An Energy Optimal Clustering Algorithm

Yang Zhong*, ZhiPing Huang and Yiyao Xu
National University of Defence Technology, China

*Email: zyangzero@163.com

Abstract: For a network, two key evaluation indicators are the network lifetime and network throughput. Different clustering algorithms can be promoted for different aspects. A reasonable clustering algorithm can effectively improve the load balancing of the network and can make the distribution of CHs (cluster heads) more uniform. Aiming at the problem of cluster preference, this article presents an improved algorithm based on the classical LEACH (Low Energy Adaptive Clustering Hierarchy) algorithm, which takes the residual Energy of nodes and the uniformity of node distribution as the primary consideration for selecting CH.

1. Introduction
Wireless sensor networks consist of a number of micro-sensor network nodes, all of which are deployed inside the detection zone. They form in the form of wireless correspondence to create multiple self-organizing networks. The purpose of this network is to work together to perceive, to collect and process the information of the sentient objects in the network and to send them to the viewer. There are typically one or more base stations (called sinks) in such networks that concentrate data collected from small sensors.

Wireless sensor networks have a large number of nodes, most of which are fixed nodes; at the same time, the node structure is simple and the power storage is extremely limited. It is precisely because of these characteristics of wireless sensor networks that scholars from various countries have proposed many routing protocols and algorithms, the most classic of which is LEACH algorithm. This algorithm is not perfect enough in clustering engineering, so network performance cannot be optimized. Based on the idea of periodicity of LEACH algorithm, this article presents a new optimization algorithm based on energy, which can effectively improve the performance of the network.

2. Problem description
2.1. Energy consumption model
Consider a network including random distribution of N sensor nodes, and its application is periodic data collection. This algorithm uses the same model as literature [1], which describes the energy consumption of wireless communication. The consumption of the node energy to transmit 1-bit data to the position at distance d is $E_{Tx}(l, d)$, which consists of the transmit circuit loss and the loss of power amplification. Depending on the distance between the two sensor nodes, the loss of power amplification uses the free space model and a multi-path decay model, as follows:

$$E_{Tx}(l, d) = \begin{cases} l \cdot E_{elec} + l \cdot \epsilon_{fs} d^2, & d < d_0 \\ l \cdot E_{elec} + l \cdot \epsilon_{amp} d^4, & d \geq d_0 \end{cases}$$

(1)
Among them, $E_{elec}$ is the dissipation energy of the transmit circuit. $\epsilon_{fs}$ and $\epsilon_{amp}$ are the coefficient under the two-channel model, the coefficients describe the energy needed to amplify the power. If the distance between two nodes d is smaller than the threshold value $d_0$, the energy amplification consumption adopts a free space model; if d is bigger than or the same as $d_0$, a multipath decay model is adopted, where $d_0$ is determined by equation (2).

$$d_0 = \frac{\epsilon_{fs}}{\epsilon_{amp}}$$  (2)

The energy consumed by the node to accept 1 bit data is as shown in equation (3).

$$E_{Rx} = l * E_{elec}$$  (3)

In addition, data needs to consume a certain amount of energy when it is fused. The energy consumption of the CH for fusing the member signal and self-signal into a valid signal is as shown in equation (4).

$$E_C = (M + 1) * E_{DA} * l$$  (4)

Where $E_{DA}$ means the energy’s consumption for the fusion unit data signal, and M is the members’ number in the cluster. Therefore, the total energy consumption of the CH can be expressed as equation (5):

$$E_C = (M + 1) * E_{DA} * l + E_{TX} + E_{RX}$$  (5)

2.2. Network Load Balancing Problem

The LEACH algorithm is a relatively classic clustering algorithm, but it has many limitations. When the LEACH algorithm selects the CH, since the CH is obtained by comparing the random number with the threshold value, neither the CHs’ number nor the CH distribution can be the same as the preset optimal value[2]. If the selected CHs’ number is too small or the CHs are unevenly distributed, the CHs in some regions will be too sparse, so that the CH will die very quickly according to the relationship between energy consumption and distance, which is very unfavorable for network survival. In addition, the choice of CHs can’t fully consider the remaining energy of the node, then this choice will lead to the death of nodes with lower energy, and the survival time and throughput of the network will be limited.

3. Introduction to the algorithm

3.1. Introduction to LEACH algorithm

The LEACH algorithm is divided into 2 steps: the cluster establishment step and the ready step[3]. In the establishment phase, every node produces a random number between 0 and 1, and then compares numbers with the threshold. If the number is lower than the threshold, it is the CH node, otherwise it is a normal node. The threshold is determined by equation (6), as follows:

$$T(n) = \begin{cases} \frac{p}{1-p\times(r \mod \frac{1}{p})} & \text{if } n \in G_r \\ 0 & \text{others} \end{cases}$$  (6)

Where $p$ is the ratio of the network’s number of expected clusters to the number of total nodes in the network (generally the best value is about 5%), r is the number of rounds completed, and $G_r$ is a group of nodes that don’t become CH per $1/p$ round.

After the CH selection is completed, the ordinary node joins the nearest CH cluster by judging the distance between itself and the CH, and then the network enters the ready step. In the ready phase, every member node transmits data via the time slot devided by the CH, and goes to sleep state for the rest of the time. The CH always listens and receives data from each member, and integrates the received data after a certain time interval and then sends it to the Sink node.

$$k_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \frac{\epsilon_{fs} M}{\epsilon_{amp} d_{toBS}^2}$$  (7)

Where $M$ is the side length of the network coverage area, $N$ is the total number of nodes, and $d_{toBS}^2$ is the average distance from all nodes to the sink node. $k_{opt}$ is the calculated optimal number of
CHs of the current network. Each round will update the value of $k_{opt}$ once, which will decrease as the decreases of the number of network nodes.

3.2. Improved algorithm – Lan

The same as the LEACH algorithm, the execution process of the improved algorithm is also periodic, and each round is also divided into the above two stages. To reduce resource overhead, the duration of the stabilization step is a lot longer than this of the setup step, so the overhead of the setup phase is negligible. The energy of remaining nodes and the distribution of CH should be considered in selecting CH. Moreover, the energy consumption of the CH fusion information should be considered. Therefore, the selection process of the CHs is as follows:

Estimate the optimal number of CHs of the current network according to the number of total nodes [4].

The entire network area is partitioned according to the CHs’ number, and each area is square. Since each area is square, the CH needs to be adjusted according to the number of areas, and a number closest to the most suitable number of CHs is selected as the best number of clusters. The resulting best cluster number is:

$$\text{Number of CHs} = (\text{round}\sqrt{\text{Optimal number of cluster heads}})^2$$

Where round refers to rounding off an integer.

Selecting the node with the most energy as the CH by comparing the remaining energy of the nodes in the already divided region (if the distance from the ordinary node to the sink node in the area is smaller than the distance to the CH, then these ordinary nodes directly send their own information to the sink node).

After waiting for all the partitions to select their own CH, terminate the current round of cluster selection process. The remaining energy is calculated in each round and the number of CHs and partitions are updated, and then the CH selection is repeated.

The process of CH chosen is shown in Figure 1.

**Figure 1.** CH selection process
4. Algorithm simulation and analysis

In the simulation, we made a 100×100 square area and randomly generated 100 nodes inside it. The Sink node is inside the network area with a coordinate of (50,50) and we can ignore its energy consumption[5]. The energy of 100 nodes is randomly generated, and the network simulation result is shown in Figure 2.

![Figure 2(a). Distribution of the Lan algorithm with 100 nodes](image)

![Figure 2(b). Distribution of the LEACH algorithm with 100 nodes](image)

Among them, the "+" node represents the CH node and the "o" node represents the ordinary node. It can be seen from the figure (a) and (b) that the number of CH nodes of the Lan algorithm and the LEACH algorithm are similar, but the CH distribution of the Lan algorithm is superior to the CH distribution of the LEACH algorithm.

The parameters of the simulation are shown in Table 1.

| Parameter name | Parameter value |
|----------------|-----------------|
| Packet length  | 500Bytes        |
| $d_0$          | $\sqrt{(\epsilon_{fs}/\epsilon_{amp})} \approx 87$ |
| $\epsilon_{fs}$| 10nJ/bit/m²      |
| $\epsilon_{amp}$| 0.013pJ/bit/m⁴  |
| $E_{DA}$       | 5nJ/bit/signal  |
| $E_{elec}$     | 50nJ/bit        |

Table 1. Lan algorithm simulation parameter configuration

Figure 3 shows the total energy of the remaining nodes as a function of the number of rounds when the CH ratio is 0.1.
Figure 3. Total energy of remaining surviving nodes

From Figure 3, we can see that although the total energy of the Lan algorithm is lower than the total energy of the LEACH algorithm at the beginning (the LEACH algorithm defines the energy values of the advanced node and the common node at the beginning, where the energy of the advanced node is 0.6J and the energy of the ordinary node is 0.5J), but because the CH distribution is relatively uniform, the distance from each node to the CH is relatively small, and the energy loss will be less, so the energy will gradually accumulate during the round robin process, and finally the energy will be more than the LEACH algorithm.

Figure 4 shows the change of the number of remaining nodes with the number of rounds when the optimal CH ratio is 0.1.

Figure 4. Number of remaining nodes

From Figure 4 we can see that the number of dead nodes in the LEACH algorithm has a mutation process, and the death node curve of the Lan algorithm is relatively flat. This is because the LEACH algorithm assigns a relatively high energy value to each node at the beginning, and the energy value of the LAN algorithm is a random number between 0 and 1, and the energy is relatively uneven. Because the node initialization energy is higher in the LEACH algorithm, the number of node deaths of LEACH is relatively small at the beginning of the round robin. However, as the number of round robin increases, the unevenness of the distribution caused by the randomness of the CH selection in the LEACH algorithm leads to the unreasonable utilization of the energy nodes (because the distance is not optimal), so the wasted energy more and then the number of deaths of the remaining nodes will
have a cliff-like decline. However, due to the uniformity of the CH node distribution and the optimization of the CH selection (choose the CH according to the remaining energy), the LAN algorithm will consume more energy preferentially, so that the node with low energy is protected. The node presents a state of uniform death.

Figure 5 shows the change of network throughput with the number of rounds.

![Figure 5. Network throughput](image)

From Figure 5 we can see that as the number of rounds increases, the throughput of the Lan algorithm will gradually exceed the classical LEACH algorithm. This is because with the increase of the rounds’ number, the remaining nodes of the Lan algorithm will be more than the LEACH algorithm, and the throughput of the network is exactly proportional to the number of nodes in the network. Therefore, the throughput of the Lan algorithm will gradually exceed the LEACH algorithm in the later stage.

5. Conclusion
In comparison with the LEACH algorithm, the Lan algorithm prolongs the network lifetime, solves the problem of uneven CH distribution and increases the network throughput, but shortens the network stability period (the first death node appears earlier). Each algorithm has its own strengths and weaknesses, and we can make reasonable choices according to different actual situations.

Although the Lan algorithm has been improved, there are still many aspects that have not been considered. For example, although the CH is partitioned before it is selected, the density of node distribution is not considered. Moreover, this algorithm only shows better network characteristics in a network with a smaller coverage area, but is not suitable for a network with a relatively large coverage area. In areas where nodes are densely distributed, the Lan algorithm can show more obvious advantages.

Acknowledgments
This work was supported by the Natural Science Foundation of Hunan Province under Grant No.2018JJ3607, National Natural Science Foundation of China under Grant No.51575517, National Technology Foundation Project under Grant No.181GF22006, and Frontier Science and Technology Innovation Project in the National Key Research and Development Program under Grant No.2016QY11W2003.
References

[1] Heinzelman W B, Chandrakasan A P, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks[J]. IEEE Transactions on Wireless Communications, 2002, 1(4):660-670.

[2] Heinzelman W R, Kulik J, Balakrishnan H. Adaptive protocols for information dissemination in wireless sensor networks[C]/ Acm/ieee International Conference on Mobile Computing & Networking. 1999.(under review; accepted; in press).

[3] Younis O, Fahmy S. HEED: A Hybrid, Energy-Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks[J]. IEEE Transactions on Mobile Computing, 2004, 3(4):366-379.

[4] Kim D H, Lee S H, Cho W D. Energy-Efficient Clustering Algorithm in Wireless Sensor Networks[J]. Transducer & Microsystem Technologies, 2008, 2(7):525-527.

[5] Rui W, Guo-Zhi L, Hong-Jie Z. Adaptive clustering algorithm based on node partition for wireless sensor networks[J]. Journal of Dalian Maritime University, 2008, 34(1):45-44.