A Multi-hop Routing Mechanism Based on Local Competitive and Weighted Dijkstra Algorithm for Wireless Sensor Networks

Qingwei Liu and Mandan Liu
Department of Automation, East China University of Science and Technology, 130 Meilong Road, Shanghai, China
Email: liumandan@ecust.edu.cn

Abstract. Extending the lifetime of the networks with limited energy has always been an important issue in the research of wireless sensor networks. Many researchers have contributed lots of routing techniques to improve energy efficiency and extend lifetime of the networks. In this paper, we proposed a Multi-hop Routing mechanism Based on local Competitive and Dijkstra algorithm (MRBCD). In MRBCD, CHs are elected by local competitive and CHs send data to the base station by their shortest energy cost path. By running local competitive mechanism, CHs in the network can be evenly distributed, which will balance energy consumption between sensor nodes. The inter-cluster routing mechanism can minimize the energy consumption in the inter-cluster communications. By combining the two mechanisms, MRBCD algorithm can balance energy consumption and improve energy efficiency, which will extend the lifetime of wireless sensor networks. Simulation results show that MRBCD algorithm extended the lifetime of the networks than LEACH and Multi-LEACH.

1. Introduction
A typical Wireless Sensor Networks (WSNs) consists large number of low cost, low power sensor nodes and a Base Station (BS). The sensor nodes are usually deployed either inside or very close the event and capable of sensing the physical environment and sending the data to the base station though routing mechanism [1]. Usually, when a node is running out of energy, there is low possibility of recharging or replace the node because of the rough environment and random deployment. Hence, extending the lifetime of the networks with limited energy has always been an important issue in the research of WSNs. According to the energy model of WSNs [2], communicating part consumes more energy than other parts, which makes routing algorithm an important role in extending lifetime of WSNs. Many routing algorithms has been proposed for wireless sensor networks [3-6]. Low-Energy Adaptive Clustering Hierarchy (LEACH) [7] is one of the most popular routing algorithms in WSNs. It provides a very good routing architecture. One of the main limitations of LEACH is that it is a single-hop network. All the CHs send data to the base station directly, which is not practical in real large scale wireless sensor networks.

For a large scale wireless sensor networks, there exist many restrictions and limitations in the design of routing algorithms. First, the sensing field are always very large. Not all the sensor nodes can communicate with the base station directly because of their communication ability. Data packets produced by sensor nodes far away from the base station needs to be relayed by other nodes. Second, since all the nodes need to transit their data to the base station, there will exist congestion close to the
base station. Sensor nodes that relaying more packets will consume more energy and dies quickly than the others. It is very important for the network that nodes consume energy evenly. The relay nodes need to be evenly distributed in the network.

In terms of the upper issues, we proposed a Multi-hop Routing mechanism Based on local Competitive and Dijkstra algorithm (MRBCD) for large scale wireless sensor networks. In MRBCD, CHs are elected by local competitive and CHs send data to the base station by their shortest energy cost path. By repeating this process, all the sensors in the network can almost consume energy synchronously and the lifetime of the networks can be extended.

2. MRBCD Algorithm

2.1. A Probability Description of Wireless Sensor Networks
Consider a homogeneous wireless sensor network. All the sensor nodes can be chosen as cluster head. Suppose sensor node $i$ have a probability $p$ to be chosen as a CH. Then the probability of it will not be chosen as CH is $1-p$. Whether sensor node $i$ will be chosen as CH can be considered as a Bernoulli distribution. Let $X_i$ denotes the random variable of whether node $i$ will be chosen as CH. Then $X_i \sim Bern(p)$. Suppose there are $N$ sensor nodes in a network. Every sensor node has the same probability to be chosen as CH. They all follow the same Bernoulli distribution. Every node can be chosen to be CH or not independently. Hence, the number of CHs (We denoted it as a random variable $X$) is distributed as a Binomial distribution. Here, $X \sim Bin(N, p)$. The expectation of $X$ is computed as follows:

$$E(X) = N * p$$

$E(X)$ presents the average number of CHs in the network. Suppose we want $k$ cluster heads in each round, i.e., $E(X) = k$. Then,

$$p = k / N$$

By setting the probability $k/N$ to every sensor node, we can get $k$ CHs after the selection process.

2.2. An Energy Efficient CH Election Mechanism
However, the last model is just an ideal one. The positions of CHs are totally randomly distributed. If the distance between CHs are very big or very small, some CHs will consume energy quickly than the rests, which will lead to the quickly death of the whole network. In order to maximize the lifetime of the networks, one of the efficient methods is to let the CHs evenly distributed. If CHs are chosen through totally randomization, the distribution can be very uneven, such as in figure 1. Instead of choosing $k$ CHs from the whole sensing field (In this paper, we consider a square sensing area with side $a$), we can divide the sensing area into $k$ circling area with radius $R_c$ and chose one CH from each area, such as in figure 2. $R_c$ is computed through equation 3. We assume the whole area is completely covered by the $k$ circles. Through this mechanism, the CHs can have a rather evenly distribution.

$$R_c = \sqrt{a^2 / k * \pi}$$

In order to construct an architecture similar to figure 2, we adopted a local way to choose CHs in this paper. Each node is assigned a radius $R_c$. The node within a node’s radius is its neighborhood. We use $n_i$ to denote node $i$’s active neighbourhood number (including itself). A node is considered active if it is able or permit to choose for CH. If a node has been CH in last round, it will not be considered for CH in next few rounds until all its neighbourhood has been CH in last epoch. If a node is running out of energy, it will be considered non active forever. Hence, $n_i$ changes with rounds. Instead of using $N$, node $i$ uses $n_i$ to compute its probability $p_i$ to becoming CH through equation 2 and expected CH number of each local competitive area is 1. We call this method 1. However, in order to choose
CH more energy efficiently, we expect 2 candidate CHs in a local circle and choose the final CH by comparing their residual energy instead of expecting one final CH directly. Hence,

$$p_i = \frac{2}{n_i}$$  \hspace{1cm} (4)

We call this method 2. Method 2 can decrease the probability of no CHs in a local competitive area. By applying method 2, wireless sensor networks will choose CHs more energy efficiently and well distributed.

Figure 1. A random distribution of CHs.  \hspace{1cm} Figure 2. Evenly distributed CHs.

2.3. An Inter-Cluster Routing Mechanism

Inter cluster routing mechanism are very important when the networks scale is very large. CH transmit data to base station directly will consume a lot of energy in this situation. We apply dijkstra algorithm to find the shortest energy path from the CH to base station. Once CHs are elected, each of them will broadcast a small message to declare itself. Other CHs will transmit this message until it reached base station. Since this message is very small, the energy increase by transmitting this message will be tolerable compared to the energy conservation by using the shortest path to transmit data. Once base station received all the CHs message, it will form a connection graph $G=\langle V,E \rangle$ with all the CHs and base station as vertex. Then it finds the shortest energy consumption path of every CH to base station based on dijkstra algorithm and broadcasting the shortest path information to every CH. However, instead of using Euclidean distance, we use an energy cost weight specified for WSNs. In this paper, we apply an energy consumption model which has been referred from [2]. Suppose the distance between cluster head $i$ and cluster head $j$ is $d_{ij}$. If $d_{ij} < d_o$, then the weight on side $\langle i,j \rangle$ is $\varepsilon_{fs}d^2$; If $d_o < d_{ij} < R_{max}$, then the weight on side $\langle i,j \rangle$ is $\varepsilon_{fs}d^4$; If $d_{ij} > R_{max}$, we think $i$ and $j$ are unconnected and the weight on side $\langle i,j \rangle$ is infinite. $R_{max}$ is sensor's maximum communication radius. By running dijkstra algorithm to find shortest cost path between each CH and base station in the graph, every CH will find its best relay node. Algorithm 1 described the shortest cost path finding algorithm.

2.4. Algorithm Operation Process

Similar to LEACH, time line of MRBCD is divided into rounds. In each round, CHs are re-elected by cluster formation mechanism and multi-hop routes are remade by inter-cluster routing mechanism.

Cluster formation mechanism: After initialization, each node broadcasts a NODE_INFO message (contains its identity, residual energy and its distance from BS) with signal large enough to reach its neighbouring nodes within competing radius $R_c$. When a node receives NODE_INFO messages from other node, it calculates distance between them by RSSI and stores the message into its neighbour table. Every node uses this table information to calculate its probability to be candidate CH by equation 4 and decides to be candidate CH or not. Once a node decides to be a candidate CH, it
broadcast a CANDIDATE message to its neighbour nodes. If there are other candidate nodes within its radius $R_c$, they will compete for the real CH by comparing their residual energy and the winner declare itself to be real CH. The real CH broadcast CLUSTER_HEAD (containing its id, residual energy and distance from BS) message to the nodes within its radius $R_{adv}$ and non-CH nodes choosing the nearest CH to join. Flowchart of process is shown in figure 3.

Algorithm 1 Shortest Cost Path Finding Algorithm

Input: Connection Graph formed by all the CHs and the BS as vertices and energy consumption cost as edges.

Output: Every CHs shortest energy cost to BS and the relative best relay node.

1: Let cost of all CHs from BS=Infinity;
2: Visit the unvisited CHs with the smallest known cost from the start vertex (BS or CH);
3: For the current CH, examine its unvisited neighbours;
4: For the current CH, calculate cost of each neighbour from the BS;
5: If the calculated cost of a CH is less than the known cost, update the shortest cost;
6: Visit the unvisited CHs with the smallest known cost from the BS;
7: Update the previous CH for each of the updated costs;
8: Add the current CH to the list of visited CHs;
9: Repeat from Step 3 until all the CHs are visited.

Inter-cluster routing mechanism: In this paper, every CH’s shortest path to base station is calculated by base station. Hence, all the CHs need to send their CH information to base station. As we described upper, once elected as real CH, it will broadcast a CLUSTER_HEAD message to the nodes within its radius $R_{adv}$. Non-CH nodes uses this information to decide which cluster to join. If a CH $j$ receives other CH’s CLUSTER_HEAD message, it will calculate their distance by RSSI and store the CH message into its neighbour CH table. CHs in this table will be CH $j$ tentative relay node. Theoretically, the largest distance between two CHs is $2 * R_c$. Here, we set $R_{adv} = 3 * R_c$ to ensure that every CH at least have one tentative relay CH. After clusters formed, each CH sends its neighbour CH table to its tentative relay node closest to BS. Finally, the BS will get all the CH information and their connection graph. The BS station runs the dijkstra algorithm to find shortest cost path for every CH and then broadcast the route information to all the CHs. The flowchart of this process is shown in figure 4.

Figure 3. Flowchart of cluster formation mechanism.

Figure 4. Flowchart of inter cluster routing mechanism.
3. Evaluation
We compare our proposed algorithm MRBCD with LEACH, Multi-LEACH [8]. We consider a scenario with 200 nodes randomly distributed in a 200×200 m sensing area. The BS is located at (200,100)m. Initial energy is 0.5J. Same energy consumption model is used as [2]. Table 1 shows the simulation parameters. Figure 5 shows the sensing field and node distribution.

| Table 1. Simulation Parameters. |
|---------------------------------|
| Parameter              | Value           |
| Sensing field          | 200m×200m       |
| BS position            | (200,100)m      |
| Number of nodes        | 200             |
| Data packet            | 4000 bytes      |
| Signal packet          | 100 bytes       |
| $E_{elec}$             | 50 nJ/bit       |
| $\varepsilon_{fs}$     | 10 pJ/bit/Signal|
| $\varepsilon_{mp}$     | 0.0013 pJ/bit/m4|
| $\varepsilon_{da}$     | 5 nJ/bit/Signal |

We first compare the energy consumption per round, as shown in figure 6. From figure 6, we know that energy consumption per round in LEACH and Multi-LEACH varies more largely than in MRBCD. That’s because in LEACH and Multi-LEACH, CHs are randomly chosen and the number of CHs can vary largely. The number of CHs will affect the energy consumption in the network. While in MRCBD, the number of CHs is constrained by the compete radius. We can also see that MRBCD and Multi-LEACH consumes less energy per round than in LEACH. Although energy consumption per round in MRBCD is not always the least, it consumes similarly in every round, which will extend the lifetime of the networks.

![Figure 5. Sensing field.](image1)

![Figure 6. Comparison of energy consumption per round.](image2)

Figure 7 shows the comparison of number of alive nodes per round in the three algorithms. The First Node Dies (FND) of LEACH, Multi-LEACH and MRBCD in the figure are 381, 607 and 741. This shows that our proposed algorithm MRBCD can extends the lifetime of the networks 95% and 22% than LEACH and Multi-LEACH.

Figure 8 shows the residual energy of nodes in the FND round. We can see that most of the energy are exhausted in MRBCD when the first node dies. While in LEACH and Multi-LEACH, there are
still much energy in other nodes when the first node dies, which means energy are not synchronically dissipated in the nodes in these two algorithms. This also explains why MRBCD can extend the lifetime of the networks.

**Figure 7.** A random distribution of CHs.  
**Figure 8.** Evenly distributed CHs.

### 4. Conclusion

In this paper, we proposed a Multi-hop Routing mechanism Based on local Competitive and Dijkstra algorithm (MRBCD). In MRBCD, CHs are elected by local competitive and CHs send data to the base station by their shortest energy cost path. In MRBCD, CHs can be evenly distributed and the energy consumption in the inter-cluster communications can be minimized. Simulation results show that all the sensors in the network can almost consume energy synchronously in our MRBCD algorithm and the lifetime of the networks can be extended.

### References

[1] Akyildiz I, Su W, Sankarasubramaniam Y and Cayirci E 2002 A survey on sensor networks *IEEE Communications Magazine* 40 (8) 102-114.

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

[3] Nayak A and Devulapalli A 2016 A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime *IEEE Sensors Journal* 16 (1) 137-144.

[4] Zhang S, Tao Y, and Yang F 2018 Fuzzy logic-based clustering algorithm for multi-hop wireless sensor networks *Procedia Computer Science* 131 1095-1103.

[5] Meng X, Shi X, Wang Z, Wu S and Li C L 2016 A grid-based reliable routing protocol for wireless sensor networks with randomly distributed clusters *Ad Hoc Networks* 51 47-61.

[6] Wang C, Zhang Y, Wang X and Zhang Z 2018 Hybrid multihop partition-based clustering routing protocol for WSNS *IEEE Sensors Letters* 2 (1) 1-4.

[7] Heinzelman W, Chandrakasan A and Balakrishnan H 2000 Energy-efficient communication protocol for wireless microsensor networks *Proceedings of the 33rd IEEE Annual Hawaii International Conference on System Sciences* p 10.

[8] Fan X and Song Y 2007 Improvement on leach protocol of wireless sensor network *Proceeding of 2007 International Conference on Sensor Technologies and Applications (SENSORCOMM 2007)* pp 260-264.