Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 15, no. 5, May 2019, pp. 1445-1452
DOI: 10.23940/ijpe.19.05.p21.14451452

Improved Clustering Optimization Algorithm for Wireless Sensor Network Energy Balance

Jinyu Li* and Jun Li

School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China

Abstract

To get over the limited energy of nodes and unbalanced energy consumption in wireless sensor networks (WSN), this paper puts forward a WSN clustering routing algorithm based on weight function timing. The algorithm was applied to build the weight function between node aggregation degree and residual energy. Then, the weight function was based on producing the timing time for all nodes. Both the iteration number and the energy consumption were reduced in cluster head selection. At the same time, the node energy consumption rate and the distance from the node to the sink node were taken into consideration. Next, the reasonable cluster head was chosen according to each node's weight function value and the timing time. In the periodic clustering process, the proposed algorithm removes the aggregation degree exchange between the nodes, thus reducing the network traffic and lowering the network energy consumption. Simulation results show that the algorithm achieves excellent cluster convergence and stable cluster size.

Keywords: wireless sensor network (WSN); routing algorithm; timing of weight function; cluster head selection

(Submitted on December 18, 2018; Revised on January 12, 2019; Accepted on February 16, 2019)

1. Introduction

WSN is a large-scale, infrastructure-free network established through the self-organization of thousands of sensor nodes. With the rapid development of sensors, microelectronics, and wireless communication, recent years have witnessed the maturation of embedded systems capable of environmental perception, data processing, and data transmission, making it possible to create WSN nodes at a low cost. The sensor nodes support data collection, data processing, and information interaction through mutual cooperation, laying the basis for real-time monitoring of the target area. Hence, WSN enjoys wide application prospects in military, medical care, marine monitoring, forest fire prevention, and many other fields [1-4] Under the effect of volume, cost, and power consumption, the WSN nodes, often battery-powered, are limited in computing, storage, and communication abilities [5-8]. With the aim to prolong network life and balance energy consumption, research on WSN has been focusing on how to make full use of the limited network resources and strike a balance between network loads [9-11].

This paper adopts and improves the adaptive windowed Fourier transform (AWFT), a clustering algorithm that is based on the node aggregation degree and the timing of the weight function [12-15]. First, the aggregation degree and residual energy were calculated for each node to obtain a specific threshold. Then, the start time of each node was synchronized, and the node with the highest threshold was chosen as the cluster head. In this case, the adjacent node stops counting and joins the cluster centring on the cluster head [16-20].

2. System Model

The WSN energy consumption was measured by the energy consumption model, which focuses on sensor nodes' data transmission, reception, and fusion. The WSN nodes' energy consumption mainly takes place in wireless transmission and power amplification. The mode of power amplification should be determined based on the transmission distance [21-26].

* Corresponding author.
E-mail address: lijinyu@mail.lzjtu.cn

The free space transmission mode should be adopted if the transmission distance is shorter than d_0; otherwise, the multi-path attenuation mode should be adopted. The energy consumption in the process of transmitting a *k*-bit message between two nodes over a distance d can be expressed as follows:

$$\begin{aligned} E_{Tx}(k,d) &= kE_{elec} + k\varepsilon_{amp}d^{\beta} \\ &= \begin{cases} kE_{elec} + k\varepsilon_{fs}d^2, d \le d_0 \\ kE_{elec} + k\varepsilon_{mp}d^4, d > d_0 \end{cases} \end{aligned} \tag{1}$$

Where E_{elec} represents the node circuits energy consumption coefficient, ε_{fs} represents the loss incurred in free space transmission, ε_{mp} is the loss incurred in multipath attenuation transmission, and d_0 is the distance for switching between two modes. The threshold can be obtained by Equation (2) below:

$$d_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \tag{2}$$

The energy consumption in data reception can be calculated by Equation (3):

$$E_{Rx} = kE_{elec} \tag{3}$$

Let us denote the energy consumption of data fusion per unit length as E_{DA}. Then, the energy consumption in data fusion at the cluster head over a length of l can be expressed as:

$$E_A = E_{DA} \times l \tag{4}$$

3. Clustering Algorithm

3.1. Node Information Acquisition Process

In the initial stage, all nodes are deployed in the target region randomly, each of which broadcasts its own node information. The broadcast information contains the node ID only. After receiving information from a node via hop-to-hop communication, the adjacent node will update the adjacent node table in that node, marking each other as the adjacent node, and broadcast its own ID and the known adjacent node to other nodes. By the time all nodes have broadcasted their own information once, every node in the network will have its own adjacent node table.

Assuming that *N* nodes, with the communication radius of *R*, are distributed in the target region *S* randomly, the average number of adjacent nodes $|NBR|$ of any node can be expressed as:

$$|NBR| = \left\lfloor \frac{N\pi R_2}{S} \right\rfloor \tag{5}$$

The communication cost of the whole network in the node information acquisition can be expressed as:

$$C_{comm} = N(N-1)|NBR| \tag{6}$$

Assuming that the bandwidth of each node communication module is *W* and the packet size is *M* bits, the computation overhead C_{comp} can be expressed as:

$$C_{comp} = \frac{mC_{comm}}{w} \tag{7}$$

3.2. Node Aggregation Degree

By the definition of graph theory, make $G = (V, E)$ an undirected network graph, where V represents the set of all nodes and E represents the set of all edges. For any $i \in V$, denote the set of its neighboring nodes as NBR_i. Then, the aggregation degree of node i is $|NBR_i|$ (excluding the ring). For node $j \in NBR_i$, if and only if $i \in NBR_i$, the aggregation degree of the node i can be obtained:

$$d_i = \frac{\left|e = (\upsilon, \omega) \in E, (\upsilon, \omega) \in \{i\} \cup NBR_i\right|}{\left|NBR_i + 1\right|} \tag{8}$$

Node i and its adjacent nodes form a sub graph $G^{\cdot} = (\upsilon, \omega)$. With E' as the set of all edges in the subgraph G', the aggregation degree of node i is the ratio of the set of all edges to the set of all nodes. For node A, its set of adjacent nodes is $\mathrm{NBRA} = \{B, C, D, E\}$, its sub graph is $G' = \{A, B, C, D, E\}$, and its set of edges in the sub graph is $E' = \{(A, B), (A, C), (A, E), (B, C), (A, D), (D, E)\}$. Thus, the aggregation degree of node A is d_A.

The mathematical definition of the node aggregation degree is given by Equation (8), and the equivalent transformation is needed in practical applications.

Suppose node j is an adjacent node of node i and the coincidence degree between nodes i and j is $d_0 = \{NBR_i' \cap NBR_J'\}$. Make NBR_i' the set of adjacent nodes of node i excluding node j, and NBR_J' is the adjacent node set of node j excluding node i. Then, the number of paths between the neighboring nodes of node i is calculated as:

$$d_{Ni} = \sum_{j=1}^{k} d_{0j} / 2 \tag{9}$$

Where k represents the number of adjacent i. Therefore, the aggregation degree of node i in Equation (9) can also be expressed as:

$$d_i = \frac{d_{Ni} + |NBR_i|}{|NBR_i| + 1} \tag{10}$$

3.3. Node Clustering Algorithm

After the calculation of node aggregation degree, it is necessary to exchange the information and compare the aggregation degree of each node, thereby obtaining the cluster head. Thus, the node clustering algorithm was introduced to calculate the node aggregation degrees, initiate time synchronization, and select the cluster head according to the time of arrival. The detailed process is as follows:

(1) At the beginning of a new round of clustering, all nodes are pending nodes. The maximum time length is set to $T_{\max}$, the clustering time is T_i, each node's initial energy is E_0, each node's residual energy is E_{ri}, and the aggregation degree of each node is d_i. In the first round of clustering, $E_{ri} = E_0$.

(2) Before the clustering, the aggregation degree of each node d_i was calculated in reference to Equation (10), and the weight of each node was obtained based on d_i. The weight function of node i is as follows:

$$\omega_i = f(d_i, E_{ri}) = \alpha d_i + (1 - \alpha)\frac{E_{ri}}{E_0} \tag{11}$$

Where $\alpha \in (0,1)$ is the pre-set weighting parameter of the system.

(3) After obtaining the node weights, the time synchronization was initiated for all nodes. The clustering time T_i of node i is expressed as:

$$T_i = \frac{T_{\max}}{\omega_i} \tag{12}$$

(4) At the TOA for node i, the node was declared as a temporary cluster head and asked to send a declaration message to its adjacent node $j \in NBR_i$, creating an adjacent node table. The element number in the table is $C_m(i)$, and $C_m(i)$ is 0. In the meantime, the cluster head was initiated to confirm the timing.

(5) When the neighbor node j of node i received the cluster head declaration information sent from node i, node j immediately stopped the counting, marked node i as the cluster head node of node j, and initiated the cluster confirmation timing.

(6) When the cluster confirmed the timing arrived, node j sent the request message to i.

(7) When i received the request message from node j, it updated its adjacent node table and $C_m(i) = C_m(i) + 1$.

(8) When the cluster head of node i confirmed the timing arrived, one of the following three options was adopted based on the value of $C_m(i)$:

If $C_m(i) > 0$, node i became the final cluster head;

If $C_m(i) = 0$ and node i had received the declaration message from other nodes, node i became a cluster node, randomly selected a cluster head from the declaration message, and sent a request message to the randomly chosen cluster head.

If $C_m(i) = 0$, node i had not received the declaration message from any other node, and the number of elements in the adjacent node table was $|NBR_i|$, then node i was an isolated node and all the nodes were clustered separately. Supposing $|NBR_i| \neq 0$ and node i randomly selected the cluster head from the request message sent from the adjacent node j, the node joined the cluster of j, and the data of i was forwarded by j to the cluster head of j.

(9) After the timing of all nodes, the round of clustering was completed, and the number of rounds reached $R_c = R_c + 1$.

(10) Upon the formation of the logical cluster, the nearest adjacent node was taken as the cluster head and sent its own data to the base station.

3.4. Further Analysis

This section further explains the weighting parameter α in the clustering algorithm, details the timing of Steps (4) and (5), and discusses the three cases in Step (8), aiming to reveal how the frequency and precision of time synchronization frequency affects the performance of the algorithm.

In Equation (11), the weight parameter α is adopted to balance the influence of residual energy and node aggregation degree on weight function's value, which is mainly dependent on the network topology. The value range of α is negatively correlated with node density. When the node density increases, the value range shifts to the lower half of (0,1); when the node density decreases, the value range shifts to the upper half of (0,1). In practice, the value of α should be selected according to network topology.

3.5. Energy-based Interactive Cluster Head Selection

A new energy-based iterative cluster method was put forward to ensure the even distribution of cluster head and reduce energy consumption in cluster head selection. On one hand, the energy consumption was lowered by reducing the iteration

number; On the other hand, premature death of cluster heads was prevented by adoption the optimal cluster head considering such two factors as node energy consumption rate and the distance between the node and the sink node, thus eliminating the phenomenon of energy hole.

In the iterative process, all nodes broadcasted information on their own residual energy, energy consumption rate, and the distance between the node and the sink node. Nodes receiving information selected optimal cluster head according to Equation (13).

$$factor = \alpha \frac{E_{residual}}{E_{\max}} \times \frac{E_{avecons}}{E_{consume}} + (1-\alpha) \times (1 - \frac{d_{toSINK}}{d_{toSINK-MAX}}) \tag{13}$$

Where α is the weight of selection factors; $E_{consume}$ is the node energy consumption; $E_{avecons}$ is the average energy consumption of the network. $E_{avecons}$ represents the average energy consumed by the network.

If the energy was consumed too fast, the cluster head would die prematurely and turn into a failure node. The accumulation of failure nodes would result in an energy hole. Therefore, the energy consumption factor $\frac{E_{avecons}}{E_{consume}}$ was introduced to reflect the energy consumption rate. The *factor* is defined as the ratio of the network average energy consumption and the node energy consumption in round R. Heavy energy consumption in round R means fast energy consumption rate and small energy consumption factor $\frac{E_{avecons}}{E_{consume}}$. In this case, the *factor* of the next round would decrease.

The *factor* is also affected by the distance between the node and the sink node. If the cluster head was too far away from the sink node, the data transmission would consume lots of energy, cause premature death of cluster head, and turn the cluster head into a failure node. Then, the cluster heads around the failure node had to transmit the packets before the failure node, in addition to their own packets to the base station. Thus, the surrounding cluster heads would consume more energy and tend to fail, resulting in an energy hole.

$1 - \frac{d_{toSINK}}{d_{toSINK-MAX}}$ was introduced to solve the problem. The parameter is seen as the ratio of the distance from the node to the sink node d_{toSINK} to the maximum distance from all nodes to the sink node $d_{toSINK-MAX}$. The larger the d_{toSINK}, the smaller the value of $1 - \frac{d_{toSINK}}{d_{toSINK-MAX}}$ and the smaller the factor.

In general, the factor is too simple for the cluster radius to be reasonable. Hence, a new method of cluster radius calculation is adopted:

$$R_{NEW} = (1 + \frac{d_{toSINK} - E(d_{toSINK})}{d_{toSINK-MAX} - d_{toSINK-MIX}}) \times (1 - \frac{N_{nbr}}{(1-\eta) N_{alive} \times R}) \times R \tag{14}$$

Where d_{toSINK} is the supposed distance of all cluster heads to the sink node; N_{nbr} is the node aggregation degree, that is, the adjacent nodes number around the node; N_{alive} is the present alive nodes; and R is the maximum cluster radius.

With the introduction of $1 + \frac{d_{toSINK} - E(d_{toSINK})}{d_{toSINK-MAX} - d_{toSINK-MIX}}$, the cluster radius was automatically adjusted based on the distance from the node to sink node. If the node was far from the sink node, both d_{toSINK} and $1 + \frac{d_{toSINK} - E(d_{toSINK})}{d_{toSINK-MAX} - d_{toSINK-MIX}}$ had a large value. Then, the cluster radius was small near the sink node and large away from the sink node.

In addition, the cluster radius should also reflect the distribution density of the nodes, a guarantee for the rationality of the aggregation degree of network nodes. Since the node density is negatively correlated with cluster density, there would

be fewer nodes in the cluster and lower energy consumption in data aggregation. In this case, the cluster head was prevented from being a failure node, thus avoiding the energy hole phenomenon. Therefore, $(1-\eta)N_{alive}$ was introduced to the calculation. At a high node density, the value of $\frac{N_{nbr}}{N_{alive}}$ was large, and the value of $1-\frac{N_{nbr}}{N_{alive}}$ was small. In this way, the node aggregation degree was controlled in a proper range, laying the basis for balanced energy consumption.

4. Simulation

The ns-3 Network Simulator was employed to simulate the proposed method. All nodes in the simulation were stationary, and the effective communication distance was set to R = 35m. According to experimental evaluation, the proposed method may achieve the best performance when α falls in [0.5,0.6]. Hence, α was set to 0.55 in the simulation. The communication standard followed the 802.11b, the channel propagation was modelled in the ns-3 Network Simulator, and the Friis free space attenuation model was adopted for fixed rate transmission and attenuation. Moreover, the signal frequency was set to 2.4GHz. The proposed method was contrasted with the LEACH algorithm and the AWFT algorithm.

As shown in Figure 1, LEACH's energy consumption declined faster than that of the other two algorithms. This is because the cluster head selection strategy of LEACH widens the average communication distance of the network. In contrast, AWFT and our algorithm ensure that the distance from the node to the cluster head is one hop. Owing to the relatively short average communication distance, the energy consumption curves of these two algorithms are smoother than that of LEACH.

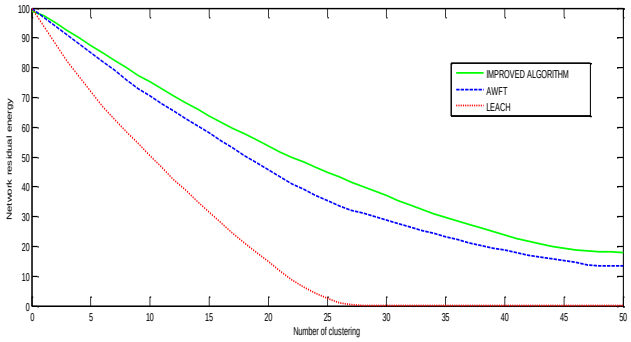

Figure 1. Comparison of network energy consumption

As shown in Figure 2, many nodes died when the energy of LEACH remained at a low level, while many nodes were still alive for AWFT and our algorithm at a low energy level. This means the latter two algorithms can balance each node's energy consumption and prolong the network lifecycle. Compared with AWFT, our algorithm further extended the network lifecycle by saving the energy in information exchange between the nodes in the clustering stage.

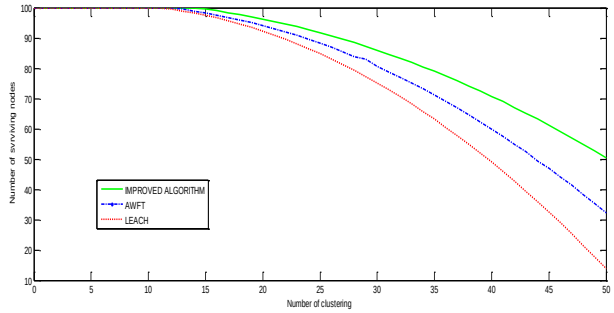

Figure 2. Comparison of network lifecycle

According to the energy model, the energy consumption in communication is proportional to the communication distance between nodes. Keeping the other parameters unchanged, it is discovered that the average distance between nodes increased within the deployment area of nodes. As shown in Figure 3, the total network energy consumption gradually flattened with the expansion of the deployment area. It is obvious that the algorithm performance was better at a lower node density, despite the increase in node communication distance. The phenomenon indicates that our algorithm is better in

sparse networks.

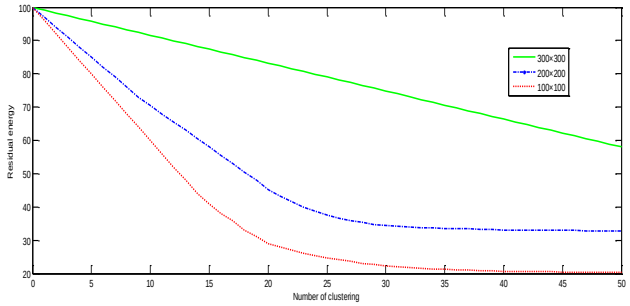

Figure 3. Effect of the size of the deployment area on the algorithm

5. Conclusions

The cluster head selection was rationalized based on the different values of the weight function based on TOA difference. The node with a large weight function value usually has a short TOA, and is more likely to become a cluster head. Thus, the weight function was dynamically adjusted by periodic clustering, and the cluster head, as the centre of data collection and forwarding, was properly changed to optimize the network topology and balance the energy consumption of each node.

During periodic clustering, the node aggregation degree, the weight function, and the clustering time determined by the weight function were calculated independently by the node itself. Both the network traffic and energy consumption in communication were reduced because it is not necessary to compare the aggregation degree and weight function of the node with those of other nodes. When the cluster head arrived on time, the clustering mechanism declared all nodes in the one hop range of the cluster head as the cluster nodes. The mechanism prevents multiple cluster heads in the cluster and stabilizes the cluster scale.

To ensure the even distribution of cluster head, the author proposed an energy-based iterative cluster head selection method to reduce the cluster heads' energy consumption. The algorithm works well on the basic hardware of the node and does not require complex computation. The simulation results show that the proposed algorithm can further reduce the overall energy consumption of the network compared to LEACH and the AWFT.

Acknowledgments

This work was supported by the Foundation for Sci & Tech Research Project of Lanzhou (No. 2015-2-74) and the Natural Science Foundation of Gansu Province of China (No. 1506RJZA084).

References

1. K. Akkaya and M. A. Younis, "Survey of Routing Protocols for Wireless Sensor Networks," *Ad Hoc Networks*, Vol. 3, No. 3, pp. 325-349, 2005
2. P. K. Batra and K. Kant, "LEACH-MAC: A New Cluster Head Selection Algorithm for Wireless Sensor Networks," *Wireless Networks*, Vol. 22, No. 1, pp. 49-60, 2016
3. C. Boler and S. Yenduri, "Resilient Multi Sink Networks using Simplistic Hop based Routing," in *Proceedings of 11th International Conference on Information Technology*: *New Generations* (*ITNG*), pp. 192-195, 2014
4. L. M. Borges, A. S. Lebres, and F. J. Velez, "Survey on the Characterization and Classification of Wireless Sensor Networks Applications," *IEEE Communications Surveys & Tutorials*, Vol. 16, No. 4, pp. 1860-1890, 2014
5. S. Cho, K. H. La, and B. Shrestha, "An Energy-Efficient Cluster- based Routing in Wireless Sensor Networks," *Communication and Networking*, pp. 57-64, 2011
6. W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application Specific Protocol Architecture for Wireless Micro Sensor Networks," *IEEE Transactions on Wireless Communication*, Vol. 1, No. 4, pp. 660-670, 2002
7. S. Hu, J. Han, and X. Wei, "A Multi-Hop Heterogeneous Cluster-based Optimization Algorithm for Wireless Sensor Networks," *Wireless Networks*, Vol. 21, No. 1, pp. 57-65, 2014
8. O. Iova, F. Theoleyre, and T. Noel, "Using Multiparent Routing in RPL to Increase the Stability and the Lifetime of the Network," *Ad Hoc Network*s, Vol. 29, pp. 45-62, 2015
9. J. H. Jeon, H. J. Byun, and J. T. Lim, "Joint Contention and Sleep Control for Lifetime Maximization in Wireless Sensor Networks," *IEEE Communications Letters*, Vol. 17, No. 2, pp. 269-272, 2013
10. D. Kumar, T. C. Aseri, and R. B. Patel, "EEHC: Energy Efficient Heterogeneous Clustered Scheme for Wireless Sensor

Networks," *Computer Communications*, Vol. 32, No. 4, pp. 662-667, 2009
11. J. S. Leu, T. H. Chiang, and M. C. Yu, "Energy Efficient Clustering Scheme for Prolonging the Lifetime of Wireless Sensor Network with Isolated Nodes," *IEEE Communications Letters*, Vol. 19, No. 2, pp. 259-262, 2015
12. H. Lin and H. Uster, "Exact and Heuristic Algorithms for Data Gathering Cluster-based Wireless Sensor Network Design Problem," *IEEE/ACM Transactions on Networking*, Vol. 22, No. 3, pp. 903-916, 2014
13. Y. H. Liu, Y. F. Zhao, and K. H. Xu, "Improvement of Leach in Wireless Sensor Networks," *Computer Engineering and Applications*, Vol. 46, No. 17, pp. 117-120, 2010
14. A. S. K. Mammu, A. Sharma, and U. Hernandez-Jayo, "A Novel Cluster-based Energy Efficient Routing in Wireless Sensor Networks," in *Proceedings of IEEE 27th International Conference on Advanced Information Networking and Applications* (*AINA*), pp. 41-47, 2013
15. S. Ping, "Delay Measurement Time Synchronization for Wireless Sensor Networks," Intel Research Center, 2013
16. W. R. Shi, D. Bai, and P. Gao, "Cluster Head Range Adaptive Adjustment Clustering Rounding Protect for Wireless Sensor Networks," *Chinese Journal of Scientific Instrument*, Vol. 33, No. 8, pp. 1779-1785, 2012
17. Y. Lin, Y. Li, and X. Yin, "Multisensor Fault Diagnosis Modeling based on the Evidence Theory," *IEEE Transactions on Reliability*, Vol. 99, pp. 1-9, 2018
18. C. Shi, Z. Dou, and Y. Lin, "Dynamic Threshold-Setting for RF-Powered Cognitive Radio Networks in Non-Gaussian Noise," *Physical Communication*, Vol. 27, pp. 99-105, April 2018
19. J. Sun, W. Wang, and L. Kou, "A Data Authentication Scheme for UAV Ad Hoc Network Communication," *Journal of Supercomputing*, Vol. 10, No. 8, pp. 1-16, 2017
20. Y. Lin, C. Wang, and J. X. Wang, "A Novel Dynamic Spectrum Access Framework based on Reinforcement Learning for Cognitive Radio Sensor Networks," *Sensors*, Vol. 16, No. 10, pp. 1-22, 2016
21. A. K. Singh and N. Purohit, "An Optimized Fuzzy Clustering for Wireless Sensor Networks," *International Journal of Electronics*, Vol. 101, No. 8, pp. 1027-1041, 2014
22. H. Taheri, P. Neamatollahi, and M. Naghibzadeh, "Improving on HEED Protocol of Wireless Sensor Networks using Nonprobabilistic Approach and Fuzzy Logic (*HEED-NPF*)," in *Proceedings of the 5th International Symposium on Telecommunications* (*IST*), pp. 193-198, 2012
23. Y. K. Tamandani and M. U. Bokhari, "SEPFL Routing Protocol based on Fuzzy Logic Control to Extend the Lifetime and Throughput of the Wireless Sensor Network," *Wireless Networks*, Vol. 22, No. 2, pp. 647-653, 2016
24. F. Tang, I. You, and S. Guo, "A Chain- Cluster based Routing Algorithm for Wireless Sensor Networks," *Journal of Intelligent Manufacturing*, Vol. 23, No. 4, pp. 1305-1313, 2012
25. J. Wu, L. Zhang, and Y. Bai, "Cluster-based Consensus Time Synchronization for Wireless Sensor Networks," *IEEE Sensors Journal*, Vol. 15, No. 3, pp. 1404-1413, 2015
26. J. Yang, D. Y. Zhang, and Y. Y. Zhang, "Cluster-based Data Aggregation and Transmission Protocol for Wireless Sensor Networks," *Journal of Software*, Vol. 21, No. 5, pp. 1127-1137, 2010

Jinyu Li received her M.S. degree in computer software and theory in 2010. She is a teacher in the Department of Computer Science and Technology at Lanzhou Jiaotong University. Her research areas include intelligent computation and data mining.

Jun Li received her Ph.D. in intelligent transportation and information systems engineering from Lanzhou Jiaotong University in 2018. She is a teacher in the Department of Computer Science and Technology at Lanzhou Jiaotong University. Her current research interests include intelligent computing methods.