

## A Clustering Algorithm Based Geographic Location Information for Wireless Sensor Networks

<sup>1</sup> Lin SHEN, <sup>2</sup> Liqin WANG

<sup>1</sup> Jiangsu University of Technology, Changzhou, China, 213001

<sup>2</sup> Changzhou College of Information Technology, Changzhou, China, 213164

<sup>1</sup> E-mail: shenlin6789@yeah.net

*Received: 18 August 2013 / Accepted: 25 September 2013 / Published: 31 October 2013*

---

**Abstract:** One of the main objectives of wireless sensor network design is to improve the energy efficiency. How to efficiently utilize sensor nodes to prolong the lifespan of a wireless network has long been a research topic. This paper presents a location based LEACH clustering algorithm, which is an extension to the LEACH routing algorithm. Armed with distributed and local network based routing decision-making mechanism, this algorithm fully utilizes the location information of network nodes in routing to reduce the routing cost. Simulation results indicate that this algorithm can balance nodes' energy consumption and prolong the network's life span. It also has good stability and extensibility. *Copyright © 2013 IFSA.*

**Keywords:** Wireless sensor networks, LEACH protocol, Geographic location information, Network lifetime.

---

### 1. Introduction

Wireless sensor networks (WSN) are wireless network composed of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations. Due to the deployment flexibility and maintenance simplicity, WSN applications have been seen in many areas [1-4].

Energy is the scarcest resource of WSN nodes, and it determines the lifetime of WSNs. Compared with traditional ad hoc network design, one of the most important design objectives of WSNs is to minimize node energy consumption and maximize the network lifetime. Because the routing protocol design is driven by the improvement of mobile service quality, the widely applied and mature routing protocols used in traditional ad hoc networks are not suitable for WSNs [5, 6]. In recent years,

clustered routing protocol has gained increasing attention from researchers because of its potential of extending WSN lifetime. Heizelman and Kopa [9] designed and implemented the first distributed and clustered routing protocol with low energy consumption, LEACH. After that, some modified algorithms based on LEACH were proposed, such as LEACH-C, LEACH-E, and LEACH-F [7, 8]. This paper presents a new improved LEACH algorithm.

The rest of the paper is organized as follows: Section 2 presents the new protocol with its key management mechanism and operation procedures. Section 3 analyzes the performance of the new protocol. Section 4 summarizes the paper.

### 2. Problem Formulation

The routing protocol design for WSNs is challenging because WSNs differ from traditional wireless ad hoc networks in many aspects [9-12].

No global ID: The number of sensor nodes could be very large in a WSN and maintaining global IDs for them is too expensive and unrealistic. Therefore, no global ID is maintained in a WSN, which is different from traditional IP-based routing protocols.

Many-to-one communications: Almost all applications require multiple sensor nodes to send data to a specific node.

Data redundancy: In many cases many sensors nodes may obtain large amount of the same or similar data. So there is a huge data redundancy in the network.

Limited resources: Each sensor node is equipped with limited resources, such as power, computation capability and memory.

LEACH cluster head is selected using a threshold  $T(n)$ , as shown in Fig. 1, where  $T(n)$  is calculated according to:

$$T(n) = \begin{cases} \frac{p}{1 - p(r \bmod (l/p))} & n \in G \\ 0 & \text{others} \end{cases}$$

In this formula,  $p$  is the percentage of cluster heads over all nodes in the network, i.e., the probability that a node is selected as a cluster head;  $r$  the number of rounds of selection; and  $G$  is the set of nodes that are not selected in round  $l/p$ . As we can see here, the selection of cluster heads is totally randomly.

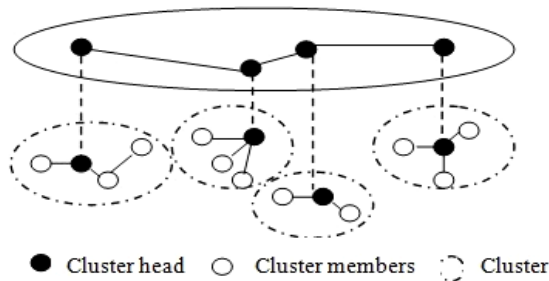


Fig. 1. Structure of clustered WSNs.

In addition, energy consumption is not considered as a factor in the above cluster head selection. In the reality, due to the environment where the nodes are deployed, the workload on data collection and transfer is different from node to node. Because node energy consumption is proportional to the amount and distance squared of data transferred, the more a node sends data and the farther it is away from the cluster head, the more energy is consumed with the node. Therefore, when a WSN has been in operation with LEACH protocol for a sufficiently long time, it will experience unbalanced remaining energy among nodes. For an energy unbalanced network, if the lifetime of cluster heads is less than the expected network lifetime, then blind nodes will appear and the network life and routing efficiency will be compromised.

Because cluster heads are randomly selected, it is possible the scenario illustrated in Fig. 2 occurs, in which two or even more cluster heads are very close to each other.

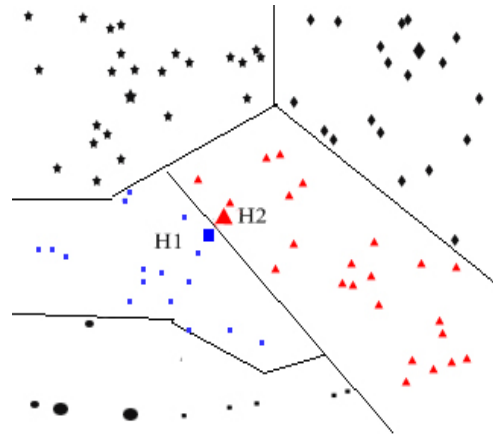


Fig. 2. Multiple cluster heads appear in a small region with LEACH protocol.

In Fig. 2, H1 and H2 are two cluster heads, nodes  $\blacksquare$  and  $\blacktriangle$  are their cluster members, respectively. H1 and H2 are very closely located. According to data communication model, the energy that a cluster head consumes is the sum of that consumed in receiving data and that in sending data, as described in Eq. (1):

$$E_{ch} = lE_{elec}N_{mem} + lE_{DA}(N_{mem} + 1) + lE_{elec} + l\epsilon_{mp}d_{toBs}^4, \quad (1)$$

where  $l$  is the length of data,  $\epsilon_{mp}$  is the power consumption of transferring 1 bit of data,  $E_{elec}$  is the power consumption of processing 1 bit of data,  $N_{mem}$  is the number of members in a cluster,  $d_{toBs}^4$  is the distance between the cluster head and node Sink,  $lE_{elec}N_{mem}$  is the power that  $N_{mem}$  cluster members consume when each of them send out length of  $l$  data to the cluster head, and  $lE_{DA}N_{mem}$  is the power that the cluster head consumes when it receives data of length  $l$  from its cluster members. It follows from (1) that the amount of energy that cluster heads H1 and H2 consume during data transfer is

$$E_{h1} = lE_{elec}N_{mem1} + lE_{DA}(N_{mem1} + 1) + lE_{elec} + l\epsilon_{mp}d_{h1toBs}^4, \quad (2)$$

$$E_{h2} = lE_{elec}N_{mem2} + lE_{DA}(N_{mem2} + 1) + lE_{elec} + l\epsilon_{mp}d_{h2toBs}^4, \quad (3)$$

where  $N_{mem1}$  and  $N_{mem2}$  are the number of members in clusters H1 and H2, respectively,  $d_{h1toBs}^4$  and  $d_{h2toBs}^4$  are the distances between the two cluster heads and node Sink, Therefore, the total energy consumed by the two clusters is

$$E_{h1} + E_{h2} = lE_{elec}(N_{mem1} + N_{mem2}) + lE_{DA}(N_{mem1} + N_{mem2} + 2) + 2lE_{elec} + l\epsilon_{mp}(d_{h1toBs}^4 + d_{h2toBs}^4) \quad (4)$$

When H1 and H2 are very close, we can have

$$d_{h1toBs} = d_{h2toBs}$$

Then (4) becomes

$$E_{h1} + E_{h2} = lE_{elec}(N_{mem1} + N_{mem2}) + lE_{DA}(N_{mem1} + N_{mem2} + 2) + 2lE_{elec} + 2l\epsilon_{mp}d_{h1toBs}^4 \quad (5)$$

As we can see, in this case the total energy consumption of two clusters is only  $lE_{elec} + l\epsilon_{mp}d_{h1toBs}^4$  greater than the case that there is only one cluster head. In addition, because  $lE_{elec} + l\epsilon_{mp}d_{h1toBs}^4$  is much greater than  $lE_{elec}(N_{mem1} + N_{mem2}) + lE_{DA}(N_{mem1} + N_{mem2} + 2)$ , therefore, the total energy consumption when there are two cluster heads is approximately twice of that when there is only one cluster head.

It is clear now that when multiple cluster heads are randomly selected within a small area, a big extra energy loss occurs. The amount of lost energy is approximately proportional to the number of cluster heads in the area. Of course, there is a precondition on this conclusion, that is, cluster heads are very closely located and the distance between them becomes negligible.

### 3. Protocol Performance

The design objective of the routing protocols for wireless sensor networks varies with the network application and operational environment. LEACH protocol is suitable for the WSNs under the following assumptions [3]:

1) All sensor nodes are identical and charged with the same amount of initial energy. All nodes consume energy at the same rate and are able to know their residual energy and control transmission power and distance. Every node has the capability to support different MAC protocol and data processing. All communication channels are identical. The energy consumption of transferring data from node A to node B is the same as that of transferring the same amount of data from node B to node A.

2) Every node can directly communicate with every other node, including the sink node.

3) The Sink node is fixed and far away from the wireless network. Thus we can ignore the energy consumed by the sink node. We assume that it always has sufficient energy to operate.

4) Every node has data to transfer in every time frame. The data transferred by sobering nodes are related and can be fused.

5) Sensor nodes are static.

WSNs are autonomous networks. Sensor nodes are independent with each other. The coordination between nodes is done through wireless communication, which costs much. This is one of the major reasons that the LEACH protocol selects cluster heads randomly. As we discussed before, this approach may cause the waste of energy because of unbalanced cluster head distribution. To solve this problem, we propose a new approach to selecting cluster heads. We assume that:

1) The network satisfies the pre-conditions of applying LEACH protocol.

2) After deployment, sensors are able to know their positions through GPS, or before deployment, their positions are accurately decided.

3) All nodes are able to adjust data transmission power. If necessary they can communicate with the base stations to acquire the initial setting information of the network.

If we modify the procedure of the calculation of  $T(n)$  during the cluster head generation such that cluster heads are produced progressively, then a node could decide if it is suitable to be a new cluster head based on the locations of existing cluster heads and its own location. More specifically, if the node is very close to any existing cluster head, then this node will give up the attempt to be a cluster head. As shown in Fig. 3, the network is divided into six parts. Nodes in region G1 will compete for being a cluster head. When a node is selected as a cluster head, it will broadcast the information to nodes nearby. Nodes in region G2 will receive the message. Thus, when nodes in this region compete for being cluster head, the location information of the cluster head in region G1 will be taken into consideration. If a node in G2 is close to the cluster head in G1, the node will be discarded. The cluster heads in all other regions will be generated in the same way.

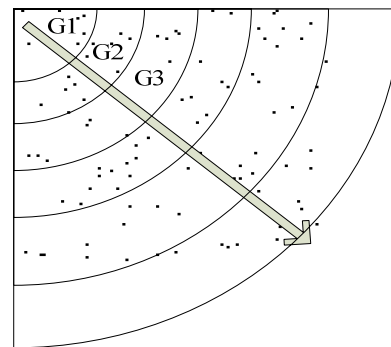


Fig. 3. Progressive approach to selecting cluster heads.

The cluster heads generated with this progressive approach will not be close to each other. However, because some nodes quit the competition for cluster head, the total number of cluster heads can be reduced, which is not good for saving the network energy. Our approach to solving this problem is,

when a node is excluded in the cluster head selection, a message is broadcast to other nodes and  $T(n)$  will be modified to increase the probability of others nodes being selected as cluster heads. The modified  $T(n)$  is:

$$T(n) = \begin{cases} \frac{p}{1 - p / r_{mod} / (1 / p) - pk} & n \in G \\ 0 & \text{others} \end{cases}, \quad (6)$$

where  $k$  is the number of nodes that are excluded from the cluster head selection due to the location reason, with an initial value of 0. When  $k$  increases,  $T(n)$  increases as well, which will ensure sufficient number of cluster heads will be generated by the progressive algorithm.

To facilitate the explanation of our improved algorithm, we introduce the following notations:

- B: The base station or node Sink
- $S_i$ : The  $i$ -th sensor node
- $H_j$ : The  $j$ -th cluster head
- ID( $S_i$ ): ID of the  $i$ -th sensor node
- Mem( $C_j$ ): Members of the  $j$ -th cluster
- Mem( $C_j$ ) $_i$ : The  $i$ -th members of the  $j$ -th cluster
- Loc( $S_i$ ): Location of the  $i$ -th sensor node
- Delay( $S_i$ ): Time delay that the  $i$ -th sensor node start to compete for a cluster head
- Num(Giveup): Number of discarded cluster heads
- ||: Operation of concatenation

#### A. Temporal distribution in cluster head selection

After the deployment of sensor nodes, we first acquire all nodes' location information (through GPS technology or known prior to its deployment) and report it to the base station. The base station decides Delay( $S_i$ ) for every node based on the geographic distribution of all sensor nodes. Delay( $S_i$ ) = 0 for those in the region to start first. As illustrated in Fig. 3, nodes in  $G_1$  start to compete for cluster heads at time 0, then nodes in  $G_2$  start with a delay, and then nodes in  $G_3$  start with a delay after nodes in  $G_2$  are finished, and so on. During the process, nodes need to send their location information to the base station:

$$S_i \rightarrow B: ID / S_i / Loc / S_i /$$

The base station needs to send the delay information to each node:

$$B \rightarrow S_i: ID / S_i / Delay / S_i /$$

#### B. Selection of cluster heads

Set Num(Giveup) to 0. Start with the nodes in  $G_1$ . If a cluster head is generated from  $G_1$ , broadcast a Hello package and Num(Giveup).

$$H_j \rightarrow \text{broadcast} : ID / H_j / \text{Hello} / \text{Num} / \text{Giveup} /$$

$$H_j \rightarrow \text{broadcast} : ID / H_j / \text{Giveup} / \text{Num} / \text{Giveup} /$$

When nodes in  $G_1$  are finished, consider nodes in  $G_2$ . Now the cluster heads generated in  $G_1$  are reference points. The distance between a node in  $G_2$  and any cluster head in  $G_1$  is a factor in selecting the node as a cluster head, as well as the random value of  $T(n)$ . If all conditions are satisfied, then broadcast the Hello message and Num(Giveup). Otherwise, only broadcast Num(Giveup). When nodes in other region receive this message, they will increment Num(Giveup) by 1, and then modify  $T(n)$  to increase the probability of being selected as cluster head.

Repeat the above process until all nodes in the network are considered.

## 4. Performance Analysis

NS2 (Network Simulator 2) is a very popular network simulation platform. It is a discrete event simulator designed for network research. To support the performance analysis of LEACH protocol, W. Heizelman et al. [13] extended NS by introducing an event-driven simulator. In the simulator the Tcl class Application/LEACH implements all functions for WSNs, including competition for cluster heads and data transmission. When the simulator is loaded with initial network settings, start() function starts to run, which invokes the DecideClusterHead() function to select cluster heads. We made a few modifications on top of the simulator extended by Heizelman.

We modified the node settings file nodes.txt. This is a text file and includes nodes' location information. We add a field to represent the time delay that a node starts to compete for cluster head. We also modified MobileNode.h/cc and Nodescen in order to read the delay information. For the Tcl class Application/LEACH, we modified start() function such that function decideClusterHead() is called based on the expiration of a timer. This way we can ensure that nodes compete for being cluster heads in a preset time order. Meanwhile, we add a distance related test in function decideClusterHead(), to decide if a node needs to give up.

#### A. Simulation Scenario and Parameters

In order to evaluate the performance of different algorithms, we use two scenarios to simulate the algorithms. In scenario 1, the region size is 100 meters by 100 meters, the number of nodes is 100, and the BS is located at (50, 175); In scenario 2, 400 sensor nodes are distributed in a 200 meters by 200 meters region and the BS is geographically located at (100, 250).

#### B. Simulation Results

##### 1) Performance measurements

In a wireless sensor network, the computing capacity and stored energy of a node is very limited. In particular, the limited energy affects the lifespan of information quality of the network. For this reason, we evaluate the algorithms based on the efficiency of the network energy consumption. We use two performance indices:

**Lifespan:** The lifespan of a sensor network is the time span from the beginning of the network operation to the instant that the network can no longer provide readable information, measured in the number of rounds. It can be measured in three ways: FND (First Node Dies), HNA (Half of the Nodes Alive, and LND (Last Node Dies).

**Data accuracy:** The accuracy of data received by the BS. The more the data is received, the high the accuracy after data fusion. The data accuracy is measured by the total data sent by all nodes in the lifespan of the network.

## 2) Analysis of simulation results

We compare the performance of the original LEACH clustering protocol and our progressive clustering protocol. Fig. 4, Fig. 5, and Fig. 6 show the change of FND, HNA and LND over the distance between cluster heads. As we can see, the lifespan of the network increases when the distance between cluster heads increases and reaches the cap when the distance is around 35 and 40. After that, when the distances increases further, the number of cluster heads goes down, and the energy consumption of the network goes up, which leads to the decline of the lifespan and data accuracy.

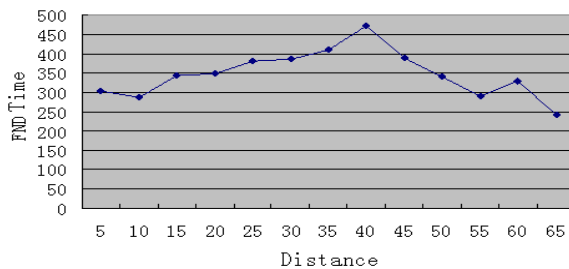


Fig. 4. FND vs. the distance between cluster heads.

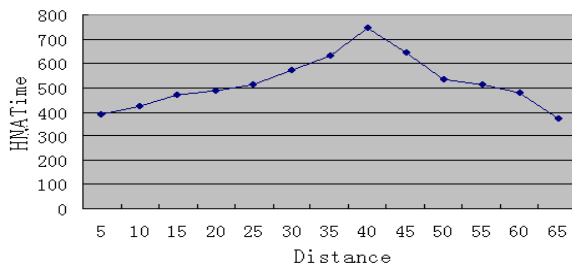


Fig. 5. HNA vs. the distance between cluster heads.

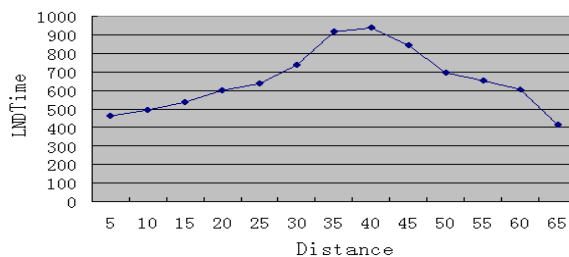


Fig. 6. LND vs. the distance between cluster heads.

It is clear from the simulation results shown in Fig. 7 that the lifespan of the new progressive clustering protocol is longer than that of the original LEACH protocol. The data transferred with the new protocol is 1/3 more than that with the old protocol, and the lifespan of the network with the new protocol is almost doubled that of the old protocol.

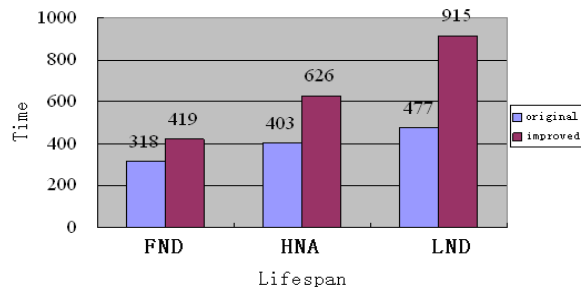


Fig. 7. Comparison of the lifespan of the two protocols.

## 5. Conclusions

The cluster head generation algorithm with the original LEACH clustering protocol can cause unbalanced distribution of cluster heads, which often leads to redundant cluster heads in a small region and thus cause the loss of energy. To solve this problem, we proposed a progressive algorithm for the cluster head selection. Simulation results show that our algorithm is much more efficient and can double the lifespan of a wireless sensor network. The algorithm can be easily implemented.

Our future work includes: (1) We simulated the performance of our algorithm in two scenarios, one is a dense network – with 100 nodes distributed in a 100 meters by 100 meters area, the other one is a less dense network – with 200 nodes distributed in a 200 meters by 200 meters area. In order to evaluate the network performance more precisely, we will consider more extreme cases. (2) Our new approach differs from the existing approaches, such as LEACH-C/E/F and DCHS, in that they all consider energy consumption as a factor in protocol improvement. We will explore the possibility of combining the strengths of these different approaches. (3) There is an assumption on the selection of new cluster head and key management scheme, which is the locations of nodes in a network are known. In reality this assumption may not be true. We will improve our protocol to deal with this situation.

## Acknowledgements

Foundation Items: The Natural Science Foundation of Jiangsu Province (BK2012584); The Natural Science Foundation of Changzhou (CJ20110025).



## References

- [1]. L. Cui, H. Ju and T. Li, Advances in wireless sensor networks, *Journal of Computer Research and Advances*, Vol. 42, Issue 1, 2005, pp. 163-174.
- [2]. J. Li and H. Gao, Research advances in wireless sensor networks, *Journal of Computer Research and Advances*, Vol. 45, Issue 1, 2008, pp. 1-15.
- [3]. L. Sun, J. Li, Y. Chen and H. Zhu, Wireless sensor networks, *Tsinghua University Publishers*, May 2005.
- [4]. F. Ren, H. Huang and C. Lin, Wireless sensor networks, *Journal of Software*, Vol. 14, Issue 7, 2003, pp. 1282-1291.
- [5]. B. Shen, S. Zhang and Y. Zhong, Clustering protocols of wireless sensor networks, *Journal of Software*, Vol. 17, No. 7, July 2006, pp. 1588-1600.
- [6]. Z. Yan and B. Zheng, Wireless sensor networks, *Journal of Computer Engineering and Application*, Vol. 15, 2005.
- [7]. Vlado Handziski, Andreas Kopk, A common wireless sensor network architecture, *Technical Report TKN-03-012 of the Telecommunications Networks Group*, July 2003, pp. 10-17.
- [8]. A. Durresi, Architectures for heterogeneous wireless sensor networks, in *Proceedings of the IEEE 16<sup>th</sup> International Symposium*, Vol. 2, 11-14 September 2005, pp. 1289-1296.
- [9]. S. Hussain, E. Shakshuki, A. W. Matin, Agent-based system architecture for wireless sensor networks, *Advanced Information Networking and Applications*, Vol. 2, April 2006, pp. 18-20.
- [10]. S. Muhammad, Z. Furqan, R. Guha, Wireless sensor network security: a secure Sink node architecture, in *Proceedings of the Performance, Computing, and Communications Conference*, April 2005, pp. 371-376.
- [11]. M. Yu, A. Malvankar, and W. Su, An environment monitoring system architecture based on sensor networks, *International Journal of Intelligent Control and Systems*, Vol. 10, Issue 3, 2006, pp. 201-209.
- [12]. C. Zhang, M. C. Zhou, and M. Yu, Ad hoc network routing and security: a review, *International Journal of Communication Systems*, Vol. 20, 2007, pp. 909-925.
- [13]. W. Heizelman, A. Chandrakasan, and H. Balakrishnan, Energy-efficient routing protocols for wireless microsensors networks, in *Proceedings of the Hawaii International Conference on System Sciences*, 2002.

2013 Copyright ©, International Frequency Sensor Association (IFSA). All rights reserved.  
(<http://www.sensorsportal.com>)

## International Frequency Sensor Association



**International Frequency Sensor Association (IFSA)** is a professional association, created with the aim to encourage the researches and developments in the area of quasi-digital and digital smart sensors and transducers.

**IFSA Membership is open to all organizations and individuals worldwide who have a vested interest in promoting or exploiting smart sensors and transducers and are able to contribute expertise in areas relevant to sensors technology.**

More than 600 members from 63 countries world-wide including ABB, Analog Devices, Honeywell, Bell Technologies, John Deere, Endevco, IMEC, Keller, Mazda, Melexis, Memsis, Motorola, PCB Piezotronics, Philips Research, Robert-Bosch GmbH, Sandia Labs, Yokogawa, NASA, US Navy, National Institute of Standard & Technology (NIST), National Research Council, etc.



For more information about IFSA membership, visit  
<http://www.sensorsportal.com>