Journal of Network and Computer Applications 46 (2014) 198–226

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Clustering in sensor networks: A literature survey

M. Mehdi Afsar [a,*], Mohammad-H. Tayarani-N [b]

[a] Young Researchers and Elite Club, Mashhad Branch, Islamic Azad University, Mashhad, Iran
[b] School of Computer Science, University of Birmingham, Birmingham, UK

ARTICLE INFO

Article history:
Received 13 March 2014
Received in revised form
13 August 2014
Accepted 16 September 2014
Available online 27 September 2014

Keywords:
Wireless sensor networks
Hierarchical architecture
Clustering
Cluster-head
Energy-efficiency

ABSTRACT

Wireless sensor networks (WSNs) have recently gained the attention of researchers in many challenging aspects. The most important challenge in these networks is energy conservation. One of the most popular solutions in making WSNs energy-efficient is to cluster the networks. In clustering, the nodes are divided into some clusters and then some nodes, called cluster-heads, are selected to be the head of each cluster. In a typical clustered WSN, the regular nodes sense the field and send their data to the cluster-head, then, after gathering and aggregating the data, the cluster-head transmits them to the base station. Clustering the nodes in WSNs has many benefits, including scalability, energy-efficiency, and reducing routing delay. In this paper we present a state-of-the-art and comprehensive survey on clustering approaches. We first begin with the objectives of clustering, clustering characteristics, and then present a classification on the clustering algorithms in WSNs. Some of the clustering objectives considered in this paper include scalability, fault-tolerance, data aggregation/fusion, increased connectivity, load balancing, and collision avoidance. Then, we survey the proposed approaches in the past few years in a classified manner and compare them based on different metrics such as mobility, cluster count, cluster size, and algorithm complexity.

1. Introduction

Micro sensors are low energy devices with small memory and low processing power. These cheap and small-sized devices have become possible with the recent advances in Complementary Metal Oxide Semiconductor (CMOS) technology and miniaturization techniques. Each sensor node usually contains a small CPU, memory, receiver/transmitter radio and a power supply unit. These tiny devices have the capability to form a network with a large number of nodes in a self-configured scheme and without a particular infrastructure. Wireless sensor networks (WSNs) are composed of a huge number of sensor nodes. There are many applications for WSNs and depending on the application, different types of sensors are used, such as sensors measuring moisture, temperature, pressure and movement. WSNs have themselves characteristics that make them different from other types of networks. One for example is that the applicability of the networks is related to energy supply of the nodes, so energy conservation is one of the most important challenges in these networks. In general, the applications of WSNs can be divided into two general groups: tracking and monitoring (Akyildiz et al., 2002; Yick et al., 2008).

WSNs are usually dispersed in harsh environments with limited access to human beings, area like battlefields, forests, and special industrial and clinical fields. Therefore, it is essential that WSNs operate in a self-configured and autonomous mode with the capabilities to form a network in an ad hoc scheme. As WSNs are energy constrained and data transmission is the most energy consumer (Anastasi et al., 2009), there is a need to an architecture in which the transmission to a Base Station (BS) is as low as possible, and all the decisions are made in the node level.

As the size of the network and the number of nodes grow, the scalability of the network proves to be important as it defines whether the network is able to be implemented in the real world or not. In this respect, hierarchical architecture is a suitable approach to increase the scalability of a network efficiently. In the hierarchical architecture, the entire network is divided into some virtual layers and the nodes that are located in the same layer have the same role. One of the first attempts in the area of hierarchical architectures in large networks is Kleinrock and Kamoun (1977) which shows that the hierarchical architecture can significantly reduce the routing table of each node so that the scalability of the network could increase. Clustering the nodes is a popular two-layered method that divides the network into two layers, and the nodes located in the same layer are grouped into some clusters. To efficiently distribute the management tasks among the nodes, some of them are elected to be the head of each group (cluster), which are usually called the cluster-heads (CHs). Since a large amount of data provided by the sensors in

* Corresponding author. Tel.: +98 915 1069234.
E-mail addresses: m.afsar@qiau.ac.ir (M.M. Afsar), tayarani@ieee.org (M.-H. Tayarani-N).

http://dx.doi.org/10.1016/j.jnca.2014.09.005
1084-8045/

WSNs are similar, the clustering utilizes the correlation among the data, and then by aggregating them, reduces the load on the network, which results in a more efficient energy consumption. The CHs are then responsible for gathering the data from regular nodes, and aggregating and transmitting them to the BS. A typical application of a clustered WSN is presented in Fig. 1. As shown in this figure, the nodes in layer 1 (the regular nodes) sense the field, generate the data, and send them to their associated CH. Then the CHs in the second layer receive these data, and after performing some processes like aggregation/fusion, transmit them to the BS in a multi-hop approach. Eventually the user receives the data from the BS through the Internet.

In the last decade, many clustering approaches have been proposed for WSNs in which the energy conservation is the common objective. The operations in clustering protocols are usually divided into three phases: CH selection, cluster formation, and data transmission. The main part of each approach (protocol) is the CH selection algorithm that defines the energy efficiency of the network. In addition to the energy efficiency, some other objectives might be targeted as main objective of the clustering approach, such as full coverage of the field (Tian and Georganas, 2002; Soro and Heinzelman, 2009) and fault-tolerance (Tai et al., 2004).

In this paper, we present a comprehensive and state-of-the-art survey on the clustering approaches in WSNs. Although some reviews have been presented in the literature, none of them has performed a coverage over all the related approaches or has proposed a proper classification on the existing approaches. Lack of a comprehensive work in this area motivated us to investigate this study. In this paper, we first aim to discuss the clustering objectives and classify the clustering characteristics in WSNs. Then we review the most important proposed approaches in details and discuss the advantages and the drawbacks of each approach, and the way their drawbacks encouraged the researchers to improve them. We also present a general taxonomy of the proposed approaches. Reviewing the extensions and the similar works with the minor focus is also in our agenda. Next, for the sake of completeness and comparison we present some tables, comparing the reviewed approaches.

The remainder of this paper is organized as follows. Related works are discussed in Section 2. The preliminaries about the clustering approaches are presented in Section 3 which explains the clustering objectives, characteristics, and classification. Clustering algorithms are surveyed in Section 4. A detailed comparison of the presented approaches is brought in Section 5 and the paper is concluded in Section 6.

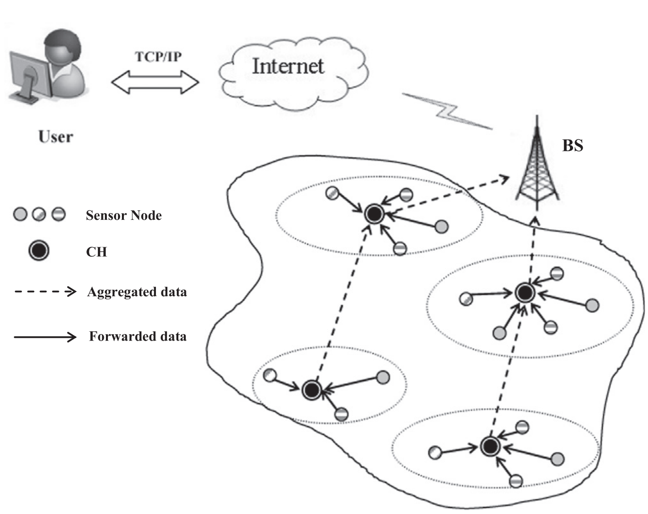

Fig. 1. A typical clustered sensor network.

2. Related work

In this section we review some survey papers in the area of clustering in WSNs. We first make a review on more complete surveys (Abbasi and Younis, 2007; Mamalis et al., 2009; Liu, 2012), and then, move to reviewing more briefly presented surveys (Ramesh and Somasundaram, 2011; Younis et al., 2006; Arboleda and Nasser, 2006; Jiang et al., 2009; Xu and Gao, 2011; Maimour et al., 2010; Joshi and Lakshmi Priya, 2011; Kumarawadu et al., 2008; Deosarkar et al., 2008; Lotf et al., 2010; Wei et al., 2011a; Aslam et al., 2012; Kumar et al., 2013; Subhai et al., 2013; Jindal and Gupta, 2013). These surveys are usually limited in scope, incomplete, or are outdated. We use 'limited in scope' for those surveys that have not covered all types of clustering, e.g. fuzzy-logic-based, heuristic-based, etc. Also, 'incomplete' indicates that the works have not surveyed all related papers or at least most valuables in their categories. Finally, we refer to a paper as 'outdated' if it is old and has not covered recent clustering methods. In particular, we consider published papers before 2011 as outdated.

One of the most important surveys on clustering algorithms has been presented in Abbasi and Younis (2007). In the work, the authors describe some important clustering approaches in WSNs and wireless networks. The paper classifies all algorithms into two major groups based on their convergence time: variable and constant convergence time. Finally in the research, the reviewed clustering algorithms are compared with different metrics. Many clustering approaches are missed in the work. More importantly, the unequal, fuzzy-logic-based and heuristic-based clustering methods have not been covered.

Another survey is presented in Mamalis et al. (2009) which tries to describe more clustering approaches by classifying them as being either probabilistic or non-probabilistic. In the probabilistic approaches, the authors mention that the most important clustering schemes are LEACH, HEED, and EEHC (Bandyopadhyay and Coyle, 2003), and other protocols are inspired from these. In non-probabilistic schemes, the works are divided into node proximity and graph-based clustering protocols, weight-based clustering protocols, and biologically inspired clustering approaches. And finally, some algorithms of reactive networks are discussed. In this work also the unequal and fuzzy-logic-based clustering approaches are missed.

A survey on clustering routing protocols is presented in Liu (2012) in which the clustering algorithms are reviewed and divided into cluster-construction routing or data-transmission routing methods. The paper focus only on 16 well-known clustering approaches, and makes no review on fuzzy-based, evolutionary-based and recently proposed approaches in this area and targets only the most important and old-presented protocols.

In addition to these reviews, there are some short surveys and minor works in this area that we review briefly in the followings. A research study on clustering approaches and their challenges is presented in Younis et al. (2006). The paper classifies the clustering approaches based on the parameters of the CH election and the execution nature of a clustering algorithm (probabilistic or iterative). Based on these criteria, some routing clustering protocols, such as LEACH (Heinzelman et al., 2000), HEED (Younis and Fahmy, 2004a), DCA (Basagni, 1999), GAF (Xu et al., 2001) and SPAN (Chen et al., 2002) are reviewed. At the end, the surveyed protocols are compared to one another with some metrics, including cluster criteria, clustering objective, and complexity. The paper misses many existing clustering approaches and reviews a small number of related works.

Another survey on clustering approaches is presented in Ramesh and Somasundaram (2011) where some important clustering approaches are reviewed, a brief report of different protocols is provided and the proposals are compared to one another.

A short survey on clustering algorithms is presented in Arboleda and Nasser (2006) which formulates the clustering approaches with three main phases: CH election, cluster formation and data transmission. The work presents a brief overview on important clustering approaches. The authors discuss the clustering approaches with four main topics: former, LEACH-based, proactive-based and reactive-based protocols.

Another attempt in this area (Jiang et al., 2009) is presented in which some popular clustering approaches are reviewed. The authors mentioned that it is hard to set a common criterion for various clustering schemes, however, the work lists some attributes as classification criteria, such as existence, count variability, and selectivity. And finally, the reviewed clustering schemes are compared.

A comparison study on hierarchical routing protocols is performed in Xu and Gao (2011) where some important hierarchical routing protocols, like LEACH, TEEN (Manjeshwar and Agrawal, 2001) and APTEEN (Manjeshwar and Agrawal, 2002), PEGASIS (Lindsey and Raghavendra, 2002), etc. are examined. Also the works are compared based on some attributes, such as proactive, data aggregation, network lifetime, and scalability.

The reference Maimour et al. (2010) is a survey on cluster-based routing protocols in which some cluster-based routing protocols are reviewed and the clustering approaches are classified into pre-established and on-demand cluster-based routing algorithms. The pre-established clustering protocols reviewed in the paper include LEACH, HEED, and EEHC, and the on-demand clustering protocols include PC (Kwon and Gerla, 2002) and CLIQUE (Forster and Murphy, 2009).

A brief survey on hierarchical routing protocols is presented in Joshi and Lakshmi Priya (2011) which surveys only eight important approaches including LEACH, HEED, TEEN, APTEEN, and EECS (Ye et al., 2005). And at the end the reviewed works are compared based on the network lifetime as a comparison criterion.

Kumarawadu et al. (2008) classify the clustering approaches into four major groups: identity-based clustering, neighborhood Information-based clustering algorithms, probabilistic clustering algorithms, and biologically inspired clustering algorithms. Some criteria used for comparing the approaches are energy-efficiency, load balancing, clock synchronization, etc.

A survey on CH election in clustering algorithms is performed in Deosarkar et al. (2008) where all the related approaches are grouped into four categories: deterministic, adaptive, combined metric, and hybrid clustering schemes. Subsequently, adaptive schemes are classified into fixed parameter probabilistic schemes and resource adaptive probabilistic schemes. The authors compare the works according to the classified groups separately and discuss some challenging issues in the clustering approaches.

Another brief survey is presented in Lotf et al. (2010) which reviews some LEACH-like clustering approaches, including LEACH, TEEN, PEGASIS, and CCS (Jung et al., 2007). The authors compare each of the improvement approach on LEACH with LEACH separately and finish the paper with some discussions on the improvements and their results in the network lifetime.

In another survey (Wei et al., 2011a), clustering schemes are categorized into deterministic and adaptive approaches. The authors discuss and overview the most popular clustering schemes based on some challenges, such as rotating the CH role, optimal cluster size, and optimum mode of communication between sensor nodes and CHs. The paper performs no comparison among the reviewed schemes.

A survey on extended LEACH-based clustering protocols is presented in Aslam et al. (2012) which discusses only four protocols: LEACH, sLEACH (Voigt et al., 2004), Multi-hop LEACH (Biradar et al., 2011), and Mobile-LEACH (M-LEACH). The work first describes LEACH and the extended protocols, then performing some simulations, the performance of the algorithms is compared to one another.

Fuzzy-logic-based clustering protocols are surveyed in Kumar et al. (2013). The work is a short survey on clustering protocols that use fuzzy-logic for CH election. Some related works in this area, including HERF (Arabi and Khodaei, 2010; Gupta and Sampalli, 2005; Lee and Cheng, 2012a; Seyyed Jalaleddin Dastgheib and Oulia, 2011), are reviewed in the paper.

A survey on neural network based clustering approaches is presented in Subhai et al. (2013) which focuses on five neural network based algorithms: ART, ART1, FUZZY ART, IVEBF, and EBCS. After a short introduction to related algorithms, the paper discuses the specifications of each algorithm.

In Jindal and Gupta (2013) LEACH and its recent advances are discussed. First the LEACH protocol is described, then some of its recent improvements, like LEACH-CC (Cui, 2007; Yueyun et al., 2012), are discussed. The paper is a short survey and does not review a wide range of contributions in this area.

After studying existing surveys in this area, we realized that there is no comprehensive paper that surveys all the clustering approaches in a classified manner. Each of the related work studied in this paper has some shortcomings in different aspects. This motivated us to perform this survey paper that targets most

Table 1
List of the previous surveys on clustering algorithms (O: outdated, I: incomplete, L: limited in scope).

Ref.	Year	Classification	Weakness
Arboleda and Nasser (2006)	2006	Former, LEACH-based, Proactive, Reactive	O, I, L
Younis et al. (2006)	2006	Probabilistic, Iterative	O, I, L
Abbasi and Younis (2007)	2007	Convergence time	O, L
Kumarawadu et al. (2008)	2008	Identity-based, neighbor-based, probabilistic, biologically inspired	O, I
Deosarkar et al. (2008)	2008	Deterministic, adaptive, combined, hybrid	O, I
Mamalis et al. (2009)	2009	Probabilistic, non-probabilistic	O, I
Jiang et al. (2009)	2009	N/A	O, I, L
Maimour et al. (2010)	2010	Pre-established, on-demand	O, I, L
Lotf et al. (2010)	2010	LEACH-based	O, I, L
Ramesh and Somasundaram (2011)	2011	Deterministic, BS-assisted adaptive, fixed parameter probabilistic, resource-adaptive probabilistic, hybrid	I, L
Xu and Gao (2011)	2011	LEACH-based	I, L
Joshi and Lakshmi Priya (2011)	2011	LEACH-based	I, L
Wei et al. (2011a)	2011	Deterministic, adaptive	I, L
Liu (2012)	2012	Cluster-construction routing, data-transmission routing	I, L
Aslam et al. (2012)	2012	LEACH-based	I, L
Kumar et al. (2013)	2013	Fuzzy-based	I, L
Subhai et al. (2013)	2013	Neural-based	I, L
Jindal and Gupta (2013)	2013	LEACH-based	I, L

of the existing clustering approaches. Table 1 lists the previous surveys on clustering methods in WSNs.

3. Preliminaries

In this section, we introduce some preliminaries of the clustering approaches. First we discuss the clustering objectives and present some characteristics of clustering, and then the clustering approaches are classified.

3.1. Clustering objectives

Clustering the nodes in WSNs is performed with different objectives and purposes. The energy conservation is the most important and common objective of all these objectives. We divide these objectives as primary and secondary. The primary objectives indicate the objectives that are the most important and substantial in the clustering process. On the other hand, the secondary objectives indicate the objectives that are not substantially important for the network and they are indirectly achieved by clustering the nodes. Fig 2 provides an overview on some most common clustering objectives and our classification of them. In the following, we list and briefly explain some of these objectives of clustering in WSNs.

1. *Scalability*: The number of sensor nodes scattered in a field, depending on the applications, can be order of hundreds, thousands or even millions (Akyildiz et al., 2002). As mentioned before, hierarchical architectures can well increase the scalability of large networks by dividing the network into some virtual layers, and then each layer into some clusters (Kleinrock and Kamoun, 1977). When a node located in a cluster decides to communicate with another node located in another cluster, the node should know some information about the associated CH of the cluster in which the other node is located. This results in an increase in the scalability of the network and significant reduction in the routing table sizes.
2. *Fault-tolerance*: WSNs are usually dispersed in harsh environments with limited access to human beings so that the fault-tolerance and self-configured characteristics are crucial for such networks. In general, the failure of some nodes should not affect the overall task of a WSN (Akyildiz et al., 2002). As mentioned in Zhou et al. (2008), clustering the nodes is an effective approach to make such networks more secure and fault-tolerant. Some examples are Zhu et al. (2006) and Tubaishat et al. (2004) which use the clustering architecture with the mentioned objective. Adaptive clustering handles fault in the CHs by re-clustering at the beginning of some predetermined periods (*rounds*, Heinzelman et al., 2000). Since the consecutive re-clustering process is costly, making backup CH, or deputy-CH, is a proper approach in fault management techniques of clustered WSNs to avoid re-clustering.
3. *Data aggregation/fusion*: Since a huge amount of data in WSNs are the same, the data aggregation/fusion is an effective approach to avoid transmitting repetitive data in the network. Data aggregation techniques are usually based on signal processing methods and a common data aggregation technique in WSNs is to aggregate all the received packets into one output packet (Krishnamachari et al., 2002). In flat architectures, all the nodes have to transmit their data to the BS by either multi-hop or direct approach, although, some data aggregation techniques are practical only for flat architectures when as they use data-centric application (Rajagopalan and Varshney, 2006). On the other hand, clustering allows the data being aggregated in the CHs which results in a more energy-efficient network, by reducing the total load of the network.
4. *Load balancing*: Clustering should be considered to ensure low-energy data processing and intra-node communication (Pantazis and Vergados, 2007); however, the CHs in the clustering architecture are responsible for long range communications, data gathering, aggregating and transmitting/forwarding, etc. This results in a faster energy depletion of the CHs. Thus, it is better to rotate this role among all the nodes in the network. Utilizing the load balancing results in a more energy-efficient network.
5. *Stabilized network topology*: Since the nodes are divided into some clusters, the changes in the location of the nodes can be managed by the CHs in the cluster level. Thus, managing the changes in topology of the network is more convenient than in flat architecture which is composed of a huge number of mobile nodes. In clustered WSN, each CH has information about its members, such as their energy and location; so if a node dies or moves to other clusters (in a mobile network), these changes are immediately registered and reported by the CHs.
6. *Maximal network lifetime*: As mentioned earlier, the main challenge in WSNs is to extend the network lifetime as much as possible. A clustering method that satisfies these objectives can extend the network lifetime. For example, if the CHs are located at the center of the node population, the CH role is properly rotated among the nodes, and the sleeping schemes are effectively utilized, then we can expect a good lifetime for the network.
7. *Increased connectivity*: This objective can be either as simple as ensuring the existing of a path from every CH to the BS (Bandyopadhyay and Coyle, 2003), or be more restrictive by imposing a bound on the length of the path (Dai and Wu, 2005). In the worst case scenario, a k-connected network requires k node failures to disconnect the network (Bredin et al., 2010; Hajiaghayi et al., 2003). Clustering the nodes improves the performance, specially in large-scale WSNs. This is because comparing to the flat architecture when the nodes are clustered, to achieve the connectivity, there has to be at least one path from every CH to the BS, not between every node in the network and the BS.
8. *Reduced routing delay*: In some applications of WSNs, specially in wireless multimedia sensor networks (WMSNs), WSN should ensure a stringent deadline (Ehsan and Hamdaoui, 2012), such as a physical event and health-care monitoring. Therefore, assuring reduced routing delay is one of the most important challenges in meeting the QoS requirements of the network (Akyildiz et al., 2007). In WSNs, which usually use the multi-hop approach for transmitting the data, clustering, compared to flat architectures, efficiently reduces the routing delay. This is

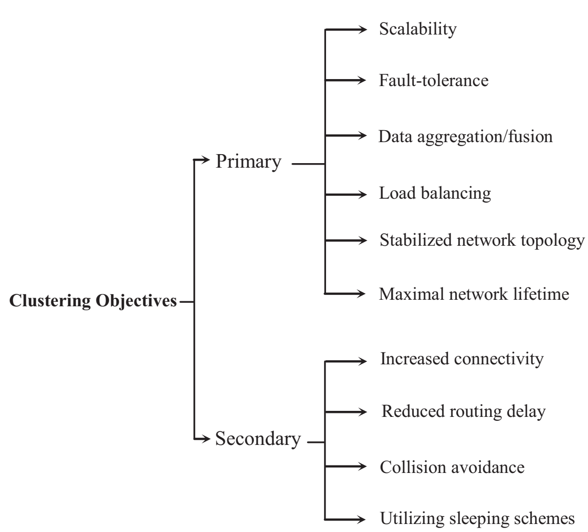

Fig. 2. Clustering objectives in a general view.

because, in the routing problem, if we have a network with n nodes and k clusters ($n > k$) in a clustered WSN, it would be more efficient if the data are routed among k CHs rather than n nodes, while in a flat WSN the data are routed among n nodes. Note that this is more considerable when $n \gg k$.

9. *Collision avoidance*: Collision avoidance in WSNs is important, because each collision causes some packets to be lost, so each node has to re-transmit the latest packets. It is more significant when the nodes are mobile and the topology is dynamic (Dong and Dargie, 2013). This increases the energy consumption in the network and is not suitable for energy-constrained networks. When the nodes are clustered, utilizing some MAC layer protocols, like Time Division Multiple Access (TDMA), makes in-cluster operations performed without collisions.
10. *Utilizing sleeping schemes*: In some applications of WSNs, there is no need for the nodes in the network to be awake in all the operational time. If all the nodes are awake for all the time, the energy of the nodes is depleted fast and the network lifetime is significantly reduced. In some other applications, the protocol selects some active nodes that cover the entire field so the other nodes located in the range of active nodes can go to sleep (Ye et al., 2003). In this scheme selecting the active nodes has some overheads such as exchanging the control messages. In a clustered WSN, it is possible that some nodes like the CHs are awake and other regular nodes go to sleep in a scheduled manner. The most popular scheme is to use the TDMA protocol, in which the CHs first send the time schedule to the regular nodes and then the regular nodes go to sleep unless they are in their time schedule. Therefore the consumed energy is significantly reduced. Note that in sleep mode the nodes switch off their radio and change their status to a low power mode. The consumed energy in sleep mode is on the order of μW, while in the idle mode it is of mW.

3.2. Clustering characteristics

In this section, we opt some clustering characteristics in order to classify different clustering approaches. In general, we define each clustered WSN to have three main characteristics: cluster properties, CH properties, and clustering process properties. In the following, we clarify each main section in detail. A proper taxonomy of clustering characteristics can be found in Fig. 3.

3.2.1. Cluster properties

Cluster properties are divided based on the specifications of clusters, like the number of clusters, cluster size, intra-cluster communication, and inter-cluster communication. A brief explanation of each is presented below:

- *The number of clusters*: The number of formed clusters can be either constant (preset) or variable. In the approaches that

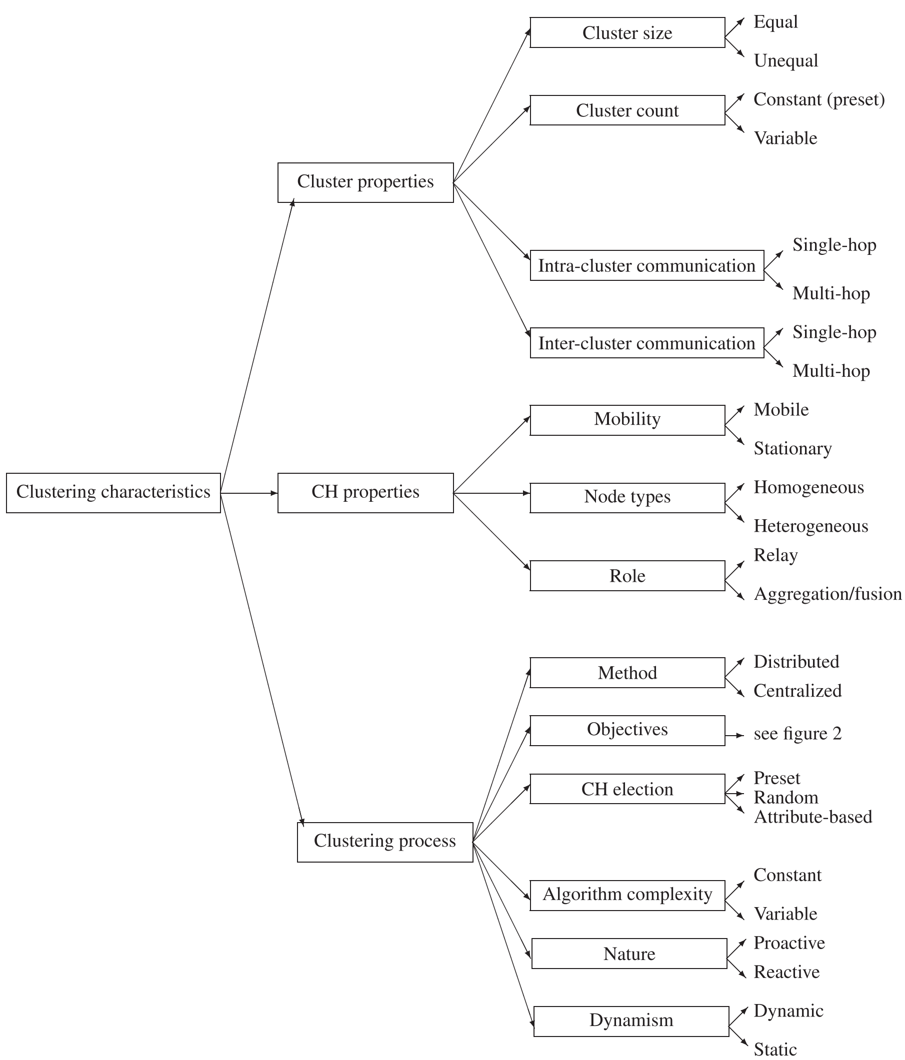

Fig. 3. The clustering characteristics in WSNs.

randomly elect the CHs among the sensor nodes, this number is variable.

- *Cluster size:* Based on the uniform distribution of the load among all the formed clusters, the size of the clusters can be equal or unequal. This inequality of clusters is based on the distance between the nodes and the BS.
- *Intra-cluster communication*: Based on the clustering algorithm, the communications within a cluster can be either direct or mullti-hop. In some clustering approaches when the number of CHs is small and the size of the clusters is large, multi-hop communication between the CH and the members may be needed.
- *Inter-cluster communication*: Since the sensor nodes are equipped with short range receiver/transmitters, so the multi-hop approach is an appropriate mechanism for WSNs. However, some applications of WSNs assume that the communication between the CHs and the BS is direct (usually in small scale networks and traditional approaches).

3.2.2. CH properties

Since the main part of each clustering algorithm is the CH election, the elected CHs have a substantial effect on the clustering algorithm performance. We list some characteristics of the CHs below:

- *Mobility*: The CHs can be either stationary or mobile. The mobile CHs can move for a limited distance, although the topology management process of mobile CHs is more difficult than in a network with stationary CHs.
- *Node type*: The dispersed CHs across the network can be rich in resources compared to the regular nodes; that is, the network supports the heterogeneity of the nodes. Or, the network can be homogeneous and the CHs are picked from the regular nodes.
- *Role*: Based on the algorithm, the elected CHs can perform different roles in the network. These roles are relay and aggregation/fusion. A CH can act as a simple relay node, such as controlling the data traffic and perform synchronization or can perform the important operations of data aggregation/fusion.

3.2.3. Clustering process

In this section we review some characteristics of a clustering algorithm.

- *Method*: A clustering algorithm can be either distributed or centralized. Since WSNs are networks with a huge number of nodes, distributed approaches have gained more popularity than the centralized approaches have.
- *Objectives*: Clustering the nodes in WSNs can have different objectives. As mentioned earlier, some objectives are more important than the others. A complete list of the clustering objectives considered in this paper are shown in Fig. 2.
- *CH selection*: Each clustering algorithm has its own CH election mechanism. However, in general, the CH election algorithms can be classified into three categories: preset, random, and attribute-based methods. In preset, the CHs are elected before the deployment of the nodes in the field. In random approaches, the CHs are selected randomly, and attribute-based algorithms select them based on some of their characteristics, like the residual energy and distance to the BS.
- *Algorithm complexity*: The algorithm complexity means the way an algorithm converges. Some algorithms converge in a variable time, depending on the network specifications like the number of CHs, and some converge in a constant time, regardless of the network specifications.
- *Clustering nature*: Many clustering algorithms have been proposed for WSNs in the literature. A small number of these approaches, known as reactive networks, are based on data-centric method. Many of the proposed approaches are proactive and do not support the reactivity and some use a hybrid of them.
- *Clustering dynamism*: A clustering approach can be either dynamic or static. In dynamic approaches, the CHs are elected based on the current conditions of the network, and most of dynamic approaches act in a real time scheme. In static approaches, the CH election and related operations are performed regardless of the current network conditions.

3.3. Clustering algorithms classification

In this section we perform a classification on the existing clustering algorithms. To do so, different metrics are used. A possible classification of clustering algorithms is to divide all approaches into distributed or centralized methods. Since centralized design is not scalable and consequently suitable for WSNs, most of existing popular approaches are distributed so this categorization is very general. Another common way is to classify the approaches based on the CH selection algorithm, as shown in Table 1. In our classification, we consider CH selection algorithm and clustering properties. We first divide the clustering approaches into equal-sized and unequal-sized clustering approaches. Then we focus on equal-sized clustering approaches and divide them into three groups: probabilistic, deterministic, and preset methods. In probabilistic approaches, we analyze the approaches based on two major categories: random and hybrid. Subsequently, we divide the deterministic approaches into four major groups: weight-based, fuzzy-logic-based, heuristic-based, and compound approaches. We also stay loyal to these categories in the case of unequal-sized clustering algorithms. Note that we aim at presenting the related works in each category in the order of publications year. Figure 4 shows our classification on the presented clustering approaches.

4. Clustering algorithms

In this section, we present a state-of-the-art and comprehensive survey on the clustering algorithms. The reviewed clustering approaches are classified into two major groups: equal-sized and unequal-sized clustering approaches.

4.1. Equal-sized clustering algorithms

Equal-sized clustering approaches have extensively been explored by many researchers. The number of proposed approaches in this area is huge and reviewing all of them in detail is not possible.

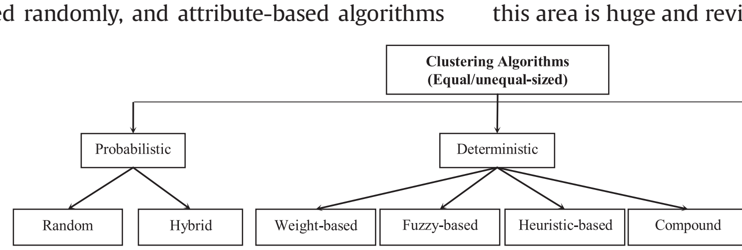

Fig. 4. Classification of the clustering approaches.

The main idea in equal-sized clustering algorithms is to form the clusters with relatively equal sizes, to keep the number of clusters as small as possible, evenly distribute them across the network, and typically, provide minimum overlapping among them. Basically, this type of clustering has a major problem: the distance between the nodes and the BS does not affect the size of clusters. Consequently, the traffic load is not evenly distributed among all the nodes, because the nodes in the vicinity of the BS have to relay more data than farther ones. Generally, traditional clustering approaches support equal-sized clustering. We divide the equal-sized clustering algorithms into three general groups: probabilistic, deterministic, and preset methods. We first target probabilistic clustering algorithms and explain the most important and basic approaches in detail and then discuss their advantages and drawbacks. We then discuss the most important extensions and similar works related to the basic approaches. Afterwards, we move to deterministic clustering algorithms. Finally, preset clustering approaches are reviewed.

4.1.1. Probabilistic clustering algorithms

The primary objective in probabilistic clustering algorithms is to prolong the network lifetime as much as possible. Some of these algorithms (particularly LEACH) aimed at randomly selecting the heads. This group conserves the simplicity and produces a near optimal overhead for clustering the nodes. In order for a clustering protocol to be efficient, the overhead of clustering, including the message and time, should be small. This overhead is incurred because the nodes need the local information to be able to organize themselves into clusters. On the other hand, others utilize some helpful metrics to achieve more goals in addition to the increased network lifetime, including reduced routing delay and fault-tolerance. However, the overhead of clustering in the latter group is accordingly increased. In this section, we describe the most important probabilistic clustering approaches and classify them into random and hybrid approaches.

4.1.1.1. Random algorithms. *Low Energy Adaptive Clustering Hierarchy* (*LEACH*): The first attempts in the area of clustering the nodes in WSNs is LEACH (Heinzelman et al., 2000, 2002). The main idea behind LEACH is to rotate the CH role among all the nodes to achieve load balancing. In LEACH, the operational time is divided into some *rounds* and each round is divided into two phases: *setup* phase in which the clusters are formed and *steady-state* phase in which the data are directly transmitted to the BS by the CHs. The timeline of operations in LEACH is depicted in Fig. 5.

Note that the CH selection in LEACH is distributed with low overhead imposed by CH selection. LEACH uses a random approach for selecting the CHs and assures that all the nodes in the network get selected as CH for at least once in an predetermined epoch. The length of the epoch depends on the number of the nodes and clusters (n/k rounds). At the beginning of the CH selection phase, all the nodes generate a random number between 0 and 1. Then each node compares its number with

$$T(n)=\begin{cases}\dfrac{p}{1-p\times(r \bmod \dfrac{1}{p})} & \text{if } n\in G\\ 0 & \text{otherwise,}\end{cases} \tag{1}$$

where p is the desirable percentage of the CHs, r is the current round, and G is the set of nodes that have not been selected as CHs in the last $1/p$ rounds. All the nodes in G compare their number with $T(n)$. If they found their number less than $T(n)$, elect themselves as CH and broadcast the *CH-ADV* message to all the nodes within R_c. If not, they join the nearest CH by sending the *Join-Req* message to the CH. Each node finds the nearest CH based on received strength signal indicator (RSSI). After cluster formation, the CHs aggregate the received data from the regular nodes and send them to the BS in a single-hop. LEACH utilizes TDMA protocol for gathering the data from regular nodes so that each regular node goes to sleep except in its time slot. Also, code division multiple access protocol is used in LEACH in order to avoid the collisions. Note that a multi-hop version of LEACH is M-LEACH (Mhatre and Rosenberg, 2004) that simply investigates the effect of multi-hop versus single-hop one communication with the BS. The authors show that M-LEACH has a better performance than LEACH.

In addition to distributed clustering in LEACH, a centralized clustering approach or LEACH-C is proposed by the same authors in Heinzelman et al. (2002). In LEACH-C, the BS is responsible for cluster formation. At the beginning, each node sends its information including its location and energy level to the BS. The BS then computes the average of the node energies and the nodes with energy below this average are not selected as CH for the current round. When the BS selects the CHs for the current round, it broadcasts the node ID of the CHs to all the nodes in the entire network. The nodes that are not selected as CH join the nearest CH. Note that because of its centralized architecture LEACH-C has the scalability problem.

LEACH protocol is simple, it is distributed, generates low overhead for CH selection, load is balanced, and the percentage of the CHs in the network is appropriate and can be defined; however, LEACH has some defects. The communication between the CHs and the BS is direct, so the power of the CHs, specially farther CHs to the BS, is depleted at a faster rate and thus the network cannot be implemented in large scales. Also, since the CHs are selected randomly, two important problems emerge. First, the distribution of the CHs across the network is not performed properly. Second, the energy of the CHs is not considered in the CH selection so the nodes with low energy have may get elected as CHs. These problems encourage the researchers to improve the protocol. In the following, the most important ones are discussed.

Threshold sensitive Energy Efficient sensor Network protocol (*TEEN*): TEEN (Manjeshwar and Agrawal, 2001) is a threshold-based

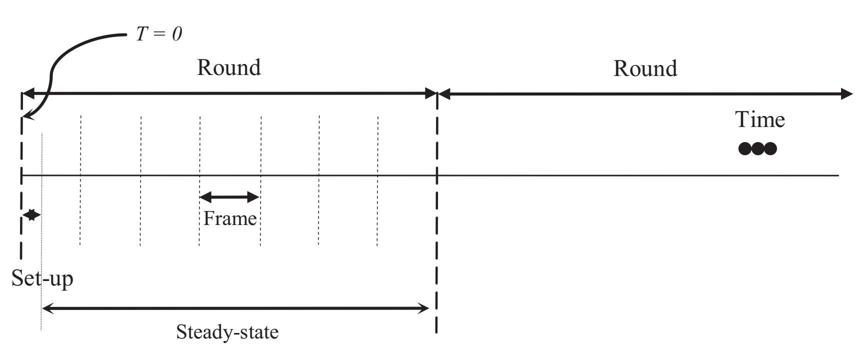

Fig. 5. Timeline of operations in LEACH.

clustering protocol for WSNs. Unlike LEACH in which all the nodes always have data for transmission, TEEN is designed for applications where the data should be sent to the BS when a specific event occurs. The main idea of TEEN is to use the data-centric protocols in a hierarchical structure. There are two thresholds in TEEN, soft and hard, which are used to define when the data should be transmitted to the BS. Each node should switch on its transmitter and report its data to its CH whenever it sensed the hard threshold. On the other hand, the soft threshold just causes the node to switch on its transmitter without any report to the CH, because there is no change or is a little change in the attribute. Using the thresholds, the protocol can define the traffic that should be sent to the BS. The architecture of TEEN is depicted in Fig. 6. As shown in this figure, the data are gathered by the first-level CHs, in a hierarchical scheme. In this scheme, the data of each level CHs are gathered by the next level CHs, and are transmitted to the BS. TEEN has an advantage of reducing the number of transmissions to the BS so that the approach is more energy-efficient. Also, data-centric nature of TEEN makes it suitable for time-concerned applications in which a quick response from the network is urgent for user. However, there are some problems in TEEN; firstly, the user has no feedback from the field of interest unless the thresholds are reached. Consequently, some nodes may die while the user is not aware of their death because it does not receive feedback. Secondly, defining the exact value of the thresholds according to the application is not very easy. Finally, TEEN is not suitable for the applications in which a periodical feedback from the region is needed, like the monitoring of a forest.

An improved version of TEEN is proposed in APTEEN (adaptive threshold sensitive energy efficient sensor network) (Manjeshwar and Agrawal, 2002) by the same authors. In fact, APTEEN is a hybrid method that targets both types of data acquisition approaches, i.e. reactive, like TEEN, and proactive, like LEACH. After CH election, the CHs broadcast four parameters to their members: attributed, thresholds, schedule, and count time. According to these factors, each node sends its data to its CH only if its sensed data reach the hard threshold. The nodes that have not transmitted any data for a predefined duration should sense the field and transmit their data to the CH. These parameters can get different values that make APTEEN very flexible.Some advantages of APTEEN are that the data are periodically gathered from the regular nodes, aggregated in the CHs, and the thresholds are defined and sent to the cluster members via the CHs. However, APTEEN has the additional complexity of implementing the threshold functions and the count time that imposes some overheads to the network. Also, CH election in APTEEN is performed by the BS which has practical problems, because of the centralized nature.

Power-Efficient GAthering in Sensor Information Systems (*PEGASIS*): Chain-based protocol is proposed in PEGASIS (Lindsey and Raghavendra, 2002; Lindsey et al., 2002) which is an improvement on the LEACH protocol. PEGASIS has two main objectives: first,

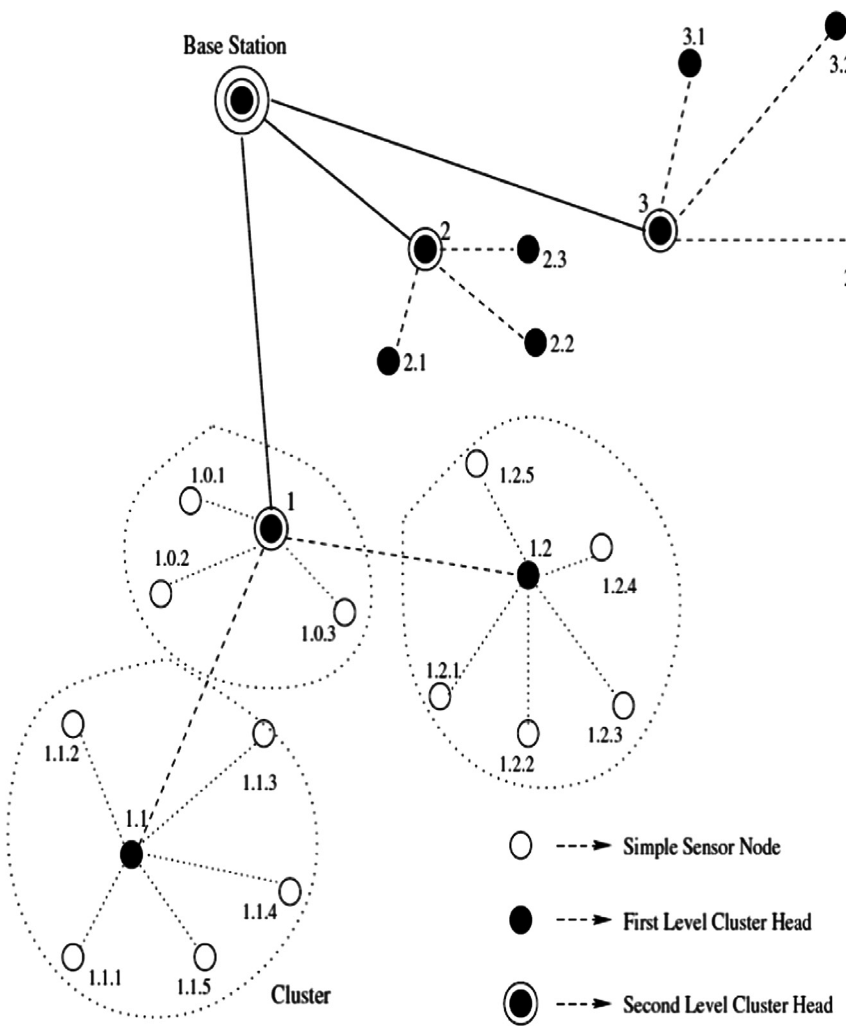

Fig. 6. Architecture of TEEN (redrawn from Manjeshwar and Agrawal, 2001).

improving the network longevity and uniform energy consumption among the nodes, and second, using a chain-based multi-hop path which tries to reduce the delay between the source and the BS. Using a greedy algorithm, the chain is established from the farthest node to the closest node to the BS. In order to build the chain, each node sends its sensed data to its closest neighbor node in the chain, then the next-hop node receives the data, aggregates them with its own data and sends them to the next-hop node in the chain. The closest node to the BS is responsible for transmitting the data to the BS. In PEGASIS, it is guaranteed that each node is a leader once in an epoch. The number of epochs equals the number of nodes so that the load is uniformly distributed among all the nodes. More importantly, unlike LEACH, PEGASIS efficiently handles different network scales using multi-hop chain-based approach among the nodes. However, this is assumed in PEGASIS that all the nodes have global knowledge of the network; this is not practical, because knowing the entire network topology is not practically possible in large distributed systems, particularly in WSNs with restricted resources. Furthermore, in chain-based architectures, it is probable that some nodes get bottleneck. Also, the single chain approach diminishes the network reliability such that if a node in the chain dies the nodes are disjointed.

An extension on PEGASIS is Concentric Clustering Scheme (CCS) (Jung et al., 2007), the main idea of which is to consider the location of the BS to enhance the performance and to prolong the lifetime of the network. As shown in Fig. 7, in CCS, the network is divided into some tracks (levels). Each level contains a chain and a head of the chain. Like PEGASIS, using a greedy algorithm, the chain in each track is constructed from farthest node to the BS. The data transmission is then started from the farthest node to the BS which sends its data to its nearest node in the chain. The next node receives the data, aggregates them with its own data and sends them to the next hop. This is continued until the data reach the CH. The CH, after data gathering and fusing, sends the data to a lower level CH. The authors show that using this method some problems of PEGASIS, such as lack of information about the BS location and being bottleneck of the CH can be solved.

CLUBS: A clustering algorithm based on local density of the nodes has been proposed in CLUBS (Nagpal and Coore, 1998). The algorithm takes the advantage of local communication to aggregate the nodes into clusters. In CLUBS every node in the network must be connected to a cluster, the diameter of all clusters in the network should be the same and the clusters should support the intra-cluster communication (the nodes in a cluster must be able to communicate with on another). In CLUBS, all the nodes count down starting from a random number generated between zero and R. Once a node counts to zero and

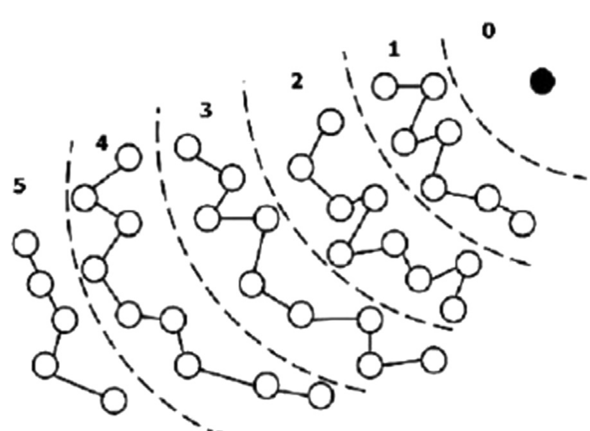

Fig. 7. Constructing the chains in a tracked WSN by CCS (redrawn from Jung et al., 2007).

is not stopped by other nodes, the node declares its status as the CH and broadcasts a "recruit" message to all the other nodes. When a node receives this message, it should join the sender and become a "follower," even if the node is still counting. If two CHs to be in the same neighboring, 1-hop away from each other, so all the operations of CH election should be started again. The authors show that the CLUBS algorithm can handle both synchronous and asynchronous distributed systems. Furthermore, CLUBS satisfies some other limitations on large distributed systems, like the algorithm does not require global IDs, global knowledge of all the nodes, or the diameter of the network. However, since CLUBS is designed to be performed in WSNs, which are very dense networks, CH re-election in the case of two CHs are in the same neighboring would increase energy consumption.

Energy Efficient Hierarchical Clustering (*EEHC*): Another clustering protocol is EEHC (Bandyopadhyay and Coyle, 2003) which divides the network into a hierarchy of layers. The operations in EEHC are classified into the initial and the extended stages. In the initial stage, the data are gathered from regular nodes by the CHs (level-1, the lowest level). Then, the data are aggregated and transmitted to the CHs of the next layer (layer-2). The operations are recursively repeated until the data reach the BS. In order to elect the CHs, a probabilistic algorithm, based on the node density in neighboring of the node, is served. In the algorithm each node announces itself as a CH based on its probability p (called *volunteer CHs*). Then, all the nodes that have received this announcement from the node by either direct or forwarded communication join the nearest CH based on the signal strength. Also, there are some nodes that receive this announcement from none of the CHs in the neighborhood of the node and elect themselves as new CHs (*forced CHs*). The authors present some analysis on the optimal values of clustering parameters and find the best ones, resulting in reduced energy consumption. EEHC improves the network lifetime and makes large-scales WSNs more scalable because of its hierarchical architecture; however, data aggregation in multi-layered clustering might increase the delay, because the data should be stored in intermediate nodes until other data arrive and then are aggregated and transmitted to the BS (Krishnamachari et al., 2002).

Fast LOcal Clustering service (*FLOC*): FLOC (Demirbas et al., 2004, 2006) is a distributed clustering mechanism which divides the network into non-overlapped and equal-sized clusters. In this method, based on the assumed double-band radio model, a node can reliably communicate with the nodes in its inner-band (i_band) range, or can unreliably communicate with the nodes in its outer-band (o_band) range. Clustering algorithm (called *program*) in FLOC is composed of a set of variables and actions. A state is defined by a value for every variable in the program, and an action is said to be enabled if its boolean expression (named *guard*) is true. The main algorithm of FLOC is outlined in Fig. 8. A node stays idle for a random time until it receives a candidacy message from a potential CH. If a node receives no such message, it broadcasts itself as a candidate (action 1). An idle node, by receiving a candidacy message, can also become either an i_band or o_band based on its proximity to the concerned sender (action 5). A candidate node may turn to an o_band node if it receives a conflict message (action 3). The conflict message demonstrates that if a node forms a cluster, then its i_band nodes overlap with i_band nodes of the sender of the conflict message. A candidate can also become a CH if it receives no conflict message (action 4). An o_band node may become an i_band node of another cluster if it receives a candidacy message from a closer potential CH.

FLOC is fast (in $O(1)$) and performs well on large WSNs. The algorithm produces equal-sized clusters with minimum overlapping. Mobility of the nodes is supported in FLOC, that is the protocol has self-healing capabilities in adding new nodes to the network and managing the mobility of the nodes. However, the

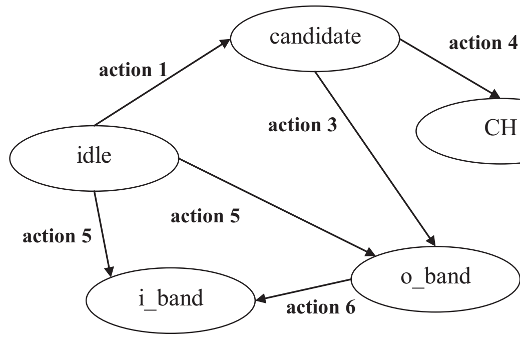

Fig. 8. Actions and state transitions in the FLOC clustering algorithm (redrawn from Demirbas et al., 2006).

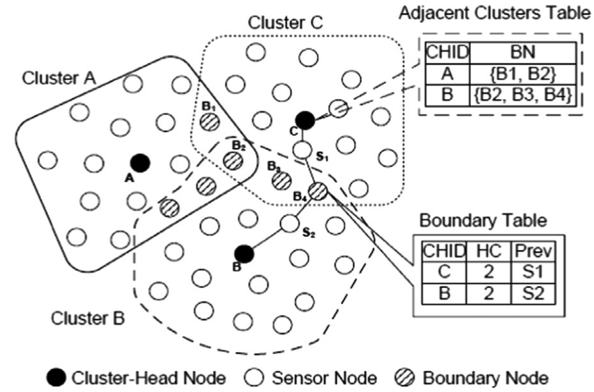

Fig. 9. Overlapping clustering by KOCA (redrawn from Youssef et al., 2009).

clustering process in FLOC is so complicated, the CH selection is performed randomly, and it is unclear how the data are transmitted among the nodes.

Two-Level LEACH (*TL-EACH*): A two-level hierarchy protocol for low energy WSNs is proposed in TL-LEACH (Loscri et al., 2005). In the approach, two types of CHs are defined: primary and secondary CHs. The primary CHs are located at the outer layer, and the secondary CHs are at the inner layer. The CH election is performed in a similar way to LEACH, with a minor modification to the CH probability. First the primary CHs are elected, and then the secondary CHs are elected from the remaining nodes. Each node communicates with its associated CH in a single-hop. TL-LEACH constructs a hierarchical tree where the primary CHs transmit their data to the secondary CHs, the secondary CHs transmit to the sub-cluster nodes up until the data reach the BS. Other operations in TL-LEACH like cluster formation, data aggregation, using TDMA schedule, etc. are performed as LEACH. TL-LEACH reduces the number of CHs in the network by two-level clustering so that the percentage of the nodes that should send data to the BS decreases. Thus, the energy is conserved more.

Multi-hop Overlapping Clustering Algorithm (*MOCA*): An overlapping based clustering approach has been proposed in MOCA (Youssef et al., 2006) which uses a random method for CH selection. Each node produces a probability p, based on which announces itself as a CH within its cluster range. This announcement is forwarded to all the nodes within the range of k hops from the CH. Then each node sends a request to all the CHs from which it has received the announcement. The authors propose a method to support overlapping in order to achieve some objectives such as inter-cluster routing, topology discovery and node localization, and recovery from CH failure. An extended version of the paper appears in KOCA (Youssef et al., 2009) which tries to solve the overlapping clustering problem. KOCA, with a specific average overlapping degree, achieves an equal-sized clustering. An example of formed clusters by KOCA is depicted in Fig. 9. As shown, there are two tables: *adjacent clusters table* and *boundary table*. The adjacent clusters table holds the information about the neighbor CHs and common boundary nodes. The boundary table holds the information for boundary nodes about the CHs to which they belong and the hops number from them. However, MOCA imposes some overheads in terms of time, due to the waiting time in node membership to form the overlapping clusters, and the messages, particularly in boundary nodes.

Clustering Communication Based on number of Neighbors (*CCN*): Another approach is proposed in Shigei et al. (2010) in which a clustering communication method (CCN) for WSNs is proposed. The algorithm consists of four main phases: calculating the number of neighbors, CH election, cluster formation, and determining TDMA schedules. CH election in CCN is randomly performed and each node decides to be a CH and broadcasts the candidacy if it has not received any candidacy. Other nodes that receive this candidacy join the CH. The CHs adjust their broadcasting range in order to reduce energy consumption. The main idea of cluster formation in CCN is to adjust the cluster range based on the number of neighbors, in order to balance the consumed energy in the CHs. For example, the CHs with a larger number of neighbors have a smaller cluster range and vice versa. Fig 10 depicts the main idea of cluster formation in CCN. Using simulation, the authors show that compared to LEACH CCN significantly improves network lifetime.

4.1.1.2. Hybrid algorithms. Deterministic CH Election in LEACH: Another extension on the LEACH protocol is proposed in Handy et al. (2002) where the residual energy of the sensor nodes is included in the CH selection. This causes the algorithm conserves the simplicity and the distributed nature of LEACH, and at the same time, considers the residual energy of the nodes for CH election,

$$T(n)_{new} = \frac{P}{1-P\left(r \bmod \frac{1}{P}\right)} \frac{E_{n\_current}}{E_{n\_max}}, \tag{2}$$

where $E_{n\_current}$ and E_{n_max} are the current energy and the initial energy of the nodes, respectively. Including the residual energy of the nodes in the new CH probability has a basic problem: after a certain number of rounds, while there are still alive nodes with enough energy to transmit the data to the BS, the network is suspended. In order to solve this problem the authors modify the probability as

$$T(n)_{new} = \frac{P}{1-P\left(r \bmod \frac{1}{P}\right)} \left(\frac{E_{n\_current}}{E_{n\_max}} + r_s \operatorname{div}\frac{1}{P}\left(1-\frac{E_{n\_current}}{E_{n\_max}}\right)\right), \tag{3}$$

where r_s is the number of consecutive rounds in which a node has not been CH. Other operations like cluster formation and data transmission are the same as LEACH. Using this method, the authors show that some improvements on LEACH, in terms of the network lifetime, are possible. However, the approach still suffers from some problems of LEACH; for example, the communication of the CH with the BS should be direct, and the assumption that all the nodes are able to reach the BS is not practical.

Hybrid Energy-Efficient Distributed Clustering (*HEED*): A hybrid clustering approach called HEED has been proposed in Younis and Fahmy (2004a,b) which is an iterative-based clustering scheme and uses a hybrid of the residual energy and communication cost, such as AMRP (the minimum power level required by a node to

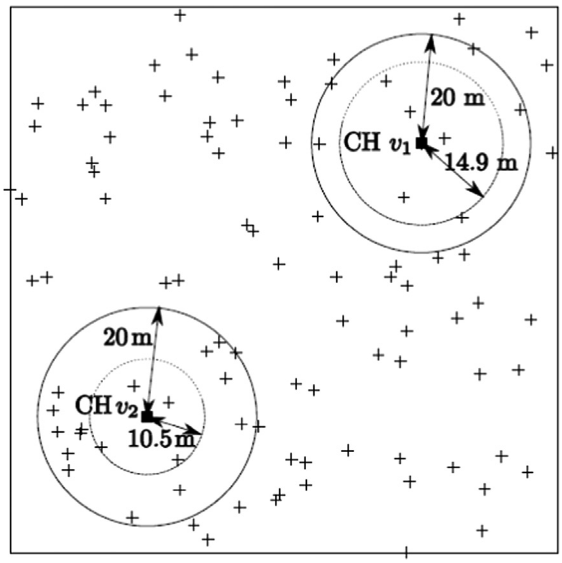

Fig. 10. Different cluster sizes formed by CCN (redrawn from Shigei et al., 2010).

communicate with its CH) or node degree, to elect the CHs. Distribution of the CHs across the network is properly done in HEED. After CH election, the CHs form a backbone in the network and through a multi-hop approach the sensed data of the regular nodes are transmitted to the BS by the CHs.

CH election in HEED is made of three main phases: initialization, main processing (repeat), and finalization. At the beginning of the operations, all the nodes generate a probability as

$$CH_{prob} = C_{prob} \times \frac{E_{residual}}{E_{max}}, \tag{4}$$

where C_{prob} is an initial percentage of CHs and is set to 5% and $E_{residual}$ and E_{max} are the current residual energy and the initial energy (fully charged battery) of the nodes, respectively. In the CH election process of HEED, each node, through several iterations, becomes either a CH or a regular node and picks up a low cost CH. Each node, proportional to its CH_{prob} probability, can elect itself as *final* CH if its CH_{prob} reaches 1, or *tentative* CH if it is less than 1. A tentative CH can turn to a regular node in the upcoming iteration if it finds a low cost CH in its neighborhood. After selected as final CHs, the nodes broadcast their status a message, with their communication cost included, to all the nodes within cluster range (R_c). Afterwards, each node picks up the lowest cost CH to which it periodically sends its data. As mentioned, the communication cost can be either AMRP or node degree. AMRP is defined as

$$AMRP = \frac{\sum_{i=1}^{M} MinPwr_i}{M}, \tag{5}$$

where M is the number of nodes within cluster range that try to reach the CH. Note that the nodes that receive no CH advertisement elect themselves as tentative or final CH depending on their CH_{prob}. At the end of each iteration, each node doubles its CH_{prob} and goes to the next iteration.

In HEED, the clusters are evenly distributed across the network, the data are transmitted to the BS by the CHs in a multi-hop manner, and the energy-efficient design is desirable so that the CHs have relatively high residual energy. Despite these advantages, HEED has some shortcomings. For example, it is an iterative-based clustering approach and the generated overhead for CH selection is high, because each node has to broadcast many messages in each iteration. Since the final CHs are selected by the secondary metric (i.e. the communication cost) and some nodes with low residual energy and high communication cost may get selected as CHs for consequent rounds, HEED has load balancing problem. Finally, in this method the number of formed clusters across the network is large, so energy consumption also increases.

As mentioned earlier, the number of formed clusters in HEED is large and the energy consumption also increases. An extension on HEED is proposed in Huang and Wu (2005) which offers re-execution of the HEED algorithm by the nodes. In HEED, some nodes detect themselves as alone nodes and elect themselves as final CHs. This makes the number of clusters in HEED large, so the extension on HEED (Huang and Wu, 2005) enforces the alone nodes to re-execute the main HEED algorithm. This results in a reduction in the number of CHs in the network so the routing latency is reduced and the network lifetime is increased.

Stable Election Protocol (*SEP*): Researchers use LEACH in different aspects. As an example, SEP (Smaragdakis et al., 2004) uses LEACH in heterogeneous WSNs. SEP studies the impact of heterogeneity, in terms of energy of the nodes. To elect the CHs, SEP uses a weighted probability method based on remaining energy in the nodes. This could prolong the stability period of the networks (stability is defined as the time from the beginning of the network process until the first node dies). In SEP an adjustable percentage of the nodes have higher energy than the other nodes. Accordingly, a modified probability is defined to consider the residual energy of the nodes. Based on this probability, the length of used epoch in LEACH is increased. The authors show that, compared to LEACH, SEP can increase the stability period of the network.

Energy-Efficient Clustering Scheme (*EECS*): A competition-based clustering approach is proposed in EECS (Li et al., 2005a). In this scheme in order to select the CHs, based on a probability each node becomes a candidate and announces its status to all its neighbor nodes within a competition range. Each candidate node, after waiting to receive the announcement from other competition nodes, checks for if there is a candidate with a greater residual energy. If there is one, the node leaves the competition, otherwise, the candidate elects itself as new CH. In EECS it is assumed that the communication with the BS is direct. Beside the competition based CH selection, a distance-based cluster formation is proposed in the paper. In order to achieve a better load balance, the nodes join the CHs that are closer to the BS. This is because the farther CHs consume more energy to send a message to the BS (Heinzelman et al., 2002). The paper shows that, using this cluster formation method, a small improvement on the network lifetime is possible. Similarly, the approach has the following problems: the communication of the CHs with the BS is direct and the competition-based CH election incurs some overhead in terms of the message and time.

Energy-Driven Adaptive Clustering Hierarchy (*EDACH*): An extension on LEACH is EDACH (Kim and Youn, 2005a) which is an enhanced version of LEACH and PEACH. The main goal of EDACH is to solve the energy problem of CHs in LEACH by using proxy nodes like in PEACH. The authors of the paper show that EDACH improves the performance of PEACH by forming more clusters in the regions farther from the BS. In EDACH, the number of CHs can vary depending on their distance to the BS. The paper assumes that the network is segmented into three segments: near, medium, and far. Each node calculates a threshold based on Eq. (1) and p is related to the segment to which the nodes belong. The value of p for near, medium, and far segments are, respectively, $(1-x)p$, p, and $(1+x)p$, where $0 < x < 1$. Using this method, the closer nodes to the BS have smaller threshold and so smaller chance to be elected as CH. Similarly, the nodes located in the farther segments to the BS have greater threshold and thus a greater chance to be elected as CH. Other operations are the same as those in LEACH. In the steady-state phase, for rotating the CH role a threshold based approach is proposed. When the energy level of a CH falls below a predefined threshold, the proxy node becomes the new CH. The authors show that using this approach some improvements on LEACH are achieved. Similarly, the approach does not consider the residual energy of the nodes in CH election. More importantly, the preliminary of CH election in the paper is network partitioning, despite the fact that the approach provides no guideline on defining partitions.

Time Controlled Clustering Algorithm (*TCCA*): In another extension on LEACH, TCCA is proposed in Selvakennedy and Sinnappan (2007). In particular, TCCA introduces a criterion for CH election in which the residual energy of the nodes and the CH election probability of LEACH are merged. The nodes first produce a random number between 0 and 1. They then compare their number with the introduced threshold (T_i) and become CH if their number is less than the threshold. T_i is computed as

$$T(i) = \begin{cases} \max\left[\dfrac{p}{1-p\times(r \bmod \dfrac{1}{p})}\times\dfrac{E_{residual}}{E_{max}}, T_{min}\right] & \text{if } n \in G \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $E_{residual}$ and E_{max} are the current residual and initial energy of the nodes, respectively, and T_{min} is a minimum threshold to avoid the possibility when $E_{residual}$ is significantly low. After CH election, the CHs broadcast advertisements containing the node ID, TTL (time to live), residual energy and a time stamp. Timer-based cluster formation is employed to manage the size of formed clusters. However, in special cases that all the nodes have a low residual energy and probability greater than T_{min}, TCCA has the same problem as Handy et al. (2002) and the network can be suspended. Also, cluster formation in TCCA is more complicated than LEACH so that the simplicity is diminished.

Clustering Algorithm via Waiting Timer (*CAWT*): CAWT (Wen and Sethares, 2005) is another distributed clustering approach in which the CHs are elected based on the node degree. More precisely, each node firstly sets a random waiting timer. When the timer is expired, the node decides itself as a new head and informs its new status to all its neighbors. This timer might be shortened whenever the node finds a new neighbor (the effect of node degree). Also, the competitive node might leave the competition by receiving a CH advertisement from its neighbors. Extensive simulations provided in the paper indicate that CAWT outperforms Max–Min heuristic (Amis et al., 2000) algorithm, in terms of time complexity and cluster formation. The authors claim that CAWT operates in asynchronous networks; however it is not generally the case. More importantly, CAWT is not energy-aware. Also, it has the load-balancing problem, because the nodes with a higher node degree are more likely to be elected as the CH in consequent rounds.

A similar CH election algorithm has been proposed in Autonomous Clustering via Directional Antenna (ACDA) (Wen, 2013). The algorithm reduces the sensing redundancy and maintains sufficient sensing coverage and network connectivity in sensor networks. The authors believe that the cluster performance is improved and sensing redundancy can be eliminated by directional antennas, random waiting timers, and local criteria. The clustering mechanism has four phases: determining the primary sensing sectors that is sectors of sensing tasks are defined, CH election using directional antennas, gateway election (inter-cluster communications) based upon deciding communication sectors and information about neighbor sectors, and renewing the CHs and gateways or stabilizing the clusters. The CH election is performed similar to CAWT in which the nodes set a random timer and update it based on receiving the packets from their neighbors in their sectors. Performance comparison using simulations show that the algorithm outperforms LEACH and CAWT, in terms of the network lifetime.

Clustering Method for Energy Efficient Routing (*CMEER*): Another extension on the LEACH protocol is CMEER (Kang et al., 2007). In CMEER, using Eq. (1) each node elects itself as a candidate. Then the CHs are selected from these candidates. Each candidate broadcasts an advertisement and declares its intention to becoming a CH to all the nodes within its radio range. If another candidate node is located within $\alpha \times$ *Advertisement_Range*, $(0<\alpha<1)$, it gives up its candidacy and stops joining the competition. This results in reduction of the number of CHs in the network and a better network energy-efficiency. Besides, data transmission to the BS in CMEER is performed by multi-hop CH-to-CH. CMEER well distributes clusters across the network. However, CMEER also suffers from random CH election. Furthermore, it has the problems of competition-based approaches, i.e. the message and time overheads.

Energy-Efficient Multi-level Clustering algorithm (*EEMC*): A proper hierarchical clustering approach and an extension on EEHC is Energy-Efficient Multi-level Clustering algorithm (EEMC) (Jin et al., 2008) which is designed to achieve minimum energy consumption in sensor networks. The authors show that EEMC terminates in $O(\log \log N)$ iterations (in a network with N nodes), and when the path loss exponent is 2, EEMC achieves the minimum latency. EEMC uses a probability based approach to elect the CHs, where the number of CHs in each level is computed by an analytical approach. Firstly, each node sends its information, including its residual energy and location, to the BS. Having received all the messages from nodes, the BS broadcasts a 'command' message to all the nodes that contains the total remaining energy and distance of the nodes to the BS. According to a probability, the nodes with a higher residual energy or shorter distance to the BS are elected as level-1 CHs. The CHs in level-2 are elected based on a certain probability according the command messages broadcasted by the CH level-1. These operations are repeated until level-*i*. Some assumptions in EEMC are not very realistic or appropriate in WSNs; for example, the nodes in each phase of CH election need to send their information to the BS and receive back the information. This increases energy consumption.

LEACH with Distance-based Thresholds (*LEACH-DT*): In another recent work (Kang and Nguyen, 2012), a distributed CH selection (LEACH-DT) algorithm is proposed which takes into account the distances from the sensors to the BS. Using this method, the proposed scheme optimally balances the energy consumption among the sensors. In fact, LEACH-DT is a modified version of LEACH in which the CH election probability is modified such that the distance of the nodes to the BS affects this probability. A multi-hop routing path is also proposed in the paper; however, multi-hop path establishment in LEACH-DT is complicated and needs many message flooding.

Discussion: In the last decade, random approaches have shown to be a popular clustering method for WSNs. This is because random approaches are simple with reduced overhead, self-organize, and energy-efficient with a long network lifetime. These approaches solve the load-balancing problem in clustering by rotating the role of CH among all the nodes, and as a result, the network is more energy-efficient. Above all, unlike iterative-based protocols, the performance of random methods does not depend on the network diameter and packet loss. This characteristic gains more importance when considering the fact that the packet loss in wireless medium is high. However, the approach has some problems in terms of the form and distribution of the clusters. Also, lack of energy consideration in CH election reduces the reliability in these approaches, because if the nodes with low residual energy get elected as the CHs, the data gets lost as they die. The most valuable random clustering protocol is LEACH that preserves the simplicity near optimal. Some researchers have tried to overcome the shortcomings of LEACH. To do so, PEGASIS with chain-based clustering, TEEN and APTEEN with reactive clustering, EEHC with hierarchical multi-level clustering, and deterministic LEACH (Handy et al., 2002), HEED, and EECS with hybrid clustering are proposed. In addition to LEACH and its extensions, other valuable random protocols include CLUBS, FLOC, and MOCA. Each one of these protocols introduces a novel clustering method. Among the extension proposals, hybrid clustering has gained more popularity than others. The major hybrid algorithms are HEED, EECS, and EEMC. Although hybrid algorithms are energy-aware and are more reliable in forming the clusters, they are not as simple and low overhead as former approaches. This is because in hybrid approaches locally exchanging status messages among the nodes are required. Finally, some protocols, such as TEEN and APTEEN, could be considered as the leading in a new aspect of clustering. As mentioned earlier, TEEN and APTEEN introduce a novel reactive-based clustering method for WSNs. Reactive-based clustering is more suitable method for WSNs, because these networks are naturally data-centric and most of their applications are query-based.

4.1.2. Deterministic clustering algorithms

Unlike probabilistic clustering algorithms, some approaches use more confident metrics to elect the CHs. Usually, these metrics are achieved locally and based upon node conditions. The most conventional metrics in CH election used in published works are the residual energy, node degree, centrality, proximity (to neighbors/BS), etc. The nodes achieve these information via message exchange with their neighbors. We call these methods *deterministic* clustering algorithms, because in these methods, the elected CHs, and consequently, the formed clusters are more controllable. There are different types of deterministic algorithms. As a well-known method of clustering in WSNs, some protocols combine some metrics into a weight and use them to produce balanced clusters. Some other protocols employ fuzzy-logic to handle uncertainties in CH election problem. Furthermore, heuristic-based clustering methods have increasingly gained popularity, because of their optimal solutions. Other methods use different metrics, including the node degree, proximity, mobility, link conditions, etc., to achieve their objectives. We present this group as *compound* algorithms. In the following, we survey the most important deterministic approaches in an organized manner.

4.1.2.1. Weight-based algorithms. Some attribute-based clustering algorithms utilize some metrics in a weight in order to elect the CHs. Typically, collecting some useful metrics, including the residual energy, distance to the BS, and the number of neighbors, result in a better performance. The basis of these approaches is a competition in which the nodes with the highest weight are elected as the CHs. Although many weight-based protocols are distributed, some published works present centralized approaches. In the following some of them are reviewed.

Distributed Clustering Algorithm (*DCA*): A distributed clustering algorithm (DCA) is proposed in Basagni (1999) for ad hoc networks which is also implementable for WSNs. DCA is an iterative and weight-based clustering approach. In DCA, a node waits for all its neighbors to decide to be a CH or join the clusters. The node with the highest weight within the competition range is selected as the CH. This weight is proportional to some metrics of ad hoc networks like the mobility of the nodes. The authors argue that the approach helps the clusters to be more stable so that the overhead of forming the clusters in such mobile networks is minimized. In CH selection if there is no node to elect itself as CH (while there are some nodes with higher weight), the nodes with lower weight elect themselves as new CHs, after waiting for a predefined time. Note that the performance of the iterative-based clustering algorithms highly depends on the network diameter and packet loss. Also, since DCA has been designed for ad hoc networks in CH selection, it does not take the power of the nodes into account. This is solved in CH selection of DWEHC which is a suitable weighted-based algorithm for WSNs, described in the following.

Distributed Weight-based Energy-Efficient Hierarchical Clustering (*DWEHC*): A weight-based clustering approach has been proposed in Ding et al. (2005) which is an extension on the HEED protocol. In the CH selection phase of DWEHC each node calculates a weight as

$$W_{weight}(s) = \left(\sum_{u \in N_{\alpha,c}(s)} \frac{R-d}{6R} \right) \times \frac{E_{residual}(s)}{E_{initial}(s)}, \tag{7}$$

where R is the cluster range, d is the distance from node s to neighboring node u, and $E_{residual}$ and $E_{initial}$ are the current residual energy and the initial energy of node s, respectively. The nodes with the largest weight among their neighboring nodes are elected as *temporary* CHs. A *real* CH is then elected from the temporary CHs, if a given percentage of its neighbors elect it as their temporary CH. In order to achieve more energy-efficiency in DWEHC, the multi-hop intra-cluster communication is supported. In contrast with HEED, the number of formed clusters in DWEHC and the number of single-member clusters are smaller, so DWEHC is more energy-efficient than HEED. However, DWEHC still suffers from the iterative-based problem of HEED. Also, multi-hop intra-cluster communications may increase the total energy dissipation in the network.

Topology Adaptive Spatial Clustering (*TASC*): TASC (Virrankoski, 2005) is a weighted clustering approach in which the weight includes distance, connectivity, and density information within the locality of each node. The main idea of TASC is adopted from Ester (1996) and Zaiane Zaane (2002). TASC aims at clustering the non-uniform sensor networks where the intensity variations in each cluster are less than that of the entire network. The clustering in TASC is performed as the following. First, all the nodes compute their weights and broadcast them to all their 2-hop neighbors. All the nodes elect the node with the highest weight as their *nominee* and declare their decision to all 2-hop neighbors. Afterwards, when all the messages are received, the nodes elect the closest nominee to themselves as the leader and join them. The introduced weight is composed of two key elements: the incidence a node has found on the shortest path between pairs of nodes; and the distance contribution of the edges of that node with respect to the total length of the path. However, the main problem of TASC is that a large number of control messages should be transferred in order to select the leader nodes. As a result, since the communications is the most energy consumer unit, energy consumption is increased.

Spatial-based Clustering: In Ma et al. (2011), an application-specific and spatial correlated based clustering algorithm for sensor networks is proposed. The algorithm uses the spatial correlation between the sensed data of the sensors to build the clusters. The primary objective of the algorithm is to use the clustering for efficient data aggregation. In order to select the CHs, the algorithm uses a weight-based approach. The introduced spatial correlated weight considers the average spatial distance variance among each node and its neighbors within a predefined communication range. This weight has direct relation with spatial correlation of the nodes, that is, the higher the weight of a node, the higher the spatial correlation with its neighbors. The nodes with the highest weight are elected as the CHs (*dominators*). The non-dominator nodes (*domaintees*) join the nearest domiators. To evaluate the performance of the algorithm, a pattern recognition scenario over environmental data is presented in the paper. Using simulation, the authors show that the algorithm presents a higher degree of accuracy in terms of information description in aggregated network, in contrast with other cluster-, tree- and grid-based frameworks.

Energy Efficient Clustering Algorithm Based on Neighbors (*EECABN*): In Wei (2011), a centralized clustering approach based on neighbours status is proposed. In the method, a combined weight for electing the CHs is introduced which is composed of the following factors: the distance between the node and the BS, the distance between the node and its neighbouring nodes within communication range *R*, and the residual energy of the node. Accordingly, the weight is defined as

$$W(i) = E(i)\frac{h_{max}}{\max(h(i), \epsilon)} \sum_{j \in B} \frac{1 - \dfrac{d(i,j)}{R}}{\max\left(\dfrac{E(j)}{E_{max}}, \epsilon\right)}, \tag{8}$$

where in node i, $E(i)$, $h(i)$, and R are the residual energy, the distance to the BS, and the communication range in which other nodes can communicate to i successfully with enough signal level, respectively, B is the set of neighbors for i, j is one of the neighbors within R, $d(i,j)$ is the distance between i and j, the variable h_{max} is the distance between the BS and the farthest node in the network, E_{max} is the initial energy, and ϵ is a constant to limit the lower bound of $E(j)/E_{max}$ or $h(i)$, while a node has a very low residual energy or is too close to the BS. In the method, the nodes are divided into strong, with an energy higher than the average energy of all the nodes in the network (E_{ave}), and weak, with an energy smaller than E_{ave}. The CHs are elected from the strong nodes as the follows. The BS collects the information of the nodes, computes the weights and elects the nodes with the highest weight as the CHs. Isolated CHs are the CHs with no members. The algorithm enforces these CHs to turn to regular nodes and join other adjacent clusters. Also, if all the nodes are weak, the BS elects the CHs among them in the same way. Other operations are similar to those of LEACH. Simulation results in the paper show that EECABN outperforms LEACH and HEED, in terms of the network lifetime. As expected, since EECABN is centralized, it has the scalability problem. Also, communication with the BS increases energy consumption.

4.1.2.2. Fuzzy-based algorithms. Recently, fuzzy-logic has been used by researchers to select the best set of CHs across the network. Fuzzy logic is usually used to model the human experience and the human decision making behavior. In fuzzy-logic the input/output relationship is expressed by a set of linguistic rules or relational expressions. A fuzzy-logic system consists of four main parts, as shown in Fig. 11: a fuzzifier, a fuzzy inference engine, a defuzzifier, and fuzzy rules.

Usually, input data are crisp, so a fuzzification of input data is required which is performed by fuzzy inference engine that converts the crisp data into a set of linguistic values. Fuzzy rules define behaviors of the system. In a reverse manner, defuzzifier produces crisp data from the results generated by the fuzzy inference engine. The major problem of fuzzy-logic-based clustering approaches is their complexity as many messages are to be exchanged in order to execute the algorithm. Running complex algorithms by sensor nodes with poor resources dissipates a lot of energy. In this section, we review the most important approaches in the area of fuzzy-logic-based clustering.

One of the first major attempts in fuzzy-logic-based clustering is proposed by Gupta and Sampalli (2005) who try to resolve the problems of LEACH using fuzzy-logic approach. In this method the node degree, node residual energy, and node centrality are used as fuzzy variables. The authors show that using this mixture, the network lifetime can be improved. In the work, the BS selects the CHs based on 27 fuzzy rules. The proposed approach is centralized and suffers from the scalability problem.

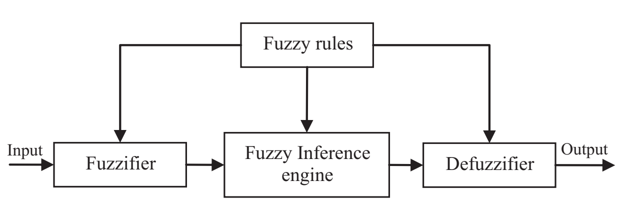

Fig. 11. Fundamental block diagram of fuzzy-logic.

Distributed fuzzy-logic-based clustering approach is proposed in Cluster Head Election mechanism using Fuzzy logic (CHEF) (Kim et al., 2008a). In this method in every round, each node generates a random number between 0 and 1. If the number is smaller than a predefined threshold, the node becomes a tentative CH. There are two fuzzy descriptors used in CH selection: the residual energy of each node and local distance. The local distance is the sum of distances between a node and other nodes within its radius *r*.

LEACH-FL (Ran et al., 2010) is an improvement on LEACH similar to Gupta and Sampalli (2005). which uses three descriptors: node residual energy, node degree and distance from the BS to compute the chance. Using 27 defined fuzzy rules, the BS selects the nodes with the higher chance as the CHs. Although this method has the same drawback as Gupta's method does, it presents better results. Each of the input functions has three membership functions which show different degrees of the functions. The defined rules in the approach are based on the following formula:

$$Probability = (Battery\ level \times 2) + Node\ density + (2 - distance). \tag{9}$$

A fuzzy-logic-based clustering approach is proposed in LEACH-ERE (Lee and Cheng, 2012b) in which using fuzzy techniques, each CH is elected based on the residual energy prediction. When the prediction is performed, the node with the highest predicted residual energy is elected as the new CH. As shown in Fig. 12, the inputs of the Fuzzy-logic Inference System (FIS) are the residual energy and the expected residual energy, and its output is a CH election probability, called chance.

Energy-aware Clustering Protocol using Fuzzy-logic (ECPF) (Taheri et al., 2012) is another recent extension on HEED protocol. ECPF uses fuzzy-logic to achieve an on-demand clustered network. In ECPF, the node degree and node centrality are used as input variables of the fuzzy system. The main structure of ECPF is the same as HEED which is made of three main phases: initialization, main processing, and finalization. At the beginning of the operations, each node computes its cost. Then, each node sets a delay timer proportional to its inversed residual energy value. This means that the node with the higher residual energy should wait less than other nodes with a lower energy. After the timer expired, each node that receives no CH announcement broadcasts a tentative CH announcement within its cluster range. The nodes can become final CH if they have the least cost among the other tentative CHs. Finally, the *uncovered* nodes elect themselves as final CHs.

There are some other fuzzy-logic clustering approaches that are listed in the following. In Mhemed et al. (2012), a fuzzy-logic based clustering approach (FLCFP) is proposed. In FLCFP, three main

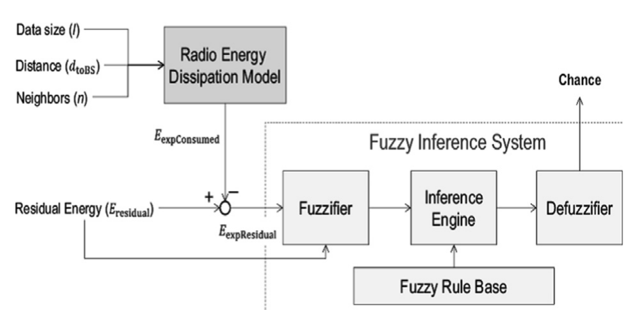

Fig. 12. Computing the CH election probability (chance) using fuzzy-logic (redrawn from Lee and Cheng, 2012b).

parameters are used in clustering the nodes as the inputs of the FIS: the energy level, the distance to the BS, and the distance between the CH and the node. In another fuzzy-logic-based clustering approach (Anno et al., 2007) the distance of cluster centroid and the residual energy of node are used as the inputs of the fuzzy system. The work is finished without any comparison between the proposed method and other works. In a similar work (Siew et al., 2011), the BS considers the energy level and the distance to the BS to elect the suitable CH. Using this method prolongs the First Node Dies (FND) time of the network, data stream is guaranteed for every round and the throughput received by the BS before FND is increased. In a work (Tashtoush and Okour, 2008) inspired ACE, the Migration Fuzzy Module takes the crisp values of node's Loyal Followers and Energy Level as an input, and the output of the defuzzifier process will be the chance of this node to become the new CH.

4.1.2.3. Heuristic-based algorithms. In the last decade, heuristic-based clustering approaches have significantly been used by researchers. These approaches provide a good distribution of the CHs across the network and offer an energy-efficient network, by finding the optimal solutions in terms of finding the best set of the nodes as CHs and best cluster sizes. Different optimization techniques have been used in this regard, including Genetic Algorithm (GA), Ant and Bee colony, Particle Swarm Optimization (PSO), Differential Evolution (DE), and Bacterial Foraging Algorithm (BFA). Each one of these methods uses different parameters in fitness function to reach their objectives. Clustering belongs to the class of NP-hard optimization problem and different algorithms have different performance in solving this problem; however, PSO algorithm has shown a better performance than other algorithms (Kulkarni and Venayagamoorthy, 2011). In these methods, in addition to energy efficiency, the convergence time is an important metric in performance evaluation. These methods are often centralized and a powerful node like the BS executes them, because the global knowledge of the network is needed; however, some approaches employ them in a distributed manner using agent nodes along side other regular nodes (Selvakennedy et al., 2007). Generally, heuristic-based methods have the scalability problem. Some of valuable works in this area are presented below.

There are some works that utilize GA to form the clusters. Genetic Clustering Algorithm (GCA) (Mudundi and Ali, 2007) uses a dynamic approach to form the clusters using GA. The main goal of GCA is to prolong the network lifetime. The parameters used for fitness function are the number of CHs and the sum of all distances of the members to their CHs. In Seo et al. (2009), LA2D-GA has been proposed in which an optimal cluster formation is presented by applying GA to the network. The authors show that LA2D outperforms LEACH; however, assuming the nodes are location-aware reduces the applicability of the method. In another method (Kuila et al., 2013), load-balanced GA-based algorithm is applied to solve the CH election problem for the equal and unequal load of the sensor nodes. The fitness function in the paper is considered on the basis of standard deviation of the CH load in the network. The simulations performed in the paper show enough performance improvement over the simple GA. Multi-objective GA-based clustering algorithm (MOGA) is proposed in Huruiala et al. (2010). The distance between clusters and the number of transmissions has been considered as the main metrics in fitness function. The main objectives in MOGA are maximizing the lifespan and minimizing routing latency. However, MOGA is centralized and is performed by the BS, so the method has the scalability problem.

In addition, some works are inspired by biology. A clustering algorithm based on social insect colonies is proposed in Cheng et al. (2011). Using social insect colonies structure, the work achieves a consistent improvement in terms of network lifetime and sensing coverage. Simulation results show that the algorithm can reduce the delay of data gathering. A bee-colony-inspired backbone selection algorithm (BEES) is proposed in AbdelSalam and Olariu (2012). BEES has four main phases: tiling, backbone selection, and clustering. First the area around the BS is divided into hexagonal shapes like a beehive. So, the area is segmented into six sectors, and each sector into rows. The first row consists of one hexagon, the second one consists of two hexagons, and so forth. Afterwards, the BS, which is located at the center of the beehive, selects six backbones in the first row. The sensors in the first row select the backbones of the next row, and these operations are recursively repeated in the entire network. When the backbone selection is finished, the clusters are formed. In the paper, some discussion about computing the angles of hexagons are provided. BEES mitigates many inherent challenges of WSNs, including localization, clustering and data aggregation. However, cluster formation in BEES is so complicated. Finally, ant-colony-based clustering method has been investigated in Kamimura et al. (2006). At first some nodes with a high rate of the residual energy are elected as the CHs. Afterwards, the clusters are formed by random meetings of the regular nodes in a repeated manner. Meetings among the sensor nodes are performed by local message exchange. The method is distributed and shows a better network lifetime than LEACH and HEED.

As another popular heuristic-based clustering algorithm, PSO has been used in some research to solve the CH election problem. In Guru et al. (2005), four different PSO-based clustering algorithms are proposed: PSO with Time Varying Inertia Weight (PSO-TVIW), PSO with Time Varying Acceleration Constants (PSO-TVAC), Hierarchical PSO with Time Varying Acceleration Constants (HPSO-TVAC) and PSO with Supervisor Student Mode (PSO-SSM) for energy aware clustering. Using simulation, the authors investigate the proposed algorithms in terms of convergence time and node distribution. The paper provides no comparison of energy consumption or network lifetime between the proposed algorithms and other methods, where they are the key metrics in WSNs. PSO-C (Latiff et al., 2007) is a centralized PSO-based clustering method with the objective of minimizing the intra-cluster distance and optimizing the energy consumption of the network. The fitness function of PSO-C is composed of $f=\beta\times f_1+(\beta-1)\times f_2$ where f_1 is the maximum average *Euclidean* distance of nodes to their associated CHs and f_2 is the ratio of total initial energy of all nodes to the total energy of the CH candidates. Simulation results provided in the paper show improvement on the network lifetime compared to LEACH and LEACH-C. In a recent method (Bennani and El Ghanami, 2012), a centralized PSO-based clustering algorithm (PSO-BC) is proposed in which the BS selects the nodes with energy higher than the average energy of all the nodes in the network as the CHs. The fitness function used in the paper is $f(P_J)=f_1(P_J)+\lambda f_2(P_J)$ where f_1 is the sum of *Euclidean* distance between the nodes and their closest CHs and f_2 is the sum of *Euclidean* distance between the CHs and the BS, and λ is a coefficient regulating the distance between the CHs and the BS. The provided simulations in the paper indicate PSO-BC outperforms LEACH-C in terms of the network lifetime.

Other methods use other heuristic-based algorithms to form the clusters. A centralized clustering approach based on the network energy and Quality of Service (QoS) is proposed in El Rhazi and Pierre (2009). In the paper, the clustering problem is modeled as a hypergraph partitioning and the search process is performed by tabu search heuristic. The approach defines moves using largest size cliques in a feasibility cluster graph. Some constrains for clustering are defined in the paper, for example, a node is called *active* if it ensures zone coverage. The authors discussed that in order for a tabu search algorithm to solve a problem there are five steps that should be managed: the algorithm should return an initial solution, a neighborhood should be defined, the tabu lists, aspiration criteria, and intensification and diversification methods should be determined. And the algorithm ends when all possible moves are prohibited by the tabu list, the maximum number of iterations is reached, or the maximum number of iterations (when the best solution is not enhanced successfully) is reached. The authors show that compared to the CPLEX-based, distributed, and simulated annealing-based methods, the tabu search algorithm based clustering offers better performance in terms of network lifetime, cluster cost and execution time. A similar work using tabu search algorithm is presented in Chamam and Pierre (2009). In the paper, each sensor node has three states: turned on, turned off and promoted CH. The work seeks an energy-efficient topology of WSNs under two main constraints: joint routing and coverage. The authors show that the problem belongs to the class of NP-Complete problems and present solutions using a tabu search heuristic algorithm.

4.1.2.4. Compound algorithms. *Hierarchical Control Clustering* (*HCC*): A hierarchical clustering scheme has been proposed in Banerjee and Khuller (2001). Fig 13 shows the main concept of HCC architecture. As shown, all the nodes in the network join the clusters at layer 0. The CHs of layer 0 then constitute a cluster at layers 1 and 2. In contrast with layer 0, the number of nodes within layer 2 significantly decreases, and as a result, the scalability increases. The algorithm has two main phases: tree discovery and cluster formation. In the tree discovery phase, a Breadth-First-Search (BFS) tree rooted at the initiator node is formed. Initiator node is the node that initiates the cluster formation process. Each node then broadcasts a signal once in every p units of time, which carries information about its shortest hop-distance to the root. Other nodes that have received this signal update their hop-count to the root, if their distance to the signal is shorter. Doing so, each node can find the best parent node. Once all the nodes know their parents, the BFS tree is formed and the routes are denoted so the data can be transmitted to the BS. HCC conserves stability of network topology, even in dynamic environments with mobile nodes. Also, HCC is a suitable approach for applications of large-scale WSNs, because of its hierarchical architecture.

Algorithm for Cluster Establishment (*ACE*): ACE (Chan and Perrig, 2004) uses an emergent algorithm in order to form the clusters. Emergent algorithm is defined as any computation that achieves formally or stochastically predictable global effects, by communicating directly with only a bounded number of immediate neighbors and without use of central control or global visibility (Fisher and Lipson, 1999). As a result, an emergent protocol for a sensor network emerges as a result of repeated local interaction and feedback between the nodes. In protocols that employ the emergent algorithm the operations evolve until the best or optimal response is achieved. The ACE protocol is composed of two main parts: *spawning* and *migration*. When a node decides to be a new CH it broadcasts an invitation to all the neighboring nodes within the desired range (spawning). The node sends this invitation to recruit its members. Each node that receives this message joins the new cluster and becomes a *follower* of the new CH. In migration process, the clusters are moved to minimize the overlapping between the clusters. The migration is performed by electing a proper candidate for the current CH. Each CH periodically checks the members conditions to find new CHs among them. A node is elected as a candidate when it has the largest number of followers and its formed cluster has the minimum overlapping with current cluster. ACE is terminated in few iterations, regardless of the network diameter and the number of the nodes.

ACE minimizes the number of formed clusters across the network by providing the minimum overlapping between the clusters. It can repair structure damages in the network caused by node failures and can also integrate new nodes in the network. However, energy is not considered in the CH selection process of ACE so the nodes with low residual energy may be selected as new CHs. The overhead of ACE is high in terms of message complexity, and because of the migration process, many control messages should be exchanged in the network.

9*Base-station Controlled Dynamic Clustering Protocol* (*BCDCP*): A centralized clustering approach, called BCDCP, is proposed in Muruganathan et al. (2005) which uses a high energy BS in order to form the clusters. The main ideas in BCDCP are the formation of balanced clusters where each CH serves an approximately equal number of member nodes to avoid CH overload, uniform placement of CHs throughout the whole sensor field, and utilization of CH-to-CH routing to transfer the data to the BS. In BCDCP it is assumed that a fixed BS is located far from the sensor field and at the beginning of the operations, it receives information about the residual energy of all the nodes in the network. Based on this feedback, the BS first computes the average energy level of all the nodes, and then chooses a set of nodes whose energy levels are above the average as the CHs. The elected CHs are among this set which have sufficient energy to perform the needed tasks of a head node. Other nodes with low energy can do other tasks that need less energy. Hence the BS, by an iterative splitting algorithm, forms the clusters. This algorithm first splits the network into two sub-clusters, and proceeds further by splitting the sub-clusters into smaller clusters. This process is repeated until the desired number of clusters is achieved.

The second major role of the BS in this protocol is to establish the multi-hop path among the CHs. Once the clusters and the CH nodes have been identified, the BS chooses the lowest-energy routing path and forwards its information to the sensor nodes along with the details on cluster groupings and selected CHs. The routing paths are selected by first connecting all the CH nodes (using the minimum spanning tree approach that minimizes the energy consumption for each CH) and then randomly choosing one CH node to forward the data to the BS. This CH is selected randomly so the task of forwarding the received data is evenly distributed among all the CHs. Once the data have been received from all sensor nodes, the CH performs data fusion and transmits them to the next-hop CH towards the BS. BCDCP has advantages in electing the CHs and forming the clusters by the BS which resolves the problem of even distribution of the CHs and the formed clusters have the equal number of members. However, the main

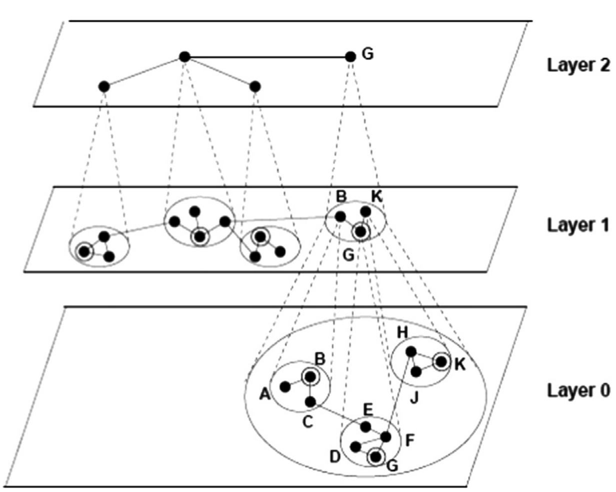

Fig. 13. An example of a three layered HCC (redrawn from Banerjee and Khuller, 2001).

problem of BCDCP is that this approach is centralized and thus suffers from the scalability problem.

Power-Efficient and Adaptive Clustering Hierarchy (*PEACH*): Using overhearing characteristics of wireless communications, PEACH (Kim and Youn, 2005b; Yi et al., 2007) forms the clusters without additional overhead, supports adaptive multi-level clustering and can be used for both location-aware and location-unaware WSNs. This is discussed in the work that most current clustering schemes dissipate a large amount of energy for re-clustering at fixed period of time. PEACH tries to solve this problem by adaptive multi-level cluster formation based on the overheard information of each sensor node. To do so, a node becomes a CH if it hears a packet destined for the node. When a node overhears the packet destined for other nodes it joins the destination node. By simulation, the authors show that the PEACH protocol significantly outperforms EEUC (Li et al., 2005b), LEACH and HEED in terms of network lifetime. Although PEACH eliminates the overhead of re-clustering, it has some problems. Firstly, the paper does not address the time synchronization issue of the nodes. If the nodes are asynchronous, the performance of PEACH is significantly degraded. Thus, PEACH strictly depends on the nodes synchronization. Secondly, the energy of the nodes is not considered in the CH election process so the reliability of PEACH is diminished. Finally, the performance of PEACH highly depends on the packet loss.

Maximum energy cluster head (*MECH*): Another improvement on the LEACH protocol is MECH (Chang and Kuo, 2006). In MECH all the nodes broadcast a 'hello' message to all the nodes within a predefined transmission range. When the number of the neighbors to a node reaches a predetermined number (*CN*), the node broadcasts an announcement demonstrating '*I am a CH*'. This broadcasting is performed to all the nodes that are one-hop away from the node. Then the nodes that receive this announcement record it and start a *back-off* timer. After the timer expiration, each node joins the nearest CH based on RSSI. Once the clusters are formed, the algorithm uses a distance-vector routing to construct the paths among the CHs to reach the BS. MECH solves some problems of LEACH: the CHs transmit their data to the BS by multi-hop, CH election in MECH is based on the node degree and the approach can decide the number of cluster members. However, the CH election in MECH is not energy-aware, and also, message exchange through the CH election and routing phases increases the load in the entire network.

Energy-efficient and dynamic clustering (*EEDC*): In Yu et al. (2007), a dynamic clustering approach has been proposed. In this method, based on the total power a node has received from all the neighboring nodes within its radio range, the number of active nodes is estimated. Then, each node dynamically computes a CH probability that determines if the node is elected as CH. The number of clusters and CHs is adjusted dynamically and autonomously where each node decides if it is elected as CH, or activates the clustering update. In addition to the dynamic clustering, an energy-efficient and power-aware routing protocol (called EEPA) is proposed in the work. In the multi-hop routing protocol, each pair of the CHs that participate in the routing are aware of the link and the nodes energy. The link energy is the necessary energy to establish a connection between a pair of CHs. The nodes energy is the current residual energy of the participant CHs in the routing which is estimated by the CHs. The authors show that the dynamic clustering approach outperforms the static clustering approaches like LEACH and HEED.

Another dynamic approach is proposed in Baek et al. (2010), where in each cluster each sensor node evaluates its relative energy consumption compared to other nodes in the cluster. Based upon the relative energy consumption in the current round, sensor nodes autonomously select a time frame where they will act as a CH in the next round. In addition, they are conditionally allowed to switch their CH depending on the signal strength of their current CH.

Energy-Aware CLustering scheme with transmission power control for sEnsor networks (*EACLE*): Another clustering approach is EACLE (Yanagihara et al., 2007) in which the CHs are elected by a waiting approach. In fact, EACLE is a clustering method based on 2-hop neighbor information which is an interesting combination of SPAN (Chen et al., 2002) and EAD (Boukerche et al., 2003) routing protocols. First all the nodes wait for T_1 seconds, where T_1 is a repetitive decreasing function on the residual energy of the nodes. When the timer expires, the nodes become CH and broadcast two packets with different transmission powers: *power low* and *power high*. The nodes that receive this packet in power low range become members and join the clusters (*slave* nodes). The nodes receiving the packet in power high range decide to become a CH or join the clusters by checking the neighbors list. Since the nodes in the power low range cannot become CH, so EACLE properly distributes the clusters across the network. When a node gets elected as a CH, it sets its timer to a larger value in order to not be selected as the CH in upcoming round. Thus, the CH role is rotated among the nodes. Experimental results provided in the paper show better performance of EACLE over EAD. However, the major problem of timer-based clustering methods is their dependency on time synchronization of the nodes. Full synchronization of the nodes in large-scales is not practically possible; on the other hand, local synchronization might be achieved by message exchange that is energy consumer.

Recent compound algorithms: In another work (Deng et al., 2011), a mobility based clustering (MBC) is proposed, of which the main factors of CH election are the residual energy and mobility of the nodes. The links between the CHs and cluster members are stable during the estimated connection time. After cluster formation, using TDMA protocol and estimated time schedule, the data are transmitted to the CH by the regular nodes. When a node leaves its current cluster, it joins the new cluster by sending join request to the new CH. By extensive simulations, the authors show that MBC outperforms LEACH protocol in terms of better energy consumption and also handles the mobility of the nodes.

Another work (Dahnil et al., 2012) presents an adaptive clustering algorithm (TCAC) to increase the network lifetime while maintaining the required network connectivity. In the proposed scheme, the CHs adjust their power levels to ensure the optimal degree of connectivity. The CHs are elected in a competition-based manner and the nodes with the higher residual energy are elected as the CHs. When the CHs are elected, they adjust and update their transmission ranges in order to conserve the desired connectivity. TCAC presents a well-balanced clusters across the network, and as a result, the network lifetime is improved. However, periodic transmission range updating and competition-based CH election increase the complexity of the algorithm.

EEBCDA (Yuea et al., 2012) is another approach which divides the network into unequal size rectangular grids and makes CHs rotate among the nodes in each grid. Firstly, the network is divided into unequal grids. Then, the nodes with the highest residual energy are elected as the head of each grid. The node ID breaks the ties in the case of equal residual energy in the nodes. Other operations are performed similar to those of LEACH. Network division into grids in the approach is complicated and increases the overhead.

In Liao and Qi (2013), a load-balanced clustering (DSBCA) algorithm is proposed. In the method, the clusters are formed based upon the distance to the BS and density distribution of the sensor nodes. Accordingly, in the uniform distribution of the nodes, similar to other unequal clustering methods, the size of the clusters grows up as the distance between the nodes and the BS is increased. In non-uniform distribution, the cluster radiuses are defined by two factors: the distance to the BS and connectivity density of the nodes. For example, the cluster radius is larger in the farther areas to the BS and with lower connectivity density. On the other hand, if the connectivity density is high, the clusters are smaller, even if the cluster are far from the BS. Using this method, the authors show that a significant improvement over traditional clustering algorithms, like HEED and LEACH, is possible.

LCM (Wang and Chen, 2013) is a Link-aware Clustering Mechanism for WSNs in which the CHs are elected by evaluating the status of the nodes and the condition of links. The paper introduces a metric, called predicted transmission count (PTX), to evaluate the candidate conditions. Finally, the nodes with the highest priority are elected as CHs. The PTX represents the capability of a candidate for persistent transmission to a specific neighboring node. The work considers the transmit power, residual energy, and link quality to derive the PTX of CH candidates. A large PTX indicates a high chance of becoming a CH.

Energy-Efficient and Distance-based clustering (EEDC) is another approach proposed in Afsar and Tayarani-N (2014) in which two criteria are defined for CH election: *local competition* and *distance condition*. In the local competition criterion, the nodes compete with one another in a predefined range (R_{comp}) to select the nodes with the highest residual energy as the CH candidates (CCH). When a proper set of nodes is selected as the CCHs, the algorithm checks if the selected CCHs have enough distance to one another, so the CHs are distributed evenly across the network. The CCHs with a greater (or equal) distance than a threshold distance, D_{thr}, are selected as new CHs. Using this hybrid makes EEDC energy-efficient, distributed, and simple enough to be implemented in real systems. After CH selection, the CHs gather the data of the regular nodes, aggregate them and periodically send them to the BS in a multi-hop manner. In order to discover the paths in routing phase of EEDC, the BS broadcasts a "hello" message to all the nodes in the network. Having received this message, each CH calculates its distance to the BS, produces a cost proportional to its distance, adds this cost to the message and forwards them to all the nodes within its transmission range. When a CH receives this message from another CH, it checks if its cost to the received CH is less than the cost through the direct link to the BS or other previously received CHs. If yes, it selects the received CH as next-hop node, otherwise, ignores the message. When each pair of the nodes know their cost, some shortest paths algorithms, like bellman-ford, are used. Simulation results in the paper show that EEDC can well outperform basic clustering approaches (LEACH and HEED).

Discussion: As reviewed, most of recent clustering algorithms are deterministic (Ma et al., 2011; Wei, 2011; Kim et al., 2008a; Ran et al., 2010; Lee and Cheng, 2012b; Taheri et al., 2012; Mhemed et al., 2012; Anno et al., 2007; Siew et al., 2011; Tashtoush and Okour, 2008; Kulkarni and Venayagamoorthy, 2011; Selvakennedy et al., 2007; Mudundi and Ali, 2007; Seo et al., 2009; Kuila et al., 2013; Huruiala et al., 2010; Cheng et al., 2011; AbdelSalam and Olariu, 2012; Kamimura et al., 2006; Guru et al., 2005; Latiff et al., 2007; Bennani and El Ghanami, 2012; El Rhazi and Pierre, 2009; Chamam and Pierre, 2009; Banerjee and Khuller, 2001; Chan and Perrig, 2004; Fisher and Lipson, 1999; Muruganathan et al., 2005; Kim and Youn, 2005b; Yi et al., 2007; Li et al., 2005b; Chang and Kuo, 2006; Yu et al., 2007; Baek et al., 2010; Yanagihara et al., 2007; Boukerche et al., 2003; Dahnil et al., 2012; Yuea et al., 2012; Liao and Qi, 2013; Wang and Chen, 2013; Afsar and Tayarani-N, 2014). It means that the robust clustering is increasingly gaining more popularity than simple approaches (random). Deterministic approaches are energy-aware, robust, adaptive, and more reliable. However, they increase the overhead of the algorithm, in terms of the message exchange and time. Also, this group has a major problem: since the CHs are elected based on some constant metrics, the CH rotation is not performed as well as random approaches, typically known as the load-balancing problem. Thus, the total network lifetime is probably affected and diminished. The CH election in such approaches is performed based upon some local information of the nodes, such as the residual energy, proximity to neighbors, and node degree or umber of the neighbors. Considering the proximity or node degree as the metric in CH election, since the nodes are usually stationary, such metrics are constant until the death or movement of the neighbors (unwanted movements due to environmental causes, e.g. storm, are possible). Thus, the nodes with a high value of the metrics are elected as CHs for consequent rounds, so they die at a faster rate. Note that in the approaches that consider the residual energy beside other metrics, this problem is less intense, but still exists. Among the deterministic algorithms, fuzzy-logic- and heuristic-based methods, albeit, provide the optimal solutions in terms of the best CHs or size of the clusters, which significantly increase the overhead of the network (message and time). On the other hand, weight-based and compound clustering algorithms, which are executed in a distributed manner, are more suitable in many applications of WSNs. The iterative- and competition-based clustering algorithms' performance highly depends on the network diameter and packet loss. Among the deterministic methods, DWEHC, HCC, ACE, BCDCP, and EACLE are the most valuable protocols. In general, deterministic approaches miss the simplicity and produce more overhead compared to the random clustering algorithms. On the other hand, they present more reliability and robustness as they consider the nodes conditions. Finally, the main idea of PEACH, which is to eliminate the overhead of re-clustering, is interesting and introduces novel aspects in clustering.

4.1.3. Preset clustering algorithms

In addition to the mentioned approaches, a few algorithms consider the CHs or clusters that can be elected or pre-assigned before node deployment. We call these methods *preset* approaches. Basically, preset approaches have a major problem: since the clusters are formed based on the information loaded in the nodes before deployment, these approaches are not dynamic and do not consider the network and nodes conditions. Consequently, if the conditions of the network are inconsistent with the expected assumptions, it leads to a worse performance. Therefore, most of current protocols utilize dynamic cluster formation and preset approaches are limited to some specific-applications. In the following, some preset approaches are surveyed.

4.1.3.1. GS^3. A distributed algorithm for scalable self-configuration and self-healing (GS^3) multi-hop WSN is proposed in Zhang and Arora (2003). The algorithm enables network nodes in a 2D plane to configure themselves into a cellular hexagonal structure, where the cells are tightly bounded geographic radius and the overlap between neighboring cells is low. This structure is depicted in Fig. 14. In the paper, two types of nodes are defined: *big* and *small*. The big nodes are responsible for cell formation. One of the big nodes starts the clustering process by selecting the heads of neighboring cells which in turn select their neighbors and so on. The unselected members become cell members. Upon their selection cell heads relocate to the centers of their cells and start establishing their neighboring cells by selecting their heads. The process is repeated until no more cells could be added. In GS^3, it is assumed that the nodes are geography-aware. In organizing the nodes into cells, the authors say that in practice, a system may not be able to organize itself into cells with exactly the ideal radius R due to the discrete node distribution. However, the deviation of the actual radius from R still needs to be small enough, and be a function of node distribution density. GS^3 handles the scalability problem in WSNs. Also, the self-healing capabilities of GS^3 help it to manage the dynamic networks with mobile nodes.

4.1.3.2. Position-based aggregator node election protocol (PANEL). Another clustering approach is PANEL (Buttyan and Schaffer, 2007; Buttyán and Schaffer, 2010), the main objective of which is to support reliable and persistent data storage applications, like TinyPEDS (Girao et al., 2007). PANEL assumes that the sensor nodes are deployed in a bounded area partitioned into geographical clusters. Clustering is performed before the deployment of the network, and each sensor node is pre-loaded with the geographical information of the cluster to which it belongs. The operations in PANEL are divided into some epochs. At the beginning of each epoch, a reference point is computed in a distributed scheme. This computation can be performed by each node independently and locally. Afterwards, the closest node to the reference point is elected as CH (*aggregator*) and other nodes join the CH. PANEL uses a position-based routing method for inter-CH routing which is used for routing the messages from a distant aggregator towards the reference point of a given cluster. When a message is received by nodes in a particular cluster, the nodes route and forward the received message to the CH by intra-cluster communications. The authors believe that PANEL can be integrated with any position-based routing algorithm (in the paper, Greedy Perimeter Stateless Routing (GPSR), Karp and Kung, 2000, is used). The simulation results performed by the authors indicate that PANEL can outperform LEACH and HEED significantly. However, cluster formation in PANEL is complicated which might lead to a higher energy consumption.

4.1.3.3. Energy efficient deployment and cluster formation (EEDCF). In another work (Kaur and Baek, 2009), a grid-based approach for CH election (EEDCF) is proposed. Unlike conventional clustering algorithms, the paper considers three facts: the sensor nodes are not required to be deployed in a square field, some nodes can have more energy than other nodes (heterogeneous network), and the nodes can have different residual energy in different times according to their tasks. The approach has three main phases: the deployment, set-up, and steady-state phases. The nodes are deployed based on a grid-based methodology in which first some nodes with higher power and wider ranges are deployed, and then other regular nodes are located based on the location of the richer nodes. Each rich node in the center of grids plays the role of CH in the network and other regular nodes join the nearest CH. Afterwards, the data are gathered, aggregated and transmitted to the BS. The authors show that EEDCF improves the network lifetime. The assumptions in the paper limit the applicability of the approach, including the heterogeneity and pre-defined location of the nodes.

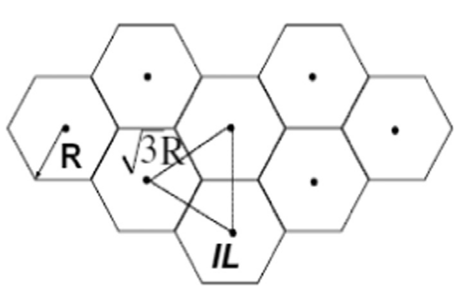

Fig. 14. Cellular hexagonal structure of the GS^3 protocol (redrawn from Zhang and Arora, 2003).

4.2. Unequal-sized clustering algorithms

Another clustering method is unequal one. Typically in unequal clustering, based on the distance between the nodes and the BS, the size of clusters is variable. That is, the closer clusters to the BS may be smaller than farther ones. The main reason is that the CHs located in the vicinity of the BS should relay more data than farther CHs, consequently the energy of them is dropped in a faster rate. This problem is typically known as the *hot spot* problem. To even the load, it is better that the closer clusters to the BS be smaller in size (it means that these clusters have a smaller number of members). The smaller the number of cluster members, the smaller the rate of intra-cluster energy consumption. Thus, such CHs could save more energy for relaying the data received from farther clusters. This is the basic idea behind all the unequal based approaches. This topic was first discussed by Soro and Heinzelman (2005). Similar to the equal-sized clustering approaches, the unequal-sized clustering approaches may be divided into probabilistic and deterministic, based on their CH selection algorithm. In the following, we first review the two major and initiative approaches proposed in Soro and Heinzelman (2005) and Li et al. (2005b), in detail, and then we move to other approaches with minor focus.

4.2.1. Preset unequal clustering algorithms

4.2.1.1. Unequal clustering scheme (UCS). The main idea of UCS (Soro and Heinzelman, 2005) is to form adaptive clusters based on their distance to the BS. Accordingly, the closer clusters to the BS are smaller than the farther ones. Fig 15 depicts the architecture of UCS. In UCS, it is assumed that the CHs are rich in the energy supply (heterogeneous network). And also, the CHs are located at the center of each cluster, so the method belongs to the preset clustering methods. For the sake of simplicity, the authors consider a pie shaped sense area at the center of which the BS is located. The formed clusters in the same layer have the same size and the CHs send the data to the BS via a two-hop path.

Compared to equal-sized clusters, UCS more fairly balances the load among the clusters. Also, UCS more extends the network lifetime compared to equal-sized clustering. However, some assumptions in UCS make it unpractical in real applications, like the CHs are located at the center of the clusters, the CHs are rich in power compared to other nodes and are located at predetermined locations.

4.2.2. Probabilistic unequal clustering algorithms

4.2.2.1. Energy-efficient unequal clustering mechanism (EEUC). Another major unequal clustering approach is EEUC (Li et al., 2005b; Chen et al., 2009) in which based on the distance to the BS an unequal clustering is proposed. In EEUC, the size of the clusters is set to be proportional to their distance to the BS, where the closer clusters to the BS are smaller than the farther ones. The assumptions in EEUC are more realistic than in UCS. For example, in this method, unlike the UCS, the BS is located outside of the field, and there is no need for the CHs to be located at the center of the clusters. In EEUC, the CHs are elected based on a competition. First some tentative CHs are elected from the regular nodes with a probability equal to T. In order to save more energy, the nodes that fail to be the tentative CHs, stay in the asleep mode until the CH election process finishes. Different competition ranges are used in order to achieve unequal clustering. A tentative CH is elected as final CH, only if it has greater residual energy than other nodes in its competition range. A greedy geographic and energy-aware routing protocol is also designed for the inter-cluster communication in UCR (Unequal Cluster-based Routing) (Chen et al., 2009) by the same authors, which considers the trade-off between the energy cost of relay paths and the residual energy of

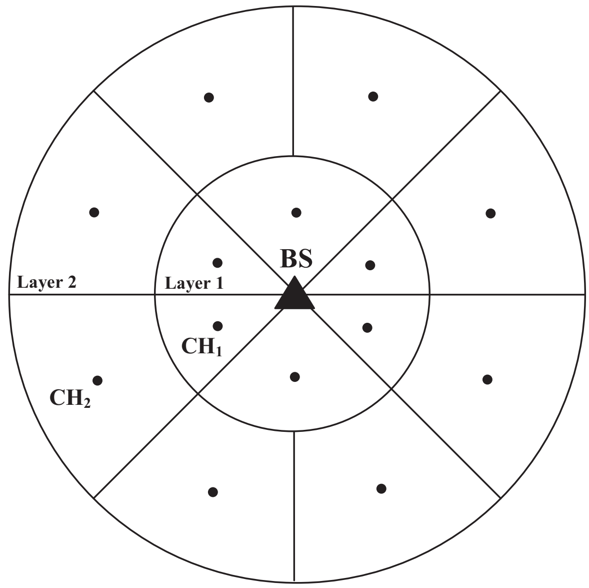

Fig. 15. Voronoi diagram of cluster formation in two layers around the BS.

relay nodes. EEUC uses the following equation for the competition ranges:

$$R_{comp} = \left(1 - c\frac{d_{max} - d_{i,BS}}{d_{max} - d_{min}}\right) R^0_{comp}, \quad (10)$$

where c is a constant between 0 and 1, d_{max} and d_{min} are the maximum and minimum distances to the BS, respectively, and R^0_{comp} is the initial cluster range used in the farther areas from the BS. In equation (10), c is the basic metric in changing the competition range. For example, if $c = 1/3$, R_{comp} varies from $2/3R^0_{comp}$ to R^0_{comp}. However, EEUC has some defects: firstly, defining the optimum value of c is not easy, specially in large-scale WSNs. Secondly, competition-based CH election increases the overhead of the network, particularly in larger competition ranges (farther areas from the BS). Thirdly, the introduced multi-hop routing scheme in Chen et al. (2009) is complicated, and more importantly, broadcasting the beacon messages by the CHs results in more energy consumption than conventional shortest path mlti-hop approaches. Fig 16 makes an overview on the EEUC architecture.

PRODUCE (Kim et al., 2008b) is another unequal clustering approach which uses a semi-centralized approach to form unequal clusters. Since the attenuation in wireless communications has direct relationship with the distance between the sender and receiver, the main idea behind PRODUCE is to keep the distance among the CHs smaller than a threshold (d_{cross}). First, the network is divided into some levels based on the node's distance to the BS, then the BS propagates the CH probabilities proportional to each level. The broadcast information via the BS includes the number of levels and the CH probability of the last layer (P_{cross}). The CH probability of each level varies between P_{cross} and P_{max} (the CH probability of the nodes in level 1). Having received these information, each node computes its probability. The CH probability in PRODUCE assures that the nodes located in the vicinity of the BS have a higher probability to become CH, on the other hand, this probability for farther nodes from the BS is smaller. As a result, the size of clusters becomes larger as the distance to the BS grows and the load is balanced among the CHs. PRODUCE suffers from the scalability problem, because dividing the network into levels and computing the CH probabilities is performed in a centralized manner.

Energy-Efficient Distributed Unequal Clustering (EEDUC) is proposed in Lee et al. (2008) which is an extension on EEUC. In this method, each sensor node sets a waiting time that is

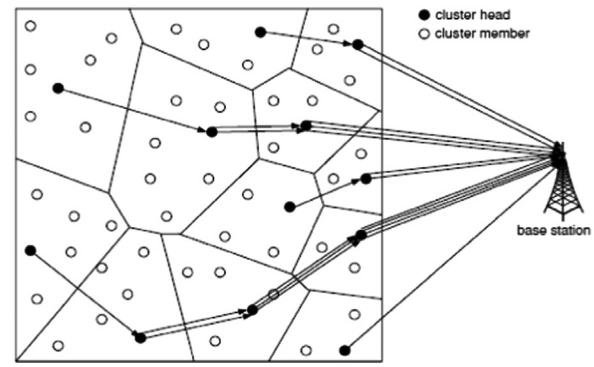

Fig. 16. An overview on the EEUC architecture (redrawn from Li et al., 2005b).

considered as a function of the number of neighboring nodes and a random number. EEDUC uses waiting time to distribute the CHs across the network. Similar to the previous mentioned approaches, in order to generate the unequal clusters, different competition ranges are applied. Similarly, unequal competition ranges are defined based upon nodes' residual energy and their distance to the BS.

Another unequal clustering is Energy-Balancing Unequal Clustering Protocol (EB-UCP) (Yang and Zhang, 2009) which uses a probability-based CH election where the closer nodes to the BS have higher probability to be elected as the CHs. Using this, the energy is more uniformly balanced among the nodes. In EB-UCP, the network is divided into some layers, and the nodes of each layer have their own probability of CH election. In general, the nodes in the layers closer to the BS have higher probabilities to be elected as CH. The probability of getting elected as tentative CHs is computed as

$$p_i = p_{min} + \frac{k-j}{k-1} \times (p_{max} - p_{min}), \quad (11)$$

where p_{max} and p_{min} are the CH probability of the first and k-th layer, respectively, and j is the layer of node i. Each node elects itself as a tentative CH based on Eq. (11). Afterwards, the tentative CHs with the higher residual energy are elected as new CHs. However, like PRODUCE, EB-UCP uses a centralized approach for dividing the network into some levels, so it has the scalability problem. Also, some important information about the system model used in this method have not been cleared in the paper, including time synchronization and mobility of the nodes.

An Energy-efficient Clustering (EC) solution is proposed in Wei et al. (2011b) which determines suitable cluster sizes depending on the hop distance to the BS, while achieving approximate equalization of node lifetimes and reduced energy consumption levels. The purpose of the EC algorithm is to determine a probability value, called p_i, to equalize and reduce energy consumption levels in the network. The energy equalization objective is to ensure that the network has similar lifetime values at different hop distances from the BS. In this method the probability of a sensor in region R_i becoming a CH is approximated as

$$p_i = \frac{1}{\pi r_i^2 \sigma} \rightarrow r_i = \sqrt{\frac{1}{\pi \sigma p_i}}, \quad (12)$$

where r_i is the radius of the circular sub-region, and σ is the node density. The paper uses an iterative-based approach to solve the corresponding value of p_i. As a conclusion, the authors show that the nodes that are closer to the BS, have higher probability p_i than the farther ones. Similar to other unequal clustering approaches, the closer clusters to the BS are smaller, and the network lifetime of all clusters is balanced. The paper also proposes a multi-hop data collection approach. EC is compared to HEED and UCR and experiments suggest that it outperforms them in terms of the network lifetime.

Location-based Unequal Clustering Algorithm (LUCA) (Lee et al., 2011) is another unequal clustering approach. In LUCA, each CH has a different cluster size based on its distance to the BS. In order to minimize the energy consumption of the network, LUCA forms the larger clusters farther from the BS. Initially, all the nodes set a back-off timer with a random value. The nodes that have not received any CH advertisement elect themselves as the CH and advertise their new status to their neighbors. LUCA assumes that the nodes have information about their distance to the BS through a GPS device, so the CHs can self-organize unequal-sized clusters regarding their distance to the BS. However, assuming the nodes are location-aware makes LUCA unpractical for many applications.

A distributed unequal clustering protocol, called Energy-Aware Distributed Unequal Clustering (EADUC), has been proposed in Yu et al. (2011). EADUC is designed to support both the homogeneous and heterogeneous networks. The cluster formation in the protocol is performed in three sub-phases: collecting neighbors information, CH competition, and cluster formation. Each node waits for a predefined time and if it receives no head announcement, it broadcasts its status as head to all the nodes within cluster range. In order to form unequal clusters, the competition range is varied. Uneven competition ranges are achieved based upon weighted function of the residual energy and the distance to the BS. However, this is unknown how the weighting factors are determined.

An unequal version of LEACH is found in Ren et al. (2010). Firstly, the BS broadcasts a distance matrix throughout the network such that all the nodes can receive it. The distance matrix contains the distance between each pair of the nodes in the network. The BS builds this matrix by broadcasting 'hello' message within the network and receiving the report messages from the nodes. Using this matrix the nodes can adjust their transmission power. Then the CHs are elected based on a modified version of LEACH CH election's probability. In the new probability, the residual energy and distance to the BS is included. However, constructing the distance matrix by the BS needs many message exchange and increases energy consumption.

Energy-Balancing unequal Clustering Approach for Gradient-based routing (EBCAG) (Liu et al., 2012) approach uses the hop-count to control the distribution of CHs. Each node maintains a gradient value that is defined as its minimum hop-count to the BS. Candidate CHs are randomly picked for each gradient value. The data gathered from the cluster members follow the direction of descending gradient to reach the BS. The size of a cluster is determined by the gradient value of its CH which is estimated based upon the received data from the nodes with a higher gradient. EBCAG has two problems: first, the CHs are elected randomly. Second, it is assumed that all the nodes of gradient i are within the communication range of all the nodes with gradient $i+1$, which is not generally the case.

An unequal clustering version of HEED is proposed in UHEED (Ever et al., 2012). In UHEED, the authors try to avoid death of the CHs that are closer to the BS. The main contribution of UHEED is to modify the competition range of the HEED protocol to achieve the unequal clusters. Different competition ranges are defined based on Eq. (10). UHEED conserves the limitations of HEED.

4.2.3. Deterministic unequal clustering algorithms

Multi-hop Routing Protocol with Unequal Clustering (MRPUC) is another unequal clustering (Gong et al., 2008). MRPUC uses three factors to balance the energy of nodes; first, it elects the nodes with greater residual energy as CHs, where clusters closer to the BS are smaller so the energy is preserved during intra-cluster communication. Second, when regular nodes join the clusters, in addition to the distance to the CH, they consider the residual energy of the CHs. And third, the CHs select the nodes with minimum energy consumption as relay nodes and maximum residual energy for forwarding the data. In order to vary the competition ranges, a modified version of Eq. (10) is used.

Partition Energy Balanced and Efficient Clustering Scheme (PEBECS) has been proposed in Wang et al. (2009). PEBECS divides the network into several equal-sized partitions, and each partition into some unequal clusters. As mentioned earlier in the previous unequal approaches, PEBECS have smaller clusters that are closer to the BS, and larger clusters in the farther areas from the BS. The CH election in PEBECS is performed by a heuristic algorithm which includes the node degree and location of the nodes as its main criteria for CH election.

Energy Aware Fuzzy Unequal Clustering (EAUCF) (Bagci and Yazici, 2010, 2013) uses a fuzzy-based approach to generate the unequal clusters. EAUCF adjusts the CH radius with respect to the residual energy and the distance to the BS. The work utilizes fuzzy-logic to handle the uncertainties in the CH radius estimation. First the tentative CHs are elected via a probabilistic approach as follows. Each node produces a random number between 0 and 1 and then compares it to a predefined threshold T. If the node finds its number less than T, it elects itself as a tentative CH. Then, the tentative CHs with higher residual energy are elected as the final CHs. The competition radius of each tentative CH in EAUCF changes dynamically, because EAUCF uses residual energy and distance to the BS to calculate the competition radius. Fuzzy input variables used in EAUCF are the distance of the nodes to the BS and the node residual energy.

4.2.3.1. Discussion. Although both equal-sized and unequal-sized clustering try to improve the network lifetime, the design challenge in these methods is basically different. In order to solve the hot spot problem, unequal clustering mechanisms attempt to form more clusters with smaller sizes in the vicinity of the BS. As mentioned earlier, in multi-hop sensor networks, this helps these CHs save some energy for relaying the received data from farther clusters to the BS. Accordingly, unequal protocols should speculate an approach to modify the uniform distribution of the CHs across the network. The main factor in unequal clustering is the proximity to the BS. Many approaches, such as PRODUCE, EB-UCP, and EBCAG, adopt to use a tracked or leveled network in which they try to form unequal clusters in each track or level. These methods often have the scalability problem, because their tracking is performed in a centralized manner (by the BS). Other approaches typically use different competition ranges in order to vary the cluster sizes, such as EEUC, EEDUC, and UHEED. In these methods, the nodes' information are used to vary the competition ranges, including the residual energy and the location of the node. However, the overhead of competition in these methods is high. Among the reviewed protocols, UCS, EEUC (UCR), EC, and EBCAG are the best unequal clustering designs.

5. Comparison

In this section, we compare the reviewed clustering approaches. We perform comparison in three main groups. We first list the clustering approaches that are the first attempts and present specific innovations in clustering, in Table 2. Then, since a significant number of clustering algorithms in WSNs are inspired from LEACH, we individually compare LEACH and its extensions, in Table 3. Finally, a comprehensive comparison on the presented clustering approaches is brought in an organized manner based on equal- and unequal-sized clustering, in Tables 4 and 5.

6. Conclusion and future work

WSNs are composed of hundreds or thousands of sensor nodes that are randomly dispersed in harsh environments. Due to the limited access to the nodes, self-organization and topology management are essential characteristics in these networks (Younis et al., 2014). More importantly, sensor nodes are drastically energy constrained so that preserving the energy is one of the most important challenges in WSNs. Researchers have proposed many algorithms to solve these challenges. One of the best and popular solutions is to cluster WSNs.

Clustering the nodes is a popular two-layered hierarchical architecture in which the regular nodes form the first layer, and the CHs constitute the second layer. In a clustered network, the CHs gather and aggregate the data from regular nodes. Then, these data are transmitted to the BS, usually through a multi-hop path among the CHs. Many objectives are pursued in clustering the nodes, including energy-efficiency, fault-tolerance, and topology management (see Fig. 2); nonetheless, the most important advantage of hierarchical clustering is that they distribute the management tasks among the CHs. In general, it is shown that a clustered WSN is more energy-efficient than non-clustered ones (Noori et al., 2011).

In this paper, we presented a comprehensive and state-of-the-art survey on the clustering approaches. First, we explained the clustering problem and its applications, objectives and characteristics. Then we reviewed and classified the most important clustering approaches and their extensions. In our classification, we divided the clustering algorithms into two major categories: equal-sized and unequal-sized clustering algorithms. We have limited ourselves to this general category, in order to comprehensively cover the clustering methods from the architecture point of view. Although both methods generally try to improve the lifespan of the network, the main design challenge in these methods is basically different. Equal-sized clustering has been designed to from equal-sized, well-distributed clusters with relatively minimum overlapping. Unequal-sized clustering methods, on the other hand, try to evenly balance the traffic load among all the CHs. Subsequently, each of these categories is divided into probabilistic, deterministic, and preset approaches, based on their CH election algorithms.

The probabilistic clustering approaches have gained the most popularity in WSNs, because of their advantages in simplicity, low overhead, high energy-efficiency, and fast convergence. We have divided probabilistic approaches into random and hybrid methods. LEACH is the most famous random clustering method that has arose many challenges for researchers. The best extensions on LEACH are TEEN, PEGASIS, EECS, and TL-LEACH. Table 3 provides a comparison between LEACH and its extensions. Although random approaches are simple with an overhead near optimal, they are not

Table 2
Comparison of the first attempts clustering approaches.

Protocol	Ref.	Innovation
LEACH	Heinzelman et al. (2000)	Random clustering
LEACH-C	Heinzelman et al. (2002)	Centralized clustering
TEEN	Manjeshwar and Agrawal (2001)	Reactive-based clustering
HCC	Banerjee and Khuller (2001)	Hierarchical (multi-layered) clustering
PEGASIS	Lindsey and Raghavendra (2002)	Chain-based clustering
GS^3	Zhang and Arora (2003)	Cellular hexagonal-based clustering
ACE	Chan and Perrig (2004)	Emergent-based clustering
FLOC	Demirbas et al. (2004)	Range-based clustering
HEED	Younis and Fahmy (2004a)	Hybrid clustering
DWEHC	Ding et al. (2005)	Weight-based clustering
Gupta et al.	Gupta and Sampalli (2005)	Fuzzy-logic-based clustering
UCS	Soro and Heinzelman (2005)	Unequal clustering
MOCA	Youssef et al. (2006)	Overlapping clustering
PEACH	Yi et al. (2007)	Adaptive/multi-level clustering

Table 3
Comparison of LEACH-based clustering protocols.

Protocol	Ref.	Method	Cluster count	Inter-cluster topology	CH election	Mobility	Location awareness	Load balancing	Node type	Nature
LEACH	Heinzelman et al. (2000)	Distributed	Variable	Direct	Random	Stationary	Not required	Good	Homogeneous	Proactive
LEACH-C	Heinzelman et al. (2002)	Centralized	Variable	Direct	Deterministic (by BS)	Stationary	Not required	Good	Homogeneous	Proactive
TEEN	Manjeshwar and Agrawal (2001)	Distributed	Constant	Multi-hop	Random	Stationary	Not required	Good	Homogeneous	Reactive
APTEEN	Manjeshwar and Agrawal (2002)	Centralized	Variable	Multi-hop	Deterministic (by BS)	Stationary	Not required	Moderate	Homogeneous	Reactive/ proactive
PEGASIS	Lindsey and Raghavendra (2002)	Distributed	Variable	Multi-hop	Random	Stationary	Required	Moderate	Homogeneous	Proactive
CCS	Jung et al. (2007)	Distributed	Variable	Multi-hop	Random	Stationary	Required	Not good	Homogeneous	Proactive
Deterministic	Handy et al. (2002)	Distributed	Variable	Direct	Hybrid	Stationary	Not required	Good	Homogeneous	Proactive
SEP	Smaragdakis et al. (2004)	Distributed	Variable	Direct	Hybrid	Stationary	Not required	Good	Heterogeneous	Proactive
TL-LEACH	Loscri et al. (2005)	Distributed	Variable	Multi-hop	Random	Stationary	Not required	Moderate	Homogeneous	Proactive
EECS	Li et al. (2005a)	Distributed	Variable	Direct	Hybrid	Stationary	Not required	Moderate	Homogeneous	Proactive
EDACH	Kim and Youn (2005a)	Distributed	Variable	Direct	Hybrid	Stationary	Not required	Good	Homogeneous	Proactive
CMEER	Kang et al. (2007)	Distributed	Variable	Multi-hop	Hybrid	Stationary	Not required	Moderate	Homogeneous	Proactive
TCCA	Selvakennedy and Sinnappan (2007)	Distributed	Variable	Multi-hop	Hybrid	Stationary	Not required	Good	Homogeneous	Proactive
LEACH-DT	Kang and Nguyen (2012)	Distributed	Variable	Multi-hop	Hybrid	Stationary	Not required	Good	Homogeneous	Proactive

Table 4
Comparison of equal-sized clustering algorithms.

Protocol	Cluster Properties				CH Properties					Clustering Process			
	Cluster size	Cluster count	Intra com.	Inter com.	Mobility	Node type	Role	Method	Objectives	CH Election	Alg. Complexity	Nature	Dynamism
LEACH (Heinzelman et al., 2000)	Equal	Variable	1-hop	1-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max Lifetime	Random	Constant	Proactive	Static
LEACH-C (Heinzelman et al., 2002)	Equal	Variable	1-hop	1-hop	Stationary	Homogeneous	Relay/ aggregation	Centralized	Max Lifetime	Deterministic (by BS)	Constant	Proactive	Static
CLUBS (Nagpal and Coore, 1998)	Equal	Constant	2-hop	*k*-hop	Movable	Homogeneous	Relay/ aggregation	Distributed	Management & scalability	Random	Variable	Proactive	Static
EEHC (Bandyopadhyay and Coyle, 2003)	Equal	Variable	*k*-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max lifetime	Random	Variable	Proactive	Static
FLOC (Demirbas et al., 2004)	Equal	Variable	2-hop	1-hop	Movable	Homogeneous	Relay/ aggregation	Distributed	Scalabiity & fault-tolerance	Random	Constant	Proactive	Static
MOCA (Youssef et al., 2006)	Equal	Variable	*k*-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Overlapping & connectivity	Random	Constant	Proactive	Static
CCN (Shigei et al., 2010)	Variable	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max Lifetime	Random	Constant	Proactive	Static
HEED (Younis and Fahmy, 2004a)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max Lifetime	Hybrid	Constant	Proactive	Static
ExHEED (Huang and Wu, 2005)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max Lifetime	Hybrid	Constant	Proactive	Static
CAWT (Wen and Sethares, 2005)	Equal	Variable	2-hop	*k*-hop	Stationary	Homogeneous	Aggregation	Distributed	Max lifetime & Connectivity	Hybrid	Constant	Proactive	Static
ACDA (Wen, 2013)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Aggregation	Centralized	Connectivity & stabilized topology	Hybrid	Constant	Proactive	Static
EEMC (Jin et al., 2008)	Equal	Variable	*k*-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max lifetime & reduced delay	Hybrid	Variable	Proactive	Static
DWEHC (Ding et al., 2005)	Equal	Variable	*k*-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max Lifetime	Weight-based	Constant	Proactive	Static
TASC (Virrankoski, 2005)	Equal	Variable	2-hop	*k*-hop	Stationary	Homogeneous	Aggregation	Distributed	Max lifetime	Weight-based	Variable	Proactive	Static
Spatial-clustering (Ma et al., 2011)	Equal	Variable	2-hop	N/A	N/A	Homogeneous	Aggregation	Distributed	Aggregation/fusion	Weight-based	Constant	Proactive	Static
EECABN (Wei, 2011)	Equal	Variable	1-hop	1-hop	Stationary	Homogeneous	Aggregation	Centralized	Max lifetime	Weight-based	Constant	Proactive	Static
Gupta and Sampalli (2005)	Equal	Variable	1-hop	1-hop	Stationary	Homogeneous	Relay/ aggregation	Centralized	Max lifetime	Fuzzy-based	Constant	Proactive	Static
HCC (Banerjee and Khuller, 2001)	Equal	Variable	*k*-hop	*k*-hop	Movable	Homogeneous	Relay/ aggregation	Distributed	Management & scalability	Compound	Variable	Proactive	Dynamic
ACE (Chan and Perrig, 2004)	Equal	Variable	*k*-hop	1-hop	Movable	Homogeneous	Relay/ aggregation	Distributed	Scalabiity & load balancing	Compound	Constant	Proactive	Static
BCDCP (Muruganathan et al., 2005)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Centralized	Max lifetime	Compound (by BS)	Constant	Proactive	Static
PEACH (Yi et al., 2007)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max lifetime	Compound	Variable	Proactive	Dynamic
EEDC (Yu et al., 2007)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max lifetime	Compound	Variable	Proactive	Dynamic
EACLE (Yanagihara et al., 2007)	Equal	Variable	2-hop	*k*-hop	Stationary	Homogeneous	Aggregation	Distributed	Connectivity & scalability	Compound	Variable	Proactive	Static
MBC (Deng et al., 2011)	Equal	Variable	1-hop	*k*-hop	Mobile	Homogeneous	Relay/ Aggregation	Distributed	Max lifetime & stabilized topology	Compound	Constant	Proactive	Dynamic
TCAC (Dahnil et al., 2012)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ Aggregation	Distributed	Max lifetime & connectivity	Compound	Constant	Proactive	Static
EEBCDA (Yuea et al., 2012)	Equal	Variable	1-hop	1-hop	Stationary	Homogeneous	Relay/ Aggregation	Distributed	Max lifetime	Compound	Constant	Proactive	Static
DSBCA (Liao and Qi, 2013)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Aggregation	Centralized	Max lifetime & load-balancing	Compound	Constant	Proactive	Static
LCM (Wang and Chen, 2013)	Equal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ Aggregation	Distributed	Max lifetime & reduced delay	Compound	Constant	Proactive	Static

EEDC (Afsar and Tayarani-N, 2014)	Equal	Variable	1-hop	k-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Max lifetime & fault-tolerance	Compound	Constant	Proactive	Static
GS^3 (Zhang and Arora, 2003)	Equal	Constant	k-hop	1-hop	Movable	Heterogeneous	Relay	Distributed	Scalability & fault-tolerance	Preset	Variable	Proactive	Dynamic
PANEL (Buttyán and Schaffer, 2010)	Equal	Constant	1-hop	k-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & fault-tolerance	Preset	Constant	Proactive	Static
EEDCF (Kaur and Baek, 2009)	Equal	Constant	1-hop	k-hop	Stationary	Heterogeneous	Relay/ aggregation	Distributed	Max lifetime	Preset	Constant	Proactive	Static

very reliable, in terms of the form of clusters and energy consideration in CH election. Furthermore, many traditional random-based clustering approaches use direct communication with the BS, instead of multi-hop communication. To address these shortcomings, hybrid methods have been employed in which a mixture of probability and some other metrics, including the residual energy or node degree, is utilized in order to form more balanced clusters. The most popular hybrid approach is HEED. Hybrid methods are usually iterative-based (Younis and Fahmy, 2004a), timer-based (Selvakennedy and Sinnappan, 2007) or competition-based (Ye et al., 2005). This makes the clustering algorithm more complicated compared to the random approaches, in terms of message and time complexity. Therefore, this depends on the application to tell which approach is suitable. We believe that the literature lacks a study that evaluates the extra overhead incurred by hybrid approaches and compares it with the amount of gained energy. Also, an important problem in previously published works is that most of them have not considered the dissipated energy during CH election and cluster formation. This makes the simulation less accurate in evaluating the performance.

On the other hand, deterministic clustering algorithms use a non-probabilistic method and certain metrics to elect the CHs. We have divided this group into four sub-groups: weight-based, fuzzy-logic-based, heuristic-based, and compound approaches. The most important advantage of deterministic methods is that they are more reliable than probabilistic approaches as the size and number of clusters, the energy level of the CHs, the location of the CHs and nodes, etc. are controlled. However, this group misses the simplicity as some of them have a slow convergence time, and some are not able to be implemented in large-scales. More precisely, weight-based protocols are typically iterative-based (Basagni, 1999; Ding et al., 2005) that significantly increases the message complexity. This is because in each iteration the nodes have to exchange the status messages. This is the case for fuzzy-logic-based approaches, where they consume a lot of energy for message exchange or executing the algorithm. Also, heuristic-based methods, since they need to have the global topology of the network and require to be implemented by the BS. The centralized and slow nature makes heuristic methods unpractical for many applications. As a result, there is a trade-off between provisioning simplicity and Quality of Service (QoS) requirements of the network. Typically, achieving one of these requirements is concurrent with losing another one. Overall, this is the application that determines what approach is suitable. A few number of research have used preset method for CH election and node deployment, including GS^3 and PANEL. This type of clustering is not practical for many applications of WSNs, where a dynamic method is required to handle network conditions.

According to the reviewed literature and presented tables and discussions, the design of an appropriate clustering protocol depends a great deal on the application, and in particular, user's requirements. In probabilistic random clustering methods, which are sufficiently simple and fast are probably the most match methods for WSNs with a large number of nodes, the primary objective is the energy-efficiency. An ideal application for these methods could be the environmental monitoring in which the user needs a periodic feedback from the play field. On the other hand, those applications that need a reliable and certain response or more robustness can employ deterministic algorithms. In small-scale specific-applications, such as high-tech applications, that need an optimal solution, utilizing heuristic-based clustering algorithms could be the best option. Furthermore, the traffic load should be considered in the design level, specially in large-scales applications. That is, the clustering protocol should support different size of clusters regarding the *command* node or BS location.

Table 5
Comparison of unequal-sized clustering protocols.

Protocol	Cluster Properties				CH Properties					Clustering Process			
	Cluster size	Cluster count	Intra com.	Inter com.	Mobility	Node type	Role	Method	Objectives	CH Election	Alg. Complexity	Nature	Dynamism
UCS (Soro and Heinzelman, 2005)	Unequal	Variable	1-hop	*k*-hop	Stationary	Heterogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Preset	Constant	Proactive	Static
EEUC (Li et al., 2005b)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
PRODUCE (Kim et al., 2008b)	Unequal	Constant	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Semi-centralized	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
EEDUC (Lee et al., 2008)	Unequal	Variable	1-hop	*k*-hop	N/A	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
EB-UCP (Yang and Zhang, 2009)	Unequal	Variable	1-hop	*k*-hop	N/A	Homogeneous	Relay/ aggregation	Semi-centralized	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
EC (Wei et al., 2011b)	Unequal	Variable	1-hop	*k*-hop	N/A	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
LUCA (Lee et al., 2011)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Random	Constant	Proactive	Static
EADUC (Yu et al., 2011)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
Unequal LEACH (Ren et al., 2010)	Unequal	Variable	1-hop	1-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
EBCAG (Liu et al., 2012)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Random	Constant	Proactive	Static
UHEED (Ever et al., 2012)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Hybrid	Constant	Proactive	Static
MRPUC (Gong et al., 2008)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Deterministic	Constant	Proactive	Static
PEBECS (Wang et al., 2009)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Heuristic-based	Constant	Proactive	Static
EAUCF (Bagci and Yazici, 2010)	Unequal	Variable	1-hop	*k*-hop	Stationary	Homogeneous	Relay/ aggregation	Distributed	Load balancing & max lifetime	Fuzzy-based	Constant	Proactive	Static

Although clustering has extensively been explored by researchers in different areas, some aspects of clustering are not properly investigated yet. Here we propose some areas for future work.

First, most of existing clustering approaches are static so they do not have the ability to adapt to the network changes. By *static* we mean that the clustering approach adopts some constant assumptions about the network, and re-clusters the network at fixed periods, without considering the nodes and environment conditions. The approach can significantly improve its performance by considering the nodes conditions. Selecting *active* nodes according to the phenomena and keeping other nodes asleep makes the approach more energy-efficient. One of the major reviewed approaches in this area is Yu et al. (2007). Selecting active nodes could be either as simple as Ye et al. (2003) in which the nodes become active if there is no active node within their probing range or more complicated like Soro and Heinzelman (2009) with the objective of fully covering the entire network. In addition to selecting active nodes, reducing the overhead of re-clustering in current clustering methods is an interesting challenge, that is introduced by PEACH. Most of existing methods consider a fixed re-clustering epoch, while wisely and adaptively adjusting these epochs can improve the energy-efficiency. Furthermore, considering the energy level of the CHs before upcoming round and then re-clustering periods (Kim and Youn, 2005a) is an effective method to increase the reliability. Therefore, dynamic clustering can be more investigated in the future.

Second open research challenge in this area is to investigate the effect of mobility in the network. According to the architecture of clustering approaches, three parts of a network can be mobile: the regular nodes, the CHs, and the BS. There is a small number of research in this area. TTDD (Luo et al., 2005), for instance, studies the effect of mobile BS in a two-tier network, while other nodes in the network are stationary. In another valuable analysis, Lotfinezhad et al. (2008) study the effect of a mobile BS in cluster-based data collection. FLOC (Demirbas et al., 2006) supports the mobility of the regular nodes and (Deng et al., 2011) selects the CHs regarding the mobility of the nodes. The main challenge in the networks supporting the mobility is that the protocol should be able to handle the overhead of the node mobility and topology changes in the network. This overhead is incurred by periodic updates of the nodes locations. Also, estimating a stable time for link establishment and cluster formation in mobile networks is so challenging. WSNs with mobile nodes have many applications, including robotics, ecology, and battlefields (Arampatzis et al., 2005).

Sensor networks are naturally data-centric (Al-Karaki and Kamal, 2004); thus providing schemes that support the data-centric routing seems essential in these networks. Apart from a limited number of published works in the area of reactive networks, such as Manjeshwar and Agrawal (2001, 2002), Xu and Qi (2004), Guo and Li (2007), almost all other clustering methods have been designed for proactive networks. Again, many applications of WSNs need a reactive method. A good illustration of that can be military applications in which the nodes should detect an intrusion. In this type of applications, the nodes send their data to the BS only when an object is detected. Thus, in the remaining of the network time, the nodes can stay in the low power mode so that the energy is conserved more. Therefore, designing clustering methods for reactive networks should be more considered in the future.

Heuristic-based clustering approaches need to be more investigated by researchers. Although heuristic-based methods are time consuming and centralized, some applications need an optimal solution before being deployed to reduce the costs. For example, in small scales heterogeneous networks, defining the exact location of rich CHs (in terms of power supply) is essential, so the best performance in terms of energy consumption, connectivity, etc. can be achieved. Most of heuristic-based clustering methods utilize PSO to achieve the best results. Employing other types of optimization algorithms is an open research direction.

The main plan of most existing clustering algorithms is to prolong the network lifetime; however, clustering can be used in other network challenges. Meeting the QoS requirements of a WSN is another open challenge in these networks. For example, cluster-based protocols are exposed to different types of attacks, including *Hello* flood, Sybel, etc. (Karlof and Wagner, 2003). Link-layer encryption and authentication can be considered in the design level and the CHs can perform the security protocols and data acquisition. Some services make the protocols fault-tolerant (Tai et al., 2004) which can be executed by the CHs. Another QoS requirement that is full-coverage of the network can be promised and pursued in cluster-based protocols (Soro and Heinzelman, 2009).

Finally, most recently, energy-harvesting sensor networks have gained more attention. In energy-harvesting WSNs, the nodes are equipped with some harvesters from the environment, such as solar and wind. Harvester nodes with a higher energy rate can be placed in the network as relay nodes (Zhang et al., 2013), or they directly play the role of the CH (Thiemo Voigt et al., 2004) in clustered networks. In such networks, the CH election metrics that are used in non-harvesting networks are not appropriate and this can be an interesting research challenge.

Acknowledgments

The authors would like to thank the reviewers for their helpful comments. The first author would also like to thank Dr. Mohamed Younis for his constructional comments and suggestions.

References

Abbasi A, Younis M. A survey on clustering algorithms for wireless sensor networks. Comput Commun 2007;30:2826–41.
AbdelSalam HS, Olariu S. Bees: bioinspired backbone selection in wireless sensor networks. IEEE Trans Parallel Distrib Syst 2012;23:44–51.
Afsar MM, Tayarani-N M-H. A novel energy-efficient and distance-based clustering approach for wireless sensor networks. Adv Intell Syst Comput 2014;223:177–86 (Springer).
Akyildiz I, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Commun Mag 2002;40:102–14.
Akyildiz IF, Melodia T, Chowdhury KR. A survey on wireless multimedia sensor networks. Comput Netw 2007;51:921–60.
Al-Karaki JN, Kamal AE. Routing techniques in wireless sensor networks: a survey. IEEE Wirel Commun 2004;11:6–28.
Amis AD, Prakash R, Vuong TH, Huynh DT. Max–min d-cluster formation in wireless ad hoc networks. In: IEEE conference on communications (INFOCOM), vol. 1, Tel Aviv, Israel; 2000. p. 32–41.
Anastasi G, Conti M, Di Francesco M, Passarella A. Energy conservation in wireless sensor networks: a survey. Ad Hoc Netw 2009;7:537–68.
Anno J, Barolli L, Xhafa F, Durresi A. A cluster head selection method for wireless sensor networks based on fuzzy logic. In: IEEE TENCON conference, Taipei, Taiwan; 2007. p. 1–4.
Arabi Z, Khodaei Y. Herf: a hybrid energy efficient routing using a fuzzy method in wireless sensor networks. Int J Distrib Parallel Syst 2010;1.
Arampatzis T, Lygeros J, Manesis S. A survey of applications of wireless sensors and wireless sensor networks. In: IEEE international symposium on control and automation, Limassol, Cyprus; 2005. p. 719–24.
Arboleda L, Nasser N. Comparison of clustering algorithms and protocols for wireless sensor networks. In: IEEE Canadian conference on electrical and computer engineering (CCECE); 2006. p. 1787–92.
Aslam M, Javaid N, Rahim A, Nazir U, Bibi A, Khan ZA. Survey of extended leach-based clustering routing protocols for wireless sensor networks. In: International conference on high performance computing and communication (HPCC), Liverpool, UK; 2012. p. 1232–8.
Baek J, An SK, Fisher P. Dynamic cluster header selection and conditional re-clustering for wireless sensor networks. IEEE Trans Consum Electron 2010;56:2249–57.
Bagci H, Yazici A. An energy aware fuzzy unequal clustering algorithm for wireless sensor networks. In: IEEE conference on fuzzy systems (FUZZ), Barcelona, Spain; 2010. p. 1–8.
Bagci H, Yazici A. An energy aware fuzzy approach to unequal clustering in wireless sensor networks. Appl Soft Comput 2013;13:1741–9.
Bandyopadhyay S, Coyle E. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In: IEEE conference on communications (INFOCOM), California, USA; 2003. p. 1713–23.

Banerjee S, Khuller S. A clustering scheme for hierarchical control in multi-hop wireless networks. In: IEEE conference on communications (INFOCOM), vol. 2, Anchorage, Alaska, USAl; 2001. p. 1028–37.
Basagni S. Distributed clustering for ad hoc networks. In: International symposium on parallel architectures, algorithms, and networks (I-SPAN), Fremantle, Australia; 1999. p. 310–5.
Bennani K, El Ghanami D. Particle swarm optimization based clustering in wireless sensor networks: the effectiveness of distance altering. In: International conference on complex systems (ICCS), Omaha, Nebraska; 2012. p. 1–4.
Biradar S, Sawant R, Mudholkar R, Patil V. A survey of hierarchical routing protocols in wireless sensor network. Int J Comput Sci Issues 2011;8.
Boukerche A, Cheng X, Linus J. Energy-aware data-centric routing in microsensor networks. In: ACM conference on modeling analysis and simulation of wireless and mobile systems (MSWiM), San Diego, CA, USA; 2003. p. 42–9.
Bredin JL, Demaine ED, Hajiaghayi MT, Rus D. Deploying sensor networks with guaranteed fault tolerance. IEEE/ACM Trans Netw 2010;18:216–28.
Buttyan L, Schaffer P. Panel: position-based aggregator node election in wireless sensor networks. In: IEEE conference on mobile adhoc and sensor systems (MASS), Pisa, Italy; 2007. p. 1–9.
Chamam A, Pierre S. On the planning of wireless sensor networks: energy-efficient clustering under the joint routing and coverage constraint. IEEE Trans Mob Comput 2009;8:1077–86.
Chan H, Perrig A. Ace: an emergent algorithm for highly uniform cluster formation. In: European conference on wireless sensor networks (EWSN), Berlin, Germany; 2004. p. 154–71.
Chang R-S, Kuo C-J. An energy efficient routing mechanism for wireless sensor networks. In: IEEE conference on advanced information networking and applications (AINA), Vienna, Austria; 2006. p. 308–12.
Chen B, Jamieson K, Balakrishnan H, Morris R. Span: an energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks. Wirel Netw 2002;8:481–94.
Chen G, Li C, Ye M, Wu J. An unequal cluster-based routing protocol in wireless sensor networks. Wirel Netw 2009;15:193–207.
Cheng C, Tse C, Lau F. A clustering algorithm for wireless sensor networks based on social insect colonies. IEEE Sens J 2011;11:711–21.
Cui X. Research and improvement of leach protocol in wireless sensor networks. In: International symposium on microwave, antenna, propagation and EMC technologies for wireless communications (MAPE), Hangzhou, China; 2007. p. 251–4.
Dahnil DP, Singh YP, Ho C-K. Topology-controlled adaptive clustering for uniformity and increased lifetime in wireless sensor networks. IET Wirel Sens Syst 2012;2:318–27.
Dai F, Wu J. On constructing k-connected k-dominating set in wireless networks. In: IEEE parallel and distributed processing symposium (IPDPS), Denver, Colorado, USA; 2005. p. 810–9.
Demirbas M, Arora A, Mittal V. Floc: a fast local clustering service for wireless sensor networks. In: Workshop on dependability issues in wireless ad hoc networks and sensor networks (DIWANS/DSN), Florence, Italy; 2004. p. 1–6.
Demirbas M, Arora A, Mittal V, Kulathumani V. A fault-local self-stabilizing clustering service for wireless ad hoc networks. IEEE Trans Parallel Distrib Syst 2006;17:912–22.
Deng S, Li J, Shen L. Mobility-based clustering protocol for wireless sensor networks with mobile nodes. IET Wirel Sens Syst 2011;1:39–47.
Deosarkar B, Yadav N, Yadav R. Clusterhead selection in clustering algorithms for wireless sensor networks: a survey. In: International conference on computing, communication and networking (ICCCN), St. Thomas U.S. Virgin Islands; 2008. p. 1–8.
Ding P, Holliday J, Celik A. Distributed energy-efficient hierarchical clustering for wireless sensor networks. In: International conference on distributed computing in sensor systems (DCOSS), Los Angeles, USA; 2005. p. 322–39.
Dong Q, Dargie W. A survey on mobility and mobility-aware mac protocols in wireless sensor networks. IEEE Commun Surv Tutor 2013;15:88–100.
Ehsan S, Hamdaoui B. A survey on energy-efficient routing techniques with qos assurances for wireless multimedia sensor networks. IEEE Commun Surv Tutor 2012;14:265–78.
El Rhazi A, Pierre S. A tabu search algorithm for cluster building in wireless sensor networks. IEEE Trans Mob Comput 2009;8:433–44.
KH-PSJXX Ester M. A density-based algorithm for discovering clusters in large spatial databases with noise. In: International conference on knowledge discovery and data mining (KDD); 1996. p. 226–31.
Ever, Enver, Luchmun, Ravi, Mostarda, Leonardo, Navarra, Alfredo, Shah, Purav. UHEED - an unequal clustering algorithm for wireless sensor networks. In: Sensornets, Rome, Italy 2012;2012.
Fisher DA, Lipson HF. A new method for enhancing survivability in unbounded systems. In: Hawaii international conference on system sciences (HICSS), Maui, Hawaii, USA; 1999.
Forster A, Murphy AL. Clique: role-free clustering with q-learning for wireless sensor networks. In: IEEE conference on distributed computing systems (ICDCS), Montreal, Canada; 2009. p. 441–9.
Girao J, Westhoff D, Mykletun E, Araki T. Tinypeds: tiny persistent encrypted data storage in asynchronous wireless sensor networks. Ad Hoc Netw 2007:1073–89.
Gong B, Li L, Wang S, Zhou X. Multihop routing protocol with unequal clustering for wireless sensor networks. In: International colloquium on computing, communication, control, and management (CCCM), vol. 2, Guangzhou, China; 2008. p. 552–6.
Guo Bin, Li Zhe. A dynamic-clustering reactive routing algorithm for wireless sensor networks. Wirel Netw 2009;15(4):423–30.
Gupta D, Sampalli S. Cluster-head election using fuzzy logic for wireless sensor networks. In: IEEE conference on communication networks and services research (CNSR), Halifax, Novia Scotia, Canada; 2005. p. 255–60.
Guru S, Halgamuge S, Fernando S. Particle swarm optimisers for cluster formation in wireless sensor networks. In: International conference on intelligent sensors, sensor networks and information processing (ISSNIP), Melborn, Australia; 2005. p. 319–24.
Hajiaghayi M, Immorlica N, Mirrokni VS. Power optimization in fault-tolerant topology control algorithms for wireless multi-hop networks. In: ACM conference on mobile computing and networking (MobiCom), San Diego, California, USA; 2003. p. 300–12.
Handy M, Haase M, Timmermann D. Low energy adaptive clustering hierarchy with deterministic cluster-head selection. In: International workshop on mobile and wireless communications network (MWCN), Stockholm, Sweden; 2002. p. 368–72.
Heinzelman WB, Chandrakasan AP, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In: Hawaii international conference on system sciences (HICSS), Wailea Maui, Hawaii; 2000. p. 10–19.
Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans Wirel Commun 2002;1:660–70.
Huang, H., Wu, J., 2005. A probabilistic clustering algorithm in wireless sensor networks. In: IEEE vehicular technology conference (VTC), vol. 62, Dallas, TX, USA; 2005. p. 1796.
Huruiala P-C, Urzica A, Gheorghe L. Hierarchical routing protocol based on evolutionary algorithms for wireless sensor networks. In: International conference on Romanian educational network (RoEduNet), Sibiu, Romania; 2010. p. 387–92.
Jiang C, Yuan D, Zhao Y. Towards clustering algorithms in wireless sensor networks-a survey. In: IEEE wireless communications and networking conference (WCNC); 2009. p. 1–6.
Jindal P, Gupta V. Study of energy efficient routing protocols of wireless sensor networks and their further researches: a survey. J Comput Sci Commun Eng 2013;2.
Jin Y, Wang Y, Kim L, Yang X. Eemc: An energy-efficient multi-level clustering algorithm for large-scale wireless sensor networks. Comput Netw 2008;52:542–62.
Joshi A, Lakshmi Priya M. A survey of hierarchical routing protocols in wireless sensor network. MES J Technol Manag 2011:67–71.
Jung S, Han Y, Chung T. The concentric clustering scheme for efficient energy consumption in the pegasis. In: Conference on advanced communication technology, Gangwon-Do; 2007. p. 260–5.
Kamimura J, Wakamiya N, Murata M. A distributed clustering method for energy-efficient data gathering in sensor networks. Int J Wirel Mob Comput 2006;1:113–20.
Kang Sang H, Nguyen Thinh. Distance based thresholds for cluster head selection in wireless sensor networks. IEEE Commun Lett 2012;16(9):1396–9.
Kang T, Yun J, Lee H, Lee I, Kim H, Lee B, et al. A clustering method for energy efficient routing in wireless sensor networks. In: World scientific and engineering academy and society (WSEAS); 2007. p. 133–8.
Karlof C, Wagner D. Secure routing in wireless sensor networks: attacks and countermeasures. Ad Hoc Netw 2003;1:293–315.
Karp B, Kung HT. Gpsr: greedy perimeter stateless routing for wireless networks. In: ACM conf on mobile computing and networking (MobiCom), Massachusetts, USA; 2000. p. 243–54.
Kaur T, Baek J. A strategic deployment and cluster-header selection for wireless sensor networks. IEEE Trans Consum Electron 2009;55:1890–7.
Kim KT, Youn HY. Energy-driven adaptive clustering hierarchy (edach) for wireless sensor networks. In: International conference on embedded and ubiquitous computing (EUC), Nagasaki, Japan; 2005a. p. 1098–107.
Kim K, Youn H. Peach: proxy-enable adaptive clustering hierarchy for wireless sensor network. In: International conference on wireless network (ICWN), Las Vegas, Nevada, USA; 2005b. p. 52–7.
Kim J-M, Park S-H, Han Y-J, Chung T-M. Chef: cluster head election mechanism using fuzzy logic in wireless sensor networks. In: International conference on advanced communication technology (ICACT), vol. 1; 2008a. p. 654–9.
Kim J, Sajjad Hussain C, Yang W, Kim D, Park M. Produce: a probability-driven unequal clustering mechanism for wireless sensor networks. In: IEEE conference on advanced information networking and applications (AINA), Okinawa, Japan; 2008b. p. 928–33.
Kleinrock L, Kamoun F. Hierarchical routing for large networks, performance evaluation and optimization. Comput Netw 1977;1:155–74.
Krishnamachari L, Estrin D, Wicker S. The impact of data aggregation in wireless sensor networks. In: International conference on distributed computing systems (ICDCS), Vienna, Austria; 2002. p. 575–8.
Kuila P, Gupta SK, Jana PK. A novel evolutionary approach for load balanced clustering problem for wireless sensor networks. Swarm Evolution Comput 2013;12:48–56.
Kulkarni RV, Venayagamoorthy GK. Particle swarm optimization in wireless-sensor networks: a brief survey. IEEE Trans Syst, Man, Cybern, Part C: Appl Rev 2011;41:262–7.
Kumarawadu P, Dechene DJ, Luccini M, Sauer A. Algorithms for node clustering in wireless sensor networks: a survey. In: International conference on information and automation for sustainability (ICIAFS); 2008. p. 295–300.
Kumar P, Oulia M, Sadeqi Ghassami H. Survey of clustering algorithms using fuzzy logic in wireless sensor network. In: International conference on energy efficient technologies for sustainability (ICEETS), Nagercoil, India; 2013. p. 924–8.

Kwon TJ, Gerla M. Efficient flooding with passive clustering (pc) in ad hoc networks. SIGCOMM Computut Commun Rev 2002;32:44–56.
Latiff NA, Tsimenidis CC, Sharif BS. Energy-aware clustering for wireless sensor networks using particle swarm optimization. In: IEEE conference on personal, indoor and mobile radio communications (PIMRC), Athens, Greece; 2007. p. 1–5.
Lee J-S, Cheng W-L. Fuzzy-logic-based clustering approach for wireless sensor networks using energy predication. IEEE Sens 2012a;12.
Lee J, Cheng W. Fuzzy-logic-based clustering approach for wireless sensor networks using energy predication. IEEE Sens J 2012b;12:2891–7.
Lee S, Lee J, Sin H, Yoo S, Lee S, Lee J, Lee Y, Kim S. An energy-efficient distributed unequal clustering protocol for wireless sensor networks. World Acad Sci, Eng Technol 2008;48:443–7.
Lee S, Choe H, Park B, Song Y, Kim C. Luca: an energy-efficient unequal clustering algorithm using location information for wireless sensor networks. Wirel Pers Commun 2011;56:715–31.
Levente Buttyán, Péter Schaffer. Position-Based aggregator node election in wireless sensor networks, Int J Distrib Sen Netw 2010;2010:15. (Article ID 679205), http://dx.doi.org/10.1155/2010/679205.
Liao HLW, Qi Y. Load-balanced clustering algorithm with distributed self-organization for wireless sensor networks. IEEE Sens J 2013;13:1498–506.
Li C, Ye M, Chen G. An energy-efficient unequal clustering mechanism for wireless sensor networks. In: IEEE conference on mobile adhoc and sensor systems (MASS), Washington, USA; 2005a. p. 3–17.
Li C, Ye M, Chen G, Wu J. An energy-efficient unequal clustering mechanism for wireless sensor networks. In: IEEE conference on mobile adhoc and sensor systems (MASS), Washington, USA; 2005b. p. 8–15.
Lindsey S, Raghavendra CS. PEGASIS: power efficient gathering in sensor information systems. In: IEEE Aerospace Conference; 2002. p. 1125–30.
Lindsey S, Raghavendra C, Sivalingam K. Data gathering algorithms in sensor networks using energy metrics. IEEE Trans Parallel Distrib Syst 2002;13:924–35.
Liu X. A survey on clustering routing protocols in wireless sensor networks. Sensors 2012;12:11123–53.
Liu T, Li Q, Liang P. An energy-balancing clustering approach for gradient-based routing in wireless sensor networks. Comput Commun 2012;35:2150–61.
Loscri V, Morabito G, Marano S. A two-level hierarchy for low-energy adaptive clustering hierarchy. In: IEEE vihicular technology conference (VTC), Dallas, TX, USA; 2005. p. 1809–13.
Lotf J, Hosseinzadeh M, Alguliev RM. Hierarchical routing in wireless sensor networks: a survey. In: International conference on computer engineering and technology (ICCET), vol. 3, Chengdu, China; 2010. p. 646–50.
Lotfinezhad M, Liang B, Sousa E. Adaptive cluster-based data collection in sensor networks with direct sink access. IEEE Trans Mob Comput 2008;7:884–97.
Luo H, Ye F, Cheng J, Lu S, Zhang L. Ttdd: two-tier data dissemination in large-scale wireless sensor networks. Wirel Netw 2005;11:161–75.
Ma Y, Guo Y, Tian X, Ghanem M. Distributed clustering-based aggregation algorithm for spatial correlated sensor networks. IEEE Sens J 2011;11:641–8.
Moufida Maimour, Houda Zeghilet, Francis Lepage. Cluster-based routing protocols for energy-efficiency in wireless sensor networks, In: Yen Kheng Tan (Ed.), sustainable wireless sensor networks, InTech 2010; ISBN: 978-953-307-297-5 http://dx.doi.org/10.5772/13274.
Mamalis B, Gavalas D, Konstantopoulos C, Pantziou G. Clustering in wireless sensor networks. Zhang/RFID, Sens Netw 2009:324–53.
Manjeshwar A, Agrawal D. Teen: a routing protocol for enhanced efficiency in wireless sensor networks. In: IEEE parallel and distributed processing symposium (IPDPS), San Francisco, USA; 2001. p. 2009–15.
Manjeshwar A, Agrawal D. Apteen: a hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In: IEEE parallel and distributed processing symposium (IPDPS), Florida, USA; 2002. p. 195–202.
Mhatre V, Rosenberg C. Homogeneous vs heterogeneous clustered sensor networks: a comparative study. In: IEEE international conference on communications (ICC), vol. 6, Paris, France; 2004. p. 3646–51.
Mhemed R, Aslam N, Phillips W, Comeau F. An energy efficient fuzzy logic cluster formation protocol in wireless sensor networks. Proced Comput Sci 2012;10:255–62.
Mudundi S, Ali HH. A new robust genetic algorithm for dynamic cluster formation in wireless sensor networks. In: Proceedings of the wireless and optical communications, Montreal, Quebec, Canada; 2007.
Muruganathan SD, Ma DC, Bhasin RI, Fapojuwo AO. A centralized energy-efficient routing protocol for wireless sensor networks. IEEE Commun Mag 2005;43: S8–S13.
Nagpal R, Coore D. An algorithm for group formation in an amorphous computer. In: International conference on parallel and distributed computing systems (PDCS), Louisville, Kentucky, USA; 1998.
Noori M, Ardakani M. Lifetime analysis of random event-driven clustered wireless sensor networks. IEEE Trans Mob Comput 2011;10:1448–58.
Pantazis NA, Vergados DD. A survey on power control issues in wireless sensor networks. IEEE Commun Surv Tutor 2007;9:86–107.
Rajagopalan R, Varshney P. Data-aggregation techniques in sensor networks: a survey. IEEE Commn Surv Tutor 2006;8:48–63.
Ramesh K, Somasundaram K. A comparative study of clusterhead selection algorithms in wireless sensor networks. Int J Comput Sci Eng Surv 2011;2.
Ran G, Zhang H, Gong S. Improving on leach protocol of wireless sensor networks using fuzzy logic. J Inf Comput Sci 2010;7:767–75.
Ren P, Qian J, Li L, Zhao Z, Li X. Unequal clustering scheme based leach for wireless sensor networks. In: International conference on genetic and evolutionary computing (ICGEC), Shenzhen, China; 2010. p. 90–3.
Selvakennedy S, Sinnappan S. An adaptive data dissemination strategy for wireless sensor networks. J Distrib Sens Netw 2007;3.
Selvakennedy S, Sinnappan S, Shang Y. A biologically inspired clustering protocol for wireless sensor networks. Comput Commun 2007;30:2786–801.
Seo H-S, Oh S-J, Lee C-W. Evolutionary genetic algorithm for efficient clustering of wireless sensor networks. In: IEEE consumer communications and networking conference (CCNC), Las Vegas, NV, USA; 2009. p. 1–5.
Seyyed Jalaleddin Dastgheib MRSG, Oulia Hamed. An efficient approach for clustering in wireless sensor network using fuzzy logic. In: International conference on computer science and network technology (ICCSNT), Harbin, China; 2011. p. 24–6.
Shigei N, Morishita H, Miyajima H. Energy efficient clustering communication based on number of neighbors for wireless sensor networks. In: International multi-conference on engineers and computer scientists (IMECS), Hong Kong; 2010.
Siew Z, Kiring A, Yew HT, Neelakantan P, Teo KTK. Energy efficient clustering algorithm in wireless sensor networks using fuzzy logic control. In: IEEE Colloquium on Humanities, Science and Engineering (CHUSER), Penang, Malaysia; 2011. p. 392–7.
Smaragdakis G, Matta I, Bestavros A. SEP: A Stable Election Protocol for clustered heterogeneous wireless sensor networks. In: International conference on sensor and actor network protocols and applications (SANPA), Boston, USA; 2004.
Soro S, Heinzelman W. Prolonging the lifetime of wireless sensor networks via unequal clustering. In: IEEE parallel and distributed processing symposium (IPDPS), Denver, Colorado, USA; 2005.
Soro S, Heinzelman WB. Cluster head election techniques for coverage preservation in wireless sensor networks. Ad Hoc Netw 2009;7:955–72.
Subhai C, Malarkan S, Vaithinathan K. A survey on energy efficient neural network based clustering models in wireless sensor networks. In: International conference on emerging trends in VLSI, embedded system, nano electronics and telecommunication system (ICEVENT), Tamil Nadu, India; 2013. p. 1–6.
Taheri H, Neamatollahi P, Younis O, Naghibzadeh S, Yaghmaee M. An energy-aware distributed clustering protocol in wireless sensor networks using fuzzy logic. Ad Hoc Netw 2012;10:1469–81.
Tai A, Tso K, Sanders W. Cluster-based failure detection service for large-scale ad hoc wireless network applications. In: IEEE conference on dependable systems and networks (DSN), Florence, Italy; 2004. p. 805–14.
Tashtoush YM, Okour MA. Fuzzy self-clustering for wireless sensor networks. In: International conference on embedded and ubiquitous computing (EUC), vol. 1, Shanghai, China; 2008. p. 223–9.
Thiemo Voigt, JSAD, Ritter Hartmut, Alonso J. Solar-aware clustering in wireless sensor networks. In: IEEE symposium on computers and communications; 2004.
Tian D, Georganas ND. A coverage-preserving node scheduling scheme for large wireless sensor networks. In: ACM conference on wireless sensor networks and applications; 2002. p. 32–41.
Tubaishat M, Yin J, Panja B, Madria S. A secure hierarchical model for sensor network. ACM SIGMOD Rec 2004;33:7–13.
Virrankoski ASR. Tasc: Topology adaptive clustering for sensor networks. In: IEEE conference on mobile adhoc and sensor systems (MASS), Washington, USA; 2005.
Voigt T, Dunkels A, Alonso J, Ritter H, Schiller J, 2004. Solar-aware clustering in wireless sensor networks. In: International symposium on computers and communications, Alexandria, Egypt; 2004. p. 238–43.
Wang S, Chen Z. Lcm: a link-aware clustering mechanism for energy-efficient routing in wireless sensor networks. IEEE Sens J 2013;13:728–36.
Wang Y, Yang T, Zhang D. An energy efficient and balance hierarchical unequal clustering algorithm for large-scale sensor networks. Inf Technol J 2009;8:28–38.
Wei Z. Energy efficient clustering algorithm based on neighbors for wireless sensor networks. J Shanghai Univ 2011;15:150–3 (Springer).
Wei C, Yang J, Gao Y. Cluster-based routing protocols in wireless sensor networks: a survey. In: Conference on computer science and network technology, vol. 3, Harbin, China; 2011a. p. 1659–63.
Wei D, Jin Y, Vural S, Moessner K, Tafazolli R. An energy-efficient clustering solution for wireless sensor networks. IEEE Trans Wirel Commun 2011b;10:3973–83.
Wen C, Sethares W. Automatic decentralized clustering for wsns. EURASIP J Wirel Commun Netw 2005;3:686–97.
Xu D, Gao J. Comparison study to hierarchical routing protocols in wireless sensor networks. In: International conference on environmental science and information application technology (ESIAT); 2011. p. 595–600.
Xu Y, Qi H. Decentralized reactive clustering for collaborative processing in sensor networks. In: International conference on parallel and distributed systems (ICPADS), Newport Beach, California, USA; 2004. p. 54–61.
Xu Y, Heidemann J, Estrin D. Geography-informed energy conservation for ad hoc routing. In: ACM conference on mobile computing and networking (MOBICOM), Rome, Italy; 2001. p. 70–84.
Yanagihara K, Taketsugu J, Fukui K, Fukunaga S, Hara S, Kitayama K. Eacle: energy-aware clustering scheme with transmission power control for sensor networks. Wirel Pers Commun 2007;40:401–15 (Springer).
Wen, Chih-Yu, Sethares, William A. Distributed clustering with directional antennas for wireless sensor networks. IEEE Sens J 2013;13:2166–80.
Yang T, Zhang D. An energy-balancing unequal clustering protocol for wireless sensor networks. Inf Technol J 2009;8:57–63.
Ye F, Zhong G, Cheng J, Lu S, Zhang L. Peas: a robust energy conserving protocol for long-lived sensor networks. In: IEEE international conference on distributed computing systems (ICDCS), Providence, Rhode Island, USA; 2003. p. 28–37.
Ye M, Li C, Chen G, Wu J, Al MYE. Eecs: an energy efficient clustering scheme in wireless sensor networks. In: IEEE conference on performance, computing, and communications (IPCCC), Phoenix, Arizona, USA; 2005. p. 535–40.

Yick J, Mukherjee B, Ghosal D. Wireless sensor network survey. Comput Netw 2008;52:2292–330.

Yi S, Heo J, Cho Y, Hong J. Peach: power-efficient and adaptive clustering hierarchy protocol for wireless sensor networks. Comput Commun 2007;30:2842–52.

Younis O, Fahmy S. Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach. In: IEEE conference on communications (INFOCOM), Hong kong; 2004a.

Younis O, Fahmy S. Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Trans Mob Comput 2004b;3:366–79.

Younis O, Krunz M, Ramasubramanian S. Node clustering in wireless sensor networks: recent developments and deployment challenges. IEEE Netw 2006:20–5.

Younis M, Senturk IF, Akkaya K, Lee S, Senel F. Topology management techniques for tolerating node failures in wireless sensor networks: a survey. Comput Netw 2014;58:254–83.

Youssef A, Younis M, Youssef M, Agrawala A. Distributed formation of overlapping multi-hop clusters in wireless sensor networks. In: IEEE glocal communications conference (GLOBECOM), San Francisco, USA; 2006. p. 1–6.

Youssef M, Youssef A, Younis M. Overlapping multihop clustering for wireless sensor networks. IEEE Trans Parallel Distrib Syst 2009;20:1844–56.

Yuea J, Zhang W, Xiao W, Tang D, Tang J. Energy efficient and balanced cluster-based data aggregation algorithm for wireless sensor networks. Proced Eng 2012;29:2009–15.

Yu M, Leung KK, Malvankar A. A dynamic clustering and energy efficient routing technique for sensor networks. IEEE Trans Wirel Commun 2007;6:3069–79.

Yu J, Qi Y, Wang G, Guo Q, Gu1 X. An energy-aware distributed unequal clustering protocol for wireless sensor networks. J Distrib Sens Netw 2011:1–8.

Yueyun C, Yue-Long Z, Jianwei A, Qian L. An energy-saving routing protocol based on leach. In: International conference on multimedia information networking and security (MINES), Nanjing, China; 2012. p. 608–11.

Zaiane, Osmar R, Lee, Chi-Hoon. Clustering spatial data in the presence of obstacles: a density-based approach. In: International database engineering and applications symposium (IDEAS); 2002. p. 214–23.

Zhang H, Arora A. Gs^3: scalable self-configuration and self-healing in wireless sensor networks. Comput Netw 2003;43:459–80.

Zhang P, Xiao G, Tan H-P. Clustering algorithms for maximizing the lifetime of wireless sensor networks with energy-harvesting sensors. Comput Netw 2013;57:2689–704.

Zhou Y, Fang Y, Zhang Y. Securingwireless sensor networks: a survey. IEEE Commn Surv Tutor 2008;10:6–28.

Zhu S, Setia S, Jajodia S. Leap+: efficient security mechanisms for large-scale distributed sensor networks. ACM Trans Sens Netw 2006;2:500–28.