Wireless Netw (2018) 24:1451–1475
https://doi.org/10.1007/s11276-016-1417-6

A survey of clustering algorithms for cognitive radio ad hoc networks

Mahassin Mohamed Ahmed Osman[1] · Sharifah Kamilah Syed-Yusof[1] ·
Nik Noordini Nik Abd Malik[1] · Suleiman Zubair[1]

Published online: 26 November 2016

Abstract The dynamic spectrum nature of cognitive radio challenges the connectivity and the stability of cognitive radio ad hoc networks (CRAHNs). Clustering is considered as an appropriate technique to overcome these issues. Various algorithms for clustering formation in CRAHN have been proposed, in which different system models and metrics are considered. This paper introduces an extensive survey of the most important published algorithms. This survey classifies the proposed algorithms based on their objectives. Moreover, it presents a detailed description of their techniques, evaluations of their performance, and discussion of the features and shortcomings of each algorithm. Furthermore, it provides and discusses the open issues for future research.

Keywords Cognitive radio · Ad hoc network · Clustering

1 Introduction

Fast growth in wireless communication networks in the last decade led to a dramatically increasing in demand for spectrum. Result in severe spectrum shortage. On the other hand, it has been observed that large portions of the licensed spectrum are highly underutilized [1, 2]. In order to tackle the problem of spectrum shortage, the Federal Communications Commission (FCC) has endorsed the unlicensed devices to work in licensed frequency bands [1]. Consequently, opportunistic spectrum access (OSA) techniques and cognitive radio are proposed as smart wireless communication technologies to handle the spectrum inefficiency utilization problem [1, 2].

The idea of cognitive radio was introduced by Mitola III [3]. Cognitive radio technology is the key technology that enables a wireless device to utilize or access spectrum in a dynamic opportunistic manner. In CR networks, there are two kinds of users: licensed users (also known as primary users) have full rights to access their appointed spectrum band whenever there is data to be transmitted and unlicensed users (also called cognitive radio users or secondary users) can only access license channels after ensuring the channel is unoccupied by the primary user [3]. From the perspective of network structure, CRNs can be classified into (1) a centralized network, which consists of base stations and wireless clients and (2) a distributed network (ad hoc network) which consists of a number of nodes that directly communicate with each other without any existence of a fixed network infrastructure such as a base station or an access point [4].

For effective operation of an ideal cognitive radio ad hoc network (CRAHN), many issues need to be tackled, which include: Spectrum hole detection (Spectrum sensing) conducted individually or collaboratively, MAC protocol to support cognitive radio based opportunistic spectrum access, neighbor discovery and topology management [5]. The topology management of the CRAHNs is impacted by three main factors: first, a common global control channel does not exist for the entire network, second; the topology of the network varies over time due to the activities of license users, and finally; for large-size networks, the volume of routing table will be large, and it will take a long time for routing information to spread in the whole network. Hence, a distributed control technique becomes inevitable for such a network. Clustering is considered as an effective topology management technique in

✉ Mahassin Mohamed Ahmed Osman
mief2@hotmail.com

[1] Universiti Teknologi Malaysia, Johor Bahru, Johor, Malaysia

CRAHNs for its capability of guaranteeing system performance by implementing a virtual network backbone and making the network smaller and more manageable. Clustering partitions network nodes into logical groups named clusters in order to improve the basic network performances such as routing delay, bandwidth consumption, and throughput [6]. Moreover, clustering may provide a simple and feasible power control mechanism [7, 8].

Ever since the idea of using clustering as a topology management scheme for wireless ad hoc networks is introduced in [9], a tremendous amount of clustering algorithms for ad hoc networks has been proposed, whereby different algorithms focus on different aspects. Therefore, different algorithms are applicable to different ad hoc network paradigms.

In mobile ad hoc networks (MANETs), the nodes are characterized by their mobility as well as limited power resources. Therefore, most of clustering algorithms based MANET have focused on the mobility behavior and energy of nodes during clustering formation [10–15]. Typically, the clustered structure in MANETs has been used to achieve various goals such as enhancing MAC protocol performance [16–18], improving and simplifying routing protocols [19–21], and ensuring network security [22, 23]. For example, in [16] the authors introduce a new MAC protocol that uses the clustered network coupled with both lightweight dynamic channel allocation and cooperative load balancing algorithms, in order to improve the throughput, the energy consumption, and the inter-packet delay variation. For the purpose of ensuring the network security, the authors in [22] introduce a cluster-based certificate revocation with a vindication capability scheme to prevent attackers from involving in network activities.

The vehicular ad hoc networks (VANETs) are featured as dense, an energy-rich and extremely highly mobility network. Therefore, clustering scheme in VANETs concentrated on the vehicle speed during the clustering process [24–26]. For example, in [25] the authors use the affinity propagation technique to introduce a mobility-based clustering scheme, which uses the vehicles' position and velocity information to form clusters that minimize both the relative mobility and the distance from each cluster head to its members in order to improve cluster stability. In VANETs the dynamic and dense network topology, lead to routing complexity as well as congestion from flooding, Moreover, the dense network causes the hidden terminal problem. Therefore, clustering schemes in VANETs have been concentrated on developing cluster-based MAC protocols [27–29], and cluster-based routing protocols [30, 31].

In energy-limited wireless sensor networks (WSNs), the emphasis of clustering has been on improving energy efficiency and prolonging the network lifetime by minimizing the energy consumption through the data aggregation [32, 33], and evenly distributing the energy consumption between all the CR nodes [32–36]. Recently, breakthrough in wireless power transfer (WPT) techniques [37] has introduced an alternative solution to the limited power capacity problem, which allows recharging the node battery for prolonging the network lifetime. Based on these techniques, the authors in [38] introduced the definition of the Wireless Rechargeable Sensor Networks (WRSNs). For the purpose of enhancing the charging efficiency in WRSNs, the authors in [39] propose a hybrid clustering charging algorithm, which organizes the sensor nodes into clusters based on location relationship, so multiple nodes can be simultaneously charged.

In CR ad hoc networks, the distinguished feature of CR nodes is the opportunistic dynamicity of spectrum availability, in which the spectrum availability of each CR node is different and it varies based on primary user activities. This feature has brought additional challenges to clustering in CRAHNs such as, the lack of a static common control channel for control information exchange, and spectrum sensing to avoid causing any interference to the primary users that have full rights over the channels. Hence, to cope with these challenges clustering algorithms for CRAHNs should be a spectrum-aware. Clustering in CRNs has been investigated in the context of facilitating fundamental CR network functions such as cooperative spectrum sensing [40–44], control channel assignment [5, 45–48], multi-channel MAC protocol implementation [5, 49–52], dynamic spectrum management [53], and routing improvement [54–58]. Furthermore, clustering in cognitive radio sensor networks (CRSNs) has been addressed in the context of energy saving [59–62].

The main contribution of this paper is to introduce a comprehensive overview of the different aspects of clustering algorithms in cognitive radio ad hoc networks (CRAHNs), including, clustering objectives, clustering attributes such as the approaches (assumptions) for modeling the clustering formation problem, the performance metrics, the used techniques to fix the problem, as well as open issues. Specifically, this paper concentrates on clustering formation algorithms, how these algorithms work, which parameters are used to construct clusters in each algorithm and the pros and cons of these algorithms. The application schemes that use the clustered network such as cluster-based MAC protocols that focus on the improvement of MAC protocol rather than clustering [50], cluster based cooperative spectrum sensing which their main emphasis is to enhance the spectrum outcome [44], and cluster-based routing that concentrate on the enhancement of routing instead of clustering process [54, 57] are out of scope of this paper. Also, this paper does not focus on clustering algorithms for the traditional wireless ad hoc network paradigms such as MANETs, VANETs, and WSNs. Additionally, the contributions are to present the basic concept of cognitive radio and shows the essential role of clustering and identifies its functionalities in CRAHNs.

The rest of the paper is structured as follows: in Sect. 2, an overview of CRNs is presented. In Sect. 3 the concept of clustering is explained. The problem of clustering formation in CRNs is discussed in Sect. 4. In Sect. 5, the classification of clustering algorithms based on their objectives is presented. The open issues are discussed in Sect. 6. Finally, Sect. 7 concludes the paper.

2 Cognitive radio network concept

The cognitive radio technology is intended to take advantage of the underutilization license spectrum by easing CR users to access the licensed band in an opportunistic behavior. CR technology is built on the basis of Software Defined Radio (SDR) that was introduced to release wireless systems from relying on hardware properties such as frequency bands, channel coding, and bandwidth [63]. SDRs insert software properties to wireless devices, resulting in their ability to work on various frequency bands using various modulation techniques. A CR is a device that can scan the surrounding environment and dynamically adjust its operating parameters in real-time through SDR mechanisms. The Federal Communications Commission (FCC) in 2003 formally defined the idiom "Cognitive Radio" as follows:

> A cognitive radio (CR) is a radio that can change its transmitter parameters based on its interaction with the environment in which it operates. This interaction may involve active negotiation or communications with other spectrum users and/or passive sensing and decision making within the radio. The majority of cognitive radios will probably be SDRs, but neither having software, nor being field reprogrammable are the requirements of a cognitive radio. [64]

CR users can share the primary user spectrum by using two different access techniques these are:

I. Overlay, a CR user uses the spectrum that has not been occupied by primary users when the license user came back to this unoccupied band the CR user should immediately vacate it and move to another unoccupied band.
II. Underlay which authorizes the coexistence of the license and cognitive users. However, an upper interference limit is set up for a given spectrum band in a specific geographic location such that the CR users are not permitted to transmit above that limit to avoid cause any detrimental interference to the license user while using the specific band in the specific area [63].

To detect unoccupied channel CR user scans spectrum bands at any location periodically. The unoccupied portion of the spectrum is defined by the idiom spectrum hole or white space which is a frequency band that is appointed to a primary user, but in a particular location and time is not used [66]. This is depicted in Fig. 1.

Based on the network structure (see Fig. 2), CRNs can be classified as follows:

I. Infrastructure-based CR network, which possesses a central network entity such as a base station in cellular networks or an access point in wireless local-area networks (LANs). An example of infrastructure CRN is the IEEE 802.22 standard which uses white spaces in TV frequency bands.
II. Ad hoc CR network (CRAHNs) is a distributed multi-hop wireless communication network without any fixed infrastructure.
III. Mesh network, which is a combination of infrastructure and ad hoc [4].

Recently, the cognitive radio ad hoc network has gained a great popularity since the multi-hop connections become substantial to provide a high degree of network connectivity and achieve high data rates for large distances. Moreover, the hardness of establishing a fixed infrastructure in specific situations such as emergency service operations, disaster recovery, and battlefield communication imposes the network to have an ad-hoc structure. However, the multi-hop nature of the CR network coupled with the heterogeneous spectrum availability provides many design challenges such as spectrum access management and topology management. Therefore, the clustered structure has emerged as a good solution to handle these challenges.

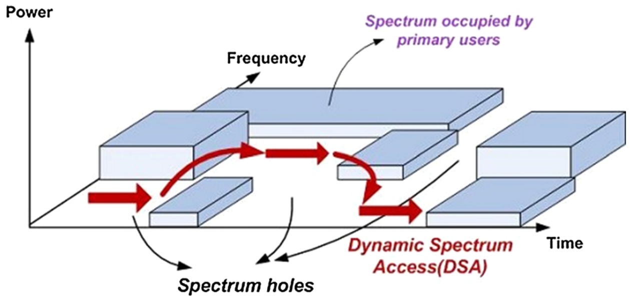
Power
Spectrum occupied by primary users
Frequency
Time
Dynamic Spectrum Access(DSA)
Spectrum holes

Fig. 1 Spectrum hole concept [4]

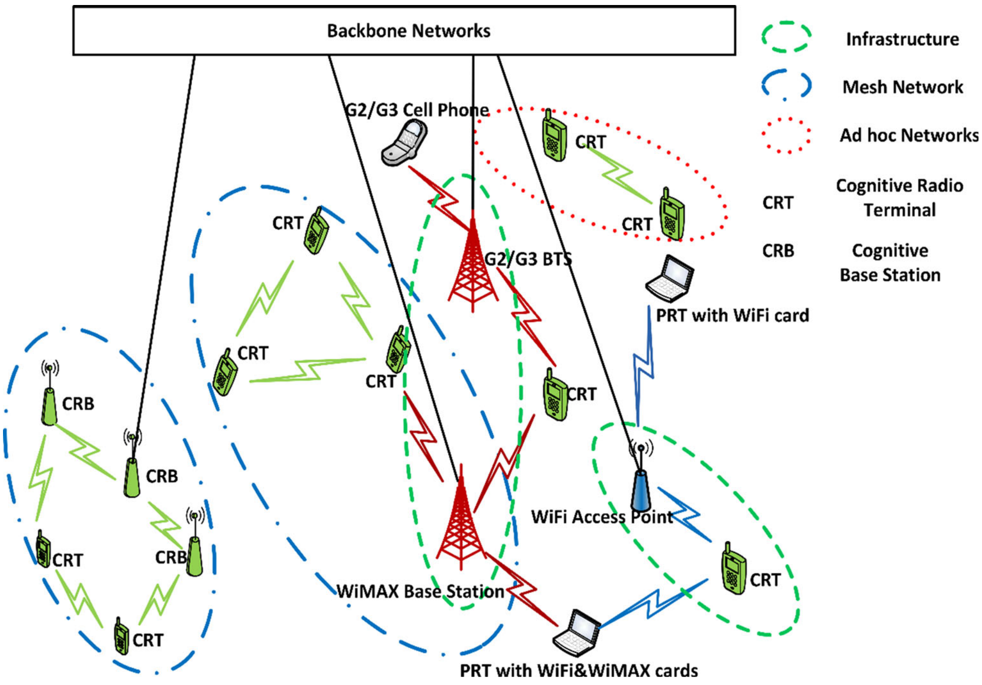

Fig. 2 Architecture of a CRN co-located with primary networks

3 Clustering basic idea

Clustering involves organizing network nodes into logical groups named clusters. Under a clustered network, CR nodes have a different role, such as cluster head, cluster gateway, or cluster member. A cluster head (CH) acts as a coordinator or a temporary base station within its cluster. Cluster gateway is a node with inter-cluster links, so it can act as a relay between neighboring clusters. A cluster member is a normal node without any privileges. A typical clustering structure is illustrated in Fig. 3.

After the clustering process, two types of communication channels are created in the network: (1) inter-clustering link, which deals with communication between clusters and (2) Intra-clustering link, which deals with communication between CH and members of the cluster.

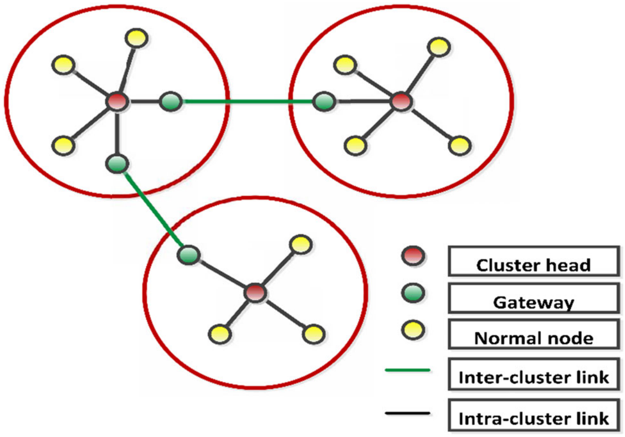

Fig. 3 Clustering architecture

Clustering process is summarized into two phases:

I. Cluster formation which handles the distribution of the CR users into virtual groups and election of appropriate user to serve as coordinator in every group.
II. Cluster maintenance, which, deal with the preservation of the clustering structure as long as possible.

4 Clustering formation in CRAHNs

4.1 Why do CRAHNs require clustering?

Clustering is considered as an effective network topology management technique which provides some benefits, which include: (1) Control channel assignment: In spite of spectrum heterogeneity in CR networks, CR users have high similarity in their list of available channels with neighbors. By grouping adjacent CR users having similar spectrum in the same cluster and using this common channel as the CCC in the cluster, clustered structure solves the problem of lack of common control channel [45, 65]. (2) Enhancement of sensing outcomes: CR users within the same cluster cooperate to decide which channels are idle. The cluster based collaborative sensing minimizes the probabilities of miss-detection and false alarm [40]. (3) Enhances the network stability: a clustered structure makes the CR ad-hoc network somewhat smaller and more manageable in the sight of each CR user. When CR user moves out of its cluster, only CR users belong to that cluster is required to refresh its information. Therefore, regional variations caused by spectrum mobility or CR user mobility do not note and update by the whole network [68]. (4) Network scalability. While the flat structure restricts the growth of network because the control overhead, such as routing establishment messages consume a large percentage of the bandwidth [69], clustered structure allows the growth of network as much as necessary by simplifying, routing and reducing the control overhead. This is because the cluster heads and cluster gateways constitute a virtual backbone for inter-cluster communication, thus the generation and dissemination of routing information can be limited in these nodes [68, 69].

4.2 Clustering formation design

Clustering formation in cognitive radio networks has additional challenges that distinguish it from the clustering in traditional wireless networks. Traditional wireless networks, even in multi-channel wireless systems usually use a dedicated channel to exchange control information. However, it is not the case in the cognitive radio networks since the available channels dynamically vary over time and location due to primary user activity. Moreover, the rapid topology changes not only due to the node mobility as in mobile ad-hoc networks (MANETs) but also due to the spectrum mobility. These challenges increase the complexity of the clustering formation design in of CR networks compared to conventional wireless networks such as mobile wireless networks and wireless sensor networks.

To overcome these challenges and ease the cluster formation design in CRAHNs three attributes must be defined. These attributes are performance metric which defines the target objective, the approach (assumptions) for modeling the problem and the appropriate technique that will be used to fix the problem. Figure 4 provides an overview of these attributes. There exist several performance metrics for clustering formation in cognitive radio networks, and these vary based on the target objectives of each algorithm. Table 1 introduces these performances metric. Basic approaches for clustering formation in cognitive radio networks are presented in Table 2 which briefs the characteristics, pros, and cons of these approaches. The most important techniques that are used for clustering formation and maintenance in CRNs are briefly presented in Table 3 which gives an abstract of these techniques, listing their characteristics, pros, and cons.

5 Classification of clustering algorithms

Clustering algorithms of CRAHNs can be classified according to different criteria. For example, depending on whether a cluster-head is elected first or the cluster is built first, clustering algorithms can be classified as cluster-head-first [55, 56, 59, 72, 77, 78] and cluster first [45, 46, 73]. Based on the hop distance between the cluster-head and its members, clustering algorithms can be divided into 1-hop clustering [5, 45–48, 56, 71–73, 77] and multi-hop clustering [55, 58, 59]. Based on the nature of the algorithms itself and how it operates, clustering algorithms can be classified as a centralized algorithm, a central unit (server or node) receives information from CRs and makes decisions or distributed algorithm, CRs makes decisions either as standalone or in cooperation with other CRs. No central entity exists. In this paper, we classify the clustering algorithms based on their objectives. Based on this criterion, clustering schemes for CRAHNs can be classified into five groups, as showed in Table 4.

5.1 Dominating-Set-based

Dominating-Set-based (DS-based) which means that each CR node is either a cluster head or is within communication range of a cluster head [5, 70]. The DS algorithm tries to find a minimum DS for a CRAHNs aiming to reduce the number of nodes that is engaged in the main network functions such as resource management, route search, and routing table maintenance. The results of such a technique include; simplifying, routing and reducing control overhead.

5.1.1 CogMesh: a cluster-based cognitive radio network

In [5] Chen et al. proposed a decentralized cluster based CR network framework to form a large scale network named Cog-mesh. Cog-mesh introduces mechanisms for neighbor discovery, cluster formation, and network topology management. Clusters are formulated based on the local spectrum availability. CogMesh consists of two phases:

Initial clustering setup phase (ICS): In this phase, the neighbor discovery and cluster formation process are presented together since they are extremely correlated. Each CR node listens to one of its idle channels for a given interval of time, waiting for beacons (messages from cluster head) on that channel. One of three events happens during a listening duration: (1) No message arrives: in this case, the CR node builds a cluster on the listening channel and becomes the cluster head. (2) A beacon comes in during the listening interval: in this case, the CR node requests to join the cluster. (3) No beacon comes but neighbor messages come: in this case, the node detects neighbor clusters, to determine if it is 2-hop away from cluster heads. On verification, the node exchanges neighbor information with the detected neighbor and moves to the next channel to listen. If the CR node cannot find a channel satisfying the first or second case after listening to all available channels, it builds a cluster on a randomly chosen channel. A cluster head discovers its neighbor clusters from

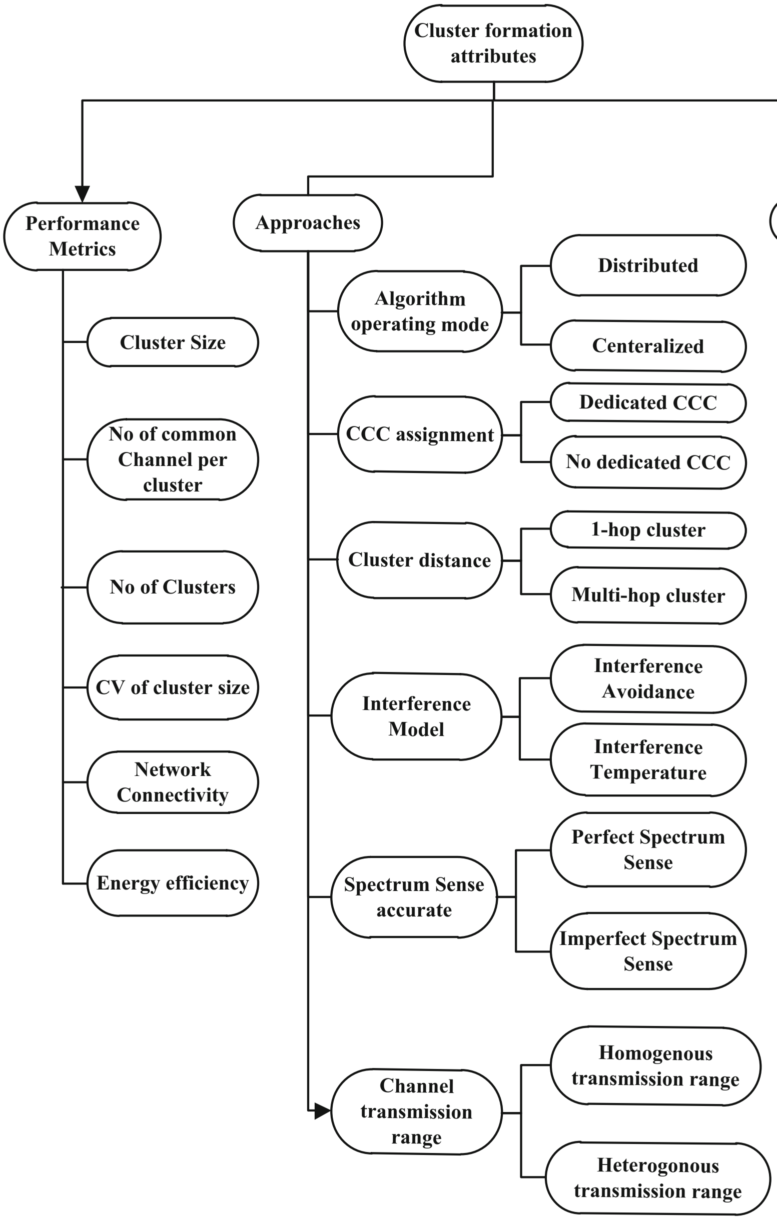

Fig. 4 Taxonomy of clustering formation attributes

the gathered neighbor information and makes interconnection by choosing gateway nodes. The ICS phase ends when all CR nodes join clusters, and clusters form interconnections.

Optimization phase: A cluster merging algorithm based on a minimum dominating set (MDS) in graph theory is used to reduce the cluster number. Clusters are merged based on the constraint, that there is at least one common idle channel for all CRs in the same cluster.

This algorithm optimizes the cluster number while it guarantees one CCC in each cluster. Nevertheless, the merging process incurs large control overhead among clusters. Moreover, the stability of the cluster is not counted, and re-cluster is easily caused by variation in spectrum availability due to the primary user (PU) activity.

5.1.2 *Efficient clustering of cognitive radio networks using affinity propagation*

In [70] Baddour et al. introduces a decentralized approach for the cluster formation process in CRN based on affinity propagation (AP) message-passing techniques. Clusters are formed based on graph domination principles, which implies that each CR is either CH or in the 1-hop neighbor

Table 1 Performance metrics for CRAHNs clustering algorithms

Performance metric	Target objective	Issues	References
Number of clusters	Minimize the number of clusters which improve network performance by simplifying routing table and reducing control overhead	Small number of clusters means large cluster size, which implies a large number of nodes share the bandwidth for intra-cluster communication	[5, 40, 46, 56, 59, 70, 72–74]
Cluster size	Maximize the number of nodes in the cluster, which reduce the intercommunication overhead	If there are no enough channels for intra-cluster communication. Large size reduces the QoS performance for the network significantly e.g. throughput	[45, 48, 74]
Number of CC per cluster	Maximize the number of CC per cluster, which allows for agile channel switching when license user activity is sensed. So the re-clustering will be avoided. It also increases intra-cluster communications throughput	If spectrum availability is highly heterogeneous between CR users, a high number of common available channels in each cluster result in small cluster sizes and a large number of clusters. Thus, the inter-cluster communications suffer from high overhead	[45, 48, 74]
Coefficient of variation of cluster size	Minimize the coefficient of variation of the cluster size which reflects the equality of the formed clusters in term of number of nodes in each cluster. A small CV implies more load balance among clusters	It is difficult to achieve, especially in a distributed approach	[45, 46]
Network connectivity	It intended to maintain the network connectivity by maximizing both the number of available channels for intra-cluster and inter-cluster communication	It deals with two contrast objectives so a fair tradeoff between the number of available channels for intra-communication and intercommunication should be done	[46, 60]
Energy efficiency	Minimize energy consumption of CRs, which extend the life of the network	It is difficult to reduce the total network power without introducing a significant effect on the network performance e.g. connectivity and throughput	[56, 59, 60]

for CH. Clusters are formed based on two measures: The similarity which is computed based on the number of common idle channels between CRs and the preference (self-similarity) which impact on the number of cluster heads.

The AP algorithm is initiated by considering all CRs in the network as a candidate cluster head.

Then each CR exchange's messages locally with their 1-hop neighbors iteratively until a subset of eligible cluster heads arises. During the iterations, two kinds of message exchange between the nodes: responsibilities $\mathrm{r(i,j)}$ sent from $\mathrm{CR_i}$ to nominee cluster head $\mathrm{CR_j}$ represents the accumulated evidence for how perfectly $\mathrm{CR_j}$ is to serve as the CH for $\mathrm{CR_i}$, taking into considering other eligible CH for $\mathrm{CR_i}$. The availability $\mathrm{a(i,j)}$ sent from the nominee cluster head $\mathrm{CR_j}$ to $\mathrm{CR_i}$ represents the accumulated evidence for how perfect it would be for $\mathrm{CR_i}$ to select $\mathrm{CR_j}$ as its CH, taking into considering the support from other nodes that $\mathrm{CR_j}$ should be a CH. At the first iteration all availability is equal to zero. Responsibility and availability are updated at each iteration using the following equations:

$$r(i,j) \leftarrow S(i,j) - \max_{j' \notin j}\{a(i,j') + S(i,j')\} \tag{1}$$

$$a(i,j) \leftarrow \min\{0, r(j,j) + \sum_{i' \notin \{i,j\}} \max\{0, r(i',j)\}\} \tag{2}$$

$$S(i,j) = |c_i \cap c_j| \tag{3}$$

where $S(i,j)$ is similarity between $\mathrm{CR_i}$ and $\mathrm{CR_j}$ and C_i, C_j are the set of available channel for $\mathrm{CR_i}$ and $\mathrm{CR_j}$ respectively. Also each $\mathrm{CR_i}$ computes its preference p(i) as

$$\mathrm{p(i) = S(i,i)} \tag{4}$$

Then each $\mathrm{CR_i}$ uses its 1- hop neighbor N_i to tune their preference as

$$p(i) = S(i,i) - \max_{k \in N_i} S(k,k) \tag{5}$$

CR node which has the highest preference among their neighbors is elected as CH.

The AP algorithm has a hard constraint that forces the CH to point to itself as CH. This may lead to invalid cluster configuration if a node chooses another node as a CH without

Table 2 Clustering formation approaches

Approach	Characteristics	Advantages	Disadvantages	References
Centralized algorithm	A central unit (server or node) receives information from CRs and makes decisions	Optimal global solutions through the overall view of the entire network performance. Can accomplish load balance between clusters	High signaling between CRs and the central unit. Unprotected from central unit failures	[41]
Distributed algorithm	CRs make decisions either as standalone or in collaboration with other CRs. Neighbor CRs exchange information to achieve optimum solutions. No central unit exists	Faster decisions Quickly adjust to network changes. Low signaling overload	It fails to provide global optimal solutions (only local). Very hard to get the load balance among the clusters	[5, 43, 45–48, 52, 56, 58–60, 70, 72–75]
Dedicated CCC	Assumes the existence of a predetermined channel for the control purposes. By utilizing license spectrum or utilizing an unlicensed band, such as the Industrial, Scientific and Medical (ISM) band, or the unlicensed ultra-wide band (UWB)	A simple approach	A fixed control channel apportionment is conflicting with the opportunistic access feature. Moreover unlicensed bands are crowded and exposed to uncontrollable interference from other unlicensed users	[41, 48, 52, 56, 58, 60]
No dedicated CCC	No dedicated CCC for the interchange of control information between the CRs. All available channels can be used for control purposes	Realistic approach	It wastes too much spectrum resource on the control function	[5, 44, 46, 58, 73, 74]
1-hop	Distance between cluster-head and its members is one hop. Thus, all the member nodes remain at most two hops distance away from each other within a cluster	Provides a shorter path for intra-cluster transmission. Also CR users in same cluster very close proximity to each other so they may be affected by the same PUs activity results in high similarity between their lists of available channels	Produces a large number of clusters compared to multi hop. High intercommunication overhead	[5, 43, 45–48, 52, 56, 59, 60, 70, 72, 73, 75]
Multi-hop	Distance between cluster head and its members more than one hop	Reduce the number of clusters which reduce the inter-cluster communication overhead	CR users in the same cluster locate in a large geographical area so they may sense different PUs activities; therefore, spectrum availability is highly heterogeneous between them. As a result, it constructs a cluster with a large size and minimum number of idle channels usually one	[55, 58, 74]
Interference avoidance (Overlay)	When the PU came back to unoccupied band the CR user should immediately vacate it and move to another unoccupied band to avoid causing interference to PU	When the license user is absent CR user could use the maximum transmission range of idle channel without any limitation	Does not allow CRs to coexist with the PUs. Frequent spectrum handoff	[5, 43, 45, 47, 48, 52, 58, 60, 75]
Interference temperature (Underlay)	Allowing the coexistence of primary and CR users. An upper interference limit is set up for a given spectrum bands in a specific location such that the CR users are not permitted to transmit above that limit in order to prevent any severe interference to the license user of that band	Low spectrum handoff	Even when licensed users are absent, CR users cannot transmit above the interference limit. Despite doing otherwise would not cause any interference	[56]
Perfect spectrum sense	Assume that all CR users could determine its set of idle channel accurately without any false alarm (sense occupied while it is idle) or misdetection (sense idle while it is occupied)	A simple approach	Not realistic	[5, 43, 45, 47, 48, 52, 58–60, 70, 75]
Imperfect spectrum sense	Spectrum sensing information is not perfect. There is a probability of false alarm or misdetection	Realistic approach	More complicated since it requires techniques to compute the both the probability of false alarm and misdetection	[56]

Table 2 continued

Approach	Characteristics	Advantages	Disadvantages	References
Heterogeneous channel transmission range	Various channels may be situated in vastly isolated frequencies with various bandwidths, and various propagation characteristics. Furthermore, Federal Communications Commission (FCC) regulations identify different transmission power for different frequency bands. Consequently, various channels have various transmission ranges	More realistic approach	More complicated	[55]
Homogenous channel transmission range	Assume that all channels have the same transmission range	A simple approach	The use of the lower channel range as the common range decreases the network connectivity by reducing the number of neighbors, therefore increasing the number of clusters in the network On the other hand, choosing the highest a transmission range increases the number of neighbors, but creates aggressive interference and also wastes the power. Moreover, it's contrary to FCC regulations	[5, 43, 45, 47, 48, 52, 58, 60, 70, 74, 75]

that node point to itself as CH. The incremental cluster formation scheme is produced to remove the invalid clusters.

This algorithm provides a lower number of clusters but due to the hard constraint, it involves too many rounds of message passing for convergence, results in wasting too much time and bandwidth. Furthermore, the existence of a CCC cannot be guaranteed in the clusters.

5.2 Common control channel establishment based

The existence of common control channel in CRNs is very important since it enables CR users to exchange the sensing data in cooperative spectrum sensing, broadcast routing information, and coordinate the spectrum's access. Unfortunately, due to the dynamic spectrum feature of CRNs, there is no idle channel common to all CRs. However, CRs in the same geographical area may have high similarity in their list of available channels. Hence, different control channels have to be assigned to different neighborhoods. This leads to dividing the CRN into clusters, whereby the CRs of a given cluster share a common control channel [5, 45–48].

5.2.1 Distributed coordination in dynamic spectrum allocation networks

In [46] Zhao et al. proposed a distributed coordination protocol to form groups according to spectrum heterogeneity in the CR network. In this protocol, CR nodes self-organize into groups based on the similarity between their lists of available channels. Only members of the same group can directly communicate with each other. CR nodes on the boundary of groups may belong to multiple groups, and act as relays for inter-group communication.

To build coordination groups, CR users are required to gather information about its neighbors. This is achieved through neighbor discovery. After a neighbor discovery, each CR user has got a table of its neighbors and their idle channels. Depending on this information, CR users select their coordination channels. The process of coordination channel selection is a decentralized elective procedure where each CR user chooses a channel with the biggest connectivity degree as the coordination channel. To adapt the primary user activity, the second largest connectivity degree channel is chosen as a backup coordination channel. When the primary user occupies a coordination channel, CR users migrate to the backup coordination channel. Before moving to the backup channel, CR users broadcast the decision on the main channel to notify group members about the variations.

This protocol provides large-group size while it guarantees one CCC for each group. However, when a license user uses the CCC, the declaration of new CCC is sent on the main channel, which causes interference to the license user.

5.2.2 A spectrum opportunity-based control channel assignment SOC and C-SOC

In [45] Liu et al. proposed two distributed clustering algorithms named Spectrum-Opportunity Clustering (SOC)

Table 3 Design techniques used for clustering

Technique	Characteristics	Advantages	Disadvantages	References
Heuristics	Generally speaking, heuristic means to learn or to find out by trial and error. Iterative algorithms, used to find a good feasible solution that is at least close to being optimal	Simplicity, easy implementation. Have acceptable time and space complexity	Most of developed approaches are designed to fit a specific problem and cannot be used for other problems. Obtain local optimum solution, not the global optimum solution. There is no analytical model for studying their convergence	[47, 56, 58]
Greedy Heuristic	A step-by-step formula for solving a problem. A potential solution is constructed iteratively. At each iteration; the superior element is added into a partial solution, until a full solution is constructed. It implies that, for a specific problem, one has to identify a solution as a set of elements. This set is sorted based on a greedy function that computes the gain of choosing each element	They are easy to implement, they require less computing resources, they are much faster to execute	Greedy algorithms usually fail to achieve the globally optimal solution, because they do not work comprehensively on all the data. They make obligations to specific options too early (During the search for the solution they take the best element next without regard for the rest of the solution) which stop them from achieving the best full solution later	[43, 45, 48, 60, 75]
Graph theory	CRNs are visualized as undirected graphs, where the vertices represent the CR nodes and edges represent the links between CR nodes	Simplified assumptions Use existing solutions of graph theory such as minimum dominating set (MDS), bipartite and biclique graphs	The computational problem of finding a minimum dominating set and the maximum edge biclique of a bipartite graph is NP-complete problems, which mean that no polynomial time algorithm can guarantee an optimal solution	[5, 43, 45, 47, 52, 70, 75]
Game theory	Game theory is a theory of decision making under conditions of uncertainty and interdependence. Clustering is modeled as a game where the CRs are the players. Solutions are found through Nash equilibrium	A very effective analytical tool for studying and predicting the manner of reasonable and selfish entities	Complicated. Hard to construct the game in a manner that assure equilibrium is always accomplished. The time of convergence to Nash Equilibrium, even under the best response moves of the players, can be very high	[48]
Affinity propagation (AP)	AP is a clustering technique based on the concept of message passing between data points. It accepts as input measures of similarity between pairs of data points which are selected based on application demands and the aim of the grouping procedure. The AP algorithm is initiated by considering all data points as potential exemplars. Real-valued messages are interchanged between data points until a high-quality set of exemplars and corresponding clusters gradually emerges	More accurate and efficient compared to other clustering techniques	It needs too many rounds of message passing for convergence, results in wasting too much time and bandwidth	[70]

and Constrained-SOC (C-SOC) to fix the problem of control channel assignment in cognitive radio networks. In these algorithms, the clustering is formulated as a maximum edge biclique construction problem. Similar concept also has been used in [70–73]. The SOC algorithm aims to make a fair balance between two conflicting

Table 4 Summary of five clustering schemes

Category	Objectives
DS-based clustering	Identifying a minimum DS for a CRAHN aiming to reduce the number of nodes that take part in the main network functions such as resource management, route search and routing table maintenance
Common control channel establishment based	Assigning local control channel. Due to the dynamic spectrum feature of CRAHNs there is no common global control channel among CR nodes in the entire network. However, CRs do share significant spectrum with local neighbors. Hence, by dividing the CRAHN into clusters and associate neighbors node in the same cluster, a common control channel could be assigned for each cluster
Stability based clustering	Improving the stability and the robustness of clustering by making tradeoffs between cluster size, Number of common idle channels per cluster and number of common idle channels between the neighbor clusters
Energy-efficient clustering	Eliminating needless energy consumption or equalizing energy consumption for CR nodes in order to extend the lifetime of the network
Cooperative spectrum sense based	Improving the outcome of spectrum sensing by decreasing the probabilities of miss-detection and false alarm caused by propagation effects such as fading and shadowing

factors, the cluster size and the number of common channels in the cluster. While C-SOC algorithm aims to create clusters with maximum cluster size while keeping the number of common idle channels in a cluster equal to or greater than a predefined value. SOC and C-SOC consist of three steps: maximum edge biclique computation, updating cluster membership information and finalizing cluster membership.

Step 1: Maximum edge biclique computation. A greedy heuristic algorithm is used to compute the maximum edge biclique as following: Each $\mathrm{CR_i}$ individually determined its set of idle channel $\mathrm{C_i}$. After neighbor's discovery, each $\mathrm{CR_i}$ knows its one-hop neighbors $\mathrm{CR_j}$ and their idle channels list $\mathrm{C_j}, \forall \mathrm{CR_j} \in \mathrm{N_i}$ as is shown in Fig. 5. Then each $\mathrm{CR_i}$ uses this information to construct a bipartite graph.

A graph $\mathrm{g(V, E)}$ is a bipartite graph if the set of vertices V can be divided into two separate sets A and B with $\mathrm{A \cup B = V}$ such that every edge in E links a vertex in A to a vertex in B. For $\mathrm{CR_i}$, set $\mathrm{A_i}$ represents its neighbor set $\mathrm{N_i}$ plus $\mathrm{CR_i}$ itself ($\mathrm{A_i = N_i \cup CR_i}$), while $\mathrm{B_i = C_i}$. an edge (x, y) exists between vertices $\mathrm{x \in A_i}$ and $\mathrm{y \in B_i}$ if $\mathrm{y \in C_i}$.

Using its bipartite graph, each $\mathrm{CR_i}$ builds the maximum edge biclique graph which contains its cluster membership information. A bipartite graph $\mathrm{Q(V = X \cup Y, E)}$ is a biclique if for each $\mathrm{x \in X}$ and $\mathrm{y \in Y}$ there is a link between x and y. For $\mathrm{CR_i}$ a biclique graph $Q_i(X_i, Y_i)$ is extracted from its bipartite graph g_i. This biclique represents a cluster of nodes X_i that have channels $Y_i \subseteq C_i$ in common. Figure 6(a) displays the bipartite graph built by $\mathrm{CR_a}$ from the Fig. 5. The set of vertices $\mathrm{A_a}$ represents the set of one hop neighbors $\mathrm{N_a = \{b, c, d, e\} \cup a}$, while the set of vertices $\mathrm{B_a}$ represents the available channels set of $\mathrm{CR_a}$, which is $\mathrm{C_a = \{1, 2, 3, 4, 6, 7, 8\}}$. Here, $\mathrm{CR_a}$ is connected to all vertices in $\mathrm{B_a}$. The maximum edge biclique graph for $\mathrm{CR_a}$ is shown in Fig. 6(b).

Step 2: Updating cluster membership information: Each $\mathrm{CR_i}$ broadcast the computed maximum edge biclique to its one-hop neighbors. Each $\mathrm{CR_i}$ compare its biclique with the neighbor's bilique in terms of channel number and cluster size to check if there is one provides better clustering and update accordingly. New biclique is rebroadcasted.

Step 3: Validating cluster membership: Each CR tests the cluster membership to ensure the cluster validation. After the clusters are constructed a CR with 1-hop neighbor to all cluster members elected as CH.

This algorithm uses the hopping sequence on the common idle channels in a cluster as CCC of the cluster, results in making the cluster more robustness to primary user activities since the cluster preserves until all common channels are unavailable. However, the hopping needs a perfect synchronization between the cluster members.

The main criticism of the SOC scheme comes from its tendency to create clusters with a high variance in the size; many clusters may have only one member. Furthermore, the C-SOC scheme may lead to invalid clustering configuration if the constraint threshold is high.

5.3 Stability based clustering

The principal drawback of clustered network comes from the needs for additional message interchange between CR nodes for repairing the cluster structure. CR network topology varies rapidly not only due to CR node's mobility, but also due to spectrum mobility, causing repeated cluster structure changes. Subsequently, control overhead for cluster repairing grows significantly. Thus, the clustering structure may waste a huge amount of network bandwidth and deplete CR node's energy fast [67]. Hence, it is needful to minimize the communication overhead resulted from cluster repairing. Almost all the stability based clustering algorithms in CRNs intend to produce a steady cluster structure by improving intra-cluster and inter-cluster connectivity, resulting in decreasing the re-affiliation rate and reducing re-clustering cases [45, 48, 52, 55, 58, 74, 75].

5.3.1 ROSS (robust spectrum sharing) algorithm

In [48] Li et al. proposed an algorithm to address the problem of how to group CR users into clusters and preserve the connectivity of the CRN. ROSS forms clusters based on one hop neighborhood. ROSS forms a spectrum-aware cluster based on the game-theoretic framework. ROSS algorithm consists of two phases: Phase I: cluster formation and Phase II: membership determination.

Phase I: Cluster formation has two steps:

Step 1: Determining cluster heads: After spectrum sensing and neighbor discovery, each CR computes two values: spectrum connectivity degree, which is the number of links between CR and its neighbors and local connectivity degree, which is the number of common channels between CR and its 1-hop neighbors. A CR node which has the least spectrum connectivity degree among its neighbors is selected as a cluster head.
Step 2: Cluster Formation: when CR node being the cluster head, it builds the initial cluster, including all 1-hop neighbors. Initial clusters may have no common idle channels. This fixed by eliminating nodes from the cluster until there is at least one common channel in each cluster.

Phase II: Membership determination: In this phase, overlaps nodes join one cluster and dissociate from the other clusters. Each overlapped node determines its membership with the aim of increasing the number of CCC in each cluster.

ROSS authors presented an algorithm that takes the connectivity between clusters (inter-cluster communication) as the main objective and this affected on important parameters such as a common idle channel per cluster and cluster size, which are considered as secondary issues.

5.3.2 COMBO algorithm

In [74] Asterjadhi et al. proposed a distributed algorithm (Combo) which aims at creating non overlapping clusters of a given size (in the number of hops) that takes into account the number of common available channels among CRs when making decisions. COMBO provides mechanisms for K-hop neighbor discovery and cluster formation.

After the neighbor discovery, all CRs run the clustering algorithm independently as follows: Each CR computes a weighted priority key based on the number of common channels between the node and its k-hop neighbors, k-degree of connectivity (the no of its k-hop neighbors) and the ID of the node. Then each CR node broadcast its weighted priority value to its k-hop neighbors. A CR node whose has the biggest weighted priority key in their neighborhood creates a cluster and broadcast a join message to their neighbors. Nodes that receive the message join the cluster, else, if they do not receive a message from any cluster head, they declare themselves as cluster heads. The algorithm ends when all nodes have been associated with a cluster as a cluster head or normal node.

In this algorithm, a reasonable number of common idle channels are ensured in each cluster. However, since the number of intra-cluster common channels is favored in the optimization, the size of created clusters is tending to be small.

5.3.3 Stability based clustering for high-mobility multi hop cognitive radio networks

In [55] Huang et al. proposed spectrum-aware clustering algorithm, which aims to create two hop cluster in high-mobility, multi-hop, and heterogeneous CRN environments. In this scheme, a novel metric named a node importance degree $d_{i,c}$ is introduced which is defined as

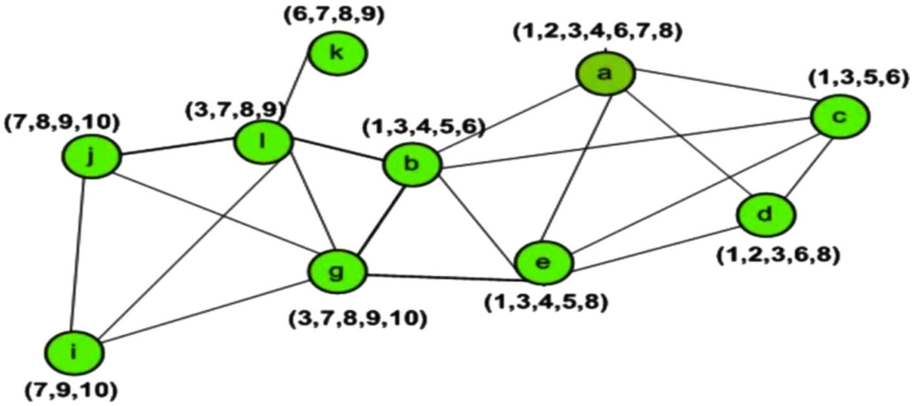

Fig. 5 Connectivity graph of a cognitive radio network with available channels set [71]

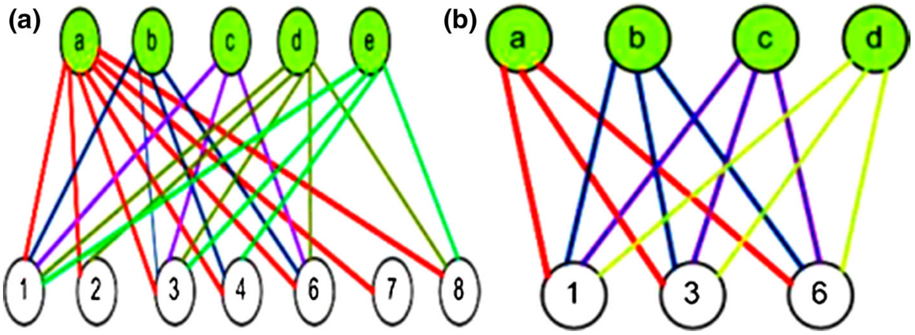

Fig. 6 **a** Bipartite graph constructed by node a, **b** maximum edge biclique graph of node a [71]

$$d_{i,c} = \frac{n_{i,c}}{1 + \frac{1}{n_{i,c}}\sum_{j=1}^{n_{i,c}} H_{ij} + \frac{1}{n_{i,c}}\sum_{j=1}^{n_{i,c}} Sw_{ij}} \quad (6)$$

where $n_{i,c}$ equal to the number of two hop neighbors of CR_i on channel c, H_{ij} is the hop count between CR_i and $CR_j\, j \in n_i$ and Sw_{ij} is channel switch count between CR_i and $CR_j \in n_i$

Initially, each node broadcasts its idle channel set, location, speed, best position, and mobility characteristics in each available channel. Thus, each node gets the information about its one hop and two-hop neighbors. Then, each node computes its node importance degree in each available channel by using Eq. (6) and chooses the largest one as its importance degree. The node with the highest importance degree among its two-hop neighbors is selected as the CH. Then it chooses the channel that has the largest importance degree as the intra-cluster control channel. Each non-cluster head node selects the cluster head with the largest node importance degree within its two-hop neighborhood as its CH.

This algorithm tends to produce stable clusters with large cluster size results in low intercommunication overhead. However, the intra-communication may suffer from serious delay due to the intra-cluster distance (two hops between CH and its members and more than two between the members themselves) and the channel switching due to the lack of common channel between the cluster head and the members of the cluster.

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

In [52] the authors proposed a distributed clustering algorithm for CRSN. The proposed clustering mechanism divides the network into clusters based on three values: spectrum availability, node power level, and node speed. In this algorithm, a novel metric named CHEV is used to select the cluster head. This algorithm has two steps as follows:

Step 1: Maximum vertex biclique graph construction: After the neighbor discovery, each CR node builds its own neighbors list. Using this information each node constructs its own undirected bipartite graph, which is used to create the maximum vertex biclique graph. This is done in the same way as in [44].

Step 2: Cluster head election: The cluster head election process is interpreted as a maximization problem as follows:

$$CH_j = \max_{i_j} CHEV_{i_j}, \quad 1 \le i_j \le CM_j \quad (7)$$

where $CH_j\, and\, CM_j$ are the cluster head and the cluster member for cluster j respectively. Here, the maximization problem is solved by using the descending sorting algorithm. The factor CHEV is calculated by each CR independently. After the maximum vertex biclique graph construction, each CR_{i_j} uses the maximum vertex biclique graph information, coupled with its energy and speed information to calculate its CHEV as follows:

$$CHEV_{i_j} = log\left(W_{i_j} * N_{i_j}^{ch_{i_j}}\right) \quad (8)$$

$$W_{i_j} = \frac{\gamma_{i_j}}{\sum_{i_j} \gamma_{i_j}} \quad \text{where} \quad \gamma_{i_j} = \frac{E_{i_j}^{\alpha}}{V_{i_j}^{\beta}} \quad \text{and} \quad 0 < \alpha, \quad \beta \le 1 \quad (9)$$

$N_{i_j}, ch_{i_j}, E_{i_j}\, and\, V_{i_j}$ are the numbers of neighboring nodes, the number of common channels the energy and the speed for node i_j respectively, and α, β are determined based on the application requirement. The idea behind the calculation of CHEV factor using the above equations is to give the preference for the CR_i node with the highest number of common channels, the highest number of neighbors, the highest energy, and the lowest speed in its neighborhood to be the cluster head.

In this algorithm, the authors present the cluster head selection process, but how the clusters are constructed is not clearly mentioned.

5.3.5 SMART (A SpectruM-Aware ClusteR based routing scheme for distributed cognitive radio networks

In [58] Saleem et al. proposed a distributed clustering scheme for CRAHNs named a SpectruM-Aware ClusteR based routTing (SMART). SMART allows CR nodes to organize into clusters and enables each CR source node to identify a route to its destination CR node in a clustered network. SMART uses an artificial intelligence scheme named reinforcement learning (RL) to assist CR nodes to make right decisions on routing. Clearly, SMART is routing based clustering algorithm. Here, our concern on the clustering algorithm rather than routing protocol. SMART introduces new metric to measure the channel quality called the channel capacity metric. This metric is used by CR nodes to rank the available channels in the clustering and routing processes, and it is defined as the probability of the channel being in off state at time t. The channel capacity metric for channel k at time t (φ_k^t) is computed as follows:

$$\varphi_k^t = \frac{\lambda_{ON,j}^k}{\lambda_{ON,j}^k + \lambda_{OFF,j}^k} + \left(\frac{\lambda_{OFF,j}^k}{\lambda_{ON,j}^k + \lambda_{OFF,j}^k}\right) * \left(e^{-\left(\lambda_{ON,j}^k + \lambda_{OFF,j}^k\right)^t}\right) \tag{10}$$

where $\lambda_{ON,j}^k$ and $\lambda_{OFF,j}^k$ are rate parameters of the exponential distribution for Primary user j. A channel k assigned to PU j has the higher φ_k^t is considered as a better channel, and so it gave a higher rank.

In order to maximize the cluster stability, SMART has a constraint value for the minimum number of common channels per each cluster, which is verified by the cluster heads. SMART consists of two phases cluster formation and cluster maintenance.

Phase I: Cluster formation: It has three steps as follows:

Step 1: Cluster head election step: The CR node with the highest number of available channels in its neighborhood declares itself as cluster head by sending cluster head information message (CHinfo) in its all available channels. This CHinfo message includes the cluster head information such as ID, available channels, the main channel, and backup channel.

Step 2: The node joining step: Node joining is the process of associating a non-clustered CR node with a cluster. It is interaction process between the cluster heads and non-clustered nodes. In this process, each CR node scans all of its available channels in a sequential manner, and each channel is scanned for a specific duration. During scanning duration, if CR node receives CHinfo message, it stores the ID of the sender and the respective information in its neighbor information table. At the end of scanning duration, there are two possible cases in which the CR node chooses to join a cluster. First, CR node has received CHinfo message from only one cluster head, and so it chooses its cluster. Second, CR node has received more than one CHinfo messages from different cluster heads, then it ranks these cluster heads based on their main channel capacity metric. The cluster heads are ranked such that the higher the main channel metric has the higher rank. Then, the CR node chooses the cluster head with the highest rank. After that, the CR node sends joining request message to its selected cluster head and waits for its response for a specific duration. Also here, there are two cases: CR node receives an acceptance message from its selected cluster head and joins it as a member. Otherwise, the next highest rank cluster head is chosen. This continues until each CR node joins a cluster. From the cluster head viewpoint, once it receives joining request, it sends back acceptance message if the number of common channels with the requested node satisfies the threshold for the minimum number of common channels in a cluster. Otherwise, it sends a rejection message to the requested CR nodes.

Step 3: Gateway election step: After cluster formation, neighboring clusters may have a different main channel (operated channel), this may cause a problem in the inter-cluster connectivity. Particularly, because the cluster heads do not switch from their main channel. Hence, the need for relay node between neighboring clusters arises. Therefore, each member node periodically scans each of its available channels in order to detect neighboring clusters. If it receives CHinfo message from any neighboring cluster, it sends gateway role request to its own cluster head. The cluster head accepts its member request to act as a gateway if it has no gateway to that cluster or the requested node has lesser number of hops to that cluster than the existed gateway. If the requested node and the existed gateway have the same number of hops to the adjacent cluster, the tie is broken by selecting the node with the highest number of available channels.

Phase II: Cluster maintenance phase: It consists of cluster merging and cluster splitting

Clustering merging: is the process of merging to adjacent clusters into one cluster. This is done only if the number of common channels between the two merged clusters satisfies the threshold for the minimum number of common channels per cluster.

Cluster splitting: is a process of dividing one cluster into two. This happens only when the cluster head detects that the number of common channels in its cluster does not satisfy the threshold for the minimum number of common channels per the cluster. Hence, the cluster head defines two sets of common channels for the new created clusters. Then, it associates its own member nodes to these clusters based on their list of available channels. Finally, it elects a cluster head for each of the clusters and quits its role as a cluster head.

This algorithm tries to form large stable clusters in order to minimize intercommunication overhead and short routing path. Therefore, it forms the clusters in a way that CR nodes are associated with same cluster as long as they fulfil the constraint of the number of common channels. However, this algorithm ignores the fact that there is an inverse relationship between the cluster size and number of common channels per cluster. As a result, it creates a stable small size cluster or large unstable clusters based on the value of common channels threshold.

5.3.6 *Graph cut based clustering for cognitive radio ad hoc networks*

The authors in [75] proposed a distributed spectrum-aware clustering algorithm, which aims to create clusters in a way that maximizing the common channels per clusters and minimizing the common channel between neighboring clusters. Maximizing the common channels per cluster lead to increase the cluster immunity against PU activity. While minimizing the common channels between adjacent clusters reduces the interference between them. The proposed algorithm defines a new hybrid metric to measure the similarity between CR nodes, which is used to model the CR local topology as a simple weighted graph. The similarity metric based on two parameters:

I. The ratio of common channels (RCC), measures the degree of overlap between CR nodes channel availabilities. It is computed by

$$RCC_{ij} = \frac{C_i^T \times C_j}{M} \tag{11}$$

where C_i,C_j are the available channels for CR_i and CR_j respectively, and M is the total number of channels

II. Relative position similarity (RPS), represents the relative position between two CR nodes. It is estimated based on the relative distance, which is computed based on received signal power as follows:

$$d_{ij} = 10^{\frac{RSSI_0 - RSSI_{ij}}{10 \times \alpha}} \tag{12}$$

$$PRS_{ij} = 1 - \frac{d_{ij}}{d_{max}} \tag{13}$$

where α is the propagation path loss exponent and $RSSI_0$ is the received signal strength at $1m$ of distance from CR node.

Based on these two parameters, the similarity metric (S_{ij}) between CR_i and CR_j is computed as

$$S_{ij} = \frac{RCC_{ij} + PRS_{ij}}{2}. \tag{14}$$

This algorithm consists of two phases:

Phase I: Initial cluster formation: In this process, first, each CR_i constructs a weighted graph $G(N_i, E_i, S)$ where N_i is the CR_i neighbors set, E_i is the CR_i link set, and S (similarity metric) is the weight of link set. Using its weighted graph each CR_i formed its own cluster. Thus, the clustering problem is modeled as a graph cut problem, where the graph cut of G for CR_i is $(X_i, \bar{X}_i)\, and\, X_i \subseteq V_i$ are CR nodes in the same cluster with CR_i and $\bar{X}_i$ are the excluded CR nodes. Due to the fact that, the clustering goal is to maximize the intra-cluster and to minimize inter-cluster, the graph cut $(X_i, \bar{X}_i)$ should minimize $cut(X_i)$ and maximize $U(X_i)$. Therefore, the graph cut is interpreted as the following optimization problem:

$$min(cut(X_i) - U(X_i)) \tag{15}$$

$$\text{given}\, cut(X_i) = \sum_{j \in X_i, k \in X_i^-} S_{jk} \quad \text{and}\, U(X_i) = \sum_{j \in X_i} S_{ij} + \prod_{j \in X_i} C_j \tag{16}$$

where $cut(X_i)$ and $U(X_i)$ are the cut cost and the utility of cluster X_i respectively, and C_j is the channel availability vector for CR_i

Phase II: Cluster synchronization: is the process of synchronization the independently formed clustering. Thus, all CR nodes have consistent clustering information. This process contains three steps as follows:

Step 1: Broadcasting cluster membership: Each CR_i node broadcasts its constructed cluster (X_i) in its available channels using hoping sequence.
Step 2: Updating cluster membership: Each CR_i compares its cluster with received clusters. If there is a cluster with better cluster utility and contains CR_i, CR_i replaces its cluster with it, and informs its neighbors with the new clustering membership X_i'.
Step 3: Validating cluster membership: Each CR_i checks if there is $CR_j \in X_i'$ and $CR_i \notin X_j'$, then CR_j is removed from X_i' to have the final cluster X_i''.

This algorithm tends to create clusters with a reasonable number of common channels in each cluster. However, this is achieved at the expense of the cluster size. Moreover, decreasing the number of common channels between the adjacent clusters may lead the network to be disconnected, specifically in the case of the high primary user's activities.

5.4 Energy-efficient clustering

In energy efficient based clustering CR nodes form clusters with the aim of getting rid needless energy consumption or equalizing energy consumption for CR nodes in order to prolong the lifetime of the entire network. This can be done by reducing the transmission power, intra-cluster distance, and the Euclidean distance between the cluster head and its member nodes [76]. Furthermore, since a cluster head works as a local coordinator, it has extra tasks compared to normal nodes so it dies early because of extravagant energy consumption, the role of the cluster head is rotated among all nodes in a cluster in order to avoid node failure and achieve equalized energy consumption among nodes throughout the entire network [58, 75]. Furthermore, since the cluster formation and maintenance consumes energy, temporal event-driven clusters are formed between event and sink when an event is detected in CRSN [60, 61].

5.4.1 Event-driven spectrum-aware clustering (ESAC) protocol

In [60] Ozger et al. proposed event-driven spectrum-aware clustering (ESAC) protocol for cluster formation in CRSNs which aims to minimize energy consumption. ESAC builds a temporal one hop cluster on the basis of event detection. Clusters are formed only in the area between event and sink and are no longer available after the end of the event. This avoids the energy consumption required to unnecessary cluster formation and maintenance. ESAC protocol consists of two phases: Phase I Determination of eligible nodes for clustering and Phase II Cluster formation.

Phase I: Determination of eligible nodes for clustering

The event, detecting nodes become eligible for clustering directly. Eligible nodes send Eligibility for Clustering REQuest (EFC REQ) messages to their 1-hop neighbors through the common control channel. Nodes that receive this message become eligible nodes for clustering only if they locate closer to the sink and farther to the event than the EFC REQ sender. The new eligible nodes send eligibility for clustering REQuest (EFC REQ) messages to their non-eligible 1-hop neighbors, and the process repeats until EFC REQ reaches the sink.

Phase II: Clusters formation: Cluster formation has two steps

Step 1: Determining cluster heads: After determining of eligible nodes for clustering, each eligible node is assigned a weight which is calculated based on the number of available channels, the eligibility degree and the distance between the node and the sink as follows:

$$W_i = |C_i| \times D_i^e + 10 \div d_{i,sink} \quad (17)$$

where $\mathrm{W_i}$ is the weight, $|\mathrm{C_i}|$ the number of available channels, and $\mathrm{D_i^e}$ = eligible node degree for $\mathrm{CR_i}$ respectively. $\mathrm{d_{i,sink}}$ is the Euclidean distance between $\mathrm{CR_i}$ and sink.

A CR node which has the highest weight among its neighbors is elected as a cluster head.

Step 2: Cluster Formation: After becoming cluster head, CR node forms cluster by finding the maximum products of the three terms $|N_{i1}| \times |C_{ic}| \times N_{i2}$ as follows: Firstly cluster-head $\mathrm{CR_i}$ determine the weight of every channel in $\mathrm{C_i}$ which is the number of one- hop neighbors that $\mathrm{CR_i}$ can reach through that channel. Secondly, the channel $j \in C_i$ that has the highest weight is added to C_{ic} and the nodes that have this channel in their available channel list are added to N_{i1} and the two- hop neighbors that can reach through this channel are added to N_{i2}. If two or more channels have the same weight, the tie is broken by choosing the lowest ID one. This process repeats until the list of available channels of $\mathrm{CR_i}$ are all added in C_{ic} or $|N_{i1}|$ is equal zero. After each iteration, the weight is calculated as the $|N_{i1}| \times |C_{ic}| \times N_{i2}$. Finally, cluster-head i constructs its cluster by N_{i1} and C_{ic} of the iteration, which has the highest weight.

where C_{ic} is common channel list between $\mathrm{CR_i}$ and its neighbors, $|N_{i1}|$ = The number of 1-hop neighbors of $\mathrm{CR_i}$ that they have the channels in C_{ic} in their available channel list, and $|N_{i2}|$ = The number of two hop neighbors of $\mathrm{CR_i}$ that they have the channels in C_{ic} in their available channel list.

The event-driven approach has been demonstrated to provide lower energy consumption. Moreover, it tried to make a compromise between cluster size, common channels per cluster and two hop neighbors that can be reachable by cluster-head through its members.

5.4.2 Mobility-aware Event-to-sink Spectrum-Aware Clustering (mESAC)

In [61] Ozger et al. proposed a Mobility-aware Event-to-sink Spectrum-Aware Clustering (mESAC), which is a mobility, enhanced version of their previous clustering protocol (ESAC) in [60]. Same as in [60] clusters are built only in the corridor between event and sink and are no longer exist after the end of the event. mESAC consists of two phases. An eligibility corridor determination phase which aims to find the eligible nodes for clustering using the same procedure as in [60], and the cluster formation phase which includes two steps cluster head determination and cluster member's determination.

Step 1: Determining cluster heads: In this step, each eligible node CR_i computes its weight value P_i based on the eligible node degree D_i, the number of available channels c_i, distance to the sink $d_{i,sink}$, remaining energy E_i, and the node speed v_i as follows

$$P_i = w_1|D_i| + w_2|c_i| + w_3|E_i| + w_4|d_{i,sink}| + w_5\left|\frac{1}{v_i}\right| \tag{18}$$

For each parameter there exists a corresponding weight value w_j which is assigned according to system requirements and the sum of the weight of the parameters is 1. Thus

$$\sum_{j=1}^{5} w_j = 1 \tag{19}$$

The election process is executed at each eligible node in a distributed iterative way such that an eligible node CR_i decides its own role (cluster-head or potential member) depending only on the decision of its neighbors with the bigger weight value. Thus, at each iteration, only those eligible CRs with the biggest weight in their neighborhood becomes cluster-heads. These cluster heads and their one hop neighbors are removed from the eligible nodes list. Hence, the set of eligible node is updated after each iteration. Then, the remaining eligible nodes again execute the election process. This process continues until all eligible nodes being either a cluster-heads or a potential cluster member.

Step 2: Cluster member's determination: Each cluster head checks if there is available channel common between itself and all other eligible neighbors. If there is at least one common channel, then the cluster head sends cluster request message to all its eligible one-hop neighbors. If there is no common channel among the eligible one-hop neighbors, the cluster head finds the channel with the largest number of eligible one-hop neighbors and defines it as the favorite cluster channel. Then, the cluster head sends cluster request message to the eligible one-hop neighbors which are having that channel in their channel availability list. If there is more than one channel have the same number of one-hop eligible neighbors, the tie is broken by selecting the channel with smaller channel ID as the favorite cluster channel. After receiving the cluster request message from their neighbor cluster heads, each potential member node sends a reply message to its cluster head. If the potential member has received more than one cluster request messages from multiple neighboring cluster heads, it sends the reply message to the cluster head with smallest weight. This is done to reduce the load of the cluster-heads with bigger weight due to the assumption that the cluster heads with bigger weight have a greater number of potential members.

The mESAC has been demonstrated to provide a lower energy consumption. However, in the presence of high primary user activity, one common channel per cluster may cause frequent re-clustering. Consequently, a significant amount of saving energy may consume in cluster maintenance. Moreover, it may suffer from the delay caused by the spontaneous cluster formation.

5.4.3 A spectrum-aware cluster-based energy efficient multimedia (SCEEM) routing protocol

In [56] Ghalib et al. proposed a spectrum-aware cluster-based energy-efficient multimedia (SCEEM) routing protocol for CRSN. In SCEEM, a clustering process begins with spectrum scanning on the potential list of frequency bands. After that, information sharing phase starts in which each CR node exchanges its available channel, mean available time of each of the available channels, which is computed based on the past channel statistics and its residual energy. This is achieved by a broadcast Info message on a predefined common control channel. Then each CR node calculates its spectrum energy rank as follows:

Let's $\mathrm{V_i(t)}$ be the channel availability matrix of $\mathrm{CR_i}$ at time t for M channels $\mathrm{V_i(t)} = [v_{i1}v_{i2}v_{i3}\ldots v_{iM}]$ where

$$v_{ic} = \begin{cases} 1 & \textit{if channel c is available for } CR_i \\ 0 & \textit{otherwise} \end{cases} \tag{19}$$

Likewise, $T_i(t)$ the expected channel availability vector for CR_i which is an average available time of each available channel and is represented as $T_i(t) = [T_{i1}T_{i2}T_{i3}\ldots T_{iM}]$. CR_i Computes the associated channel availability matrix A_{ij} with CR_j that permits them to access the channels between them as $A_{ij} = \left[A_{ij1}A_{ij2}A_{ij3}\ldots A_{ijM}\right]$ where

$$A_{ijc} = \left(v_{ic} * v_{jc}\right) * \min\left(T_{ic}, T_{jc}\right) \tag{20}$$

Then CR_i computes the relative spectrum availability rank $\mathrm{Y_{ij}}$ with neighboring CR_j as

$$\mathrm{Y_{ij}} = \frac{1}{\mathrm{M}}(\mathrm{A_{ij}} * \mathrm{I}) \tag{21}$$

where $\mathrm{I} = [\mathrm{I_1 I_2 \ldots I_M}]^{\mathrm{T}}$ and $\mathrm{I_c} = 1$. Finally CR_i calculates its spectrum energy rank $\mathrm{Y_i}$ using Eq. (11)

$$\mathrm{Y_i} = \sum_{\mathrm{j \in N_i}} \mathrm{Y_{ij}} * \frac{\mathrm{E_i}}{\max\limits_{\forall \mathrm{k \in N_i}} \{\mathrm{E_k}\}} \tag{22}$$

where $\mathrm{E_i}$ is residual energy for CR_i

CR_i compares its rank value with all the neighbors' values, and if it has one of the top three ranks, it considers itself as a potential cluster head. Then each potential cluster head runs a random timer and waits. If it does not receive any CH message until its timer expired, then it declares itself as CH and the broadcasts CH message to inform the neighbors to join its cluster as normal nodes. Else, if it receives a cluster head declaration message from one of its neighbors before its timer expires, then it clears its timer and associates to that cluster as a normal node.

If a non-cluster head node receives more than one join messages, it decides to join the CH which has the highest rank.

5.4.4 Distributed Spectrum-Aware Clustering (DSAC) protocol

In [59] Zhang et al. proposed Distributed Spectrum-Aware Clustering (DSAC) protocol, which forms one hop clusters in static cognitive radio sensor networks. DSAC forms a cluster with a goal to reduce energy consumption by minimizing the distance between member nodes and their respective cluster heads and by achieving the optimal number of clusters in the network. Furthermore, since the cluster head depletes its energy quickly compared to normal nodes, the cluster head role is shifted among all nodes in a cluster with equal probability to equalize the energy consumption within a cluster. DSAC forms a cluster under the spectrum-aware constraint which is considered as a groupwise constraint. The main idea of DSAC is to set each node as a separate cluster at the start and then combines two closest clusters, which have at least one common channel. This process is repeated until the cluster number reduces to the optimal number. The optimal cluster number that can minimize the network energy consumption is calculated by the following equation:

$$K_{opt} = \frac{N}{R \times \sqrt{3} \times \rho} + 0.5 \tag{23}$$

where N is the number of nodes in the network, R is the transmission range of CR node and ρ is the node density. Equation (22) is shown that the optimal number of clusters in DSAC is dependent on the number of nodes in the network, node density, and the maximum transmission range.

5.5 Spectrum sense improvement based

Spectrum sensing is the process of sensing the primary user activities in order to detect the idle channel. Many techniques are introduced for an individual CR user to sense the license user activities such as energy detection, matched filter detection and cyclostationary feature detection [77]. Unfortunately, an individual CR user cannot always make accurate decisions, whether the primary users are present or not, due to multipath fading or shadowing. Therefore, cooperative spectrum sensing has been introduced to enhance the sensing outcome. In traditional cooperative spectrum sensing systems, each CR user makes spectrum detection individually, and then sends its sensing outcome to the coherence center through a specific channel named reporting channel. At this coherence center, a decision on, whether the spectrum is idle or not is made. However, this reporting channel may suffer from the channel congestion problem when a large number of nodes access the reporting channel. Moreover, this channel has no immunity against the propagation effects. Beside that if all CR nodes involve in the sensing process of all PUs channels, this means even the CR nodes that are allocated outside the detection range of PU can cooperate in sensing its channel, which in turn, causes a negative effect on the sensing outcome [43]. To overcome all these issues and enhance sensing outcome a cluster-based cooperative spectrum sensing scheme is introduced [40–44].

5.5.1 Cooperative spectrum sensing with cluster-based architecture in cognitive radio networks

In [41] Guo et al. proposed a cluster-based cooperative sensing scheme to reduce the overhead and delay of sensing. In this scheme CR, users are grouped into different clusters based on geographical location. The network consists of a base station (BS), leader and normal nodes. To obtain accurate decisions on sensing outcomes the leader analyzes the received signal power of its members and takes a decision depending on the highest strength among them. Leaders forward their decisions to the BS. BS makes a global decision, whether the spectrum is available or not and broadcasts the decision to the leaders. In this scheme, a fair balance between communication overhead and sensing accuracy is done to get the optimum cluster size.

The clustering technique in this scheme is to group CR users according to their geographical locations so CR users who lay in the same area are associated with the same cluster. It has two stages: cluster head selection and cluster construction.

Stage 1: leader selection: the leaders are chosen by BS in a centralized manner as follows: Firstly, cognitive BS computes the Euclidean distance between it and all nodes. Secondly, BS nominates n CR nodes with minimum distance as potential leaders. It then arbitrarily chooses half of them as leaders. Finally, BS disseminates the selection outcomes to all CR nodes. The dissemination message includes the IDs of selected leaders and the cluster size. The cluster size is predetermined to avoid the congestion in one cluster.

Stage 2: cluster construction. Clusters are built in a decentralized manner as follows:

Every leader sends a beacon to normal nodes, instructing them to choose him as a cluster head. When CR node receives the beacons, it measures the strength of the received signal power (RSP) of each beacon and chooses the leader, which its beacon has, the highest received signal power, as its favorite leader (FL). The rest of cluster heads are considered potential leaders (PL). Then each CR node sends an associate message to its FL consisting of the node ID and RSP. After receiving associate messages from normal nodes, the leaders put nodes in a pool in descending order based on their received signal powers and count the number of nodes. If the number of demanding node is less than or equal the maximum cluster size (let it equal z), the leader approves all demands and sends an ACK to all demanding nodes. Else, the leader sends ACK to the first z nodes in the pool and sends a NACK to the rest of the nodes. If the normal node gets the ACK from its FL, it marks the FL as its leader and associates into this cluster. Otherwise, the node chooses the leader with the highest received signal power from its PL list as FL. This procedure ends when all normal nodes associate into the cluster.

5.5.2 *Cluster-based CSS (cluster-based cooperative spectrum sensing)*

In [43] the authors proposed a cluster-based CSS to improve spectrum utilization by reducing the false alarm and misdetection of spectrum availability. This is achieved by allowing only the CR nodes that are allocated in the detection area of a specific primary user to corporate in the sensing process of its channel. In this scheme, the clusters are built in a way that, CR nodes are associated with the same cluster as long as they locate within the detection range of the same set of PU channels. Therefore, in each cluster, all its members cooperate in sensing the same set of PU channels. Hence, clusters are created in a way that maximizing the total throughput for all clusters while keeping the number of CR nodes in each cluster is equal to or greater than a predefined value. The threshold value of the number of CR nodes in each cluster (cluster size) is determined in a way that keeps the corporative detection probability above a given threshold. Thus, for each channel j the cooperative detection probability $D_{d,j}$

$$D_{d,j} = \left(1 - \prod_{i=1}^{S}\left(1 - P_{d,(i,j)}\right)\right) \geq D_{th} \tag{24}$$

where $P_{d,(i,j)}$ is the detection probability for channel j at CR_i, and D_{th} is the corporative detection probability threshold.

This is done to provide a sufficient protection to the primary user. Thus, the desired minimum cluster size ($\bar{S}$) is calculated as follows:

$$S \geq \frac{log(1 - D_{th})}{log\left(1 - P^{min}_{d,j}\right)} = \bar{S} \tag{25}$$

Considering that the detection probability is monotonically increasing with respect to the SNR for a fixed sensing time, thus

$$P^{min}_{d,j} = min\left\{\left(P_{d,(i,j)}|SNR^{min}_{j}\right) = \min_{1\leq i\leq S}\left\{SNR_{i,j}\right\}\right\} \tag{26}$$

where z is the ceiling function, which returns the smallest integer not less than z, and $P^{min}_{d,j}$ is the minimum detection probability for channel j among all the cooperative CR nodes.

The total throughput for each cluster k is given by

$$R_k(S_k, CH_k) = \sum_{j\in CH_k}\frac{T-\tau}{T}\times\rho_{idle,j}\times C_j \times\left(1 - Q^{k}_{f,j}(S_k, CH_k)\right) \tag{27}$$

where $T$is the length of a time slot, τ is the total sensing time, $\rho_{idle,j}.C_j$ are the idle probability and the capacity for channel j respectively, and $Q^{k}_{f,j}$ is the false alarm probability for channel j when it sensed by cluster k members S_k.

A greedy heuristic algorithm is used to construct the maximum-weight one-sided biclique as follows: Each CR_i node measures the received signal strength for all the PUs. If the received signal strength for primary user j $\left(PU_j\right)$ is above a predefined threshold, CR_i is located inside the detection range of PU_j. Then, each CR node broadcasts its received signal strength information. Then, each CR_i node constructs its own weighted bipartite graph $G_i(X\cup Y, E)$, vertex set X represents the CR nodes in the network, set Y includes the sensed PU channels, and E is the set of existing edges. An edge exists between $(x, y), x\in Xandy\in Y$ if and only if the SU x is within the detection range of the PU channel y. Each edge is associated with weight equal to the throughput of the PU channel. Considering the cluster size threshold and the throughput of PU channels, each CR node uses its own bipartite graph to form its maximum-weight one-sided biclique $Q^{*}_{k}(S_k, CH_k)$, which contains the cluster membership information.

After CR nodes organize into clusters, in each cluster one of CR nodes will be chosen as a cluster head. Thereafter, each CR node individually performs spectrum sensing and sends the decision results to its cluster head. Then, the cluster head combines all decision results and makes a final decision and then inform the channel availability to all its cluster members.

All the clustering formation algorithms addressed in this paper are tabulated in Table 5. In the table, the classified type, the algorithm operating behavior in terms of being distributed or centralized, CCC assignment method, and the cluster distance are listed for each clustering formation algorithm. In addition, the used, interference model and whether the spectrum sense accurate perfect or not are listed next. The next three columns indicate whether the proposed algorithm, consider the challenges of channel heterogeneity in terms of transmission range, load balance, and robustness to PU activity. The last column shows the used techniques.

Table 5 A comparative summary of surveyed clustering algorithms for CRAHNs

References	Type	Approaches						Load balance based	Robustness to PU activity	Technique
		Distributed algorithm	Dedicated Control Channel	Cluster distance	Interference model	Spectrum sensing accurate	Heterogeneous Transmission Ranges			
Ozger et al. [61]	Energy-efficient	Yes	Yes	1-hop	Avoidance	Imperfect	No	No	Low	Greedy heuristic
Zhang, et al. [75]	Stability based clustering	Yes	No	1-hop	Avoidance	Perfect	No	No	High	Greedy heuristic and graph theory
Zareei et al. [52]	Stability based clustering	Yes	Yes	1-hop	Avoidance	Perfect	No	No	High	Graph theory
Saleem et al. [58]	Stability based clustering	Yes	Yes	k-hop	Avoidance	Perfect	No	No	High	Heuristic
Zhang et al. [43]	Cooperative spectrum sensing	Yes	Not mentioned	1-hop	–	–	No	No	Medium	Greedy heuristic and graph theory
Shah et al. [56]	Energy-efficient	Yes	Yes	1-hop	Temperature	Imperfect	No	Yes	Medium	Heuristic
Ozger and Akan [60]	Energy-efficient	Yes	Yes	1-hop	Avoidance	Perfect	No	No	Medium	Heuristic
Liu et al. [45]	Common control channel establishment	Yes	No	1-hop	Avoidance	Imperfect	No	No	High	Greedy heuristic
Li and Gross [48]	Stability based clustering	Yes	Yes	1-hop	Avoidance	Perfect	No	No	High	Game theory
Zhang et al. [59]	Energy-efficient	Yes	No	1-hop	Avoidance	Perfect	No	Yes	Low	Greedy heuristic
Huang et al. [55]	Stability based clustering	Yes	No	k-hop	Avoidance	Perfect	Yes	No	High	Heuristic
Baddour et al. [70]	Dominating-Set	Yes	Not mentioned	1-hop	Avoidance	Perfect	No	No	Low	Affinity propagation
Asterjadhi et al. [74]	Stability based clustering	Yes	No	k-hop	Not mentioned	Imperfect	No	No	High	Heuristic
Guo et al. [41]	Cooperative spectrum sensing	Hybrid	Yes	Not mentioned		Imperfect	–	Yes	–	Heuristic
Chen et al. [5]	Dominating-Set	Yes	No	1-hop	Avoidance	Perfect	No	No	Low	Heuristic
Zhao et al. [46]	Common control channel establishment	Yes	No	k-hop	Avoidance	Perfect	No	No	Medium	Heuristic

6 Open issues

In spite of the previous research which, addressed clustering techniques from different point of views, it is clear that numerous issues still remain open and need to be handled in the future works.

A. Simple assumptions
Almost all clustering algorithms in previous works have been designed using a network model with simple assumptions. However, these assumptions may not be suitable in a practical environment; therefore, they should be reconsidered in future work. This includes:

i. Quasi-static network, such as node mobility, spectrum availability change at a relatively slow rate or even static during the duration of cluster formation. Herein, clusters are formed based on static and the latest information. However, this is not applicable in real scenarios. Since CR network is characterized as an extremely dynamic network. Designing Adaptive clustering algorithms is an open issue that needs to be investigated.

ii. Heterogeneity of channel transmission range
Practically in CR networks the channel homogeneity assumptions are overridden since different channels may be located on vastly separated frequencies with different bandwidths, and different propagation characteristics. Moreover, Federal Communications Commission (FCC) regulations may define different transmission range for different channels. As a result, different channels correspond to different transmission ranges [78].

iii. Fixed common transmission range for CR nodes
Considering a common fixed transmission range are not acceptable for two reasons. First, the common transmission range in for CR nodes will neutralize the unique feature of CR which its capability to interact and adjust with the surrounding environments. Second, the value of the radio transmission range effects in the network connectivity and energy consumption significantly. Using a common large transmission range increases the connectivity of the network by increasing the number of direct links. However, this is at the expense of high-energy consumption. Moreover, CR nodes cannot use any arbitrarily high level of transmission power in all channels, because the transmission power for each spectrum band is restricted identified by FCC regulations to prevent interference to other users. In contrast, using a short transmission range considerably reduces the power consumption as well as the interference but, at the expense of network connectivity. Consequently, a new approach that exploits the capability of CR to adapt its transmission parameters based on the operating environment must be considered in clustering formation.

B. Cluster maintenance
Almost all of existing works focused on the clustering formation. However, as mentioned earlier, clustering consists of two processes, cluster formation, and cluster maintenance. An open issue is the design of accurate clustering maintenance algorithms that address the migration of cluster head, cluster merging, cluster splitting, member node joining and member node leaving.

C. Load balance per cluster
The uniformity of the constructed clusters in terms of the number of nodes per cluster is an important factor. On one hand, an extremely large cluster size may impose a heavy load on the cluster heads, causing a cluster head to become the bottleneck of CRAHNs and reduce network throughput. Moreover, cluster heads with overload members deplete their energy fast result in the entire network might be disconnected. On the other hand, a small cluster size, mean a large number of clusters and thus increase the length of hierarchical routes, causing large control overhead and high end-to-end delay.

D. Performance metric
Most of the previous work evaluated only the clustering algorithms in terms of the number of clusters, cluster size, and number of common channels per cluster. These metric do not reflect the effect of clustering in the overall network performance. Therefore, newer performance metrics that clarify the enhancement of network performance such as throughput and delay should be used for evaluating the clustering algorithms.

E. Design techniques
Most of the reviewed algorithms based on heuristic and greedy heuristic algorithms, which obtain a local optimum solution rather than the global optimum solution so more intelligent techniques such as meta heuristic algorithms and artificial neural network should be used in future work.

F. Analytical evaluation
The performance evaluation of the reviewed clustering algorithms is evaluated using a simulation tool such as C++, Matlab, and NS2. Nevertheless, there is no analytical evaluation of these algorithms; having such

evaluation will help in verifying the simulation results of these algorithms.

G. Security
The CR ad hoc network is a self-organized network without any centralized control. Which means, the nodes in the CR ad hoc network has no central authorities, so they have no immunity against attacks. Despite this fact, all previously proposed cluster formation algorithms assume that the CR nodes are trustworthy. This assumption may lead to the election of a compromised or malicious node to be the cluster head. Having a malicious cluster-head badly jeopardizes the security and usability of the network. Developing of security based clustering algorithms remains an open issue for future research.

7 Conclusion

Clustering in CR networks eases the network functions in various network layers such as allowing CR users to decide the spectrum availability in a collaborative manner, coordinate the spectrum's access and simplify routing table. Nevertheless, the foremost challenge in clustering formation is how to build a stable cluster when it is influenced by license user activity. In this paper, we presented an overview of several clustering formation algorithms in cognitive radio networks. First, we presented fundamental concepts about cognitive radio networks, including the definition of CRN, its architecture, and its main functions. Next, we displayed the essential idea of clustering, including the definition of cluster and clustering, the importance of clustering for a large CRHAN, the basic approaches for modeling clustering formation problem, the performance metric and the techniques that are utilized for clustering. Finally, we categorized the proposed clustering algorithms into five groups according to their main objectives and outlined open issues which research should focus on addressing.

References

1. Federal Communications Commission. (2002, November). *Spectrum policy tasks force report*. ET Docket No. 02-135.
2. Akyildiz, I. F., Lee, W. Y., Vuran, M. C., & Mohanty, S. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks, 50*(13), 2127–2159.
3. Mitola, J., & Maguire, G. Q. (1999). Cognitive radio: Making software radios more personal. *IEEE Personal Communications, 6*(4), 13–18.
4. Akyildiz, I. F., Lee, W. Y., & Chowdhury, K. R. (2009). CRAHNs: Cognitive radio ad hoc networks. *Ad Hoc Networks, 7*(5), 810–836.
5. Chen, T., Zhang, H., Maggio, G. M., & Chlamtac, I. (2007, April). CogMesh: A cluster-based cognitive radio network. In *2007 2nd IEEE international symposium on new frontiers in dynamic spectrum access networks* (pp. 168–178). IEEE.
6. Chinara, S., & Rath, S. K. (2009). A survey on one-hop clustering algorithms in mobile ad hoc networks. *Journal of Network and Systems Management, 17*(1–2), 183–207.
7. Kawadia, V., & Kumar, P. R. (2003, March). Power control and clustering in ad hoc networks. In *INFOCOM 2003. Twenty-second annual joint conference of the IEEE computer and communications. IEEE Societies* (Vol. 1, pp. 459–469). IEEE.
8. Liu, C. H., Rong, B., & Cui, S. (2015). Optimal discrete power control in poisson-clustered ad hoc networks. *IEEE Transactions on Wireless Communications, 14*(1), 138–151.
9. Ephremides, A., Wieselthier, J. E., & Baker, D. J. (1987). A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE, 75*(1), 56–73.
10. Basu, P., Khan, N., & Little, T. D. (2001, April). A mobility based metric for clustering in mobile ad hoc networks. In *2001 international conference on distributed computing systems workshop* (pp. 413–418). IEEE.
11. Konstantopoulos, C., Gavalas, D., & Pantziou, G. (2008). Clustering in mobile ad hoc networks through neighborhood stability-based mobility prediction. *Computer Networks, 52*(9), 1797–1824.
12. Ni, M., Zhong, Z., & Zhao, D. (2011). MPBC: A mobility prediction-based clustering scheme for ad hoc networks. *IEEE Transactions on Vehicular Technology, 60*(9), 4549–4559.
13. Hussein, A. H., Salem, A. O. A., & Yousef, S. (2008, June). A flexible weighted clustering algorithm based on battery power for mobile ad hoc networks. In *2008 IEEE international symposium on industrial electronics* (pp. 2102–2107). IEEE.
14. Leu, J. J. Y., Tsai, M. H., Chiang, T. C., & Huang, Y. M. (2006, September). Adaptive power-aware clustering and multicasting protocol for mobile ad hoc networks. In *International conference on ubiquitous intelligence and computing* (pp. 331–340). Berlin, Heidelberg: Springer.
15. Fathi, A., & Taheri, H. (2010, July). Enhance Topology Control Protocol (ECEC) to conserve energy based clustering in wireless ad hoc networks. In *2010 3rd IEEE international conference on computer science and information technology (ICCSIT)* (Vol. 9, pp. 356–360). IEEE.
16. Karaoglu, B., & Heinzelman, W. (2015). Cooperative load balancing and dynamic channel allocation for cluster-based mobile ad hoc networks. *IEEE Transactions on Mobile Computing, 14*(5), 951–963.
17. Pandey, A., & Lim, J. S. (2010, May). Ctb-mac: Cluster-based tdma broadcast mac protocol for mobile ad-hoc network. In *2010 5th international conference on future information technology* (pp. 1–6). IEEE.
18. Natesapillai, K., Palanisamy, V., & Duraiswamy, K. (2009). Reducing broadcast overhead using clustering based broadcast mechanism in mobile ad hoc network. *Journal of Computer Science, 5*(8), 548–556.
19. Pathak, S., & Jain, S. (2016). A novel weight based clustering algorithm for routing in MANET. *Wireless Networks*, *22*(8), 2695–2704.
20. Kanakala, S., Ananthula, V. R., & Vempaty, P. (2014). Energy-efficient cluster based routing protocol in mobile ad hoc networks using network coding. *Journal of Computer Networks and Communications*, *2014*, 351020. doi:10.1155/2014/351020.
21. Kumar, R., Verma, P., & Singh, Y. (2013). Mobile ad hoc networks and it's routing protocols. *World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 7*(8), 1151–1160.

22. Liu, W., Nishiyama, H., Ansari, N., Yang, J., & Kato, N. (2013). Cluster-based certificate revocation with vindication capability for mobile ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems, 24*(2), 239–249.
23. Jarang, M. R., & Nimbalkar, M. V. (2015, October). Implementation of cluster based certificate revocation in mobile ad hoc networks. In *2015 international conference on green computing and internet of things (ICGCIoT)* (pp. 610–615). IEEE.
24. Hafeez, K. A., Zhao, L., Liao, Z., & Ma, B. N. W. (2012, June). A fuzzy-logic-based cluster head selection algorithm in VANETs. In *2012 IEEE international conference on communications (ICC)* (pp. 203–207). IEEE.
25. Hassanabadi, B., Shea, C., Zhang, L., & Valaee, S. (2014). Clustering in vehicular ad hoc networks using affinity propagation. *Ad Hoc Networks, 13*, 535–548.
26. Chen, Y., Fang, M., Shi, S., Guo, W., & Zheng, X. (2015). Distributed multi-hop clustering algorithm for VANETs based on neighborhood follow. *EURASIP Journal on Wireless Communications and Networking, 2015*(1), 1–12.
27. Omar, H. A., Zhuang, W., & Li, L. (2013). VeMAC: A TDMA-based MAC protocol for reliable broadcast in VANETs. *IEEE Transactions on Mobile Computing, 12*(9), 1724–1736.
28. Bharati, S., & Zhuang, W. (2013). CAH-MAC: Cooperative ADHOC MAC for vehicular networks. *IEEE Journal on Selected Areas in Communications, 31*(9), 470–479.
29. Hadded, M., Muhlethaler, P., Laouiti, A., Zagrouba, R., & Saidane, L. A. (2015). TDMA-based MAC protocols for vehicular ad hoc networks: A survey, qualitative analysis, and open research issues. *IEEE Communications Surveys & Tutorials, 17*(4), 2461–2492.
30. Bhaumik, M., DasGupta, S., & Saha, S. (2012). Affinity based clustering routing protocol for vehicular ad hoc networks. *Procedia Engineering, 38*, 673–679.
31. Mohammed Nasr, M. M., Abdelgader, A. M. S., Wang, Z. G., & Shen, L. F. (2016). VANET clustering based routing protocol suitable for deserts. *Sensors, 16*(4), 478.
32. Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications, 1*(4), 660–670.
33. Lin, C., Wu, G., Xia, F., Li, M., Yao, L., & Pei, Z. (2012). Energy efficient ant colony algorithms for data aggregation in wireless sensor networks. *Journal of Computer and System Sciences, 78*(6), 1686–1702.
34. Younis, O., & Fahmy, S. (2004). HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing, 3*(4), 366–379.
35. Yu, J., Feng, L., Jia, L., Gu, X., & Yu, D. (2014). A local energy consumption prediction-based clustering protocol for wireless sensor networks. *Sensors, 14*(12), 23017–23040.
36. Siavoshi, S., Kavian, Y. S., & Sharif, H. (2016). Load-balanced energy efficient clustering protocol for wireless sensor networks. *IET Wireless Sensor Systems, 6*(3), 67–73.
37. Kurs, A., Karalis, A., Moffatt, R., Joannopoulos, J. D., Fisher, P., & Soljačić, M. (2007). Wireless power transfer via strongly coupled magnetic resonances. *Science, 317*(5834), 83–86.
38. Tan, R., Xing, G., Liu, B., Wang, J., & Jia, X. (2012). Exploiting data fusion to improve the coverage of wireless sensor networks. *IEEE/ACM Transactions on Networking, 20*(2), 450–462.
39. Lin, C., Wu, G., Obaidat, M. S., & Yu, C. W. (2016). Clustering and splitting charging algorithms for large scaled wireless rechargeable sensor networks. *Journal of Systems and Software, 113*, 381–394.
40. Sun, C., Zhang, W., & Letaief, K. B. (2007, June). Cluster-based cooperative spectrum sensing in cognitive radio systems. In *2007 IEEE international conference on communications* (pp. 2511–2515). IEEE.
41. Guo, C., Peng, T., Xu, S., Wang, H., & Wang, W. (2009, April). Cooperative spectrum sensing with cluster-based architecture in cognitive radio networks. In *IEEE 69th Vehicular Technology Conference, 2009. VTC Spring 2009* (pp. 1–5). IEEE.
42. Nguyen-Thanh, N., & Koo, I. (2013). A cluster-based selective cooperative spectrum sensing scheme in cognitive radio. *EURASIP Journal on Wireless Communications and Networking, 2013*(1), 1–9.
43. Zhang, W., Yang, Y., & Yeo, C. K. (2015). Cluster-based cooperative spectrum sensing assignment strategy for heterogeneous cognitive radio network. *IEEE Transactions on Vehicular Technology, 64*(6), 2637–2647.
44. Wang, Y., Lin, W., Huang, Y., & Ni, W. (2014). Optimization of cluster-based cooperative spectrum sensing scheme in cognitive radio networks with soft data fusion. *Wireless Personal Communications, 77*(4), 2871–2888.
45. Liu, S., Lazos, L., & Krunz, M. (2012). Cluster-based control channel allocation in opportunistic cognitive radio networks. *IEEE Transactions on Mobile Computing, 11*(10), 1436–1449.
46. Lazos, L., Liu, S., & Krunz, M. (2009, June). Spectrum opportunity-based control channel assignment in cognitive radio networks. In *2009 6th annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks* (pp. 1–9). IEEE.
47. Zhao, J., Zheng, H., & Yang, G. H. (2005, November). Distributed coordination in dynamic spectrum allocation networks. In *First IEEE international symposium on new frontiers in dynamic spectrum access networks, 2005. DySPAN 2005* (pp. 259–268). IEEE.
48. Li, D., & Gross, J. (2011, June). Robust clustering of ad-hoc cognitive radio networks under opportunistic spectrum access. In *2011 IEEE international conference on communications (ICC)* (pp. 1–6). IEEE.
49. Zhao, J., Zheng, H., & Yang, G. H. (2007). Spectrum sharing through distributed coordination in dynamic spectrum access networks. *Wireless Communications and Mobile Computing, 7*(9), 1061–1075.
50. Li, X., Hu, F., Zhang, H., & Zhang, X. (2013). A cluster-based MAC protocol for cognitive radio ad hoc networks. *Wireless Personal Communications, 69*(2), 937–955.
51. Pritom, M. M. A., Sarker, S., Razzaque, M. A., Hassan, M. M., Hossain, M. A., & Alelaiwi, A. (2015). A multi constrained QoS aware MAC protocol for cluster-based cognitive radio sensor networks. *International Journal of Distributed Sensor Networks, 2015*, 37.
52. Zareei, M., Islam, A. M., & Mansoor, N. (2016). Cross-layer mobility-aware MAC protocol for cognitive radio sensor network. *EURASIP Journal on Wireless Communications and Networking, 2016*, 160.
53. Alsarhan, A., & Agarwal, A. (2009, August). Cluster-based spectrum management using cognitive radios in wireless mesh network. In *Proceedings of 18th international conference on computer communications and networks, 2009. ICCCN 2009* (pp. 1–6). IEEE.
54. Altilar, A. T. D. (2011). United nodes: Cluster-based routing protocol for mobile cognitive radio networks. *IET Communications, 5*(15), 2097–2105.
55. Huang, X. L., Wang, G., Hu, F., & Kumar, S. (2011). Stability-capacity-adaptive routing for high-mobility multihop cognitive radio networks. *IEEE Transactions on Vehicular Technology, 60*(6), 2714–2729.
56. Shah, G. A., Alagoz, F., Fadel, E. A., & Akan, O. B. (2014). A spectrum-aware clustering for efficient multimedia routing in

cognitive radio sensor networks. *IEEE Transactions on Vehicular Technology, 63*(7), 3369–3380.

57. Zubair, S., & Fisal, N. (2014). Reliable geographical forwarding in cognitive radio sensor networks using virtual clusters. *Sensors, 14*(5), 8996–9026.
58. Saleem, Y., Yau, K. L. A., Mohamad, H., Ramli, N., & Rehmani, M. H. (2015). SMART: A SpectruM-Aware ClusteR-based rouTing scheme for distributed cognitive radio networks. *Computer Networks, 91*, 196–224.
59. Zhang, H., Zhang, Z., Dai, H., Yin, R., & Chen, X. (2011, December). Distributed spectrum-aware clustering in cognitive radio sensor networks. In *2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)* (pp. 1–6). IEEE.
60. Ozger, M., & Akan, O. B. (2013, April). Event-driven spectrum-aware clustering in cognitive radio sensor networks. In *2013 Proceedings IEEE INFOCOM* (pp. 1483–1491). IEEE.
61. Ozger, M., Fadel, E., & Akan, O. (2016). Event-to-sink spectrum-aware clustering in mobile cognitive radio sensor networks. *IEEE Transactions on Mobile Computing, 15*(9), 2221–2233.
62. Pei, E., Han, H., Sun, Z., Shen, B., & Zhang, T. (2015). LEAUCH: Low-energy adaptive uneven clustering hierarchy for cognitive radio sensor network. *EURASIP Journal on Wireless Communications and Networking, 2015*(1), 1–8.
63. Mitola, J. (1995). The software radio architecture. *IEEE Communications Magazine, 33*(5), 26–38.
64. FCC. (2003, December). *Notice of proposed rulemaking and order*. ET Docket No. 03-222.
65. Lo, B. F. (2011). A survey of common control channel design in cognitive radio networks. *Physical Communication, 4*(1), 26–39.
66. Haykin, S. (2005). Cognitive radio: Brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications, 23*(2), 201–220.
67. Chen, K.-C., & Prasad, R. (2009). *Cognitive radio networks, in cognitive radio networks*. Chichester: Wiley.
68. Yu, J. Y., & Chong, P. H. J. (2005). A survey of clustering schemes for mobile ad hoc networks. *IEEE Communications Surveys & Tutorials, 7*(1), 32–48.
69. Kozat, U. C., Kondylis, G., Ryu, B., & Marina, M. K. (2001, June). Virtual dynamic backbone for mobile ad hoc networks. In *IEEE international conference on communications, 2001. ICC 2001* (Vol. 1, pp. 250–255). IEEE.
70. Baddour, K. E., Ureten, O., & Willink, T. J. (2009, August). Efficient clustering of cognitive radio networks using affinity propagation. In *Proceedings of 18th international conference on computer communications and networks, 2009. ICCCN 2009* (pp. 1-6). IEEE.
71. Mansoor, N., Islam, A. M., Zareei, M., Baharun, S., & Komaki, S. (2013, December). Spectrum aware cluster-based architecture for cognitive radio ad-hoc networks. In *2013 international conference on advances in electrical engineering (ICAEE)* (pp. 181–185). IEEE.
72. Bradonjić, M., & Lazos, L. (2012). Graph-based criteria for spectrum-aware clustering in cognitive radio networks. *Ad Hoc Networks, 10*(1), 75–94.
73. ul Hasan, N., Ejaz, W., Manzoor, K., & Kim, H. S. (2013). GSM: Gateway selection mechanism for strengthening inter-cluster coordination in cognitive radio ad hoc networks. *EURASIP Journal on Wireless Communications and Networking, 2013*(1), 1.
74. Asterjadhi, A., Baldo, N., & Zorzi, M. (2010, April). A cluster formation protocol for cognitive radio ad hoc networks. In *2010 European wireless conference (EW)* (pp. 955–961). IEEE.
75. Zhang, H., Xu, N., Xu, F., & Wang, Z. (2016). Graph cut based clustering for cognitive radio ad hoc networks without common control channels. *Wireless Networks*, 1–13. doi:10.1007/s11276-016-1329-5.
76. Xu, G., Tan, X., Wei, S., Guo, S., & Wang, B. (2010, September). An energy-driven adaptive cluster-head rotation algorithm for cognitive radio network. In *2010 first international conference on pervasive computing signal processing and applications (PCSPA)* (pp. 138–141). IEEE.
77. Cabric, D., Mishra, S. M., & Brodersen, R. W. (2004, November). Implementation issues in spectrum sensing for cognitive radios. In *2004 conference record of the thirty-eighth Asilomar conference on Signals, systems and computers* (Vol. 1, pp. 772–776). IEEE.
78. Ma, M., & Tsang, D. H. (2009). Joint design of spectrum sharing and routing with channel heterogeneity in cognitive radio networks. *Physical Communication, 2*(1), 127–137.

Mahassin Mohamed Ahmed Osman received her M.Sc. degree in telecommunication and information systems form University of Khartoum, Sudan in 2002. Currently she is a Ph.D. candidate with the Faculty of Electrical Engineering, University Technologi Malaysia (UTM), Malaysia. Her research interests include cognitive radio ad hoc networks and wireless sensor networks.

Sharifah Kamilah Syed-Yusof received B.Sc. degree in electrical engineering from George Washington University, USA in 1988 and obtained her M.E.E. and Ph.D. degrees from UTM in 1994 and 2006, respectively. She is currently working as associate professor with Faculty of Electrical Engineering, UTM. Her research interest includes wireless communication, software defined radio and cognitive radio. She is a member of Eta Kappa Nu (HKN), and Phi Beta Kappa society.

Nik Noordini Nik Abd Malik received her bachelor degree in electrical engineering from UTM, her M.Eng. degree in radio frequency and microwave communication engineering form University of Queensland (UQ), Australia in 2005, and her Ph.D. degree in electrical engineering from UTM in 2013. Her research interests include collaborative beamforming, wireless sensor networks, bio-inspired optimization algorithms and array antennas.

Suleiman Zubair received his degree of Bachelor of Engineering in Electrical and Computer Engineering in 2004 and his degree of Master of Engineering in Communication Engineering from the department of Telecommunication Engineering Federal University of Technology Minna, Nigeria where he also lectures. He received his Ph.D. in 2015 from Faculty of Electrical Engineering, Universiti Teknologi Malaysia (UTM) Malaysia. He recently completed Post-Doctoral research with Universiti Teknologi Malaysia (UTM) Malaysia under the Post-Doctoral Fellowship Scheme for the Project: Development of Reliable Opportunistic Mobile Routing (ROMR) Protocol for Cognitive Radio Sensor Networks and a member of the Telematic Research Group. His research interests include wireless sensor networks, cognitive radio networks, cognitive networks, and next generation networks.