y) exists between vertices $x \in X_i$ and $y \in Y_i$. SU_i creates a bipartite graph $G_i(X_i, Y_i, E_i)$, as shown in Fig. 2(a). The set of vertices X_i corresponds to the one-hop neighbors $N_{SU_i} = [j, k, l, m, n]$ plus i and the set of vertices Y_i corresponds to IPC_{SU_i} [18], [19].

Fig. 2(b) shows the maximum edge biclique graph of SU_i, which is created from the bipartite of SU_i in Fig. 2(a). SU_i forms its maximum edge bipartite graph with one-hop neighbors of SU_j, SU_k, SU_l, SU_m and SU_n and channels [1, 3, 6]. Thus, every potential SU in the CRAHN creates its own maximum edge biclique graph [18], [19].

In [21], Huang *et al.* proposed a two-hop clustering scheme for mobile multihop CRNs. The cluster head is determined by the number of two-hop neighbors and the number of channels that are switched.

In [22], Ozger *et al.* proposed an event-to-sink spectrum-aware clustering in mobile CRSNs. The cluster head election scheme is defined as follows:

$$\gamma_i = w_n|N| + w_c|C| + w_e|E| + w_d|D| + w_s|S|, \tag{1}$$

where w_n, w_c, w_e, w_d, and w_s are the weight for node degree, available channels, reaming energy, distance to sink, and node speed, respectively. N, C, E, D, and S represent the node degree, the number of available channels, the remaining energy, distance to sink, and the node speed, respectively.

In [22], the weight coefficients in the simulations are $w_n = 0.4$, $w_c = 0.3$, $w_e = 0.1$, $w_d = 0.1$, and $w_s = 0.1$, which sum to 1.0. The sum of the node degree and available channel weights is 0.7, or 70% of all weight coefficients.

The one-hop cluster has smaller cluster members and coverage than a two-hop cluster [21], making the contention and collision in two-hop clusters more frequent. In our proposed CDMAC, cluster formation is based on a one-hop cluster. Here, β_i^m represents the degree of node importance for SU node i, which is the licensed channel m. Therefore, we have the degree of node importance as follows:

$$\beta_i^m = \frac{n_i^m}{1 + \frac{1}{n_i^m} \sum_{j=1}^{n_i^m} SW_j^m}, \tag{2}$$

where n_i^m denotes the number of one hop neighbors and SW_j^m denotes the number of switches to the licensed channel m of one-hop neighbor SU node j.

IV. CDMAC PROTOCOL

A. CDMAC CONTROL CHANNEL

Next, we describe the opportunistic spectrum access scheme of CDMAC. CDMAC can alleviate collision among SUs and has spectrum sensing capability. In the proposed CDMAC protocol, the beacon interval is divided into time slots to provide opportunistic spectrum access. Table 1 shows the symbols for our proposed CDMAC. Fig. 3 shows a CDMAC protocol control channel for a multichannel CRAHN. The two phases of CDMAC are as follows.

- Sensing window: Each SU senses the licensed channels based on the probability of sensing success. When the probability of successful sensing is larger than the threshold probability, the licensed channel is sensed by the SU. The previous three sensing outcomes are taken as calculation references. Here, we assume that each channel is different. Thus, the SUs must collect information for each channel. The estimation of probability of sensing success among SUs must be synchronized; otherwise, each SU will obtain different outcomes for each CRAHN channel.
  This synchronization is achieved using beacon exchanges during the sensing window. Channel sensing is performed based on the probability of success. Therefore, CDMAC reduces transition time and further decreases energy consumption.
- Contention window: This period involves the exchange of RTS, CTS, and licensed channel confirm (LCC) control frames. In addition, there is a cluster-head-confirm (CHC) for the cluster head to *JOIN* or to *LEAVE*

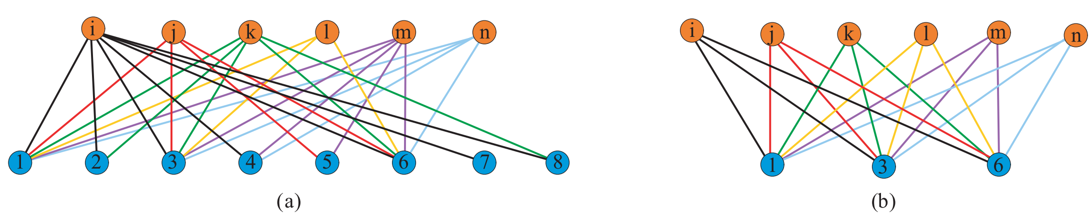

FIGURE 2. **Bipartite graph constructed by SU_i.**

TABLE 1. Symbols for our proposed CDMAC protocol.

Pr_{false}	Probability of a false alarm
Pr_{detect}	Probability of detection
H_1	Active licensed PU channel
H_0	Inactive licensed PU channel
u	Decision statistic
κ	Decision threshold
m	SU numbers for a considered cluster
Q	Q function
κ_{false}	False alarm threshold for channel sensing
κ_{detect}	Detection threshold for channel sensing
SU_i, SU_j	Any SU node
SU_{head}	Cluster head
β	Degree of node importance
$CH_{control}$	Control channel for cluster head
CH_{new}	New control channel for cluster head
$CH_{available}$	Number of available PU channels
PU_{active}	Licensed PU channels active
n	Number of PU channels
CH_{SU_i}	Number of usable channels for SU_i
RTS	Request-to-send control frame
CTS	Clear-to-send control frame
LCC	Licensed channel confirm control frame
CHB	Cluster head broadcast control frame
CHC	Cluster head confirm control frame
$Collision_{slot}$	Number of collision slots
$Empty_{slot}$	Number of empty slots
W_{rts}	Contention window size in RTS
$Pr_{succSensing}$	Success probability of sensing
P_{max}	Maximum transmission power of an SU
CH_{id}	Selected data channel
$Slot_{RTS}$	Initial frame length of RTS
$Slot_{CTS}$	Initial frame length of CTS
$Slot_{LCC}$	Initial frame length of LCC
$Slot_{succRTS}$	Number of successful slots in RTS contention period
$Slot_{free}$	Number of free slots in CTS/LCC contention period
$SIFS$	Short inter-frame space
$DIFS$	Time interval DCF inter-frame space

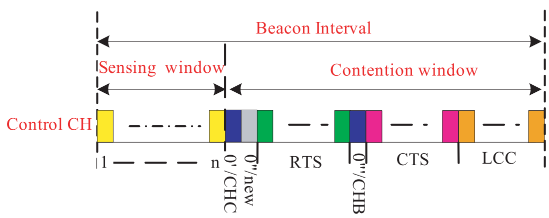

FIGURE 3. CDMAC control channel in multichannel CRAHN.

a cluster and a cluster-head-broadcast (CHB) for the cluster head to announce the sources that win the contention and assigned slots in the RTS contention window. *new* is for when a new node wants to *JOIN* or to *LEAVE* a cluster.

The frame lengths of RTS, CTS, and LCC in the contention window are initially the same, but are then adjusted according to actual traffic.

Each SU must send three control messages sequentially in the control channel before obtaining a data channel. CDMAC does not imply a global synchronized scheme because it exchanges information in the control channel during the sensing window. A new node can obtain the status of each channel by listening to the control channel. A new node must operate according to the sequence of the time slot in the sensing window. Therefore, a new node will not miss the collection of channel statuses in the one-hop transmission range.

B. SENSING CHANNEL MODEL

The probability that an SU considers a PU to be in the ON state when the PU is in the OFF state is denoted as the false alarm probability. The probability that SU considers the PU to be in the OFF state when it is in the ON state is denoted as the misdetection probability. The SU will have fewer usable PU channels in the false alarm state, thus reducing the efficiency of channel utilization. The SU will interfere with the PU as a result of misdetection [23], [24].

Let $r(k) = n(k)$ be the received signal under H_0 or $r(k) = s(k)+n(k)$ under H_1. When a licensed PU channel is inactive, we denote it by H_0. When a licensed PU channel is active, we denote it by H_1. Here, s(k), n(k), and r(k) are the PU's signal, noise, and received signal, respectively [24].

To evaluate the channel sensing performance, the probabilities of detection Pr_{detect} and false alarm Pr_{false} are defined as follows [24]:

$$Pr_{false} = Pr(H_1|actual = H_0) = Pr(u > \kappa|H_0) \quad (3)$$
$$Pr_{detect} = Pr(H_1|actual = H_1) = Pr(u > \kappa|H_1) \quad (4)$$

where u is the decision statistic and κ is the decision threshold. The value of κ is determined based on the requirements of the channel sensing performance.

Next, we derive the false alarm and detection probabilities. Here, we modify these terms based on the scheme proposed in [25]. We assume the noise and signal to be an i.i.d. random process with zero mean and variances of σ_n^2 and σ_s^2, respectively. The ratio of signal variance to noise variance is denoted as the signal-to-noise (SNR) ratio:

$$SNR = \frac{\sigma_s^2}{\sigma_n^2} \quad (5)$$
$$u = \frac{1}{m}\sum_{k=1}^{m}(r_k)^2 \quad (6)$$

where m is the SU numbers for a considered cluster.

The false alarm probability is given by the Q function under a given channel sensing threshold κ:

$$Pr_{false} = prob(u > \kappa|H_0) = Q(\frac{\kappa - \sigma_n^2}{\frac{\sigma_n^2}{\sqrt{\frac{m}{2}}}}) \quad (7)$$

$$Q^{-1}(Pr_{false}) = \frac{\kappa - \sigma_n^2}{\frac{\sigma_n^2}{\sqrt{\frac{m}{2}}}} \quad (8)$$

$$\frac{\sigma_n^2}{\sqrt{\frac{m}{2}}}Q^{-1}(Pr_{false}) = \kappa - \sigma_n^2 \quad (9)$$

$$\kappa = \sigma_n^2(1 + \frac{1}{\sqrt{\frac{m}{2}}}Q^{-1}(Pr_{false})) \quad (10)$$

The false alarm threshold of channel sensing κ_{false} can be set if false alarm probability Pr_{false} can be predetermined:

$$\kappa_{false} = \sigma_n^2(1 + \frac{Q^{-1}(P_{false})}{\sqrt{\frac{m}{2}}}). \quad (11)$$

Similarly, we can derive the channel sensing threshold to derive the detection probability based on H_1 and SNR as follows:

$$\kappa_{detect} = \sigma_n^2(1 + SNR)(1 + \frac{Q^{-1}(Pr_{detect})}{\sqrt{\frac{m}{2}}}). \quad (12)$$

The capacity loss caused by the misdetection of SUs decreases if more licensed channels can be used.

C. CONTENTION WINDOW DESCRIPTIONS

The control messages in the contention window are shown in Fig. 3. The sensing window and each control frame in the contention window occur and are transmitted sequentially when one connection is created between two SUs. Here, we provide the details of the control frames.

- CHC contains the following fields: DCC_{id}, $Head_{id}$ and $CH_1, \cdots, CH_n$. DCC_{id} denotes the ID for the control channel, $Head_{id}$ denotes the ID of the cluster head, and $CH_1, \cdots, CH_n$ denote the licensed channel IDs. CHC also performs synchronization.
- CHB contains the following fields: DCC_{id}, $Head_{id}$, and Src_{id}, $Slot_{id}$. Src_{id} denotes the source node ID that wins the contention in the RTS control frame. $Slot_{id}$ denotes the assigned slot that Src_{id} wins.
- RTS contains the following fields: CW_{node}, CH_1, CH_2, CH_3, SU_{snd}, SU_{rcv}, and $Nbr_1 \cdots, Nbr_n$. CW_{node} denotes the size of the contention window of $node$; CH_1, CH_2, and CH_3 have a higher success probability in the SU_{snd} state; SU_{snd} denotes that the SU sends an RTS; SU_{rcv} denotes that the SU receives an RTS; and $Nbr_1, \cdots, Nbr_n$ denote the IDs of the neighbors of SU_{snd}.
- CTS contains the fields CW_{node}, CH_{id}, SU_{snd}, SU_{rcv}, and $Nbr_1, \cdots, Nbr_n$, which are added to the CTS fields of IEEE 802.11. CW_{node} denotes the contention window size of the $node$. CH_{id} denotes the selected channel. SU_{snd} denotes that the SU sends a CTS, SU_{rcv} denotes that the SU receives a CTS, and $Nbr_1, \cdots, Nbr_n$ denote the IDs of the neighbors of SU_{snd}.
- LCC contains the following fields: CW_{node}, CH_{id}, SU_{snd}, and SU_{rcv}. CH_{id} denotes the coordinating channel that was agreed upon by SU_{snd} and SU_{rcv}.
- ACK contains the following fields: CH_{id}, SU_{snd}, and SU_{rcv}. CH_{id} denotes the released licensed channel between SU_{snd} and SU_{rcv}.

D. DYNAMIC CONTROL CHANNEL

In CDMAC, each SU exchanges control information on the dynamic control channel with other SUs. The SUs access the data channel from the exchange control information. The data transceiver dynamically switches to a suitable data channel and uses this channel to transmit data packets.

A separate channel for buffering and for switching MAC protocols is proposed in [10]. The SU changes to a new data channel to continue transmission while the PU of current data channel is used for switching the MAC protocol. When the PU is active, the SU remains in the current data channel until its transmission is complete, such that the MAC protocol can be buffered.

The SU releases the data channel once the transmission is terminated, irrespective of whether the MAC protocol is buffering or switching. The SU then enters the next contention window to contend for the right to use the data channel.

The buffering MAC protocol has a higher performance than the switching MAC protocol when a time-slotted MAC protocol is used [10]. Therefore, the CDMAC access scheme uses the buffering MAC protocol.

A real-time scheduling scheme in CRAHNs is proposed in [26]. Channel assignment is performed dynamically based on actual traffic and channel scheduling. The previous channel assignment and future traffic statuses determine this real-time channel assignment.

Algorithm 1 shows the dynamic control channel scheme in the proposed multichannel CDMAC protocol for multihop mobile CRAHNs.

V. DETAILED PRINCIPLES OF AND STEPS IN CDMAC

When SUs sense idle PU licensed channels, they can access them. For efficient sensing, the SUs should set a sensing success probability for each licensed channel. Each SU must record all the sensing outcomes for all licensed channels.

A. CHANNEL SENSING DURING SENSING WINDOW

To overcome interference to the PU, each SU must scan each licensed channel according to the sensing channel model. In CDMAC, each beacon interval involves sensing and has contention windows, and each SU should perform these processes sequentially to reserve a licensed channel.

Algorithm 2 shows the sensing channel model for our proposed multichannel CDMAC protocol for multihop mobile CRAHNs.

B. DYNAMIC CONTENTION WINDOW IN CDMAC

In CDMAC, the control frame MAC design and transmission procedures are based on IEEE 802.11. Each SU must exchange information through the control frame on the control channel to access a licensed channel.

In mobile CRAHNs, the frame length of the RTS field should adapt to actual traffic. Available RTS field slots in the contention window are insufficient when traffic is high.

Algorithm 1 Dynamic Control Channel of the CDMAC Protocol

Each SU sends degree of node importance β to its one-hop neighbors.
If one SU has the highest β among its one-hop neighbors, then this SU will announce itself as cluster head, SU_{head}.
A CHC control frame is performed in time slot $0'$ and sent to all cluster members.
SU_{head} determines control channel $CH_{control}$ using β.
if ($PU_{active} = CH_{control}$) **then**
    SU_{head} migrates $CH_{control}$ if PU is active on the control channel
    SU_{head} searches for new control channel CH_{new} based on β
    if ($CH_{available} = 0$) **then**
        Sense PU in next sensing window
    else
        SU_{head} announces CH_{new} as $CH_{control}$ to its cluster members
    end if
else
    continue
end if

Algorithm 2 Sensing Channel Model for the CDMAC Protocol

SU_i sets the sensing window size.
    /* n is the number of PU channels. */
for (j from 1 to n) **do**
    SU_i predetermines Pr_{detct}
    SU_i collects the neighbors' information.
    /* m is the SU numbers for SU_i cluster. */
    SU_i calculates m
    SU_i derives the spectrum sensing threshold λ_{detect}.
    SU_i senses PU channel j
    /* the sensing probability of detection in Eq.(4) */
    if ($H_1|actual = H_1$) **then**
    /* CH_{SU_i} denotes the usable channel for SU_i. */
        SU_i sets the number of usable channels, CH_{SU_i} ++
        SU_i records PU channel j in the channel status table
        if $CH_{SU_i} \geq 3$ **then**
            SU_i stops sensing
            exit
        end if
    else
    /* the sensing probability of false alarm in Eq.(3) */
        if ($H_1|actual = H_0$) **then**
            false alarm
        end if
    end if
end for

Likewise, idle slots in the RTS, CTS, and LCC fields in the contention window are wasted when traffic is low. In this case, idle time slots in all the fields are not used, increasing MAC delay and reducing system utilization.

Algorithm 3 shows the dynamic contention window for our proposed multichannel CDMAC protocol.

In TAMAC, the length of the MAC period is fixed according to the number of licensed PU channels. In RARE, the length of the MAC period is fixed according to the number of SU nodes. Thus, the consumption of idle slots increases under low traffic and MAC delays increase. Compared with RARE and TAMAC, CDMAC saves more energy per beacon interval and further reduces MAC delay; this is made possible by the dynamic lengths of the contention windows.

C. CHANNEL ASSIGNMENT IN MULTICHANNEL CDMAC

Each SU_j records the status of active PUs and occupies the licensed channels of other SUs. While one licensed channel CH_i can be used by SU_j, it must not be used by PUs and other SUs in the contention region.

$CH_i(j)$ shows whether channel i is usable by SU_j. If licensed channel i is occupied by other SUs in the contention region, then $CH_i(j)$ is set to 0. Otherwise, it is set to 1. $PU_i(j)$ indicates whether PU_i is active. If so, then $PU_i(j)$ is set to 0. Otherwise, $PU_i(j)$ is set to 1.

A transceiver for an SU exploits the spectrum holes in a set of channels, where each SU transceiver can use, at most, one available channel for communication. It efficiently utilizes spectrum opportunities to obtain a channel assignment. In non-overlapping channel assignment algorithms, a different channel is assigned to each SU. While the number of usable licensed channels is usually sufficient, maximum throughput can be achieved in this manner [27].

The reuse of assigned channels should occur as often as possible. Channel spatial reuse is increased when there are no hidden terminal problems. Each channel's usage status is tracked by exchanging a control frame. Therefore, CDMAC is an approach to determine compact channel spatial reuse patterns. A PU licensed channel status can be obtained by control frame exchange among SUs. However, this creates higher control overhead.

Algorithm 4 shows the channel assignment for our proposed multichannel CDMAC protocol for multihop mobile CRAHNs.

D. EXAMPLES OF THE CDMAC PROTOCOL IN MULTICHANNEL MOBILE CRAHNS

Fig. 4 (a) shows slots 0, 1, and 3 assigned by the SU nodes. One collision occurs in slot 2. If one RTS control frame occupies one time slot successfully, the CTS and LCC control

Algorithm 3 Dynamic Contention Window for the CDMAC Protocol

Initially, the frame lengths of RTS, CTS, and LCC are as same.
Set $Slot_{RTS} = Slot_{CTS} = Slot_{LCC}$
SU_i sends RTS to SU_j by selecting one slot randomly in the RTS contention period.
After the RTS contention period, all slots in the RTS contention period are in one of three states: successful, collision, or free.
Cluster head reassigns the slots sequence for each successful SU in CHB.
Cluster head statistics $Slot_{succRTS}$.
CHB is performed in time slot $0'''$ and sent to all cluster members.
Each SU in one-hop cluster receives the CHB.
All SUs in the cluster know the status of contention and the slot reassign sequence in the CTS and LCC contention periods.
The frame lengths of CTS and LCC are shortened to the number of successful slots in the RTS contention period.
Set $Slot_{CTS} = Slot_{succRTS}$ and $Slot_{LCC} = Slot_{succRTS}$
Set $Slot_{free} = Slot_{RTS} - Slot_{succRTS}$
All free slots in the CTS and LCC contention period are deleted.
The cluster head calculates the numbers of $Collision_{slot}$ and $Empty_{slot}$ in the RTS contention period.
if $(Collision_{slot} > 1)$ **then**
    Cluster head sends EFL to slot $0'$.
    The frame lengths of RTS, CTS, and LCC in the following beacon interval increase to twice the $Collision_{slot}$.
end if
if $(Empty_{slot} > 1$ and $Collision_{slot} = 0)$ **then**
    Cluster head sends SFL to slot $0'$.
    Set W_{rts} as the contention window size in RTS.
    The frame lengths of RTS, CTS, and LCC in the next beacon interval decrease to $\lfloor \frac{W_{rts}}{2} \rfloor$.
end if
The cluster head adjusts the new initial frame lengths of RTS, CTS, and LCC in the next beacon interval.
All members of the cluster head change their frame length.

frames occupy the same time slot with the RTS. Fig. 4 shows that the CTS and LCC contention window frames are empty in slot 2.

In Fig. 4 (b), in our proposed CDMAC protocol, the CRAHN is formed by a one-hop cluster. The cluster head knows whether the communication status of a cluster is a success or a collision. The CHB of the control frame in slot $0'''$ is then sent to all SU members. All empty slots in CTS and LCC contention window frames are then shrunk and the success slots are reordered according to the success sequence. The MAC delay is shortened after the CDMAC protocol is run.

Algorithm 4 Channel Assignment of the CDMAC Protocol

SU_i creates a connection with SU_j via handshaking
SU_i selects three idle channels that have the highest $Pr_{succSensing}$
Assume P_{max} is the same for all SUs
SU_i sends an RTS control frame in the RTS contention window to SU_j
SU_j receives the RTS from SU_i
/* CH_{SU_j} denotes the usable channel for SU_j */
if $CH_{SU_j} = 0$ **then**
    Set CH_{id} of CTS to empty
    connection fail
else
    SU_j selects one CH_{id} from candidate channels
    SU_j adds CH_{id} to CTS and sends to SU_i
    connection success
end if

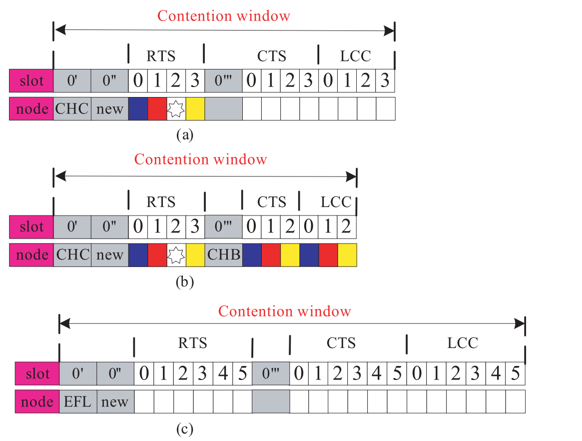

FIGURE 4. Cluster head enlarges the frame length (EFL) in a multichannel CRAHN. (a) The number of empty slots is zero and one collision occurs at slot 2 in the RTS contention window. (b) Cluster head broadcasts CHB control frame in $0'''$ slot and the idle time slots in CTS/LCC will be deleted. (c) Cluster head broadcasts EFL control frame in $0'$ slot to enlarge frame length in the next beacon interval.

In Fig. 4 (a) and (b), the RTS contention window frame has one slot collision, indicating that there are more than two RTS control frames in slot 2. To reduce the probability of collisions for the RTS control frame, CDMAC enlarges the contention window frame. In CDMAC, the number of enlarged slots is twice the number of collision slots. When the number of collision slots is one, the added time slots is two. Fig. 4 (c) shows that the contention window frames add two to four slots in the new control frame and then enlarge that to six slots in each RTS, CTS, and LCC control frame. Then, the cluster head sends the enlarged frame length (EFL)

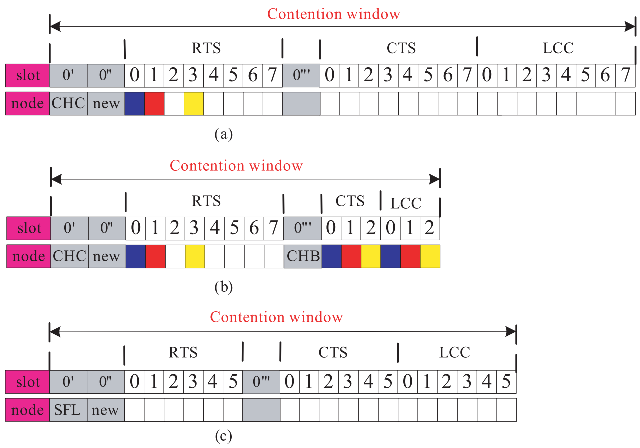

FIGURE 5. Cluster head shrinks the frame length (SFL) in a multichannel CRAHN. (a) The number of empty slots is five and no collision occurs in the RTS contention window. (b) Cluster head broadcasts CHB control frame in 0‴ slot and the idle time slots in CTS/LCC will be deleted. (c) Cluster head broadcasts SFL control frame in 0′ slot to shrink frame length in the next beacon interval.

control frame in slot 0′ in the next beacon interval and the size of the contention window changes to 6.

Fig. 5 (a) shows slots 0, 1, and 6 assigned by the SU nodes. No collision is observed in any slot. After the success of RTS handshaking, the handshaking of the CTS and LCC control frames is a success and occupies the same time slot as RTS in the CTS and LCC contention frames. From Fig. 5 (a), we know that there are five empty slots in the CTS and LCC contention window frames.

In Fig. 5 (b), the CHB of the control frames in slot 0‴ is sent to all SU members. All the empty slots in the CTS and LCC contention window frames are shrunk and the success slots reordered according to the success sequence. The MAC delay is shortened after CDMAC.

Fig. 5 (c) shows that the contention window frame reduces by two slots in the present control frame and shrinks to six slots in each RTS, CTS, and LCC control frame. The cluster head sends the shrink frame length (SFL) control frame in slot 0′ in the next beacon interval, and the size of the contention window changes to 6.

VI. THROUGHPUT ANALYSIS USING THE MARKOV CHAIN MODEL FOR A SATURATED NETWORK

Fig. 6 shows a continuous-time Markov chain (CTMAC) modeling a CDMAC system containing n PUs and m SUs. CDMAC assumes that no PU channels overlap. The Markov chain states are as follows:

- (P_i, S_j): The PUs and SUs can coexist to share the channel and the numbers of PUs and SUs are i and j, respectively.
- (P_i, S_{jw}): The number of active PUs is i and the waiting number of SUs is j.

Table 2 shows the symbols for throughput analysis and performance evaluation. The transitions in Fig. 6 are as follows.

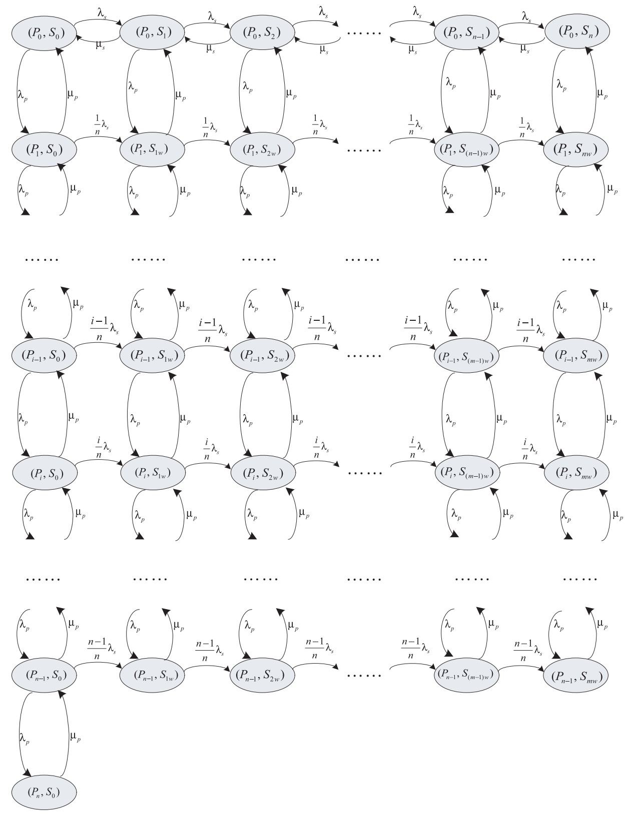

FIGURE 6. Markov chain modeling interference control with multiple PUs and multiple SUs.

When the spectrum band is occupied by an SU, if this SU determines that a PU needs to acquire the spectrum channel, it buffers the unfinished service session, sensing the licensed channel until the end of the PU's service session. For example, CTMC transits from state (P_0, S_1) to state (P_1, S_{1w}) at a rate of λ_p. If the PU finishes its service, the CTMC transits from state (P_1, S_{1w}) to (P_0, S_1) at a rate of μ_p. In addition, when the CTMC is in state (P_0, S_0), if an SU attempts to access the spectrum, it continuously senses the licensed channel until the PU vacates and the CTMC transits to state (P_1, S_{1w}) at a rate of λ_s.

If a PU finishes its service before an SU has accessed the channel, the CTMC transits from state (P_1, S_{1w}) to (P_0, S_1) at a rate of μ_p. In contrast, if a second SU requests access to the licensed spectrum before the PU completes its service, the second SU also buffers its service session and the CTMC transits to state (P_1, S_{2w}) at a rate of λ_s. In state (P_1, S_{2w}), both SUs continuously sense the spectrum. Once the PU vacates, the CTMC transits to state (P_0, S_2) at a rate of μ_p, where two SUs share the spectrum channel. Also, when the CTMC is in state (P_0, S_0), if SUs attempt to access the spectrum, they continuously sense the licensed channel until the PU vacates and the CTMC transits to state (P_1, S_{1w}) at a rate of λ_s.

The Markov chain model of CDMAC is illustrated in Fig. 6. Here, we evaluate the CDMAC Markov chain model

TABLE 2. Symbols for throughput analysis and performance evaluation.

P_i	Number of PUs
S_j	Number of SUs
S_{jw}	Waiting number of SUs
μ_p	Departure rate for PUs
μ_s	Departure rate for SUs
λ_p	Arrival rate for PUs
λ_s	Arrival rate for SUs
$\pi_{(i,j)}$	Steady state probability
ϵ	Channel utilization
$Pr_{available}(i)$	Probability for i available PU channels
$CH_{available}$	Number of available PU channels
Pr_{idle}	Idling probability
Pr_{succ}	Success probability
Pr_{coll}	Collision probability
γ	Probability of SU selecting a slot time randomly
T_s	Time for successful transmission
T_c	Time for collision transmission
T_{ms}	Mini-slot time units
$E[T]$	Average time of handshakes
$E[CH_{available}]$	Average number of available PU channels
ε	Channel spatial reuse
R_{data}	Data rate for a PU licensed channel
R_{ctrl}	Data rate for a control channel
$E_{succ}(k)$	Successful energy consumption of kth hop
PW_{rts}	RTS power consumption
PW_{cts}	CTS power consumption
PW_{LCC}	LCC power consumption
T_{rts}	RTS transmission time
T_{cts}	CTS transmission time
T_{LCC}	LCC transmission time
$E^j_{coll}(k)$	Collision energy consumption for jth trial of kth hop
C_{succ}	Total number of successful connections
n_{trial}	Number of trials for the RTS control frame
hop_i	Number of hops at the ith connection
$E_{tolconsum}$	Total energy consumption
$E[E_{consum}]$	Per-hop energy consumption
$t^j_{delay}(k)$	Hop contention delay of the jth trial for the kth hop
$E[T_{delay}]$	Delay per hop

using the following equations:

$$\begin{cases} \pi_{0,0} + \cdots + \pi_{0,m} + \pi_{1,0} + \cdots + \pi_{1,mw} + \\ \cdots + \pi_{i,0} + \cdots + \pi_{i,mw} + \cdots + \\ \pi_{n-1,0} + \cdots + \pi_{n-1,mw} + \cdots + \pi_{n,0} = 1 \\ \mu_p \pi_{1,0} + \mu_s \pi_{0,1} = \alpha_1 \pi_{0,0} \\ \lambda_s \pi_{0,j-1} + \mu_p \pi_{1,jw} + \mu_s \pi_{1,j+1} = \alpha_2 \pi_{0,j} \\ \lambda_s \pi_{0,m-1} + \mu_p \pi_{1,mw} = \alpha_3 \pi_{0,m} \\ \lambda_p \pi_{i-1,0} + \mu_p \pi_{i+1,0} = \alpha_4 \pi_{i,0} \\ \lambda_p \pi_{i-1,jw} + \frac{i}{n} \lambda_s \pi_{i,(j-1)w} + \mu_p \pi_{i+1,jw} = \alpha_5 \pi_{i,jw} \\ \lambda_p \pi_{i-1,mw} + \frac{i}{n} \lambda_s \pi_{i,(m-1)w} + \mu_p \pi_{i+1,mw} = \alpha_6 \pi_{i,mw} \\ \lambda_p \pi_{n-2,0} + \mu_p \pi_{n,0} = \alpha_7 \pi_{n-1,0} \\ \lambda_p \pi_{n-2,jw} + \frac{n-1}{n} \lambda_s \pi_{n-1,(j-1)w} = \alpha_8 \pi_{n-1,jw} \\ \lambda_p \pi_{n-2,mw} + \frac{n-1}{n} \lambda_s \pi_{n-1,m-1} = \mu_p \pi_{n-1,mw} \\ \lambda_p \pi_{n-1,0} = \mu_p \pi_{n,0} \\ i = 1, \cdots, n-2 \\ j = 1, \cdots, m-1 \end{cases} \tag{13}$$

where

$$\begin{aligned} \alpha_1 &= \lambda_s + \lambda_p \\ \alpha_2 &= \mu_s + \lambda_s + \lambda_p \\ \alpha_3 &= \mu_s + \lambda_p \\ \alpha_4 &= \mu_p + \lambda_p + \frac{i}{n}\lambda_s \\ \alpha_5 &= \mu_p + \lambda_p + \frac{i}{n}\lambda_s \\ \alpha_6 &= \mu_p + \lambda_p \\ \alpha_7 &= \mu_p + \lambda_p + \frac{n-1}{n}\lambda_s \\ \alpha_8 &= \mu_p + \frac{n-1}{n}\lambda_s \end{aligned}$$

The first normalization equation in Eq.(13) should satisfy a Markov chain. The flow balance for each state is indicated by the other ten equations. The steady state probabilities for $(P_0, S_0), \cdots, (P_n, S_m)$ are denoted by $\pi_{(i,j)}, (i, j) \in \{(0, 0), \cdots, (n, m)\}$.

Let there be a row vector with elements $\pi_{(i,j)}$ be $\pi = [\pi_{(0,0)}, \cdots, \pi_{(0,m)}, \cdots, \pi_{(i,0)}, \cdots, \pi_{(i,mw)}, \cdots, \pi_{(n-1,0)}, \cdots, \pi_{(n-1,mw)}, \pi_{(n,0)}]$. Then, the Markov chain model equations can be re-written as

$$A\pi = B, \tag{14}$$

where $B = (1, 0, \cdots, 0)^T$. Matrix A is defined in Eq.(16), as shown at the top of the next page.

Therefore,

$$\pi = A^{-1}B. \tag{15}$$

In [28] and [29], licensed channels were analyzed to derive the saturation throughput. Let ϵ be channel utilization.

The probability of i available PU channels is

$$Pr_{available}(i) = \sum_{j=1}^{m} \pi_{(i,j)} = \begin{cases} \binom{n}{i}(1-\epsilon)^i \epsilon^{n-i} \\ \text{when } m \geq n \text{ and } 0 \leq i \leq n; \\ \binom{m}{i}(1-\epsilon)^i \epsilon^{m-i} \\ \text{when } m < n \text{ and } 0 \leq i \leq m; \\ 0 \\ \text{otherwise.} \end{cases} \tag{17}$$

Therefore, the average number of available licensed channels can be derived as follows:

$$E[CH_{available}] = \sum_{i=1}^{n} iPr_{available}(i) = \sum_{i=1}^{n}\sum_{j=1}^{m} i\pi_{(i,j)}. \tag{18}$$

Let γ be the probability that one SU selected a slot time randomly.

Let Pr_{idle}, Pr_{succ}, and Pr_{coll} be the probabilities of idle, successful, and collision scenarios, respectively, for a transmitted *RTS* packet [28]:

$$Pr_{idle} = (1-\gamma)^m \tag{19}$$

$$Pr_{succ} = m\gamma(1-\gamma)^{m-1} \tag{20}$$

$$Pr_{coll} = 1 - (1-\gamma)^m - m\gamma(1-\gamma)^{m-1} \tag{21}$$

$$
\begin{bmatrix}
1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & 1 & 1 & \cdots & 1 & \cdots & 1 & \cdots & 1 & 1 & 1 \\
-\alpha_1 & \mu_s & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & \mu_p & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & \cdots & \lambda_s & -\alpha_2 & \mu_s & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & & & & \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & \lambda_s & -\alpha_3 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\cdots & & \cdots & & & & \cdots & & & & \cdots & & & & & & & \cdots \\
\vdots & & \vdots & & & & \vdots & & & & \vdots & & \vdots & & & & & \vdots \\
\cdots & & \cdots & & & & \cdots & & & & \cdots & & \cdots & & & & & \cdots \\
0 & 0 & \cdots & 0 & 0 & 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 & \lambda_p & 0 & \cdots & 0 & 0 & \mu_p
\end{bmatrix} \quad (16)
$$

Let T_s and T_c be the average time for which a transmission is successful or a collision occurs, respectively. Let RTS, CTS and LCC be the control frame sizes of RTS, CTS and LCC, respectively. Therefore, we obtain:

$$T_s = \frac{RTS + CTS + LCC}{R_{control}} + 2SIFS + DIFS \quad (22)$$

$$T_c = \frac{RTS}{R_{control}} + DIFS \quad (23)$$

While the $RTS/CTS/LCC$ of CDMAC is exchanged successfully, the average time of handshakes can be expressed as follows [28]:

$$E[T] = \frac{Pr_{idle}T_{ms} + Pr_{succ}T_s + Pr_{coll}T_c}{Pr_{succ}}, \quad (24)$$

where T_{ms} is the mini-slot time units in a sensing window. When $E[CH_{available}]$ and $E[T]$ are completely calculated, we use them to obtain the throughput, as described in the section that follows.

VII. PERFORMANCE EVALUATION OF MULTICHANNEL MOBILE CRAHNS

In this section, we present simulation results for our proposed CDMAC-based one-hop cluster and dynamic control channel, and dynamic contention window size. The simulation is implemented using the C programming language. The main difference among CDMAC, RARE, and TAMAC is the contention window design. For CDMAC, the contention window size dynamically adjusts with the actual traffic. The contention window size of TAMAC is based on the number of licensed PU channels. The contention-free period in RARE is based on the number of SUs.

In TAMAC, the frame length of the contention window is determined by the number of licensed PU channels. Each node exchanges control frames with other nodes in available licensed PU channels. If the PU is active/nonactive dynamically, the system performance will be affected significantly. In addition, the contention window size cannot adjust according with SUs' actual traffic. Thus, the end-to-end delay of TAMAC increases when collisions occur under high traffic loads, which decreases system performance.

In this paper, the system model has one control channel and idle licensed PU channels are data channels. Each node must exchange control frames and then negotiate one idle licensed PU channel to transfer data. In RARE, each SU node in the contention-free period uses a preassigned time slot to transfer data. The contention-free period in RARE includes the intra-cluster and inter-cluster contention periods. Each node occupies one time slot in the intra-cluster and one time slot in the inter-cluster contention period, respectively.

To compare CDMAC, TAMAC and RARE fairly, we set each SU node in the contention-free period in RARE to first use a preassigned time slot to exchange control frames. The contention-free period includes two contention-free subperiods. All SU pairs negotiate the idle licensed PU channels by exchanging control frames in the preassigned time slots. After control frame negotiation in the contention-free period, the SUs pair transfer data using one obtained idle licensed PU channel. In addition, the frame length of the contention-free period for each cluster in RARE is based on the number of SUs in that cluster. Therefore, idle time slots in RARE increase under low traffic loads.

In addition, due to the time-varying characteristics of traffic loads and PU state, RARE and TAMAC schemes result in low resource transmission efficiency. TAMAC is also less adaptive to network topology changes due to node mobility.

In this section, we compare the performances of CDMAC with RARE and TAMAC. For our proposed CDMAC protocol, the cluster formation is one-hop and the contention window is dynamic, changing according to actual traffic. The transmission rates of each data channel and the control channel are 2 and 1 *Mbps*, respectively. The power consumption of the control frames for transmitting and receiving are 1,675 and 1,425 *mw*, respectively. Because the contention window is different for our CDMAC protocol and other MAC protocols, we consider the following metrics under different contention window design schemes. In this simulation process, we set 1 slot as 1 s. Table 3 shows the parameters of our proposed CDMAC multichannel MAC protocol in mobile CRAHNs.

The random waypoint model is used as a mobility model in this paper [30]. An SU randomly selects a destination point and moves with constant velocity in a straight line to the destination. Then, the SU pauses for a time before it selects a new destination. The mobility velocity is selected between $[v_{min}, v_{max}]$ under a uniform distribution.

TABLE 3. Parameters for our proposed CDMAC scheme.

Simulation time	10000 s
Number of SUs	400
Bounded region	600 $m \times$ 600 m
Transmission range of SU	200 m
Transmission range of PU	300 m
Number of PUs	6, 8, 10
Number of active PUs	0, 1, 2, 3
Mean of "PU ON" duration	300 s
Departure rate	0.5
Arrival rate	$1, \cdots, 512$
PU sensing error	0%, 30%
Minimum velocity of mobility, v_{min}	1 m/s
Maximum velocity of mobility, v_{max}	20 m/s
Mobility interval	300 s
Number of mobile SUs for each time	8
Maximum trial times before success	3

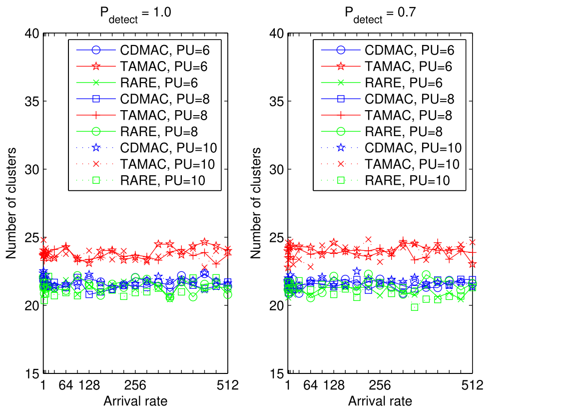

FIGURE 7. Comparison of the number of clusters in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

In our simulation, ten topologies are created using different seeds. All presented simulation results are the average values of the ten seeds. Traffic is assumed to be uniformly distributed across all nodes with various overall loads. The number of newly created connections per second is denoted by arrival rate. The number of terminated connections per second is denoted by departure rate. The inverse of departure rate is also the average lifetime of a connection. "PU ON" denotes that a PU is in the active state. "PU OFF" denotes that a PU is in the idle state.

Here, we take the following items as our performance evaluation metrics: number of clusters, re-clustering effect, throughput, per-hop energy consumption, and delay per hop for MAC contention.

A. NUMBER OF CLUSTERS

The number of clusters can be an efficacy index for cluster-based CRAHNs. Fewer clusters are preferable [19].

Fig. 7 shows that the number of clusters in CDMAC, TDMAC, and RARE ranges from 20 to 25, with averages 21.5, 23.9, and 21.4, respectively. The main cause of this difference is that cluster formation in TAMAC is based on licensed PU channels.

B. RE-CLUSTERING EFFECTS

Topology changes as SU nodes move in or out of a cluster. Re-clustering is performed when a cluster head leaves the cluster. The amount of re-clustering determines the stability of a cluster-based CRAHN, which is more robust with less re-clustering. In RARE, re-clustering is decreased by using a second cluster head.

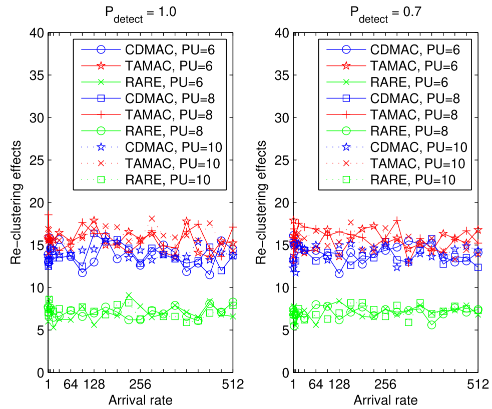

FIGURE 8. Comparison of the re-clustering effects in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

Re-clustering is affected by cluster head mobility. In Fig. 8, re-clustering occurs 13.8, 15.8, and 7.1 times on average in CDMAC, TAMAC, and RARE, respectively. In TAMAC, cluster formation is based on licensed PU channels and the PU state is dynamic. Therefore, re-clustering effects in TAMAC are larger than in CDMAC and RARE. In RARE, a cluster head leaving a cluster is simply replaced by the second cluster head. Therefore, RARE has less re-clustering than CDMAC and TAMAC.

C. THROUGHPUT

Assume there are n licensed channels and m SUs. When SU_j can use idle licensed channel i, we set $C_{single}(i,j)$ to 1. Otherwise, we set $C_{single}(i,j)$ to 0. If at least one SU can use idle licensed channel i, we set $C_{multiple}(i)$ to 1. Otherwise, we set $C_{multiple}(i)$ to 0.

Channel spatial reuse ε is defined as the number of SUs using one idle licensed channel simultaneously. Hence, ε is defined as follows:

$$\varepsilon = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} C_{single}(i,j)}{\sum_{i=1}^{n} .C_{multiple}(i)}. \tag{25}$$

In [29], Bianchi derived a performance analysis of IEEE 802.11. Tan and Le [27], Sadreddini [28] analyzed the saturation throughput of licensed channels.

The idle time slots of the CTS and LCC contention window of CDMAC are deleted. The RTS contention window size is adapted to actual traffic. The throughput for this CRAHN is defined as ζ, as follows:

$$\zeta = \frac{\varepsilon R_{data} T_s E[CH_{available}]}{(n-1)R_{data} + R_{ctrl}}. \tag{26}$$

Therefore, the throughput per contention window size for this CRAHN is defined as $\zeta_{windowSize}$, as follows:

$$\zeta_{windowSize} = \frac{\varepsilon R_{data} T_s E[CH_{available}]}{((n-1)R_{data} + R_{ctrl})E[T]}, \tag{27}$$

where ε denotes channel spatial reuse, R_{data} denotes the data rate for a PU licensed channel, and R_{ctrl} denotes the data rate for a control channel. T_s and $E[T]$ are defined in Eq.(22) and Eq.(24), respectively.

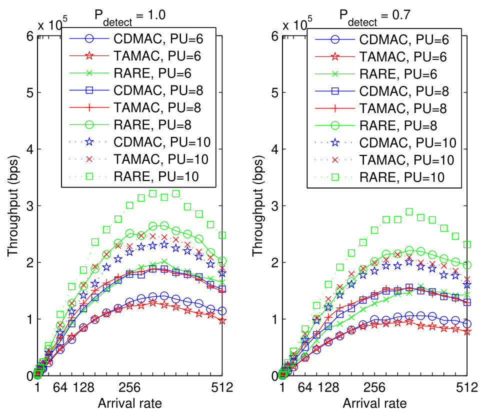

FIGURE 9. Comparison of throughput in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

Fig. 9 shows the throughput index ζ in CDMAC, TAMAC and RARE under mobility velocities of 1−20 m/s and two active PUs in CRAHNs with 6, 8, and 10 licensed PU channels. When using two PU channels in the active ON state and considering $Pr_{detect} = 1.0$, the largest throughput in CDMAC is 231,219 *bps* for an arrival rate of 350 with 10 PU channels. The throughput of TAMAC when using two active PUs and considering $Pr_{detect} = 1.0$ is 247,207 *bps* for an arrival rate of 288 with 10 PU channels. The throughput of RARE when using two active PUs and considering $Pr_{detect} = 1.0$ is 321,259 *bps* for an arrival rate of 384 with 10 PU channels. RARE has higher throughput than the CDMAC and TAMAC schemes because of its contention-free period with preassigned time slots for each cluster member, which decreases collision probability. From Fig. 9, we know that the throughputs of CDMAC, TAMAC, and RARE increase with the number of PU channels and decrease at lower detect probability.

Fig. 10 shows the throughput per contention window size index $\zeta_{windowSize}$ in CDMAC, TAMAC, and RARE under mobility velocities of 1−20 m/s with two active PUs in CRAHNs with 6, 8 and 10 licensed PU channels. When using two PU channels in the active ON state and considering $Pr_{detect} = 1.0$, the largest throughput per contention window size in CDMAC is 19,363 *bps* for an arrival rate of 288 with 10 PU channels. The throughput per contention window size of TAMAC when using two active PUs and considering $Pr_{detect} = 1.0$ is 8,240 *bps* for an arrival rate of 288 with 10 PU channels. The throughput per contention window size of RARE when using two active PUs and

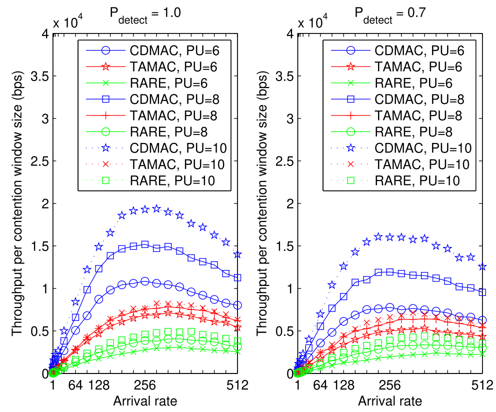

FIGURE 10. Comparison of throughput per contention window size in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

considering $Pr_{detect} = 1.0$ is 4,919 *bps* for an arrival rate of 320 with 10 PU channels. CDMAC has higher throughput per contention window size than TAMAC and RARE schemes because it has a lower contention window size and end-to-end delay. From Fig. 10, we know that the throughput per contention window size of CDMAC, TAMAC, and RARE is higher when the number of PU channels increases and is lower when detection probability decreases.

From Fig. 10, the largest improvements in throughput per contention window size in CDMAC compared with TAMAC and RARE, considering $Pr_{detect} = 1.0$ at 10 PU channels, are 135.0% and 293.6%, respectively.

D. PER-HOP ENERGY CONSUMPTION FOR MAC CONTENTION

Energy consumption occurs under three conditions: idling, successful transmission, and collision [31]. Here, we focus on the energy consumed during successful transmissions (E_{succ}) and collisions (E_{coll}). In addition, all SUs are assumed to always be powered on.

- Let $E_{succ}(k)$ denote the kth hop energy consumption of a successful transmission. Let PW_{rts}, PW_{cts},and PW_{lcc} denote the power consumed while RTS, CTS,and LCC are transmitted, respectively. Let T_{rts}, T_{cts}, and T_{lcc} be the transmission times for RTS, CTS,and LCC, respectively. Thus, E_{succ} is computed as follows [31]:

$$E_{succ}(k) = T_{rts}PW_{rts} + T_{cts}PW_{cts} + T_{lcc}PW_{lcc}. \tag{28}$$

- Let $E^j_{coll}(k)$ denote the jth trial for the kth hop energy consumption due to collision. For each connection, there are j attempts for trial sending RTS/CTS/LCC handshaking. Therefore, the j trial means that there are $(j-1)$ failures and 1 success. In CDMAC, energy consumption due to collision transmission occurs in the RTS field of the contention window. Hence, E^j_{coll} is given as follows [31]:

$$E^j_{coll}(k) = T_{rts}PW_{rts}. \tag{29}$$

The number of hops at the ith connection is denoted by hop_i. Therefore, $E[E_{consum}]$ can be computed as follows [31]:

$$E_{tolconsum} = \sum_{j=1,conn=i}^{n_{trial}} E_{coll}^{j-1}(k) + E_{succ}(k)) \qquad (30)$$

$$E[E_{consum}] = \frac{\sum_{i=1}^{C_{succ}} \sum_{k=1}^{hop_i} E_{tolconsum}}{\sum_{i=1}^{C_{succ}} hop_i}, \qquad (31)$$

where C_{succ} denotes the total number of successful connections and the n_{trial} is the number of trials for the RTS control frame in the MAC contention window before RTS/CTS/LCC handshaking is successful.

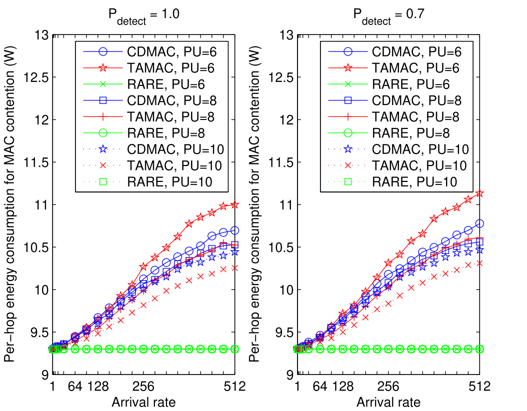

FIGURE 11. Comparison of per-hop energy consumption for MAC contention of SUs in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

Fig. 11 shows the per-hop energy consumption in CDMAC, TAMAC, and RARE under mobility velocities of 1−20 m/s and two active PUs in CRAHNs with 6, 8 and 10 licensed PU channels. We observed that the per-hop energy consumption in CDMAC ranged from 9.3 W to 10.4 W when using two PU channels in the active ON state and considering $Pr_{detect} = 1.0$ with 10 licensed PU channels. With the same parameters in TAMAC, the per-hop energy consumption ranged from 9.3 W to 10.3 W. For RARE, the per-hop energy consumption is 9.3 W. The per-hop energy consumption of CDMAC is similar to that of TAMAC. The per-hop energy consumption of RARE is always 9.3 W because of its contention-free design. In addition, per-hop energy consumption of CDMAC and TAMAC increased when PU sensing errors increased.

E. DELAY PER HOP FOR MAC CONTENTION

Let $E[T_{delay}]$ be the average MAC delay per hop. The total time taken for successful and complete reception is defined as the MAC delay per connection.

In the buffering MAC protocol, the total MAC delay before a packet is entirely transmitted denotes the MAC delay per connection. The MAC delay is the successful $RTS/CTS/LCC$ handshakes in the contention window. Let $t_{delay}^{j}(k)$ denote the jth trial for the kth hop contention delay for a connection in a MAC contention. Therefore, $E[T_{delay}]$ can be computed as follows [10]:

$$E[T_{delay}] = \frac{\sum_{i=1}^{C_{succ}} \sum_{k=1}^{hop_i} \sum_{j=1,conn=i}^{n_{trial}} t_{delay}^{j}(k)}{\sum_{i=1}^{C_{succ}} hop_i}, \qquad (32)$$

where C_{succ} denotes the total number of successful connections.

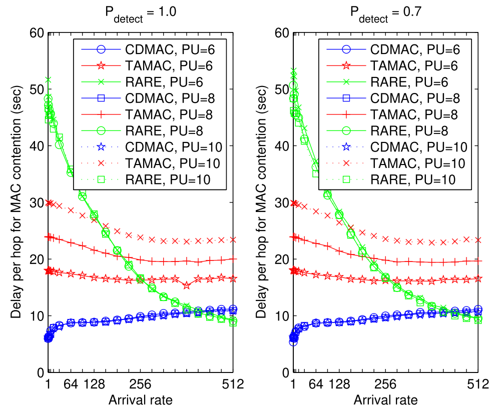

FIGURE 12. Comparison of delay per hop for MAC contention of SU in CDMAC, TAMAC and RARE versus arrival rate in a multichannel CRAHN.

Fig. 12 shows the delay per hop for MAC contention in CDMAC, TAMAC, and RARE under mobility velocities 1−20 m/s and two active PUs in CRAHNs with 6, 8 and 10 licensed PU channels. The delay per hop for a MAC contention in CDMAC ranges from 6.2 to 10.9 slots when using two PU channels in the active ON state and considering $Pr_{detect} = 1.0$ with 10 licensed PU channels. Under the same conditions, the delay per hop for MAC contention in TAMAC and RARE ranges from 30.0 to 23.4 slots and 46.6 to 8.7. The largest improvements in the delay per hop for MAC contention in CDMAC compared with TAMAC and RARE are 79.3% and 86.7%, respectively.

MAC collisions occurring in TAMAC with a greater number of active PUs are more serious than in CDMAC due to the fewer idle time slots in the contention window period. RARE has higher throughput and delay per-hop and lower throughput per contention window size. MAC collision occurring in RARE is less frequent than in CDMAC and TAMAC due to its contention-free period design, which utilizes preassigned time slots. Therefore, RARE performs better under higher traffic loads than under lower traffic loads. In addition, RARE has fewer re-clustering effects due to its cluster head mobility. TAMAC has lower throughput per contention window size than CDMAC but has higher throughput per contention window size than RARE. CDMAC has higher throughput per contention window size and lower delay per-hop than TAMAC and RARE.

VIII. CONCLUSION

This paper proposed a one-hop cluster-based dynamic control channel and dynamic contention window to achieve a high throughput per contention window size, energy-efficient, and low MAC contention delay per hop protocol. Per-hop delay reductions in CDMAC were found to be greater than those in TAMAC and RARE because the dynamic contention window in CDMAC saved more idle slots. In addition, the CDMAC protocol provides lower MAC contention collision than TAMAC. CDMAC also saved more time slots than RARE, which used preassigned time slots. The proposed CDMAC scheme effectively achieved not only lower MAC delay per hop, but also higher throughput per contention window size than TAMAC and RARE. In addition, CDMAC has higher per-hop energy efficiency than TAMAC with fewer licensed PU channels. Simulation results showed that the largest improvements in per-hop MAC delay reduction compared with TAMAC and RARE are 79.3% and 86.7%, respectively (for two active PU channels and ten licensed PU channels). Additionally, CDMAC saved idle slots, improving throughput per contention window size. In the simulation results, we observed that the maximum throughput per contention window size in CDMAC compared to TAMAC and RARE improved to 135.0% and 293.6%, respectively (for two active PU channels and ten licensed PU channels).

ACKNOWLEDGMENT

The authors would like to thank the editor and reviewers for their valuable comments and suggestions.

REFERENCES

[1] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," *Comput. Netw.*, vol. 50, pp. 2127–2159, Sep. 2006.

[2] P. Ren, Y. Wang, Q. Du, and J. Xu, "A survey on dynamic spectrum access protocols for distributed cognitive wireless networks," *EURASIP J. Wireless Commun. Netw.*, vol. 1, pp. 1–21, Dec. 2012.

[3] 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 *Proc. I-SPAN*, Dallas, TX, USA, Dec. 2000, pp. 232–237.

[4] A. Muqattash and M. Krunz, "POWMAC: A single-channel power-control protocol for throughput enhancement in wireless ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 5, pp. 1067–1084, May 2005.

[5] C.-Y. Chang, C.-T. Chang, L. Zhao, Z. Dong, and C.-C. Chen, "A stepwise multichannel MAC protocol for improving bandwidth utilization in wireless ad hoc networks," *IEEE Syst. J.*, vol. 11, no. 4, pp. 2444–2455, Dec. 2017.

[6] G. M. Júnior and L. H. A. Correia, "SM3-MAC: A multichannel collision-free MAC protocol for wireless sensor networks," in *Proc. ISCC*, Natal, Brazil, Jun. 2018, pp. 550–555.

[7] S. Debroy, S. De, and M. Chatterjee, "Contention based multichannel MAC protocol for distributed cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 12, pp. 2749–2762, Dec. 2014.

[8] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE Standard 802.1 1-1997, IEEE Inc., New York, NY, USA, 1997.

[9] Y. Saleem, K.-L. A. Yau, H. Mohamad, N. Ramli, M. H. Rehmani, and Q. Ni, "Clustering and reinforcement-learning-based routing for cognitive radio networks," *IEEE Wireless Commun.*, vol. 24, no. 4, pp. 146–151, Aug. 2017.

[10] A. Azarfar, J. F. Frigon, and B. Sansò, "Delay analysis of multichannel opportunistic spectrum access MAC protocols," *IEEE Trans. Mobile Comput.*, vol. 15, no. 1, pp. 92–106, Jan. 2016.

[11] R. N. Yadav and R. Misra, "Approximating common control channel problem in cognitive radio networks," *IEEE Syst. J.*, vol. 13, no. 1, pp. 301–312, Mar. 2019.

[12] C.-C. Tang, K.-F. Ssu, and C.-H. Yang, "A cluster-based link recovery mechanism for spectrum aware on-demand routing in cognitive radio ad hoc networks," in *Proc. ICS*, Hualien, Taiwan, Dec. 2012, pp. 601–660.

[13] X. Li, F. Hu, H. Zhang, and X. Zhang, "A cluster-based MAC protocol for cognitive radio ad hoc networks," *Wireless Pers. Commun.*, vol. 69, no. 2, pp. 937–955, 2013.

[14] S. Liu, L. Lazos, and M. Krunz, "Cluster-based control channel allocation in opportunistic cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 11, no. 10, pp. 1436–1449, Oct. 2012.

[15] T. Chen, H. Zhang, G. Maggio, and I. Chlamtac, "Cogmesh: A cluster-based cognitive radio network," in *Proc. 2nd IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw.*, Dublin, Ireland, Apr. 2007, pp. 168–178.

[16] M. Zareei, A. K. M. M. Islam, N. Mansoor, S. Baharun, E. M. Mohamed, and S. Sampei, "CMCS: A cross-layer mobility-aware MAC protocol for cognitive radio sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 1, pp. 1–15, Jul. 2016.

[17] S. Sultana, A. Hossain, M. O. Rahman, and M. Jusoh, "Traffic adaptive synchronized cluster based MAC protocol for cognitive radio ad hoc network," in *Proc. ICEESI*, Penang, Malaysia, Oct. 2017, pp. 1–7.

[18] N. Mansoor, A. K. M. M. Islam Islam, M. Zareei, and S. Baharun, "Cluster modelling for cognitive radio ad-hoc networks using graph theory," in *Proc. EAME*, Xiamen, China, May 2015, pp. 805–809.

[19] N. Mansoor, A. K. M. M. Islam, M. Zareei, and C. Vargas-Rosales, "RARE: A spectrum aware cross-layer MAC protocol for cognitive radio ad-hoc networks," *IEEE Access*, vol. 6, pp. 22210–22227, 2018.

[20] S. K. Yoo, S. L. Cotton, P. C. Sofotasios, S. Muhaidat, O. S. Badarneh, and G. K. Karagiannidis, "Energy detection-based spectrum sensing over Fisher–Snedecor F fading channels," in *Proc. IEEE GLOBECOM*, Abu Dhabi, United Arab Emirates, Dec. 2018, pp. 1–6.

[21] X.-L. Huang, G. Wang, F. Hu, and S. Kumar, "Stability-capacity-adaptive routing for high-mobility multihop cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 6, pp. 2714–2729, Jul. 2011.

[22] M. Ozger, E. Fadel, and O. B. Akan, "Event-to-sink spectrum-aware clustering in mobile cognitive radio sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 9, pp. 2221–2233, Sep. 2016.

[23] S. Maleki, A. Pandharipande, and G. Leus, "Energy-efficient distributed spectrum sensing for cognitive sensor networks," *IEEE Sensors J.*, vol. 11, no. 3, pp. 565–573, Mar. 2011.

[24] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Phys. Commun.*, vol. 4, no. 1, pp. 40–62, Mar. 2011.

[25] Z. Ye, G. Memik, and J. Grosspietsch, "Energy detection using estimated noise variance for spectrum sensing in cognitive radio networks," in *Proc. WCNC*, Las Vegas, NV, USA, Mar. 2008, pp. 711–716.

[26] S. Sodagari, "Real-time scheduling for cognitive radio networks," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2332–2343, Sep. 2018.

[27] L. T. Tan and L. B. Le, "Channel assignment for throughput maximization in cognitive radio networks," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 1427–1431.

[28] Z. Sadreddini, Ö. Makul, T. Çavdar, and F. B. Günay, "Performance analysis of licensed shared access based secondary users activity on cognitive radio networks," in *Proc. EBBT*, Istanbul, Turkey, Apr. 2018, pp. 1–4.

[29] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 3, pp. 535–547, Mar. 2000.

[30] A. Pramanik, B. Choudhury, T. S. Choudhury, W. Arif, and J. Mehedi, "Simulative study of random waypoint mobility model for mobile ad hoc networks," in *Proc. GCCT*, Tamil Nadu, India, Apr. 2015, pp. 112–116.

[31] D. Jung, R. Kim, and H. Lim, "Power-saving strategy for balancing energy and delay performance in WLANs," *Comput. Commun.*, vol. 50, pp. 3–9, Sep. 2014.

CHIEN-MIN WU was born in Yunlin, Taiwan, in 1966. He received the B.S. degree in automatic control engineering from Feng Chia University, Taichung, Taiwan, in 1989, the M.S. degree in electrical and information engineering from Yuan Ze University, Chungli, Taiwan, in 1994, and the Ph.D. degree from the Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan, in 2004. In 1994, he joined the Technical Development Department, Philips Ltd. Co., where he was a Member of the Technical Staff. He is currently a Professor with the Department of the Computer Science and Information Engineering, Nanhua University. His current research interests include cognitive radio networks, ad hoc networks, and MAC protocol design.

MENG-SIN WU was born in Yunlin, Taiwan, in 1996. He received the B.S. degree in computer science and information engineering from Nanhua University, Chiayi, Taiwan, in 2018. His research interests include cognitive radio networks, ad hoc networks, and MAC protocol design.

YOU-JIE YANG was born in Xinbei, Taiwan, in 1994. He received the B.S. degree in computer science and information engineering from Nanhua University, Chiayi, Taiwan, in 2018. His research interests include cognitive radio networks, ad hoc networks, and MAC protocol design.

CHENG-YUE SIE was born in Taoyuan, Taiwan, in 1995. He received the B.S. degree in computer science and information engineering from Nanhua University, Chiayi, Taiwan, in 2019. His research interests include cognitive radio networks, ad hoc networks, and MAC protocol design.