Zareei *et al. EURASIP Journal on Wireless Communications and Networking* (2016) 2016:160
DOI 10.1186/s13638-016-0652-y

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

CMCS: a cross-layer mobility-aware MAC protocol for cognitive radio sensor networks

Mahdi Zareei[1], A. K. M. Muzahidul Islam[1*], Nafees Mansoor[1], Sabariah Baharun[1], Ehab Mahmoud Mohamed[3] and Seiichi Sampei[2]

Abstract

Cognitive radio sensor networks (CRSNs) are multi-channel-capable networks that inherit some of the challenges of traditional wireless sensor networks (WSNs), such as limited power source and hardware capacity. In several CRSN applications, such as surveillance and intelligent transportation systems, node mobility is a typical assumption. However, as a node changes its physical location, spectrum mobility may also follow. Therefore, the treating of node mobility in CRSN imposes new challenges on all network layers, especially in the data link layer. In this paper, we propose a novel cross-layer mobility-aware medium access control (MAC) protocol for CRSN. We also propose an efficient spectrum-aware cluster formation and maintenance. The proposed scheme is more robust against primary users' activity as well as node mobility in a CRSN because it integrates spectrum sensing at the physical (PHY) layer with packet scheduling at the MAC layer. Simulation results show that the proposed protocol guarantees about 60 % more common channels per cluster in a higher node ratio. Moreover, the proposed MAC protocol outperforms existing protocols (e.g., CogMesh, cluster-based MAC, and KoN-MAC) in terms of the packet delivery ratio, energy consumption, and delay, by up to 5, 30, and 25 %, respectively.

Keywords: Cognitive radio sensor network, Cognitive radio ad hoc network, Cognitive radio network, Medium access control, Mobility management, Clustering

1 Introduction

Stemming from the emergence of new wireless products and services over the last decade, the problem of spectrum scarcity has attracted a lot of attention. Static spectrum allocation to wireless devices causes an abundance of temporal and geographical unused spectrum in the licensed bands [1, 2]. These temporarily unused portions of the licensed spectrum are called spectrum holes [3]. A spectrum hole is defined as a frequency band that is assigned to a licensed user, but which is unutilized at a particular time or location. To address this critical problem, the Federal Communications Commission (FCC) has approved unlicensed users or secondary users (SUs) to opportunistically use spectrum holes and co-exist with licensed users, while ensuring that the primary users (PUs) or licensed users of the spectrum are not affected [4, 5]. To this end, dynamic spectrum access (DSA) techniques are proposed, which lead to cognitive radio (CR) as a promising technology that can overcome this issue, improving the overall spectrum usage by utilizing spectrum access in both the licensed and unlicensed bands [6].

Cognitive radio networks (CRNs) can be classified as infrastructure-based CRNs and cognitive radio ad hoc networks (CRAHNs) [6, 7]. Infrastructure-based CRNs employs a central network entity to manage the network, e.g., base stations (BSs) in cellular networks. On the other hand, CRAHNs do not have any infrastructure backbone, and CR users can communicate with each other via an ad hoc connection [8]. In contrast with centralized CRNs, a distributed cognitive radio network can be a more appropriate selection because of its lower system complexity, faster align positioning, and lower implementation cost. The cognition concept has been applied to many ad hoc technologies such as mobile ad hoc networks (MANETs) [9], wireless mesh networks (WMNs) [10], and vehicular

---

*Correspondence: muzahidul.kl@utm.my
[1] Electronic Systems Engineering Department, Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM), Jalan Sultan Yahya Petra, 54100 Kuala Lumpur , Malaysia
Full list of author information is available at the end of the article

ad hoc network (VANET) [11]. To enjoy the potential benefits of CRs, the CR sensor network (CRSN) paradigm has also been investigated.

A CRSN [12] is a multi-channel capable network and has two basic differences from existing conventional wireless sensor network (WSN). The first difference is that in a CRSN, the number of available channels varies within a given time, where this attribute is fixed in existing WSNs. The second difference is that the set of available channels differs for each node in a CRSN. In traditional WSNs, all nodes of a single network are expected to use the same set of available channels. Because of these differences, protocol stacks developed for traditional WSNs may not work properly for CRSNs. Specifically, the medium access control (MAC) mechanism defines medium access approaches that should be customized to cope with CRSN challenges.

Some traditional WSN challenges, such as low energy and hardware limitation, are inherent to CRSN, increasing the complexity of spectrum management in a CRSN. Current solutions for the cognitive radio network concept do not consider the energy and hardware limitations. Moreover, node mobility plays a vital role in several applications, and dealing with node mobility in energy-limited processes involves making a careful trade-off between energy efficiency, throughput, and robustness under node mobility [13]. Recent research on CRAHN and CRSN have focused on channel management schemes [14, 15], packet size optimization [16], and reliability and congestion control system in the network framework [17].

For the successful establishment of CRSNs, we need to design a robust and intelligent MAC protocol. MAC protocol architecture for CRSNs must be able to intelligently adapt to the unique characteristics of such networks to maintain robust performance even in the presence of a dynamic environment. To achieve better performance in cooperative networks such as CRSNs, the application of cross-layer optimization is found to be useful [18, 19].

Moreover, clustering is seen as an efficient way to design network architecture to facilitate efficient network protocols by introducing self-organized cell structures, where each cell is led by a cluster head (CH). This concept reduces the communication overhead cost and increases the reliability. Moreover, clustering can be defined as a structured way to manage topology effectively and to increase the system capacity and stability [20]. For the past few years, many researchers have extensively investigated clustering techniques for WSNs and wireless ad hoc networks [21, 22].

In this paper, we introduce a novel cross-layer mobility-aware MAC protocol that integrates spectrum sensing at the physical (PHY) layer with packet scheduling at the MAC layer. The proposed cross-layer protocol enables the network to be more robust against PUs' activities as well as the mobility of SU nodes. We introduce a novel spectrum sensing strategy to keep track of the PU channel status and to increase the probability of finding the right channel for the SUs' data transmission. Using this method, SU nodes can save time and energy in the spectrum-sensing phase. We also introduce a parameter called the cluster-head election value (CHEV), which simultaneously considers the energy level of a sensor, its current speed, available PU channels, and the number of its neighboring nodes. CHEV is used as a robust metric for cluster formation, which makes the proposed clustering approach more effective. Furthermore, the use of the CHEV parameter guarantees a large number of common channels per cluster, and it also makes the clusters more robust to changes in network topology due to SU mobility as well as spectrum mobility due to PU activity, where restructuring/re-clustering is less dominant. The simulation results show that the proposed protocol outperforms similar protocols in terms of throughput, power consumption, and packet transmission delay. Table 1 summarizes the important abbreviations used in this paper.

The rest of the paper is organized as follows: Section 2 provides an overview of the mobility challenges in CRSN. In Section 3, we discuss related work on the CRSN MAC protocol, while in Section 4, we clarify the system model. In Section 5, we present the proposed cross-layer MAC protocol, and in Section 6, we introduce cluster formation and maintenance. We show the simulation results and discussion in Section 7, and in Section 8, we conclude the paper and discuss future works.

2 Mobility in CRSN

In CRSN, one of the major issues is the disruption caused by nodes' mobility. Thus, there is a need to deal with varying network topologies as mobility affects the steady state of a network. The mobility of nodes affects both the link density and the link stability of a network, and a self-correcting, self-organizing, and self-architecting CRSN has to overcome the various degrees of state disturbance caused by the nodes' mobility.

The term "link stability" refers to the stability of the links between a given node and its neighbors. The lifetime of a link or its transmission quality is a parameter that can be used to measure the stability of any link [23]. A node is connected to the broader CRSN by its neighbors, so neighborhood links are the basic building blocks of a CRSN. Neighborhood link stability is a requirement for self-organizing end-to-end protocols at the higher layers as a lack of stability at the link level hinders certain types of transactions in higher layers.

In CRSN, there are two types of mobility that can affect the link stability. The first type is spectrum mobility, and it occurs when a PU appears and occupies the licensed band. As a result, CR users have to move to another spectrum

Table 1 Abbreviation

Acronym	Description
ACK	Acknowledgment
ACL	Available channel list
ACTS	Acknowledgment clear to send
BS	Base-station
CCC	Common control channel
CHEV	Cluster-head election value
CHs	Cluster heads
CM	Cluster member
CnHS	Channel hoping sequence
CR	Cognitive radio
CRAHNs	Cognitive radio ad hoc networks
CRNs	Cognitive radio networks
CRSN	Cognitive radio sensor network
CSMA	Carrier-sensing multiple access
CSMA/CA	Carrier-sensing multiple access/collision avoidance
CTS	Clear to send
DSA	Dynamic spectrum access
FCC	Federal Communications Commission
GW	Gateway nodes
HPM	Historical prediction model
MAC	Medium access control
MANET	Mobile ad hoc network
NCC	Next control channel
OSA	Opportunistic spectrum access
PER	Primary exclusive region
PHY	Physical
PUs	Primary users
RSS	Receiver spectrum sensing
RSSI	Radio signal strength indicator
RTS	Ready to send
SNR	Signal-to-noise ratio
SUs	Secondary users
TSF	Timer synchronization function
TSS	Transmitter spectrum sensing
VANET	Vehicular ad hoc network
WMNs	Wireless mesh networks
WSN	Wireless sensor network

hole to avoid interference with the PU while maintaining the connection. This is referred to as spectrum mobility. The second type is user mobility, which occurs when a node changes its physical location, resulting in changes in the network topology. Because a network consists of static and mobile nodes, it can be challenging to maintain the connectivity between nodes. Moreover, when a node changes its physical location, spectrum mobility may also follow. Therefore, user mobility and spectrum mobility need to be considered equally when designing a mobility-management scheme for a cluster-based CRSN. Furthermore, the frequency or channel availability in CRSN varies over time and space, which makes it more difficult for mobile users to traverse and provide seamless and reliable communications across multiple clusters.

3 Related work

A large number of MAC protocols have already been published for WSN over the last decade [24–26]. However, traditional and CR-based WSNs have many differences, which makes the solution for WSN unsuitable and infeasible for application in CRSN. CRSN is also different from multi-channel WSNs. In CRSN, PUs always have priority of access to the spectrum, and there is therefore a need for some level of synchronization to ensure the detection of the presence of PUs. In contrast to CRSN, all users in multi-channel WSNs have the same priority when using a channel, which means that there is no need for synchronization. However, in a distributed environment where the sensing undertaken by the users is generally asynchronous, it is hard to achieve network-wide synchronization. To overcome the spectrum management challenges in CRSN, several nodes need to interact and cooperate using novel design techniques.

In recent years, the design of an efficient MAC protocol for CRN has attracted a lot of attention, and many studies have been proposed to attain this goal [27–30].

A MAC protocol with a mobility support (cluster member (CM)-MAC) is presented in [27, 31], which is a carrier-sense multiple access/collision avoidance (CSMA/CA)-based MAC protocol with mobility support that uses a radio signal strength indicator (RSSI) to help CRs to know and respond to their vicinity to a primary exclusive region (PER) of PUs. It uses a dedicated common control channel (CCC) in order to exchange control frames such as ready to send (RTS), clear to send (CTS), and acknowledgment (ACK) frames. Two spectrum sensing procedures are employed, namely transmitter spectrum sensing (TSS), which is performed by a CR transmitter, and receiver spectrum sensing (RSS), which is completed on the receiver side. Both the transmitter and receiver combine the spectrum information into RTS/CTS frames, and the neighboring CRs of the transmitter and receiver can therefore obtain the local knowledge of one-hop spectrum availability from the broadcasted RTS/CTS frames. The TSS/RSS procedure that is integrated in RTS/CTS/ACTS handshaking is sufficient to ensure that all neighboring CRs can receive the spectrum information, which ensures the successful next one-hop transmission. CM-MAC tries to choose the appropriate set of channels, and it subsequently splits the data payload into multiple segments and transmits on multiple channels simultaneously to improve the throughput. It implements distributed spectrum information exchange instead of using the central coordination, which is similar to IEEE 802.22 standard [32].

Decentralized predictive MAC (P-MAC) protocol [28] is a synchronization-based multi-transceiver and multi-channel MAC protocol. It assumes that a dedicated CCC is available, and each CR device is equipped with two half-duplex transceivers, namely the control transceiver and data transceiver. Distributed sensing is used to determine the network-wide spectrum opportunity. Time synchronization is performed to pause or stop the secondary transmission and to determine the network-wide spectral opportunities. P-MAC uses the timer synchronization function (TSF), as in the IEEE 802.11 MAC protocol, for synchronization. In addition, P-MAC uses a novel channel-selection algorithm that is based on the historical prediction model (HPM). In P-MAC, nodes share sensing information and maintain knowledge of the channels' status across the entire network to ensure cooperative sensing and to protect the rights of the PUs. Similar to 802.22, P-MAC follows two types of sensing: (i) fast sensing, which takes a very short time with low accuracy, and (ii) fine sensing, which improves the sensing accuracy and decreases the incidence of false alarm and increases the quality of service (QoS). The data transceiver is also responsible for fine sensing and fast sensing and remains in the doze state whenever there is no data to send or receive. It also keeps the transceiver active for synchronization 60 % of the time, and in the other 40 %, the transceiver remains in the doze state in order to increase the energy conservation.

The optimal cross-layer cognitive MAC protocol (OCC-MAC) [29] is an opportunistic spectrum access (OSA)-type MAC protocol for multi-hop CRAHNs. It adopts the slotted p-persistent CSMA algorithm, where all control messages are exchanged through the CCC. In OCC-MAC, the link's transmit power and persistence probability are jointly adjusted on the basis of the ingress rate that is regulated by sources to maximize the total net revenue of the secondary system, while keeping the licensed users' collision probability below a tolerable threshold. OCC-MAC focuses on a slotted random-access system, where the time is divided into fixed length intervals and all cognitive nodes are synchronized and start their transmissions only at the beginning of each time slot. OCC-MAC tries to balance the interference level and the contention level among cognitive links to utilize spectrum opportunities in a unified optimization framework. Despite its optimality, the overhead congestion and unsuccessful reception of control messages on the CCC may be a key bottleneck to the deployment of OCC-MAC in practice.

A cluster-based MAC protocol for CR ad hoc networks is introduced in [30], which forms clusters based on the geographical positions of nodes, available channels, and experienced statistics to maintain the cluster stability. They propose an experience database that stores spectrum-occupancy statistics to support neighbor discovery and cluster formation. The CH in each cluster broadcasts a beacon packet that contains cluster control information such as time synchronization. For each transmission pair, CHs define the channel access schedule. The nodes that do not participate in the communication tune to different channels based on the experience database to perform neighbor discovery. Therefore, cluster-based MACs do not require dedicated neighbor discovery. However, during neighbor discovery, each node gathers neighbors' information and assigns a specific value to each link to which it is connected. Then, nodes share their link values among themselves and depending on the link value nodes start forming clusters.

CogMesh, which was proposed by Chen et al. [33], is the most well-known cluster-based MAC protocol for CRN and opportunistically utilizes different spectrum holes for smooth peer-to-peer communication. CogMesh forms clusters with nodes that share local common channels, where multi-channel multi-access networks are considered. A node forms a cluster using the common channel of the cluster, which is called the master channel and invites the neighboring nodes that have the same channel to join the cluster. For intra-cluster communication, the CogMesh MAC protocol uses the guaranteed access period and the random access period for the control message exchange. Although CogMesh shows promising performance in statistic CRN, because of the absence of a mobility-handling mechanism, it faces performance degradation in dynamic networks. Because of its popularity, we consider this protocol as a benchmark reference for our simulation studies.

Despite the potential of the existing MAC protocol for CRN, they cannot be directly applied in CRSNs because of the challenges with respect to limited resources and hardware capacity, which are inherent in typical sensor nodes. In addition to the above-mentioned challenges, there are additional challenges to be dealt with when considering the heterogeneous spectrum availability. Moreover, there is also the need for a proper mechanism to handle the nodes' mobility. Therefore, the MAC protocol for CRSN should be carefully designed considering all limitations of typical sensor nodes as well as the need to fulfill the typical CRSN requirements. Recently, some researchers tried to overcome these challenges by proposing MAC protocols that were specifically designed for CRSN, such as in [34, 35].

Xu et al. [34] proposed a schedule-based protocol known as KoN-MAC, which is a cluster-based MAC protocol for multi-hop CRSN. KoN-MAC allows sensor nodes to dynamically select an interference-free channel for data communication. KoN-MAC uses a channel-weighting concept to distinguish between the channels and makes the adjacent clusters select different channels for data transmission, which consequently mitigates the multi-channel hidden-terminal problem. To achieve better energy efficiency, nodes sense a smaller set of channels. However, owing to the transmission scheduling and generated traffic classes, KoN-MAC suffers from poor QoS provisioning.

Maleki et al. [35] presented an energy-efficient distributed spectrum-sensing technique for CRSNs, which acts based on the combination of censoring and sleeping schemes. By optimally choosing the sleeping and sensing design parameters, subject to constraints on the detection performance, the authors tried to minimize the overall energy consumption. The constraint on the detection performance is defined as a minimum target probability of detection and a maximum permissible probability of false alarms. The signal-to-noise ratio (SNR) in [35] is assumed to be known and equal for all sensing nodes, which means that the probability of detection for all sensors is the same. However, this is not a realistic assumption as each node has a different distance with PU.

In this work, we propose an efficient MAC protocol by introducing a spectrum-aware clustering scheme to divide the network into several stable clusters. This enables us to reduce the network control overhead. Most of the existing clustering schemes did not consider spectrum-aware clustering [33, 34], which results in the formation of unstable clusters and the wastage of network resources owing to frequent reclustering/restructuring. Moreover, we have integrated the spectrum sensing at the PHY layer with packet scheduling at the MAC layer, and nodes can therefore have up-to-date information about the spectrum availability before starting their transmission. The proposed cross-layer protocol uses the adoptive data period to handle the nodes' mobility and to enable them to be more robust against spectrum mobility caused by PUs' activities. Furthermore, to conserve energy, the proposed protocol keeps track of the PU channel status during the time to increase the probability of finding the right channel for the SUs' data transmission. Using this method, the SU nodes can save time and energy during the spectrum-sensing phase.

4 System model

Figure 1 shows the CRSN system architecture, where we assume that there are N_{SU} cognitive radios and N_{PU} primary users deployed in CRSN. A given number of non-overlapping orthogonal channels $\{\mathrm{Ch}|Ch_i, i = 1, 2, \ldots, n\}$ are available, and each channel has a unique channel ID. Each node is aware of its location, and each node has a single half-duplex cognitive radio transceiver, which is capable of detecting and utilizing spectrum holes in a distributed and efficient way. Cognitive radios or SUs coexist with PUs, and they opportunistically and conditionally access the channels.

A discrete-time Markov chain is employed to PUs' channel-usage patterns [36], which means that the PU's may change their state (i.e., channel usage) after each process or step. To obtain a common time reference to synchronize the SUs, GPS clock has been used. A dedicated control channel is assumed available for all nodes for control packets. The channel availability for each node is related to the node's physical location. Similar to the IEEE 802.22 standard, SUs use an available channel only when it is not occupied by PUs. By detecting the PU's presence, SUs vacate the channel. To prevent interference between PUs and SUs, a simple interference avoidance model is used in the proposed network [37].

In Fig. 1, the clustering architecture for CRSN was considered. After clustering the network, to reduce the overhead in the dedicated control channel, each cluster has its own control channel that will be chosen from the available channel set. Therefore, one of the shared channels will be used for control purpose, while one of the remaining channels will be used for data transmission purpose. Nodes that form the cluster become CHs, which are responsible for inter-cluster communication as well as intra-cluster channel access control. Inter-cluster communication is relayed by gateway nodes (GWs), which are nodes that are in the border of two neighboring clusters and which can hear both cluster beacons, as shown in Fig. 1. Two SU nodes are considered as a one-hop neighbor if they are within each other's transmission range.

5 Proposed cross-layer mobility-aware cognitive radio sensor MAC protocol

By considering the challenges introduced in Section 2, we propose a new cross-layer MAC protocol for CRSN, which is called the cross-layer mobility-aware cognitive radio sensor (CMCS) MAC protocol. By considering the structure of the IEEE 802.15.4 standard superframe for low-power networks, we divide the channel access time into a sequence of superframes, which are shown in Fig. 2. In this figure, vertical slots represent independent actions that will be performed individually. Beacon section defines the information included in the beacon packet. The detailed operation of the superframe is given in the following subsections.

5.1 Beacon period

The CH first generates a random back-off time to avoid collision with the PUs (a licensed user may suddenly occupy the beaconing spectrum or another CH may generate a beacon signal). If the spectrum remains idle when

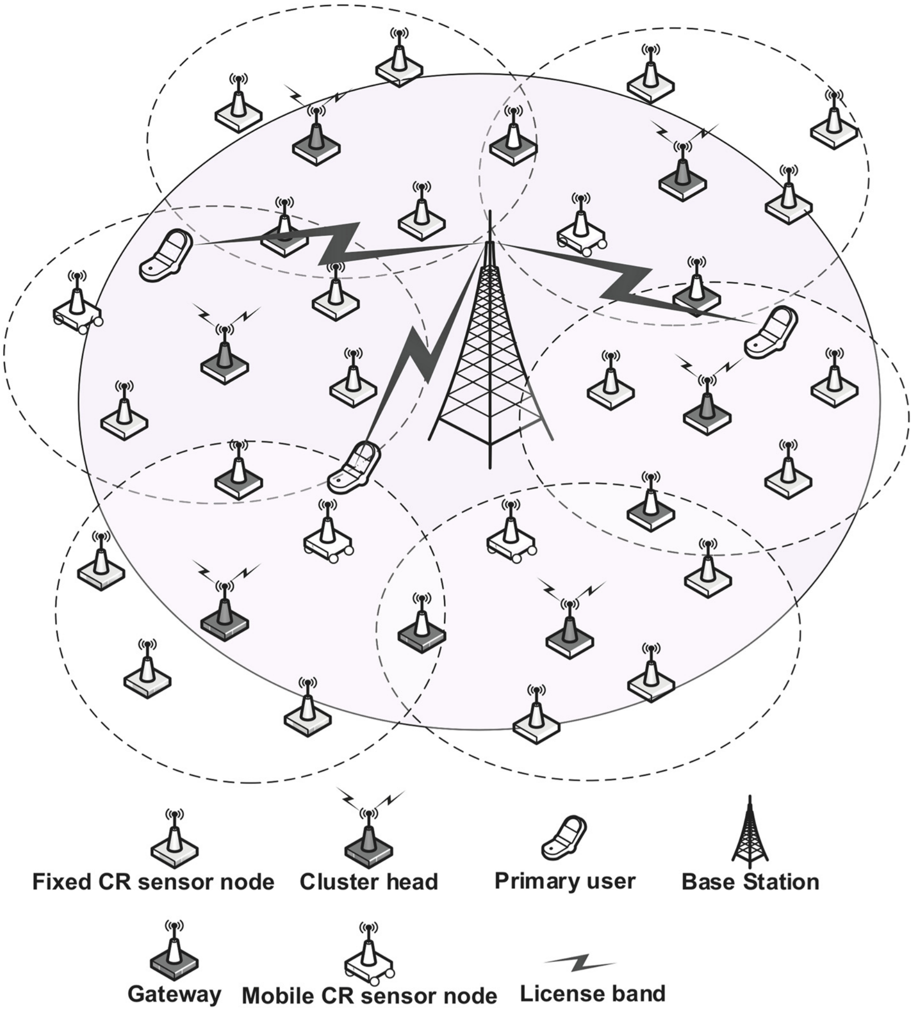

Fig. 1 System architecture for CRSN

the back-off time expires, the CH starts to send a beacon containing the CH ID, time synchronization, channel-hopping sequence (CnHS), control, and resource allocation information of the cluster. To reduce the re-clustering in the network, CH defines a reserved CH (RCH) in the network. Thus, in the case of the CH moving out of the cluster, the RCH takes the CH responsibilities of that cluster. This is presented in more detail in the explanation about the cluster formation.

If the spectrum becomes occupied by PUs during the random back-off time, the CH detects collisions and stops sending beacons using that spectrum, after which it repeats the process by tuning to the next control channel (NCC) based on the CnHS. The cluster members also

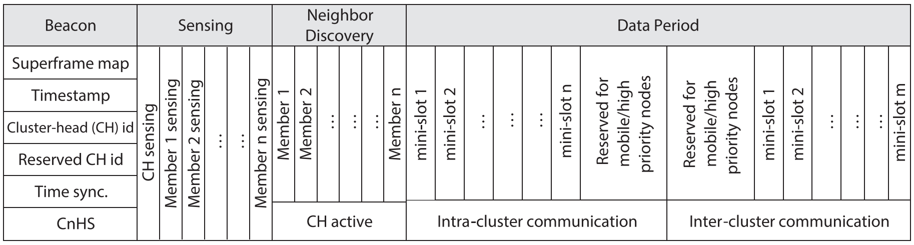

Fig. 2 CMCS MAC protocol superframe structure

tune to the CnHS if they do not receive the beacon message during a pre-specified time. This mechanism results in decreased interference of the PUs other clusters.

5.2 Spectrum-sensing period

CMCS performs channel sensing before channel selection, and nodes can therefore have up-to-date sensing information before beginning data transmission. During the sensing stage, SUs are not allowed to transmit because sensing information has to be interference free. Each SU senses the channels individually and maintains an available channel list (ACL) by the end of the sensing period.

Assuming $\mathrm{ACL}_i(t)$ be the channel availability vector of SU node i at time t for Ch channels, where Ch is the total number of channels available for CRSN. Each vector element, $\mathrm{acl}^i_{\mathrm{ch}}(t)$, can take a binary value 0 if the channel is busy and a value of 1 if it is vacant. Therefore, $\mathrm{ACL}_i(t)=[\mathrm{acl}^i_1(t)\ \mathrm{acl}^i_2(t) \ldots \mathrm{acl}^i_{\mathrm{Ch}}(t)]^T$.

In the spectrum-sensing period, all nodes remain quiet and listen to the channel. Synchronizing the spectrum-sensing period helps to reduce the false-alarm probability. If any node senses that the cluster control channel is occupied by the PU, it must convey the CCC occupancy status to the CH on the NCC at the specified time. After obtaining the status from a CM about the current CCC, a CH announces to all of its cluster members that it is tuning to the NCC. Using this mechanism, the CMCS can avoid the hidden terminal problem.

Because sensing the entire spectrum would consume a lot of power, it is proposed that the SU node in the CMCS senses only those channels that have a higher probability of being available for data transmission. Therefore, each SU node keeps track of the mean of the channel availability time, which is called $\mathrm{ChA}_i(t)$, and is defined as $\mathrm{ChA}_i(t)=[\mathrm{cha}^i_1(t)\ \mathrm{cha}^i_2(t) \ldots \mathrm{cha}^i_{\mathrm{Ch}}(t)]^T$.

The CH in each cluster aggregates the channel sensing information ($\mathrm{ChA}_i(t)$) from all of its members to create and update the channel availability rate (ChR) list, which sorts the channels based on the probability of their availability in the cluster. ChR for cluster j is given by Eq. (1), where CM_j is the number of cluster members in cluster j at time t, and i_j denotes the ith member of cluster j.

$$\mathrm{ChR}_j(t) = \sum_{i_j \in \mathrm{CM}_j} \mathrm{ChA}_{i_j}(t), \quad 1 \le i_j \le \mathrm{CM}_j. \tag{1}$$

In each superframe, the CH broadcasts the updated list of ChR to all members. Therefore, the CR nodes scan only those spectra that have higher priorities in the spectrum database. By using this mechanism, the CMCS can reduce the power consumption caused by frequent spectrum sensing. The sorted channel list is also used for the CnHS. Therefore, channels with higher rates will be tried first, and this can increase the chance of successful transmission and reduce the power consumption due to a lower number of re-transmissions.

5.3 Neighbor discovery

We merged the neighbor-discovery and cluster-formation phases as they are closely related to each other. To exchange information regarding spectrum availability, CR nodes need to discover their neighboring nodes, and this is usually done during the neighbor-discovery phase. To obtain a common time reference, we synchronized the GPS clock of the SUs. In the neighbor-discovery phase, time slots are set long enough to discover all the neighboring nodes that are operating on the same channel. Once all CRs sense the free spectrum and prepare the accessible channel set, CR nodes arrange the channels.

5.4 Data transmission period

The proposed CMCS does not define a special channel for data transmission. Therefore, all available channels of the cluster could be used for data communication. During the data transmission period, the cognitive radio nodes shift to a predefined channel in order to transmit packets. Before communication, a transmitter automatically generates a random backoff time. If the allocated channel remains idle until the timer expires, the transmission can take place. However, if a collision is detected, the node pair does not transmit or shift to any other frequency.

The data transmission period is divided into two phases: intra-cluster communication and inter-cluster communication. All communication between nodes in each cluster occurs during intra-cluster communication, e.g., data transmission, exchanging the ACL, joining the cluster, and leaving the cluster. Meanwhile, communication between neighboring clusters takes place in the inter-cluster phase. To make the data transmission period more robust against node mobility, the CMCS uses dynamic data-transmission frame times to improve the throughput mechanism as well as to reduce the power consumption.

6 Proposed cluster formation and maintenance

6.1 Outline of the proposed system

In the proposed protocol, cluster formation is performed in a distributed manner based on the available information regarding the neighbor nodes. In the initial stage, as the node may not know all of its neighbors, clusters are partially formed. However, after initial clustering, clusters are gradually reconstructed and interconnected to form a more reliable network structure by collecting more neighbor information during the neighbor-discovery phase. The proposed clustering mechanism divides the network into clusters based on three values: spectrum availability, node power level, and node speed. In the proposed clustering scheme, clusters are formed with the neighboring nodes in an ad hoc topology. This procedure is explained in the rest of this section.

Nodes in the proposed architecture exchange their $\text{ACL}_i(t)$ based on the spectrum sensing information that is obtained. Each node generates its own neighbor list N_i, where $i = 1, 2, 3, \ldots, n$ using the neighbor-discovery mechanism. Next, the cluster formation phase starts. Because the CMCS tries to generate fewer clusters, cluster formation is defined as a maximum vertex biclique problem [38] to include more nodes inside each cluster, while ensuring that there exists a sufficient number of idle channels for intra-cluster communication.

First, based on the neighbor list, N_i, and $\text{ACL}_i(t)$, every CR_i creates an undirected bipartite graph $\text{G}_i(\text{A}_i, \text{B}_i$, and $\text{E}_i)$. Graph G (V, E) is called bipartite if the vertices set V can be split into two disjoint sets A and B, where $\text{A} \bigcup \text{B} = \text{V}$, such that all edges in E connect vertices from A to B. Here, $\text{A}_i = \text{CR}_i \bigcup \text{N}_i$, and $\text{B}_i = \text{C}_i$, where C_i is the $\text{ACL}_i(t)$ of the CR_i. An edge (x, y) exists between vertices $\text{x} \in \text{A}_i$ and $\text{y} \in \text{B}_i$ if $\text{y} \in \text{C}_i$, i.e., channel y is a subset of the available channels of CR_i.

Figure 3 shows the connectivity graph of a CR network with the ACL in the brackets. Figure 4a shows the bipartite graph $\text{G}_d(\text{A}_d, \text{B}_d, \text{E}_d)$ constructed by CR_d from Fig. 3. The set of vertices A_d that corresponds to the one-hop neighbors is A_d = a, b, e, k, j, while the set of vertices B_d corresponds to the $\text{ACL}_d(t)$, which is equal to $\text{ACL}_i(t)$=1, 2, 3, 4, 5. Here, vertices 1, 2, 3 are common to all vertices in A_d. In Fig. 4b, the maximum vertex biclique graph for node d is presented. With the neighboring nodes (a, b, e, k, j) and channels (1, 2, 3), node d forms its maximum vertex biclique. Thus, every node in the network constructs its own maximum vertices biclique graph.

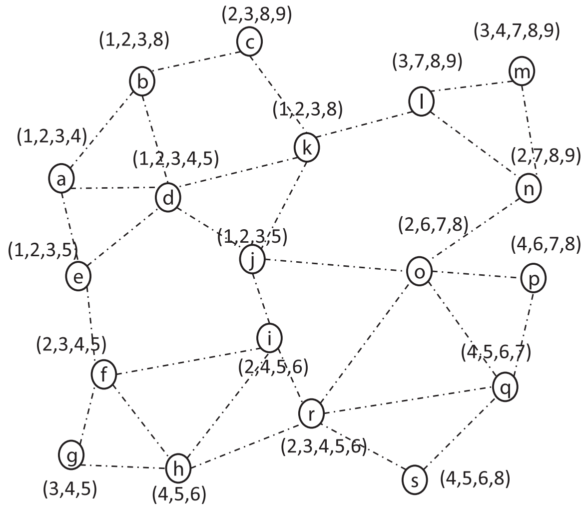

Fig. 3 Connectivity graph of a CRSN with available channel list in brackets

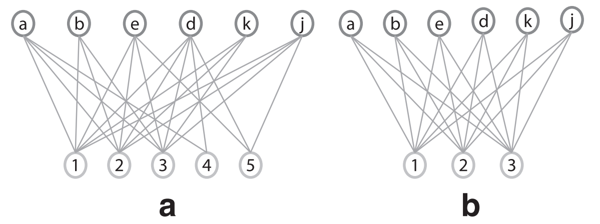

Fig. 4 a Bipartite graph constructed by node d. **b** Maximum vertex biclique graph of node d

To choose an optimal CH from among all nodes, we define a parameter, namely the CHEV. In this paper, we formulate the choice of a CH as a maximization problem, which can be defined as follows:

$$i_j^* = \max_{i_j}\left(\text{CHEV}_{i_j}\right), \quad 1 \le i_j \le \text{CM}_j, \tag{2}$$

$$\text{CHEV}_{i_j} \quad \propto \quad {N_{i_j}}^{\text{Ch}_{i_j}}, \tag{3}$$

where i_j indicates the node i in cluster j, N_{i_j} is the total number of neighboring nodes to node i in cluster j, and Ch_{i_j} is the total number of common channels for node i in cluster j.

The idea behind choosing the CHEV value as given in Eqs. (2) and (3) is to choose the node with the highest number of common channels and the highest number of neighbors to be the CH. This makes the cluster more flexible to the PU appearance and spectrum mobility as well as avoiding the need for a large number of clusters in the network.

Because the CH is responsible for cluster stability, it should be the node with the highest available power. In addition, to produce a high-mobility aware MAC protocol to avoid frequent reclustering due to CH movements, the CH should be the node with the lowest speed from among the cluster members in the network. By combining these two important features of CHs, we define the constant of the relationship given in Eq. (3) as follows:

$$\text{CHEV}_{i_j} = W_{i_j} \times {N_{i_j}}^{\text{Ch}_{i_j}}, \tag{4}$$

$$W_{i_j} = \frac{\gamma_{i_j}}{\sum_{i_j} \gamma_{i_j}} \quad \text{where} \quad \gamma_{i_j} = \frac{{E_{i_j}}^{\alpha}}{{V_{i_j}}^{\beta}}, \quad \alpha, \beta \in R+ \\ \text{and} \quad 0 < \alpha, \beta \le 1 \tag{5}$$

where W_{i_j} is a normalization factor that indicates how powerful and static node i_j is relative to the other nodes in the cluster. γ_{i_j}, which is always a positive value, is the proposed parameter to indicate the relationship between the node energy and speed. Meanwhile, α and β are design parameters for prioritising the speed and energy based on the application requirement. If more priority is given to energy, then α will be higher than β and vice versa.

To avoid having a big CHEV value, we take the log of CHEV as a final selection metric of CHs. Thus, the optimization problem in Eq. (2) is written as follows:

$$i_j^* = \max_{i_j} (\log (W_{i_j} \times N_{i_j}^{\mathrm{Ch}_{i_j}})). \qquad (6)$$

This optimization problem can be solved simply using the well-known descending sorting algorithm [39]. Therefore, a node with the highest log(CHEV) value forms the cluster and becomes the CH. For example, when node *a* and *d* have four shared channels, while node *d* has more neighbors compared to node *a* under the conditions that they are static and have the same amount of power, node *d* will be the CH. If the log(CHEV) value of a node CR_i is smaller than that of its neighbor, CR_i joins the neighbor that has the highest log(CHEV) value, as the CM. Once the clusters are formed, CHs prioritize other cluster members based on log(CHEV) for the RCH selection. The CM with the highest log(CHEV) becomes the RCH for the cluster. The RCH takes charge of the cluster whenever the current CH moves out, which reduces the possibility of re-clustering. By considering the following definitions, we define the proposed cluster formation algorithm in Algorithm 1.

i,j → Possible integer (e.g., 1, 2, 3...)
i_j → i member of cluster j
N_{i_j} → Neighbor set of i_j
ACL_{i_j}(t) → Accessible channel list of i_j at time t
CM_j → Cluster member of cluster j
CH_j → Cluster-head of cluster j
GW_j → Gateway of cluster j
F_{i_j}. → Node i_j status flag

Algorithm 1: Cluster-formation algorithm

```
Start
$\forall i_j \in CM_j \mid i_j$ start broadcasting $\mathrm{ACL}_{i_j}$ ;
$i_j$ receives $\mathrm{ACL}k_j$, where $i_j \neq k_j$ ;
$i_j$ construct bipartite and calculate $\mathrm{CHEV}_{i_j}$ ;
$i_j = \max_{i_j} (\log (w_{i_j} \times N_{i_j}^{\mathrm{Ch}_{i_j}}))$ ;
$i_j$ exchange $\mathrm{CHEV}_{i_j}$ with neighbors ;
if $CHEV_{i_j} > CHEV_{k_j} \mid i_j \neq k_j$ then
    $\mathrm{CH}_j = i_j$ ;  /* $i_j$ becomes cluster head */
else
    $\mathrm{CM}_j = i_j$ ;  /* $i_j$ becomes cluster member */
    if $i_j$ receives any other beacon then
        $GW_j = i_j$ ;  /* $i_j$ becomes gateway */
    end
end
End
```

Figure 5 illustrates the different steps involved in cluster formation. As shown in Fig. 5b, the network is divided into several clusters based on Eq. (4). After the cluster formation, CH determines and maintains a list of operating frequencies for the cluster. Then, as depicted in Fig. 5c, member nodes check the neighbor list to determine the presence of any other CHs in the neighborhood. If any are found, the member node informs the CH about a new neighbor belonging to a different cluster. The CH will receive this information from all the nodes, and then assign the node with the highest CHEV as a GW with each particular cluster to maintain the inter-cluster communication. As clusters in the architecture are independently formed in a distributed way, it is possible to have loops in the architecture as they increase the robustness of the architecture in the case of link failure.

6.2 Topology management

6.2.1 Nodes joining the network

As shown in Algorithm 2, each node starts by channel hopping to receive a beacon message. If the joining node receives a beacon from a cluster, it first compares its own free channel set with the information obtained from the beacon. If a joining node has at least γ channels in common with the cluster, it sends a request to join the cluster during the intra-cluster communication phase. CH defines a mini-slot to the new node if it has a free slot available; otherwise, it rejects the new node's request. If the request is rejected by a CH, the node has to change its channel and listen to other channels until it obtains a beacon message from one of the channels. If it cannot join any existing cluster, it forms a new cluster and becomes a CH.

6.2.2 Nodes leaving the network

In our proposed scheme, the management of nodes that leave the network is relatively simple. There are three types of movement that can occur in each cluster: (i) the CH moves out of the cluster, (ii) the GW node moves out of the cluster, and (iii) the CM moves out from or moves into the cluster. The CMCS handles each situation as follows:

1. The CH moves out of the cluster: When a CH node detects its own movement, it informs all of the nodes about this movement and assigns the RCH as a new CH. In the case where RCH does not cover all cluster member nodes, nodes that fail to connect to the RCH will create new cluster(s) if they could not join any existing cluster(s). It also informs the neighboring cluster through the GW node about its approach to the new cluster.
2. GW moves out of the cluster: When a GW starts to move, it informs the CH about its departure. The CH then appoints another CM as a GW node to maintain

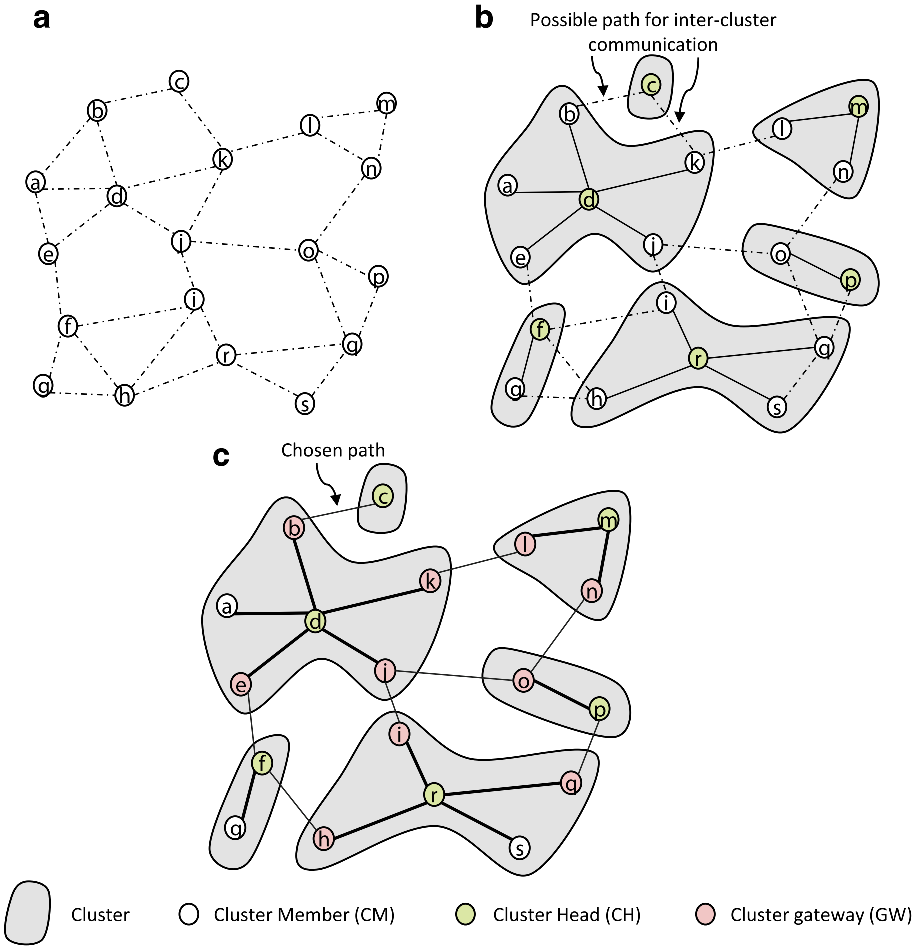

Fig. 5 Cluster-formation process. **a** Network connectivity graph. **b** Cluster formation based on CHEV value. **c** Inter-cluster connection through gateways

the connection with neighboring clusters. The CH removes the outgoing node's timeslot from the intra-cluster phase. It also informs the neighboring cluster about the new node approaching that cluster.

3. The CM moves out of the cluster: When a CM starts to move, it informs the CH, which then removes its timeslot from the intra-cluster phase and informs the neighboring cluster about the new node that is approaching. The host cluster treats all new upcoming nodes as a CM node, and it defines a timeslot for the upcoming node. Using this scheme, mobile nodes do not become disconnected from the remainder of the network.

6.2.3 Spectrum mobility

If during the sensing phase nodes detect that their channel availability has changed, they inform the CH about these changes. If one of the main channels of the cluster is not available for one node, the CH removes that channel from the ACL and no communication is performed on that channel. Using this mechanism, we can avoid hidden terminal problem in the network. Otherwise, because we form the cluster using a maximum number of shared channels, we reduce the chance of reclustering due to channel occupancy by PUs.

7 Simulation analysis

To evaluate the proposed protocol, we performed a comparison study using a MATLAB simulation with three other well-known cluster-based approaches, namely CogMesh [33], cluster-based MAC [30], and KoN-MAC [34]. For comparison, we considered two major metrics for clustering (cluster size and number of channels) and three major metrics for MAC protocol (energy efficiency, throughput, and delay).

To simulate the performance of the proposed protocol, we carried out different sets of simulations. In the first simulation, we deployed different numbers of CR sensor nodes (50 to 300) in a 1000 m × 1000 m area to study the impact of various network sizes on the performance of the protocols. The areas ranged from a low-density network (50 SUs) to a relatively dense network (300 SUs). In addition, we evaluated the response for two different communication ranges (100 and 200 m) to investigate further the effect of the network density on the performance of the protocols. There were 10 randomly distributed PUs in the network, while the number of available primary radio channels was set as 10. Although PUs are considered stationary during the network lifetime, we employed a discrete-time Markov chain to represent the pattern of each PUs' channel usage. This means that PUs may change their state (i.e., channel usage) after each process, affecting the available channels in each SU; this is considered as spectrum mobility. We adjusted the simulation time to 1000 s in order to simulate a multiple number of superframes (100 superframes) in a single run of the Monte Carlo simulation. To obtain sufficiently confident results against changes in the PUs' channel occupancy and SUs' mobility, we used 50 Monte Carlo simulation runs each lasting 1000 s. In the simulations, we assigned equal priority to the speed and energy for the cluster formation ($\alpha = \beta = 1$).

Figures 6 and 7 show the clustering performance of the proposed protocol (CMCS) compared with CogMesh, cluster-based MAC, and KoN-MAC with different sensor-node transmission ranges. Figure 6 shows the results when the transmission range is set to be 100 m, while Fig. 7 shows the results when the transmission range is set to 200 m.

Figure 6a shows the effect of increasing the number of nodes on the number of clusters. The figure indicates that as the number of SUs increases in the network, KoN-MAC creates a smaller number of clusters compared to other MAC protocols, and this is believed to be because of its clustering scheme, which is based on the connection of nodes to neighbors with lower ID without considering the spectrum availability. On the other hand, cluster-based MAC protocol creates a smaller number of clusters compared to CMCS, which is slightly higher than the number created using CogMesh. The reason for this is that CogMesh and cluster-based MAC protocol do not consider the available channels for the formation of clusters, while CMCS includes a smaller number of nodes in each cluster to maintain the higher number of shared channels seen in Fig. 6b. As indicated in Fig. 6a, the number of clusters in the CMCS increases linearly, which means that the CMCS makes clusters having the same ratio. Although neither a low or high number of clusters is defined as being a good or bad performance metric, according to [40], weight-based clustering is seen as an efficient way of producing well-balanced and stable clusters.

Figure 6b depicts the number of common channels per cluster against the number of nodes. As can be observed, KoN-MAC shows the worst performance of among all the MAC protocols. This is because the available channel has no role in its clustering scheme. Meanwhile, from Fig. 6b, we observed that CogMesh can only achieve one (1) common channel, and this is when the number of nodes reaches 300. Cluster-based MAC protocol exhibits a better performance when compared to CogMesh, but the number of common channels then drops sharply as the number of nodes increases. Meanwhile, the proposed CMCS guarantees a large number of common channels per cluster (4.5 with 300 nodes); thus, it is shown to outperform both CogMesh and cluster-based MAC protocols.

The effect of increasing the number of nodes on the number of clusters is shown in Fig. 7a. As can be observed, a higher transmission range causes a smaller number of clusters to be produced by all of the protocols. However, a similar pattern can be observed as KoN-MAC produced the lowest number of clusters. As previously

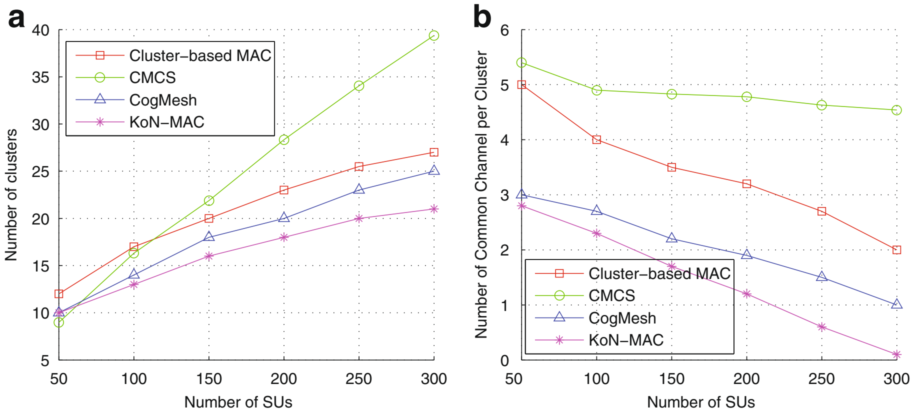

Fig. 6 Clustering performance (transmission range 100 m). **a** Average number of clusters. **b** Average number of common channels per cluster

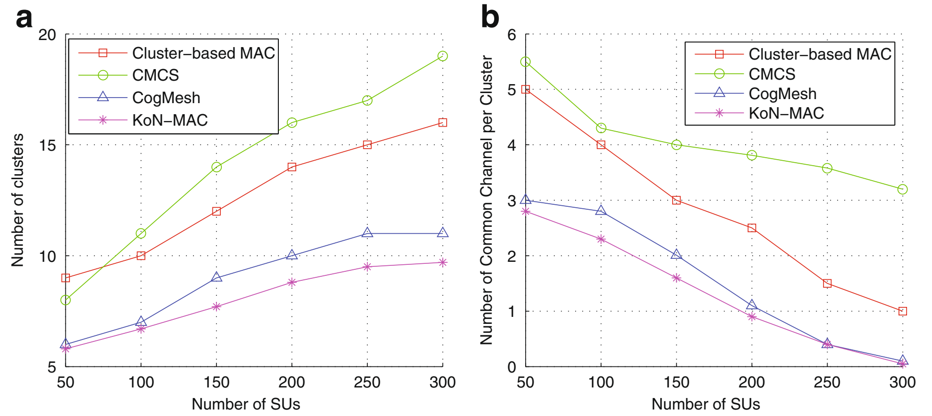

Fig. 7 Clustering performance (transmission range 200 m). **a** Average number of clusters. **b** Average number of common channels per cluster

discussed, KoN-MAC and CogMesh do not consider the available channels for cluster formation. Therefore, they include more nodes in each cluster without considering the available channels. The cluster-based MAC protocol needs to ensure only that there is one (1) similar channel in each cluster. Therefore, by increasing the number of nodes, it creates smaller clusters by ensuring that there is at least one (1) common channel in each cluster.

Figure 7b shows the effect of increasing the number of nodes on the number of common channels. As can be seen, KoN-MAC is shown to have the worst performance of all of the others, and CMCS is shown to have the best performance. Meanwhile, CogMesh performs slightly better than KoN-MAC. The number of common channels in the cluster-based MAC protocol decreases linearly as we increase the number of nodes, while it has a better performance compared to CogMesh.

Considering the analysis of the clustering performance, we observe that CMCS achieved around five common channels per cluster when the transmission range is set to 100 m. On the other hand, the cluster-based MAC protocol can achieve as low as two common channels per cluster when the network density increases to 300 nodes. Therefore, we can conclude that CMCS could improve the common channel ratio of other protocols by more than 60 % with a higher node ratio. However, in higher-density networks (Fig. 7b), where the transmission range is set to 200 m, CMCS achieves almost three common channels per cluster. However, it still performs better than the other candidate MAC protocols (up to 60 % higher common channel at 300 nodes), which are presented in the results. As previously discussed, a large number of available channels in each cluster can ensure stable cluster formation and minimize the reconfiguration/re-clustering of the network.

In the remainder of this section, we discuss the performance analysis for the MAC protocol using energy efficiency, throughput, and delay as its performance metrics. For these simulations, we considered 100 SUs with a 100-m transmission range. We considered two scenarios to determine the effect of different network dynamicity values and node speed on the performance of the proposed protocol.

Figure 8 depicts the impact of the node speed on the protocol performance when 10 % of the nodes are set to be mobile and when the speed varies from 0 (static network) to 10 m/s (dynamic network).

The impact of the mobility speed on the packet delivery ratio (PDR) is depicted in Fig. 8a. As can be seen from the figure, all of the protocols show nearly similar performance when the network is static. However, by increasing the speed of mobile nodes from 2 to 10 m/s, the PDR of KoN-MAC, CogMesh, cluster-based MAC, and CMCS protocols drop by almost 15, 13, 10, and 7 %, respectively. The increase in speed will increase the movement of nodes between clusters, and the dynamicity of the network would eventually make the formation of a stable route more difficult. Therefore, more packets will be dropped and a lower packet delivery can be seen. Again, we observe that CMCS shows a better performance because it uses an adoptive data period to handle upcoming nodes in the cluster.

Figure 8b illustrates the effect of the node speed on the energy consumption of the nodes. As shown in the figure, CogMesh shows an overall higher energy consumption (17 % in static networks and around 30 % at high speeds) compared to other protocols. By increasing the speed of the mobile nodes, the KoN-MAC protocol shows an exponential increase in energy consumption. By increasing the speed of mobile nodes, nodes leave their vicinity more rapidly, resulting in disrupted clusters and more

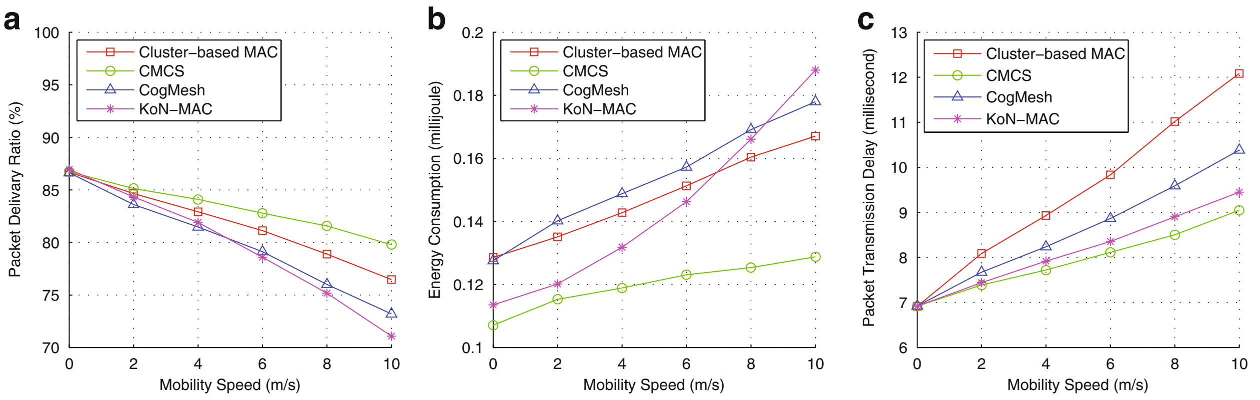

Fig. 8 Impact of mobile node speed on the protocol performance. **a** Average packet delivery ratio. **b** Average energy consumption. **c** Average packet transmission delay

frequent reclustering, which increases the energy consumption in KoN-MAC. The cluster-based MAC protocol shows a slightly lower energy consumption compared to CogMesh. By increasing the node speed, it is more likely that a CH will move out of the cluster, causing reclustering to occur. Moreover, because of this movement, the previous common channel may not be available for all nodes, so nodes need to perform channel sensing and neighbor discovery more often, which can in turn increase the energy consumption. Because CMCS utilizes RCH, providing a higher number of common channels, and it informs neighboring clusters about approaching mobile nodes, it consumes less energy.

The effect of the mobility speed on the packet transmission delay is presented in Fig. 8c. It can be noted that in static networks, protocols show similar performance. However, by increasing the speed from 2–10 m/s, CogMesh shows a sharp increase in the packet transmission delay. As can be observed, frequent reclustering at higher speeds as well as a large number of nodes in each cluster can increase the delays in CogMesh and cluster-based MAC protocol by up to 25 % (3 ms) and 12 %, respectively, when compared to the CMCS protocol. Although KoN-MAC achieves a lower PDR, it performs better than CogMesh and cluster-based MAC protocols in terms of delay. Note that the delay is only considered for successfully delivered packets. Meanwhile, CMCS uses an efficient routing protocol that considers different metrics to find the optimal route, which helps CSMS to achieve better results.

To observe the effect of an increasing number of mobile SUs in the network, the next simulation scenario considers the network dynamicity, which varies from 0 % (static network) to 10 %, whereas the node speed is fixed to 2 m/s. Figure 9 illustrates the performance of CMCS against KoN-MAC, CogMesh, and cluster-based MAC.

Figure 9a shows the effect of network dynamicity on the PDR. As can be seen, by increasing the number of mobile nodes in the network, the PDR falls sharply when using KoN-MAC, CogMesh, and cluster-based MAC contrary

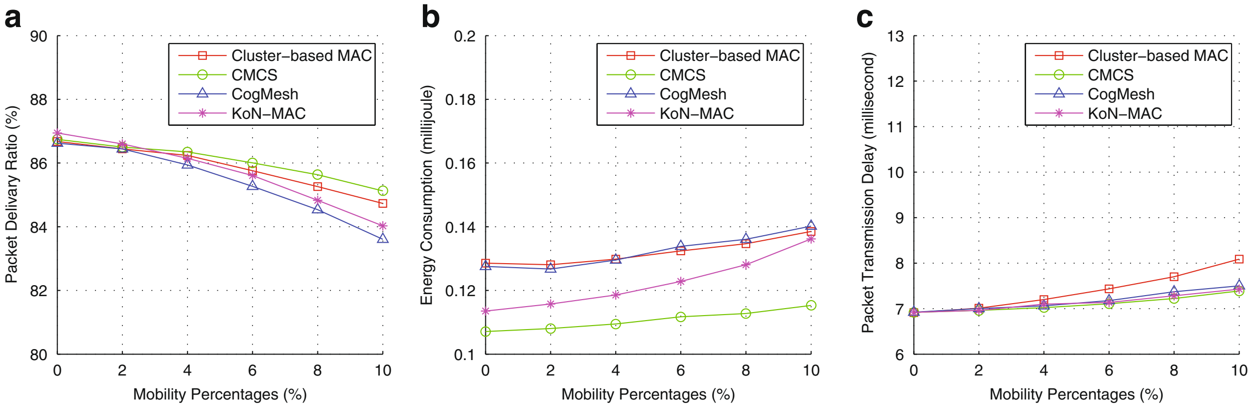

Fig. 9 Impact of network dynamicity on protocol performance. **a** Average packet delivery ratio. **b** Average energy consumption. **c** Average packet transmission delay

Algorithm 2: Topology-management algorithm

```
1  Start
2  $\forall i_j \in CM_j$ SYNC using the PU signal;
3  if $F_{i_j} = 1$ ; /* $i_j \in CM_j$, 1 means true and 0 means false */
4  then
5      $i_j$ tune to CCC ;
6      $i_j$ waits for cluster beacon ;
7      if $i_j$ receives beacon then
8          $i_j$ involved in cluster action ;
9      else
10         $i_j$ tunes to the next channel based on CnHS ;
11         if $i_j$ hears beacon then
12             $i_j$ involved in cluster action ;
13         else
14             $F_{i_j} = 0$ ; /* set the node flag to not belong to any cluster */
15             go to line 3;
16         end
17     end
18 else
19     $i_j$ starts channel hopping based on CnHS ;
20     if $i_j$ hears beacon in any channel then
21         $i_j$ obtains cluster information ;
22         if $i_j$ has min $\gamma$ common channel with cluster then
23             $i_j$ requests to join ;
24             if mini slot available then
25                 $i_j$ involved in cluster action ;
26             end
27         end
28     end
29     $CH_j = i_j$ ; /* node becomes CH itself */
30 end
31 End
```

to the use of CMCS. This is because CMCS uses adaptive scheduling to handle the movement of the mobile nodes in the network whose function is not provided by other protocols. Meanwhile, a comparison of Figs. 8a and 9a show that the increase in speed can affect the network performance of all protocols, particularly those with a higher ratio.

Figure 9b depicts the network dynamicity against energy consumption. We observed that CogMesh and cluster-based MAC protocol show similar performance, while the CMCS realizes a packet delay of up to 15 % compared to the other protocols in a higher network dynamicity (10 %). KoN-MAC shows the most rapid increase in the energy consumption compared to other protocols, and this is due to its clustering scheme, which does not consider the mobility in the network. As previously discussed, CMCS uses spectrum-aware clustering, which makes the cluster more stable against changes in the network dynamicity. In addition, the concept of a reserved CH in CMCS can reduce the reclustering issues, resulting in greater energy conservation for the protocol. By comparing Figs. 8b and 9b, we observe that increasing the speed would affect the network performance more than by introducing network dynamicity with low speed.

Figure 9c confirms that the use of the proposed novel routing helps CMCS to outperform CogMesh in terms of the transmission delay (up to a 10 % decrease in the delay), especially with a higher network dynamicity (10 % mobility), while KoN-MAC and cluster-based MAC protocols exhibit performances that are marginally similar to the CMCS protocol.

8 Conclusions

This paper proposed a novel cross-layer mobility-aware MAC protocol for CRSN, which is robust against the activities of PUs, as well as node mobility in networks. This was realized by integrating the spectrum sensing at the PHY layer with the packet scheduling at MAC layer. The proposed spectrum-aware clustering scheme was designed in such a way that it ensures that there is stability in the formation of clusters in order to avoid frequent reclustering. A greater number of common channels results in clusters that are more robust against the mobility of both SUs and spectrum because of the PUs' activity. To handle the mobility in the network, the CMCS uses an adaptive data period to handle the upcoming new nodes in the clusters. The simulation results show that the proposed protocol can achieve around five common channels per cluster in lower-density networks, and around three common channels per cluster in higher-density networks. However, it overtakes the other candidate MAC protocols by more than 60 %, where the number of SUs increases in the network. Moreover, CMCS outperforms KoN-MAC, CogMesh, and cluster-based MAC protocols in terms of the packet delivery ratio, energy consumption, and delay by up to 5, 30, and 25 %, respectively. This work focused on the design of an efficient MAC protocol for channel assignment. However, by considering the cross-layer approach, an appropriate routing protocol can also be integrated with the MAC protocol, and this will be done as future work.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

MZ, AKMMI, NM, and EMM designed the study and developed the methodology. Performance evaluations, simulation studies, and data analyses were performed by MZ and NM. MZ wrote the manuscript and AKMMI, SB, EMM, and SS revised the manuscript. All of the authors take full responsibility for the content of the paper. All authors read and approved the final manuscript.

Acknowledgements

This research was partially supported by GUP TIER 1, which is a research grant of the Universiti Teknologi Malaysia (UTM), Ministry of Higher Education, Malaysia with Vote No. 05H61 for the year 2014 to 2016. We would like to thank the Malaysia-Japan International Institute of Technology for their financial support. We would also like to thank three anonymous reviewers for their insightful comments and suggestions to improve the quality of the paper.

Author details

[1]Electronic Systems Engineering Department, Malaysia-Japan International Institute of Technology (MJIIT), Universiti Teknologi Malaysia (UTM), Jalan Sultan Yahya Petra, 54100 Kuala Lumpur , Malaysia. [2]Graduate School of Engineering, Osaka University, 565-0871 Suita-shi, Japan. [3]Electrical Engineering Department, Aswan University, Aswan, Egypt.

Received: 3 November 2015 Accepted: 7 June 2016

References

1. FCC Spectrum Policy Task Force, Report of the spectrum efficiency working group (2002). https://transition.fcc.gov/sptf/files/SEWGFinalReport_1.pdf
2. G Staple, K Werbach, The end of spectrum scarcity [spectrum allocation and utilization]. IEEE Spectr. **41**(3), 48–52 (2004)
3. R Tandra, SM Mishra, A Sahai, What is a spectrum hole and what does it take to recognize one?. Proc IEEE. **97**(5), 824–848 (2009)
4. J Mitola, GQ Maguire Jr, Cognitive radio: making software radios more personal. IEEE Pers Commun. **6**(4), 13–18 (1999)
5. AM Wyglinski, M Nekovee, T Hou, *Cognitive Radio Communications and Networks: Principles and Practice*. (Academic Press, Cambridge, 2009), p. 736
6. IF Akyildiz, W-Y Lee, MC Vuran, S Mohanty, Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. Comput Netw. **50**(13), 2127–2159 (2006)
7. N Mansoor, AM Islam, M Zareei, S Baharun, T Wakabayashi, S Komaki, Cognitive radio ad-hoc network architectures: a survey. Wirel Pers Commun. **81**(3), 1117–1142 (2015)
8. IF Akyildiz, W-Y Lee, KR Chowdhury, Crahns: Cognitive radio ad hoc networks. Ad Hoc Netw. **7**(5), 810–836 (2009)
9. O Younis, L Kant, K Chang, K Young, C Graff, Cognitive manet design for mission-critical networks. IEEE Commun Mag. **47**(10), 64–71 (2009)
10. KR Chowdhury, IF Akyildiz, Cognitive wireless mesh networks with dynamic spectrum access. IEEE J Sel Areas Commun. **26**(1), 168–181 (2008)
11. M Di Felice, KR Chowdhury, L Bononi, in *2011 IEEE Vehicular Networking Conference (VNC)*. Cooperative spectrum management in cognitive Vehicular Ad Hoc Networks, (2011), pp. 47–54. doi:10.1109/VNC.2011.6117123
12. OB Akan, O Karli, O Ergul, Cognitive radio sensor networks. IEEE Netw. **23**(4), 34–40 (2009)
13. M Zareei, AKMM Islam, A Zeb, S Baharun, S Komaki, Mobility-aware timeout medium access control protocol for wireless sensor networks. {AEU} - Int J Electron Commun. **68**(10), 1000–1006 (2014). doi:10.1016/j.aeue.2014.05.014
14. JA Han, WS Jeon, DG Jeong, Energy-efficient channel management scheme for cognitive radio sensor networks. IEEE Trans Veh Technol. **60**(4), 1905–1910 (2011)
15. N Mansoor, AKM Muzahidul Islam, M Zareei, S Baharun, S Komaki, in *International Conference on Advances in Electrical Engineering (ICAEE)*. Spectrum aware cluster-based architecture for cognitive radio ad-hoc networks, (2013), pp. 181–185. doi:10.1109/ICAEE.2013.6750329
16. MC Oto, OB Akan, Energy-efficient packet size optimization for cognitive radio sensor networks. IEEE Trans Wirel Commun. **11**(4), 1544–1553 (2012)
17. AO Bicen, OB Akan, Reliability and congestion control in cognitive radio sensor networks. Ad Hoc Netw. **9**(7), 1154–1164 (2011)
18. A Argyriou, Cross-layer and cooperative opportunistic network coding in wireless ad hoc networks. IEEE Trans Veh Technol. **59**(2), 803–812 (2010)
19. W Chen, L Dai, K Letaief, Z Cao, A unified cross-layer framework for resource allocation in cooperative networks. IEEE Trans Wirel Commun. **7**(8), 3000–3012 (2008)
20. JY Yu, PHJ Chong, A survey of clustering schemes for mobile ad hoc networks. IEEE Commun Surv Tutor. **7**(1–4), 32–48 (2005)
21. AKM Muzahidul Islam, K Wada, W Chen, Dynamic cluster-based architecture and data congregation protocols for wireless sensor network. Int J Innov Comput Inf Control (IJICIC). **9**(10), 4085–4099 (2013)
22. J Uchida, IA Muzahidul, Y Katayama, W Chen, K Wada, Construction and maintenance of a novel cluster-based architecture for ad hoc sensor networks. Ad Hoc Sens Wirel Netw. **6**(1–2), 1–31 (2008)
23. M Gerharz, C de Waal, M Frank, P Martini, in *27th Annual IEEE Conference on Local Computer Networks*. Link stability in mobile wireless ad hoc networks, (2002), pp. 30–39. doi:10.1109/LCN.2002.1181761
24. M Zareei, A Taghizadeh, R Budiarto, T-C Wan, EMS-MAC: energy efficient contention-based medium access control protocol for mobile sensor networks. Comput J. **54**(12), 1963–1972 (2011). doi:10.1093/comjnl/bxr053. http://comjnl.oxfordjournals.org/content/early/2011/06/02/comjnl.bxr053.abstract
25. MA Yigitel, OD Incel, C Ersoy, Qos-aware mac protocols for wireless sensor networks: a survey. Comput Netw. **55**(8), 1982–2004 (2011)
26. P Huang, L Xiao, S Soltani, MW Mutka, N Xi, The evolution of mac protocols in wireless sensor networks: a survey. IEEE Commun Surv Tutor. **15**(1), 101–120 (2013)
27. P Hu, M Ibnkahla, A mac protocol with mobility support in cognitive radio ad hoc networks: protocol design and analysis. Ad Hoc Netw. **17**, 114–128 (2014)
28. GP Joshi, SY Nam, SW Kim, Decentralized predictive mac protocol for ad hoc cognitive radio networks. Wirel Pers Commun. **74**(2), 803–821 (2014)
29. MV Nguyen, S Lee, S-j You, CS Hong, LB Le, Cross-layer design for congestion, contention, and power control in crahns under packet collision constraints. IEEE Trans Wirel Commun. **12**(11), 5557–5571 (2013)
30. X Li, F Hu, H Zhang, X Zhang, A cluster-based mac protocol for cognitive radio ad hoc networks. Wirel Pers Commun. **69**(2), 937–955 (2013)
31. P Hu, M Ibnkahla, in *2012 IEEE International Conference on Communications (ICC)*. CM-MAC: A cognitive MAC protocol with mobility support in cognitive radio ad hoc networks, (2012), pp. 430–434. doi:10.1109/ICC.2012.6363842
32. C Cordeiro, K Challapali, D Birru, N Sai Shankar, in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, DySPAN 2005*. IEEE 802.22: the first worldwide wireless standard based on cognitive radios, (2005), pp. 328–337. doi:10.1109/DYSPAN.2005.1542649
33. T Chen, H Zhang, GM Maggio, I Chlamtac, in *2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*. CogMesh: A Cluster-Based Cognitive Radio Network (IEEE, 2007), pp. 168–178. doi:10.1109/DYSPAN.2007.29
34. Y Xu, C Wu, C He, L Jiang, in *IEEE Global Communications Conference (GLOBECOM)*. A cluster-based energy efficient MAC protocol for multi-hop cognitive radio sensor networks, (2012), pp. 537–542. doi:10.1109/GLOCOM.2012.6503168
35. S Maleki, A Pandharipande, G Leus, Energy-efficient distributed spectrum sensing for cognitive sensor networks. IEEE Sensors J. **11**(3), 565–573 (2011)
36. Q Zhao, S Geirhofer, L Tong, BM Sadler, Opportunistic spectrum access via periodic channel sensing. IEEE Trans. Signal Process. **56**(2), 785–796 (2008)
37. B Jalaeian, M Motani, in *IEEE 9th Malaysia International Conference on Communications (MICC)*. Location Aware CR-MAC: A multi-channel cross layered PHY-MAC protocol for cognitive radio ad hoc networks, (2009), pp. 348–353. doi:10.1109/MICC.2009.5431528
38. R Peeters, The maximum edge biclique problem is np-complete. Discret Appl Math. **131**(3), 651–654 (2003)
39. DE Knuth, *The Art of Computer Programming: Sorting and Searching*, vol. 3. (Pearson Education, New Jersey, 1998)
40. O Younis, M Krunz, S Ramasubramanian, Node clustering in wireless sensor networks: recent developments and deployment challenges. IEEE Netw. **20**(3), 20–25 (2006)