Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 57 (2015) 660 – 669
Procedia Computer Science

3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)

Coverage-aware Unequal Clustering Algorithm for Wireless Sensor Networks

Nabajyoti Mazumdar[a]*, Hari Om[b],

[a,b] *Department of Computer Science and Engineering, Indian School of Mines, Dhanbad-826004, India*

Abstract

Wireless Sensor Networks (WSNs) are deployed to monitor physical conditions of various places such as geographical regions, traffic, battlefields, etc. Coverage preservation is the most important issue in such applications. Though the coverage aware clustering algorithms for WSNs have been investigated, yet the coverage aware algorithms in the context of unequal clustering have not yet been explored. In this paper, we present a 1-hop clustering algorithm, called Coverage-aware Unequal Clustering Algorithm (CUCA) for WSNs. Here, the aim is to reduce the size of clusters away from the base station/sink as the cluster heads (CHs) are burdened with large communication distance to the base station. Our proposed algorithm allows equal load distribution among the CHs. To increase the coverage lifetime of the network, the sensor nodes whose sensing area is completely overlapped by its neighbors are given higher priority to be selected as CHs. The sensor node's residual energy is considered for partial overlapping. Our algorithm outperforms over the state-of-the-arts existing algorithms, namely HEED and CA in terms of coverage lifetime.

Peer-review under responsibility of organizing committee of the 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)

Keywords: Wireless sensor networks; unequal clustering; coverage lifetime; network lifetime;

---

1. Introduction

The wireless sensor networks (WSNs) have gained a large research interest in last few years. A WSN consists of a large number of sensors, each having capability of sensing, computing, and communicating. The

---

* Nabajyoti Mazumdar. Tel.:+91-9693209028;
E-mail address: nabajyoti.ismdhanbad@gmail.com

Peer-review under responsibility of organizing committee of the 3rd International Conference on Recent Trends in Computing 2015 (ICRTC-2015)
doi:10.1016/j.procs.2015.07.437

sensor nodes are however equipped with low powered irreplaceable batteries. They also have limited processor power and low bandwidth communication channels. Therefore, an efficient technique is required for reliable communication considering these constraints. The coverage preservation is one of the most important issues in WSNs for reliable performance. The wireless sensor networks are applicable for a wide range of applications like environment monitoring, surveillance and vehicular tracking, etc. [1]. To cover the whole target area is the most vital issue in such applications. To achieve this objective, the sensor nodes are deployed densely in the target area that results in a large amount of redundancy in the network. The sensor nodes may lead to death if their energy are not utilized efficiently since the sensor nodes have limited battery power, which cannot be replaced or recharged. Hence, it may result in holes in the network, which in turn effects the coverage of the target area. Therefore, there is a need to adopt efficient coverage aware techniques for the long run operation of a WSN.

Many energy saving techniques have been proposed to maximize the network lifetime, which in turn increase the coverage lifetime. In WSNs, the network lifetime means the time when the first node dies, whereas the coverage lifetime means the time till the complete coverage of the target area is maintained irrespective of the number of nodes alive. Clustering is a very effective energy preservation technique in WSNs as it provides efficient resource utilization and minimizes the energy consumption in WSNs by reducing the number of sensor nodes that takepart in long distance transmission [9]. In clustering, a number of sensor nodes are grouped together to form a cluster and one of them is selected as a cluster head (CH) based on various parameters such as residual energy, node degree, etc. [10-16]. The CHs have more burden as they have to aggregate the packets that have been received from its cluster members (CMs) and they send the aggregated packets to the base station (BS) which is situated far away from the network. Hence, the CHs spend more energy as compared to other nodes, i.e., non-CHs. The coverage-aware clustering algorithms for WSNs have been investigated in literature [2-5]. In coverage-aware clustering algorithms, a node whose sensing area is completely covered by its neighbors is given higher priority to be selected as CH because the death of such node does not breach the coverage of the network. However, frequently selecting the same node as CH also limits the coverage and the network lifetime. So, there must be a tradeoff between the coverage and energy aware techniques. In addition to this, a non-CH node whose sensing area is fully covered by its neighbors may voluntarily go to sleep state; otherwise, such nodes lead to data redundancy that also consumes additional energy. Again, the hot spot problem is a major drawback of many clustering techniques used for WSNs. The CHs further away from the base station are overloaded as they have to send packets to a long distance. In other words, the load is not uniformly distributed among the CHs that leads to death of some CHs and hence leads to holes in the network. This problem is referred as hot spot problem.

In this paper, we propose a new distributed unequal clustering algorithm, called *Coverage-aware Unequal Clustering Algorithm (CUCA)* for WSNs to maximize the coverage lifetime of the network. In this algorithm, the nodes whose sensing area is fully covered by its neighbors are given higher priority to be selected as cluster heads over the partial overlapped sensor nodes. Here, the sizes of clusters are unequal to balance the load among the CHs, which avoids the hot spot problem. The cluster radius is inversely proportional to its distance from the sink. The CHs calculate their cluster radii based on their distance from the sink. The salient features of our proposed algorithm can be summarized as follows:

- It is a distributed coverage aware clustering algorithm that enhances the coverage lifetime of a network.
- It avoids hot spot problem by incorporating unequal clustering.
- The non-CH nodes whose area is fully covered by their neighbors are moved to sleep mode to reduce the redundancy and energy consumption.

The remaining part of the paper is organized as follows. Section 2 reviews the related works. In section 3, we present our proposed algorithm. In section 4, the simulation results of our proposed algorithm are given, which are followed by the conclusions and future work in section 5.

2. Related work

In last few decades, several works have been done on wireless sensor networks especially related to clustering. Some clustering algorithms have been proposed in recent years also [15-16]. In these algorithms, the CHs selection is done based on different parameters, such as some form of probability [10], residual energy of the node [8], distance [13], node degree [9], etc. Most of these algorithms form clusters of equal sizes [7-16]. The paper [6] investigates unequal clustering [6]. In these papers, the CHs have large cluster radius near the sink, meaning thereby they have more CMs as compared to the CHs that are far away from the sink. All these papers have mainly focussed on the network lifetime in terms of first node dead. By improving the network lifetime, the researchers believe to increase the coverage lifetime too. However, as soon as the first node dies, the coverage also gets affected. So, recently, many coverage aware clustering algorithms have been proposed in which the CHs are selected based on its coverage percentage, i.e., percentage of the sensing area of the node covered by its neighbors. In [2], Tsai has proposed a coverage preserving algorithm based on LEACH [10] in which a node with high overlapped area is assigned a higher probability of being a CH. This approach however does not consider the residual energy of the sensor nodes while selecting the CHs. Heinzelman et al. [3] have proposed a family of coverage preserving cluster head selection metrics. One of them is known as coverage-aware (CA) metric. This metric depends neither on the sensor node's energy nor the remaining energy of the neighbor nodes.

3. Proposed work

In this section, we define the network model and some terminologies which are used in our proposed algorithm followed by our proposed algorithm.

3.1. Network model

We assume a WSN model used for this work that consists of a large number of homogeneous sensor nodes deployed in the target area. All sensor nodes are stationary and they initially cover the whole target area. All sensor nodes sense their local data and then send it to their corresponding CHs. The CHs aggregate the received data to reduce the redundancy and then send it to the sink directly. The sink is also assumed as stationary, which is situated outside the target area. The sensor nodes are aware of their locations in the network. The wireless link is symmetric and bidirectional. This energy model of the WSN is adopted from the work discussed in [13].

3.2. Terminologies

Following are the terminologies that help in presenting and understanding our proposed algorithm.

- The set of sensor nodes is denoted by $S = \{S_1, S_2, \ldots, S_n\}$, where S_i is sensor node.
- The sensing area of i^th^sensor node S_i isdenoted by $Area(i)$
- The distance between the nodes S_i and $S_j$is denoted by $Dist(i, j)$
- The distance of node S_i from the sink is denoted by $D_{sink}(i, sink)$.
- The distance between the sink and its closest node is denoted as $minD_{sink}= min(D_{sink}(i, sink))$.
- The distance between the sink and its farthest node is denoted as $maxD_{sink}= max(D_{sink}(i, sink))$.
- The sensing radius of node S_i is denoted by R_s.
- The maximum cluster radius is denoted by R_{max}.
- The cluster radius of i^{th} cluster head is denoted by $C_{radius}(i)$.
- The set of neighbors of node $S_i$is denoted by $Neighbor(i)= \{S_j : Dist(i, j) \leq R_s\}$.

- The initial energy of a sensor node is denoted by$Energy_{init}$.
- The residual energy of node S_i is denoted by $Energy_{res}$.
- T_{CH} and T_{CF} denote maximum CH selection and cluster formation time, respectively

3.3. Proposed Algorithm

We describe the basic idea of our algorithm. Initially, all sensor nodes broadcast a control message within its sensing area. The message contains the node ID and its location information. Upon receiving such message, a sensor node updates its neighbor table, Neighbor (i). Then, each sensor calculates its overlapping sensing area with its neighbors, denoted by Θ, using the following equation:

$$\Theta(i) = \frac{1}{Area(i)} \bigcup_{j \in Neighbor(i)} \left(Area(i) \bigcap Area(j) \right) \tag{1}$$

It is evident from (1) that there is a node whose Θ (i) value is 1 since its sensing area is fully covered by its neighbors. Furthermore, each node introduces a time delay that is measured by timer before going to start their campaign for CH selection and it is measured as follows:

$$T(i) = \left\{ \left(1 - \frac{Energy_{res}(i)}{Energy_{init}(i)} \right) \times T_1 + \left(1 - \Theta(i)\right) \times T_2 \right\} \tag{2}$$

Here T(i) denotes the timedelay of ith node S_i. The constantsT_1 and T_2 are selected in such a way that the summation of these constants should be equal to T_{CH}, i.e., $T_1+T_2=T_{CH}$. It is clear from (2) that if a node's Θ (i) =1, then the timer depends only on its residual energy. So, the value of its T(i) will be smaller than that of those nodes that have Θ (i) $\neq$ 1. Therefore, we can say that the fully covered nodes are given higher priority as compared to the partially covered nodes. Once the timer of ith node S_i expires, i.e., T(i) =0, the node selects itself as a CH and broadcasts a CH advertisement message within its cluster radius $C_{radius}(i)$. The cluster radius is calculated as follows:

$$C_{radius}(i) = \left(1 \text{ - } \alpha \frac{D_{\text{sink}}(\text{i, sink}). - \min D_{\text{sink}}}{\max D_{\text{sink}} - \min D_{\text{sink}}} \right) \times R_{\max} \tag{3}$$

Where, R_{max} denotes the maximum allowed cluster radius in the network and α is a constant that lies in the interval [0, 1). It is clear from (3) that $C_{radius}(i)$ is inversely proportional to the distance between ith CH and the sink. A node upon receiving the CH advertisement message stops its timer as its neighbour's timer has already expired and it becomes a non-CH node for the upcoming communication round. To form the clusters, each non-CH node calculates its timer using the following formula:

$$T_{CM}(i) = \left(1 - \frac{Energy_{res}(i)}{Energy_{init}(i)} \right) \times T_{CF} \tag{4}$$

Once the timer of ith node expires, the node declares itself as a CM and joins the nearest CH by broadcasting a CH join message, which contains its ID and location information. If a neighbour of jth node S_j overhears the message from ith node S_i, then ith node S_i increments its active neighbor count by one. Furthermore, if the active neighbor nodes cover the sensing area of a non-CH, then the node moves to the sleep state for the upcoming communication round.

3.4. Algorithmic flow

The CUCA is given below in Algorithm 1.

Algorithm: Coverage-aware Unequal Clustering Algorithm (CUCA)

```
For each node i,

1.  Broadcast control message to announce its existence in Network.
2.  Update Neighbor table upon receiving control messages from different nodes.
3.  Calculate overlapping sensing area Θ (i) using (1).
4.  Broadcast CH advertisement message when $T(i) > T_{CH}$.

    4.1 If T(i) expires, calculate cluster radius $C_{radius}(i)$.
    4.2 Broadcast CH advertisement message within its cluster radius, $C_{radius}(i)$.
    4.3 If a node $S_j$ receives CH advertisement message, then it stops its timer for CH selection.

5.  Each non-CH node j

    5.1 Introduces a delay of $T_{CM}(j)$ and then joins nearest CH by advertising CH join message

    5.2 If a non-CH node's sensing area is completely overlapped with its neighbors, then it moves to sleep mode.
```

3.5. Complexity Analysis

Lemma 1. The message complexity of our proposed algorithm is $O(1)$ for each sensor node.

Proof. Each node broadcasts at most three messages per communication round that include control message to find neighbors, CH advertisement, and finally cluster join message,which are done by each node independently. Therefore, the message complexity of our proposed algorithm is $O(1)$.

Lemma 2. The time complexity of our proposed algorithm is $O(n)$ for n number of sensor nodes.

Proof. From Eq. (3) and Eq. (5), Θ and C_{radius} are calculated in $O(1)$ time, since they are calculated independently by each sensor node. To form clusters, each CM node needs to join its nearest CH for which it requires to check (n-1) CHs in worst case. Therefore, finding the nearest CHs takes $O(n)$ time. Hence, the time complexity of our proposed algorithm is $O(n)$.

4. Simulation Results

For performance evaluation, we test our proposed algorithm and compare it with the HEED [7] and CA algorithms [3]. We use the following parameters for performance evaluation: network lifetime, coverage lifetime, and energy consumption. In our experiments, we have computed the network lifetime in terms of rounds when the first node becomes dead (FND). Similarly, we have defined the coverage lifetime as the number of rounds till the target area has been fully covered, i.e., 100 percent. To support the coverage and network lifetime, we have also measured the amount of average remaining energy of the sensor nodes per round. The simulations have been carried out using Dev C++ and MatLab. The parameters with their typical values used in our simulations are given in Table 1.

TABLE.1. Parameters and their values

Parameter Name	Notation	Value
Target area	A	40 ×40 m sq.
Sink location	S	(0,0)
No. of sensors	n	60 - 100
Initial energy	E_{init}	0.02 Joules
Tx range	r	5m
Control packet size	C_p	100
Data packet size	D_p	500
Free space (ε_{fs})	ε_{fs}	10pJ/m^2/bit
Multi-path fading (ε_{mp})	ε_{mp}	0.0013 pJ/bit/m^4
Tx or *Rx* electronics	α_{tx}or α_{rx}	50nJ/m^2/bit

4.1. Network Lifetime

We have computed the network lifetime in terms of alive nodes with respect to number of rounds as shown in Fig. 1. As evident from Fig.1, our proposed algorithm performs consistently better than the CA algorithm. As compared to the HEED, it initially (for less number of rounds) performs slightly inferior sinceinitially the fully covered nodes will repeatedly get selected as CHs because the nodes whose sensing area is fully covered by its neighbors are given higher priority to be selected as cluster heads over the partial overlapped sensor nodes. However, it outperforms over HEED as the number of rounds increases since in HEED all the sensor nodes actively contribute to the network operation, which has greater deteriorate as the number of rounds increases due to data redundancy.

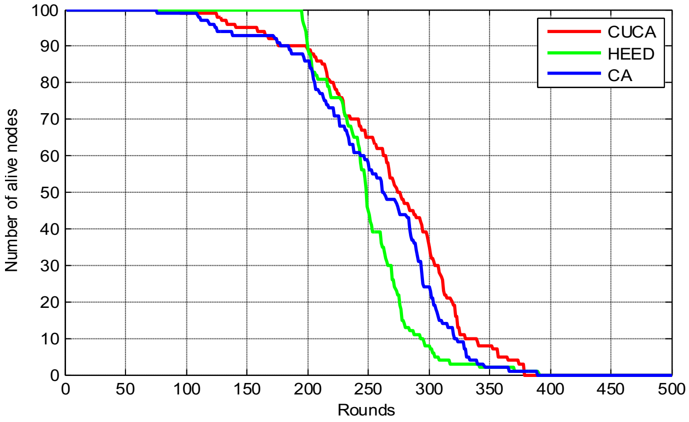

Fig. 1. Network lifetime of various algorithms (n = 100)

We have also computed network lifetime in terms of rounds when the first node dead for different number of sensor nodes (for deploying 60, 70, 80, 90 & 100 sensors)as shown in Fig.2.Here our algorithm performs slightly better than the CA, but not the HEED. The performance of CA is inferior to CUCA and HEED because CA is only coverage-aware so the same nodes are being selected as CHs in every round which exhaust the residual energy of the CHs quickly.

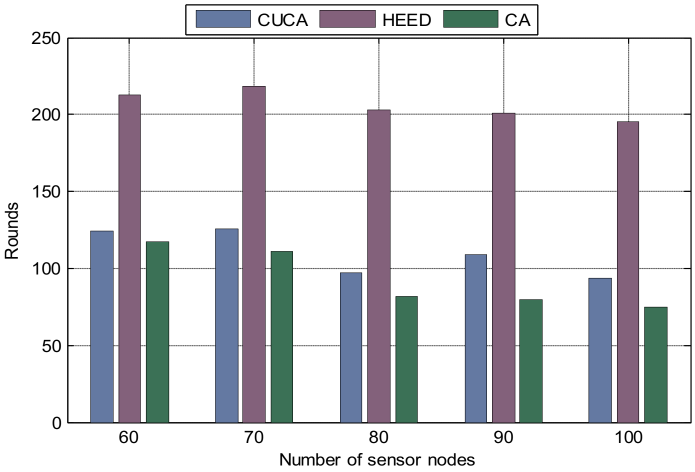

Fig.2. Network lifetime (FND)

4.2. Coverage Lifetime

We have computed the coverage lifetime of CUCA, HEED, and CA algorithms with respect to number of rounds as shown in Fig. 3. As evident from Fig. 3, our proposed algorithm performs consistently better than the HEED and CA algorithms. In our method, the CHs are selected in an efficient way, which provides better performance.

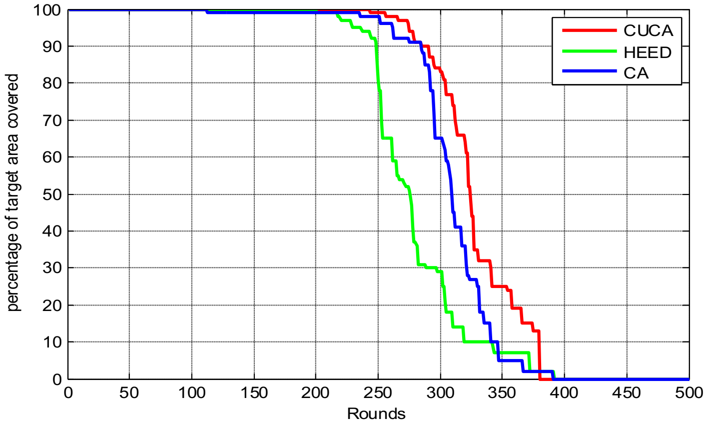

Fig. 3. Coverage lifetime of various algorithms ($n = 100$)

We have computed complete coverage for deploying 60, 70, 80, 90 and 100 sensors and the results are shown in Fig. 4. As evident from Fig. 4, the CUCA achieves complete coverage of the target area for longer time as compared to the HEED and CA algorithms. We also find that the HEED guarantees 100% coverage of the target area until the first node dies as it is only energy aware. Furthermore, we see that the network lifetime for HEED is 195 rounds when the first node dies (refer Fig. 1) and it also takes same rounds when 100% coverage lifetime is considered (refer Fig. 3). Thus,in HEED, as soon as the first node dies it reduces the coverage of target area. In case of CA, the network life time is 80 rounds whereas it is 110 rounds when 100% coverage lifetime is considered as the fast depletion of residual energy of the CHs limits the coverage lifetime and the network lifetime of the network. In case of the CUCA, the network lifetime is 95 rounds when the first node dies whereas it is 243 rounds in case of 100% coverage lifetime. So, in spite of death of certain nodes, our algorithm preserves the coverage of the target area for longer time than the HEED and CA algorithms. Thus, our algorithm is more efficient than the HEED and CA as the coverage preservation is an important factor for the reliability of a WSN.

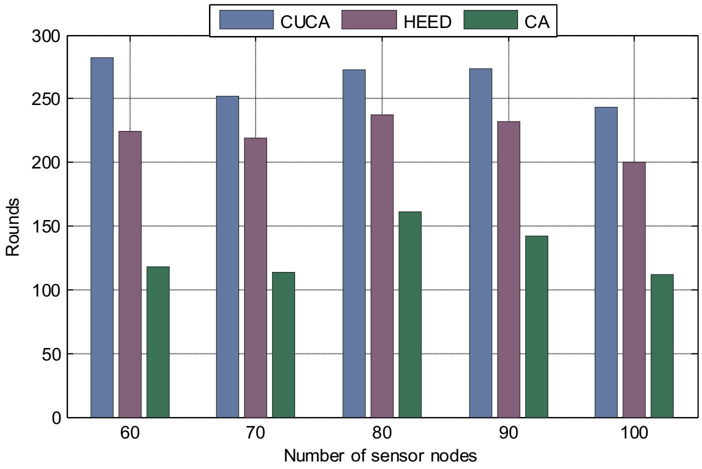

Fig. 4. Coverage lifetime when 100% coverage

4.3. Energy Efficiency

We have also measured the average remaining energy of the sensor nodes in the network to justify the lifetime of the sensor nodes as shown in Fig. 5. We find that the CUCA and HEED have very close energy consumption at the initial stage but the HEED performs poor as compared to the CUCA and CA as the number of rounds increases. The higher energy consumption of the HEED is mainly due to the active partition of all nodes in the network that consumes extra energy and leads to data redundancy.

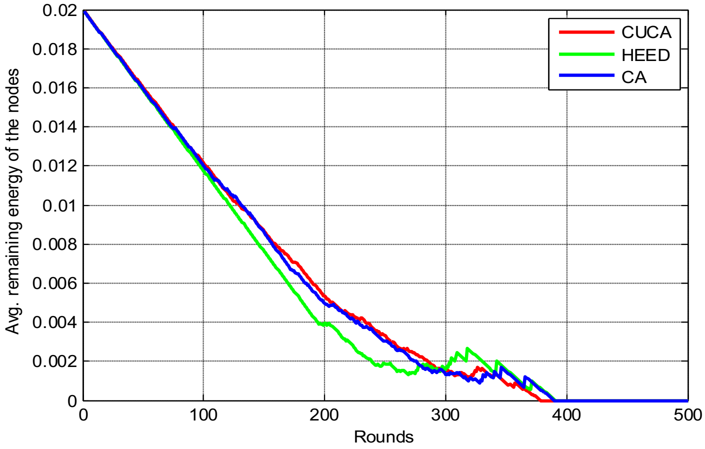

Fig. 5. Average remaining residual energy of sensor nodes

The reasons for better performance of the CUCA algorithm are summarized as follows.

- The CUCA applies two strategies for selecting the CHs. In target area where the node density is very high, the chance of overlapping of sensing area of the sensor nodes is 100% in most of the cases. In such context, those nodes are selected as CHs whose sensing area is completely overlapped with its neighbors. As a result, the death of such nodes does not breach the coverage of the target area. Furthermore, there is very less chance that the nodes' sensing area is 100% overlapped in sparse deployed region. Therefore, in such regions, the selection of CHs depends on their residual energy and the overlapped area.

- During cluster formation, those non-CH nodes are allowed to take cluster membership whose residual energy isrelatively higher than the others. In other words, the nodes having less residual energy and whose sensing areas are completely covered by their neighbors, they are moved to sleep state in order to save their energy. Therefore, this approach also contributes in enhancing the coverage and network lifetime.
- The unequal cluster sizes also play a significant role in minimizing the communication energy of the CHs.

5 Conclusion

In this paper, we have proposed a new coverage-aware unequal clustering algorithm for wireless sensor network to maximize the coverage and network lifetime. In this algorithm, we have adopted unequal clustering technique to get advantages over the equal clustering techniques. Experimental results show that our algorithm outperforms the CA and HEED algorithms in terms of coverage lifetime. In future, we will extend this work for multi-hop routing with the modified version of the CUCA in which the load of the CHs will be the main focus for relaying the data towards the sink.

References

[1] I. F. Akilidz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: survey,” Computer Networks. vol. 38, pp. 393-422, 2002.

[2] Y. R. Tsai, “Coverage-preserving routing protocols for randomly distributed wireless sensor networks,” IEEE Trans. on Wireless Communications. vol. 6, pp. 1240-1245, 2007.

[3] W. B. Heinzelman, S. Soro, “Cluster head election techniques for coverage preservation in wireless sensor networks,” Ad-Hoc Networks. vol. 7, pp. 955-972, 2009.

[4] B. Wang, H. B. Lim, and D. Ma, “A coverage-aware clustering protocol for wireless sensor networks,” Computer Networks. vol. 56, pp. 1599-1611, 2012.

[5] Y. Tao, Y. Zhang, and Y. Ji, “Flow-balanced routing for multi-hop clustered wireless sensor networks,” Ad-Hoc Networks. vol. 11, pp. 541-554, 2013.

[6] M. Ye, C. F. Li, G. H. Chen, and J. Wu, “An energy-efficient unequal clustering mechanism for wireless sensor networks” Proce. of the Intl. Conf. on Mobile Ad hoc and Sensor Systems. pp. 8, 2005.

[7] A. Navid, V. Alireza, WenyaoXu, G. Maria, and S. Majid, “Cluster size optimization in sensor networks with decentralized cluster-based protocols” Computer Communications. vol. 35, pp. 207-220, 2012

[8] O. Younis and S. Fahmy, “HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks,” IEEE Trans. on Mobile Computing, vol. 3, pp. 366–379, 2004.

[9] A. A. Abbasi and M. Younis, “A survey on clustering algorithms for wireless sensor networks” Computer Communications. vol. 30, pp. 2826-2841, 2007.

[10] W. B. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocols for wireless microsensor networks,” Proce. of Hawaii Intl. Conf. on System Sciences (2000).

[11] Y. He, W. S. Yoon and J. H. Kim, “Multi-level Clustering Architecture for Wireless Sensor Networks,” Information Technology Journal, Vol. 5, pp.188-191, 2006

[12] W. Liu and J. Yu, “Energy Efficient Clustering and Routing Scheme for Wireless Sensor Networks,” Proceedingof the IEEE International Conference on Intelligent Computing and Intelligent Systems,pp. 612-616.

[13] G. Gupta and M. Younis. “Load-balanced clustering of wireless sensor networks.” In IEEE International Conference on Communications,” Vol. No. 3, pp. 1848–1852, 2003

[14] .S. Bandhopadhyay and E. Coyle, “An energy efficient hierarchical clustering algorithm for wireless sensor networks,” Proceedings of IEEE INFOCOM, 2003.

[15] N. Dimokas, D. Katsaros, and Y. Manolopoulos, “Energy-efficient distributed clustering in wireless sensor networks. J. of Parallel Distributed Computing, vol. 70, pp. 371-383, 2010.

[16] A. Nauman, P. William, R. William, and S. Shyamala, “A multi-criterion optimization technique for energy efficient cluster formation in wireless sensor networks. Information Fusion. vol. 12, pp. 202-212, 2011.