Hindawi
International Journal of Digital Multimedia Broadcasting
Volume 2022, Article ID 5566365, 11 pages
https://doi.org/10.1155/2022/5566365

Research Article

An Energy-Efficient Clustering Routing for Wireless Sensor Networks Based on Energy Consumption Optimization

Xu Huibin [1] and Zeng Mengjia [1,2]

[1]*School of Information Engineering, Huzhou University, Huzhou, Zhejiang 313000, China*
[2]*School of Science and Engineering, Huzhou College, Huzhou, Zhejiang 313000, China*

Correspondence should be addressed to Xu Huibin; 02623@zjhu.edu.cn

Received 18 January 2021; Revised 16 June 2021; Accepted 7 February 2022; Published 10 March 2022

Academic Editor: Floriano De Rango

In order to alleviate energy consumption of wireless sensor networks (WSNs), energy consumption optimization-based clustering routing (ECOR) is proposed in this paper. In ECOR, network is gridding by hexagon. And there is only a cluster head (CH) in each hexagon, which makes the distribution of CHs more even. Residual energy of nodes and distance from the centroid of hexagon are used to elect CHs. For any a CH, dynamic time slot allocation strategy is adopted to allocate time slot for its cluster members. According to status of cluster members, the duration of time slot is dynamically adjusted. In intercluster communication, the Dijkstra algorithm is used to construct the shortest path between CHs in order to shorten the distance of transmitting data. Simulation results show that the ECOR algorithm outperforms the improved-Low Energy Adaptive Clustering Hierarchy (I-LEACH) algorithm in terms of distribution of CHs and energy consumption.

1. Introduction

In wireless sensor networks (WSNs), a sensor node (or simply a node) is a device that can gather sensory information, perform some processing, and communicate with other nodes [1, 2]. However, limited energy capacity [3] and computation limit the applications of these devices.

Therefore, energy efficiency is crucial performance of WSNs. Many studies show that routing protocol plays an important role in saving energy of nodes [4]. The clustering routing protocol [5, 6] were applied in WSNs since that it has the advantages of scalability, resource sharing, and energy saving.

One of the most popular and widely clustering protocols is the low-energy adaptive clustering hierarchy (LEACH) [7], which is presented by Heinzelman et al. It is probabilistic approach that randomly selects cluster heads (CHs) in each round. After the CH selection, CHs broadcast information toward nearby nodes to form clustering. Receiving the information, nearby nodes will select the nearest CH to join [8]. Then, member nodes (MNs) send data toward its CHs.

Though LEACH improves energy efficiency of network, some disadvantages are associated with it. One is the network lifetime. In LEACH, a sensor node with least residual energy may be selected to be a CH, which may be reduce the network lifetime. Another is distribution of CHs. Random selection of CHs results in the uneven distribution of CHs [9–11].

Therefore, energy consumption optimization-based clustering routing (ECOR) is proposed in this paper. The monitoring area is gridding by hexagon, and the CH is selected in each hexagon to make CH distribution more even. The weight of nodes to be a CH is computed. Energy of nodes and distance from the centroid of hexagon is taken into account when the weight was computed. In each hexagon, node with the biggest weight is selected to be a CH.

Intracluster communication, the duration of time slot is adjusted to reduce consumption of energy. In Intercluster

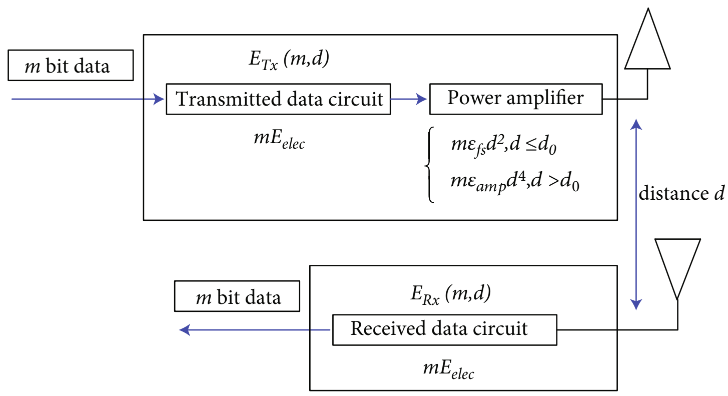

Figure 1: Energy consumption model.

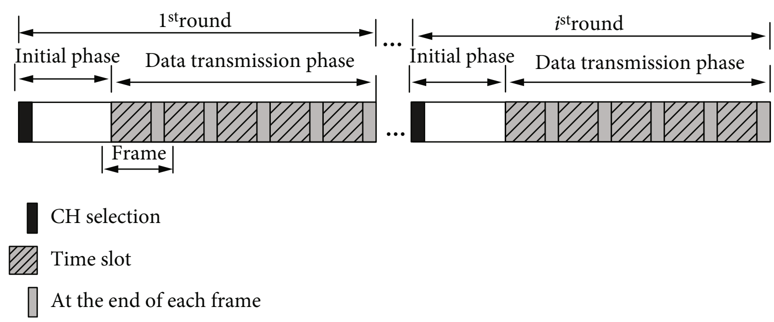

Figure 2: Round structure.

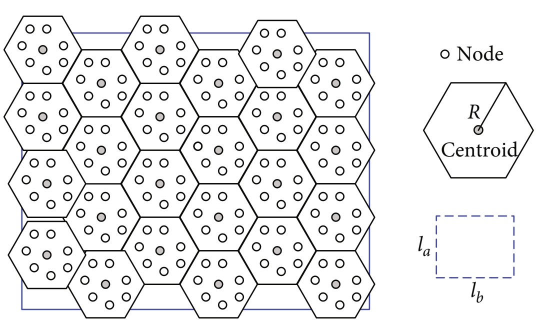

Figure 3: Hexagonal grid-based cluster model.

communication, the Dijkstra algorithm is applied to construct the shortest path between CHs, and the consumption of CHs' energy are reduced.

The rest of this paper is organized as follows: Section 2 presents related works. Section 3 describes the system model, including network model, energy consumption model, and round structure. Section 4 presents the proposed routing in detail. Section 5 explains the analysis of simulation results. Finally, we conclude this paper in Section 6.

2. Related Works

Reference [12] proposed a residual energy-based cluster-head selection algorithm so that the network lifetime is improved. The algorithm focuses on an efficient CH selection scheme. In CH selection process, energy and an optimum value of CHs are used to elect the next round CHs. Similar to work in reference [12], reference [13] also proposed a CH selection algorithm. CH election has done based on threshold energy, residual energy, and optimum numbers of clusters.

In addition, both references [14, 15] have reduced the consumption of energy in view of simplifying sensing data of nodes. Namely, the aim of reference [14] is to reduce the energy consumption by reducing the number of sensors reporting information to the sink without worsening the reliability, while basic idea of reference [15] is to weight the information detected by the sensors according to the distortion area to better estimate the event at the sink node.

Reference [16] proposed the improved LEACH (I-LEACH) algorithm, which optimized the energy consumption of the whole network to determine the optimal number of CHs. And the threshold of CH selection was set by the idea of equalization. Reference [17] proposed a particle swarm optimization-based cluster routing to make CH distribution more even. However, the single hop communication mode is still adopted in data transmission, which increases energy consumption of data transmission. In addition, reference [18] proposed wolf optimizer-based centralized cluster algorithm. The algorithm selects CHs using the grey wolf optimizer.

Reference [19] proposed fuzzy comprehensive evaluation-based energy-efficient cluster routing (FCCR). In FCCR, the improved K-means cluster is used to form clustering. The CH selection is optimized by using fuzzy

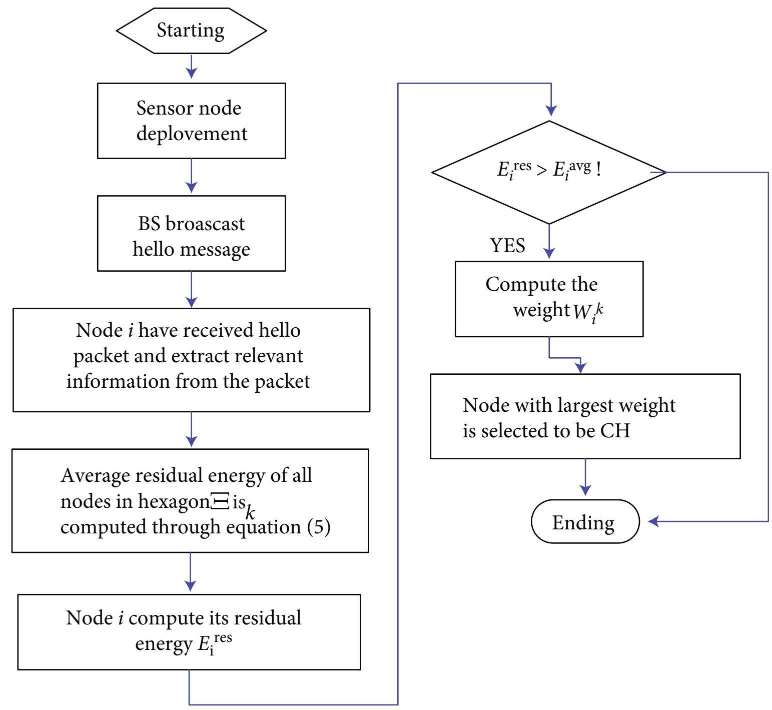

Figure 4: Flowchart of selecting CH.

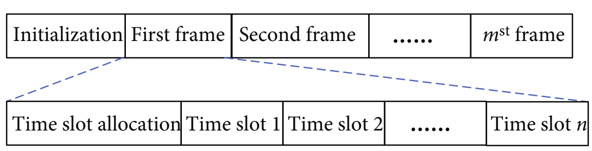

Figure 5: Initial time slot allocation.

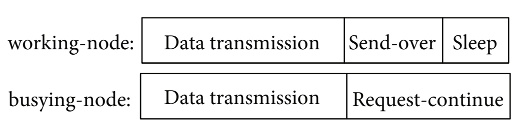

Figure 6: Working node and busying node.

comprehensive evaluation. Reference [20] proposed the improved-firefly-based clustering (IFFC) algorithm. It uses the improved firefly clustering algorithm to form clustering. The algorithm has two main advantages. One is even distribution of CHs, and another is that balance the energy consumption between heterogeneous nodes. And frequency of reclustering is dropped. However, the IFFC algorithm did not optimize the data transmission path of intercluster communication.

3. System Model

3.1. Network Model. N sensor nodes are randomly deployed in $\ell_a \times \ell_b \mathrm{m}^2$ area. The relevant constraints of network model are as follows [10]:

(1) Initial state of all nodes is same. That is, battery capacity of each node is same. The energy consumption of transferring unit data to unit distance is same for all nodes

(2) There is only a base station (BS) in the network, and its energy is not limited

(3) Physical position of all nodes is fixed after the deployment

(4) Each node has its own ID. They can know their remaining energy and position. In addition, each node is able to fuse with its received data

(5) The communication link is symmetrical. For each node, the distance between transmitter and receiver is estimated through received signal strength index (RSSI). The transmitting power and communication radius of node can be adjusted by itself

Moreover, some important notations used in this paper are summarized in Notation 1.

Notation 1. Notations and their descriptions. Notation: Description,

d:Distance between transmitter and receiver,

$E_{\mathrm{Tx}}(m, d)$:Consumed energy that the transmitter transmit m bits data toward receiver,

$E_{\mathrm{Rx}}(m, d)$:Consumed energy that the receiver received m bit data,

Ξ_k:k^{st} hexagon,

N_k:The set of nodes in Ξ_k,

S_i:i^{st} sensor node,

X_i:Position vector of s_i,

W_i^k:Weight of s_i that located in Ξ_k,

E_i^{res}:Residual energy of s_i,

E_k^{avg}:Average residual energy of all nodes in Ξ_k,

$F(t, j)$:State of s_i in frame t^{st},

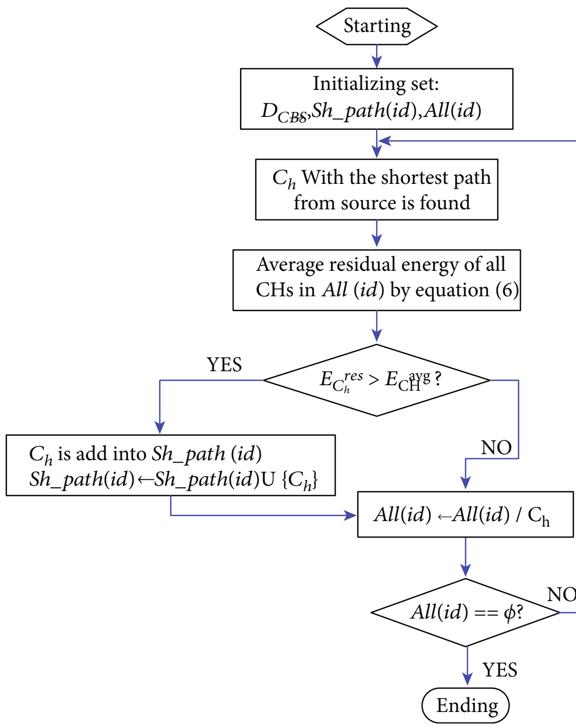

Figure 7: Flowchart of constructing the shortest path.

$\lambda(t+1, j)$:Time slot adjustment factor of s_i in $(t+1)^{\text{st}}$ frame,

C_i:Represent the i^{st} CH,

$E_{ci}{}^{\text{res}}$:Residual energy of all nodes in C_i.

3.2. Energy Consumption Model. The energy consumption model [21, 22] of the ECOR algorithm is as shown in Figure 1. Let $E_{\text{Tx}}(m, d)$ refers to consumed energy that the transmitter transmits m bits data toward receiver. And the distance between transmitter and receiver is d.

$$E_{\text{Tx}}(m, d) = \begin{cases} mE_{\text{elec}} + m\varepsilon_{fs}d^2, \text{ if } d \le d_0, \\ mE_{\text{elec}} + m\varepsilon_{\text{amp}}d^4, \text{ if } d > d_0, \end{cases} \quad (1)$$

where E_{elec} represents the consumed energy that transmitted per bit data. ε_{fs} and ε_{amp} represent energy-consuming factor under free space and double-path fading model, respectively, $d_0 = \sqrt{\varepsilon_{fs}/\varepsilon_{\text{amp}}}$.

Received node consumes energy$E_{\text{Rx}}(m)$ to receive m bit data:

$$E_{\text{Rx}}(m) = mE_{\text{elec}}. \quad (2)$$

3.3. Round Structure. Each round consists of the initial phase and data transmission phase. In the initial phase, the CH selection is completed, and cluster is formed. In data transmission phase, time is divided into different equal time slots [23]. The duration of the allocated time slot is adjusted through the node state.

Each node transmits its data toward its CH in its time slot, as shown in Figure 2. The data transmission phase is composed of multiple frames. At the end of each frame, CH transmits data toward BS.

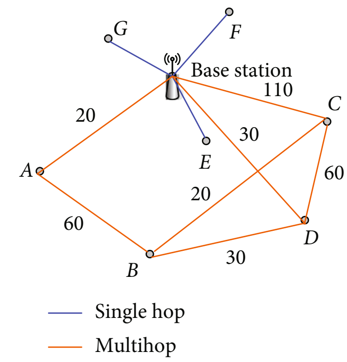

Figure 8: Improved Dijkstra algorithm.

4. ECOR Routing

4.1. Hexagonal Grid-Based Cluster. The interesting region is divided into hexagon. There is only a CH in each hexagon to make the distribution of CH more even, as shown Figure 3. Centroid is the same distance from any point on the edge of the hexagon. If the node close to the centroid is chosen as the CH, the whole network can be covered with the smallest numbers of CH.

We consider $\ell_a \times \ell_b$ area of the 2D plane. And the number of centroid is

$$M = \left\lfloor \frac{(\ell_a \times \ell_b)}{A} \right\rfloor, \quad (3)$$

where $\lfloor \cdot \rfloor$ represents floor function. $A = 3/2\sqrt{3}R^2$ is the area of a hexagon. R represents the radius of inner tangential circle of the hexagon. x_k^c is the position vector of centroid the of k^{st} hexagon Ξ_k, and $k = 1, 2, \cdots, M$.

If all CHs are located at the centroid of hexagons, the network can be covered with the least numbers of CHs, which is also the original intention of adopting hexagonal grid-based cluster scheme. BS loads all position vectors into Hello packet and broadcasts the Hello packet throughout the network.

Received from position vectors, each node computes the distance from the centroids of all hexagons. The node is in k^{st} hexagon Ξ_k, if the distance from x_k^c is not more than R, $k = 1, 2, \cdots, M$, so that these nodes know in which hexagon they are.

LetN_k represents the set of nodes in k^{st} hexagonΞ_k, $k = 1, 2, \cdots, M$, and its definition is given by

$$N_k = \{s_i \mid \|x_i - x_k^c\| < R\}, \quad (4)$$

Where x_i represents the position vector of s_i, and $i = 1, 2, \cdots, N$..

4.2. Selection of CHs and Cluster Formation. In the initial phase, the tentative CHs are initially chosen using a back-off timer [24]. Each tentative CH will collect average residual energy of all nodes in its hexagon. The timer value is inversely proportional to the residual energy level of node. For instance, the back-off timer value will be low if the energy level is high, and vice versa.

Table 1: Shortest path iteration.

Iteration	Sh_path (id)	A	B	C	D
At first	{BS}	20	∞	110	30
1	$\{BS, A\}$	20	80	100	30
2	$\{BS, A, D\}$	20	80	90	30
3	$\{BS, A, D, B\}$	20	80	90	30
4	$\{BS, A, D, B, C\}$	20	80	90	30

Therefore, back-off timer value will be expired soon for the nodes with higher energy level. Once the back-off timer reaches zero, the respective node declares itself as tentative CH and broadcast tentative CH advertisement message (*Ten_CH*) within its hexagon. When any of the nodes in hexagon receives *Ten_CH* before its timer expires, it transmits its residual energy level toward the tentative CH. When tentative CH has received residual energy level of all nodes in its hexagon, it calculates the average residual energy of all nodes in hexagon, which is given by

$$E_k^{\text{avg}} = \frac{1}{|N_k|} \sum_{h=1}^{|N_k|} E_h^{\text{res}}, s_h \in N_k, \tag{5}$$

Where E_k^{avg} is the average residual energy of all nodes in hexagon Ξ_k. E_h^{res} denotes the residual energy of node s_h. $|N_k|$ is the number of elements in the set Ξ_k.

It is worth explaining that the node with the largest remaining energy is chosen as tentative CH because energy is consumed to calculate the average energy consumption and broadcast this information. This helps balance the residual energy levels of nodes in the hexagon.

Residual energy of node and distance of node from centroid of the hexagon are used to compute the weight. Then, the node with the largest weight will be selected to be a CH. Let W_i^k denote the weight of node $s_i \in N_i$that is located in k^{st} hexagon Ξ_k, and its definition is given by

$$W_i^k = \begin{cases} \alpha \dfrac{E_i^{\text{res}}}{E_{\text{int}}} + \beta \left(1 - \dfrac{d_{i,\Xi_k}}{R}\right), E_i^{\text{res}} \geq E_k^{\text{avg}} \\ 0, E_i^{\text{res}} < E_k^{\text{avg}} \end{cases}, \tag{6}$$

whereE_i^{res} denotes the residual energy of node $s_i \in N_i$. E_{int} denotes initial energy of node $s_i \in N_i$. According to the abovementioned network model, the initial energy of all nodes is equal. d_{i,Ξ_k} is the distance between the node s_i and the centroid of the hexagon. Both α and β are the weight coefficients, which control the proportion of residual energy and distance in the weight value.

Table 2: Simulation parameters.

Parameters	Value
Simulation area	$200 \times 200\text{m}^2$
Number of nodes	100
Initial energy of node	2 J
Position of BS	(100, 100)
Transmission range	45 m
Packet size	4000 bit
E_{elec}	50nJ/bit
ε_{fs}	10pJ/bit/m^2
ε_{amp}	0.0013pJ/bit/m^4

According to equation (6), when the residual energy of a node is lower than the average residual energy of all nodes in the hexagon in which the node located, the weight of the node is zero. The node with largest weight in each hexagon will be selected to be a CH of the hexagon. The entire process is depicted in a flowchart as shown in Figure 4.

After the selection of CHs, CHs broadcast *Inv_CH* message to nearby nodes for clustering. Each node chooses the closest CH to join using optimal distance communication. Therefore, MNs send *Join_CH* message toward the closest CH. Eventually cluster formation is done.

4.3. Intracluster Communication. After CH was selected, CH will allocate time slot to its MNs in the way of time division multiple access (TDMA) [25]. MNs would transmit data toward CH during its allocated time slot.

4.3.1. Initial Time Slot Allocation. Initially, CH adopts equal time slot scheme to allocate time slot for MNs. In ECOR routing, there are m frames, and there are n time slots in each frame, as shown in Figure 5.

After time slot have been allocated, CH broadcasts time slot allocation table in the cluster. Once received the time slot allocation table, MNs know its allocated time slot and transmit its data during its allocated time slot.

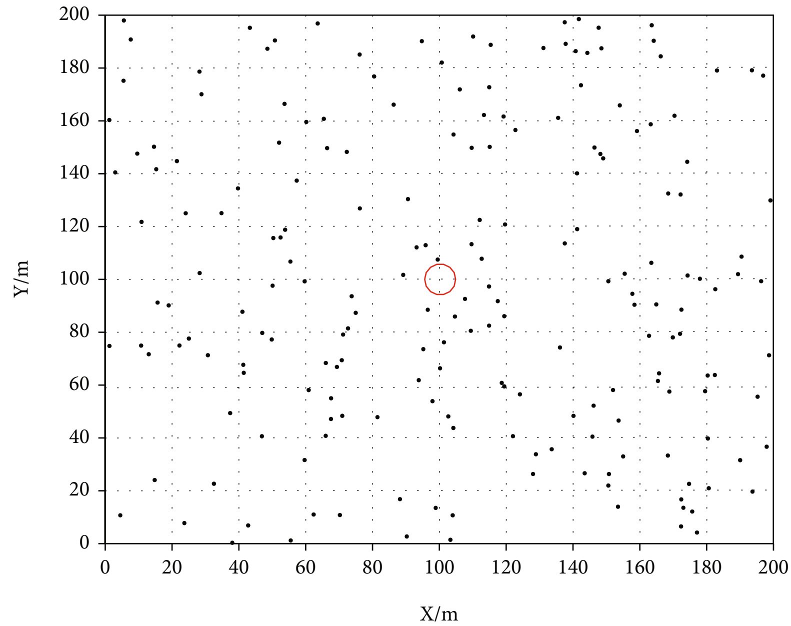

FIGURE 9: Distribution of 100 nodes in $200 \times 200\text{m}^2$.

If the data is completely transmitted during its allocated time slot, node would send a send-over control message at the completion of transmitting data. The send-over control message indicates the completion of the data transmission. Afterward, the node will go to sleep and turn off the radio. For simplicity, the type of node is called working-node.

If the data is not completely transmitted during its allocated time slot, node sends the request-continue control message at the end of the time slot. For simplicity, the node that sends the request-continue control message is called the busying node, as shown in Figure 6.

4.3.2. Time Slot Adjustment. CH will adjust the duration of time slot according to the type of node in the previous round. Therefore, time slot adjustment factorλ is introduced. Let $F(t, j)$ represents the state of node s_j in t^{st} frame. If the state of node s_j is working node, $F(t, j) = 0$. Otherwise, if the state of node s_j is busying node,$F(t, j) = 1$.

Let $\lambda(t+1, j)$ represents time slot adjustment factor of node s_j in $(t+1)^{\text{st}}$ frame, and$\lambda(t+1, j) = 1 + 0.5F(t, j)$. Assume that there are n nodes in the cluster, CH adjusts duration of time slot in next frame according to equation (7):

$$T_s(t+1, j) = \frac{n\lambda(t+1, j)}{\sum_{i=1}^{n} \lambda(t+1, i)} T_s(t, j), \tag{7}$$

where $T_s(t, j)$ represents the duration of j^{st} time slot in t^{st} frame.$T_s(t+1, j)$ represents the duration of j^{st}time slot in $(t+1)^{\text{st}}$ frame.

4.4. Intercluster Communication. According to equation (1), the consumed energy of node is proportional to the $2 \sim 4^{\text{th}}$ power of the transmission distance. If $d < d_0$, transmitter transmits data in single hop way. If $d > d_0$, transmitter still adopts the single hop way, and the energy consumed in transmitting data process is 4^{th} power of d.

In this case, consumed energy of node can be reduced by transmitting in multihop way. Based on the above analysis, the shortest path based on the improved-Dijkstra algorithm [26] is adopted to construct intercluster communication.

In the Dijkstra algorithm, the network topology is represented by weighted graph $G = (V, E)$, where V denotes the vertex set that composed of CHs and BS. The E denotes set of weights that is the distance between the vertices.

Initially, the BS is considered to be a source node, shortest path vertex set Sh_path(id), and all vertex sets All(id) are constructed. At first, there is only the BS in Sh_path(id) set, namely, Sh_path(id) = {BS}.

All(id) contains all vertices except the BS and CHs that are able to communicate directly with the source node. If the distance between a CH and BS is less than d_0, CH communicates directly with BS in single hop way without multihop communication. Let $D_{C\longrightarrow \text{BS}}$ represents CH set that distance from BS is less than d_0, and its definition is shown in Equation (8).

$$D_{C\longrightarrow \text{BS}} = \left\{ C_i \middle| d_{C_i\longrightarrow \text{BS}} < d_0 \right\}, \tag{8}$$

where C_i represents the i^{st} CH, and $i = 1, 2, \cdots, M$. $d_{C_i\longrightarrow \text{BS}}$ represents the distance C_i and BS.

Initially, $\text{All(id)} = \{V/BS/D_{C\longrightarrow \text{BS}}\}$. When Sh_path(id) and All(id) are initialized, for any a CH $C_h \in \text{All(id)}$, the CH C_h with the shortest path from the source node and its residual energy more than average residual energy of all CHs is selected to join in Sh_path(id). Let $E_{\text{CH}}^{\text{avg}}$ denotes the

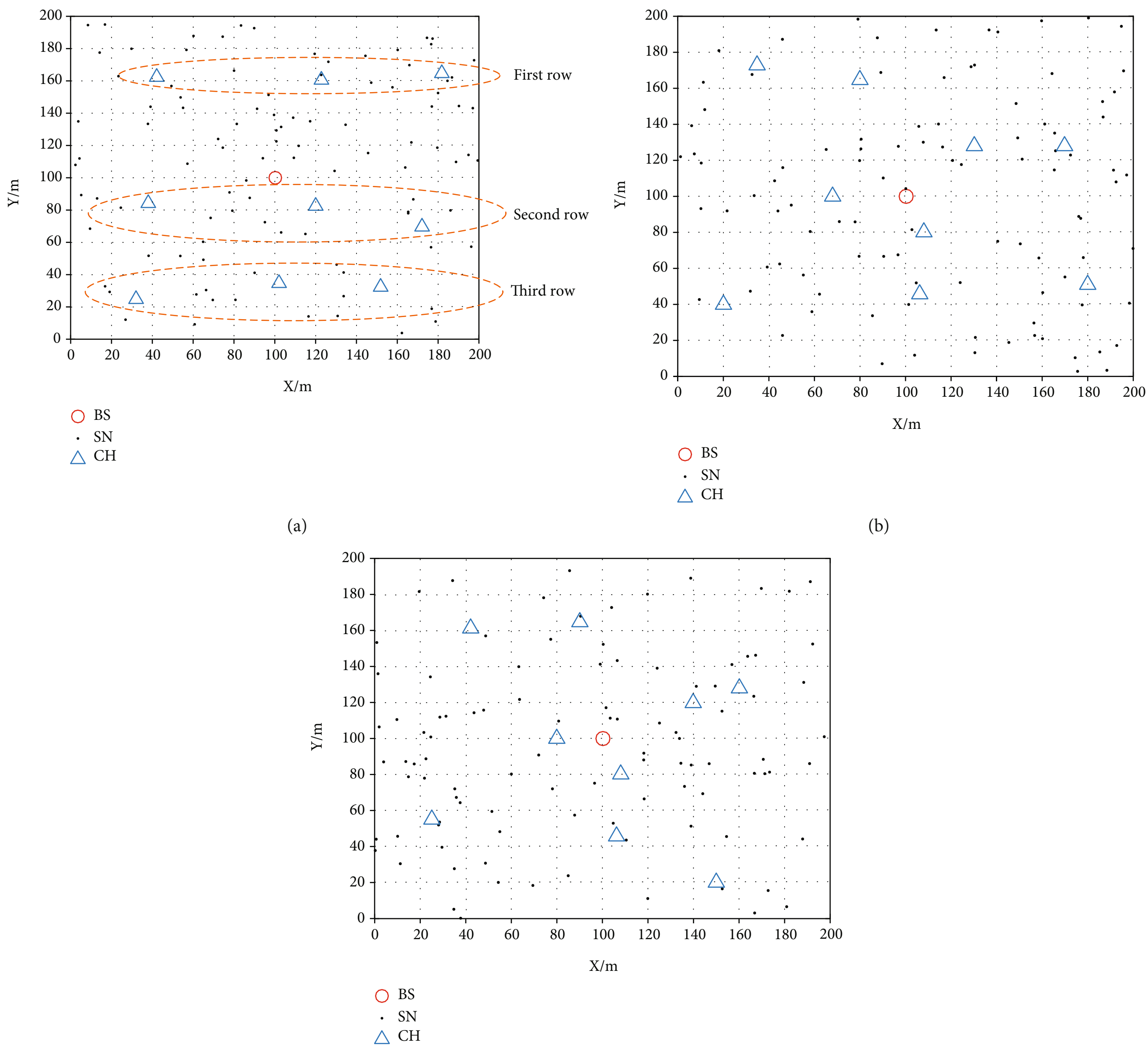

Figure 10: Performance analysis in terms of distribution of CHs.

average residual energy of CHs in All(id):

$$E_{\mathrm{CH}}^{\mathrm{avg}} = \frac{1}{|\mathrm{All(id)}|} \sum_{i=1}^{|\mathrm{All(id)}|} E_{C_i}^{\mathrm{res}}, \tag{9}$$

where $E_{C_i}^{\mathrm{res}}$ denotes the residual energy of C_i. $|\mathrm{All(id)}|$ is the number of CHs in All(id).

Afterward, the length of path between source node and any CH located in All(id) is updated. Repeat the above process until that all vertices in All(id) are join in Sh_path(id). Finally, the shortest path from source node from each CH is constructed. The steps are depicted in the flowchart given in Figure 7.

The process of constructing the shortest path is illustrated in detail with the help of Figure 8. In Figure 8, $V = \{BS, A, B, C, D, E, F, G\}$. Assume that the distance between vertex E, F, $G$and BS are less than d_0, and $D_{C\longrightarrow \mathrm{BS}}(\mathrm{id}) = \{E, F, G\}$, so that, at first, $\mathrm{All(id)} = \{A, B, C, D\}$.

After, for any a vertex in $\mathrm{All(id)} = \{A, B, C, D\}$, the vertex with the shortest path from BS is selected to join inSh_path(id). As shown Figure 8, it is the vertex A with the shortest path from BS, and its path is 20. Therefore, vertex A is joined inSh_path(id), namely, $\mathrm{Sh\_path(id)} = \mathrm{Sh\_path(id)} \cup A$.

Subsequently, the length of shortest path between source node and any CH located in All(id) is updated, as shown in Table 1. At first, the shortest path between BS and vertex B is cannot be confirmed; so, the shortest path is denoted by"∞," as shown in Table 1. But when vertex A is joined in Sh_ path(id), the shortest path between the BS and vertex B can be constructed with the help of vertex A, and the shortest path between the BS and vertex B is updated to 80, as shown in Table 1.

Repeating the above procedure until that the shortest path from BS to all vertexes is completely constructed, as shown in Table 1. As known from Table 1, the shortest path from BS to vertex A is BS $\longrightarrow A$. The shortest path from BS to vertex B is BS $\longrightarrow A \longrightarrow B$. The shortest path from BS to vertex C is BS $\longrightarrow D \longrightarrow C$. The shortest path from BS to vertex D is BS $\longrightarrow D$.

5. Performance Analysis

5.1. Simulation Parameters. In order to better analyze the performance of the ECOR algorithm, the WSN-based simulation software NS-2.35 [27] is used to analyze the network performance of the ECOR algorithm, I-LEACH, and FCCR algorithm.

In simulation, the number of nodes is 100. The specific simulation parameters are shown in Table 2. 100 nodes are randomly deployed in $200 \times 200\,\text{m}^2$ area, as shown in Figure 9. The circle in Figure 9 denotes BS. Other black solid points represent the nodes.

The number of network survival nodes and energy consumption were selected to evaluate the performance of ECOR algorithm. The evaluation criteria of network alive nodes were adopted in the reference [28], namely, first node death (FND) and half of node alive (HNA).

5.2. Distribution of CHs. The distribution performance of CHs of the ECOR algorithm, I-LEACH, and FCCR algorithm is analyzed, as shown in Figure 10. The hollow triangle denotes CH. Figure 10(a) shows the distribution of CHs of the ECOR algorithm. Figure 10(b) shows the distribution of CHs of the I-LEACH algorithm. Figure 10(c) shows the distribution of CHs of the FCCR algorithm.

As shown in Figure 10, compared with the I-LEACH algorithm and FCCR algorithm, the distribution of CHs of the ECOR algorithm is more even. In the ECOR algorithm, CHs are arranged in approximately three rows. Distribution of CHs in both the I-LEACH algorithm and FCCR algorithm is haphazard. It is because that the ECOR algorithm has used hexagon-based grid to generate a CH in each grid, making distribution of CHs more even. However, the I-LEACH algorithm selects CHs based on the energy of node, which do not consider the position of nodes. So, the distribution of CHs is uneven in the I-LEACH algorithm.

5.3. Number of Alive Nodes. Firstly, the number of alive nodes in each algorithm is analyzed, as shown in Figure 11.

From Figure 11, it is clear that the performance of number of alive nodes in ECOR is better than I-LEACH and FCCR. The I-LEACH algorithm loses 35% of the nodes at 800 rounds. And the FCCR algorithm loses 35% of the nodes after more rounds approximately at 1200 rounds. The proposed ECOR algorithm has a minimum lose rate.

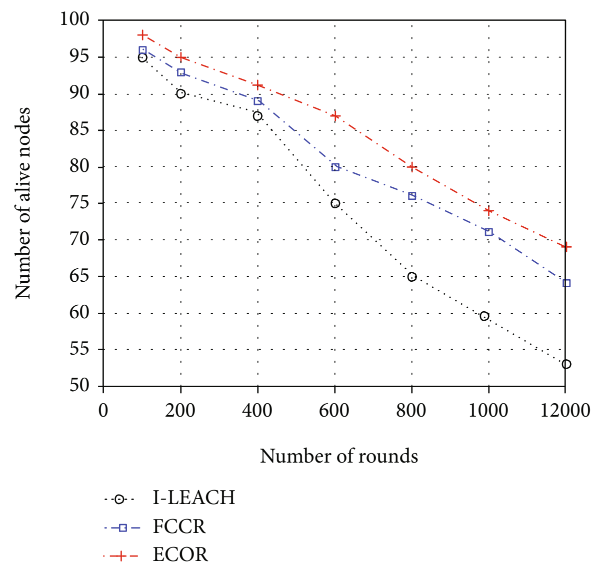

Figure 11: Performance analysis in terms of number of alive nodes.

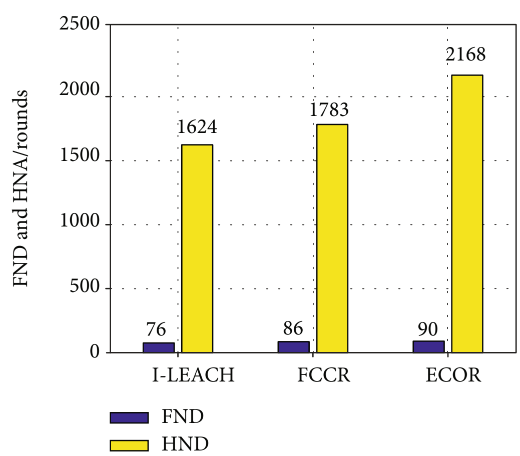

Figure 12: Performance analysis in terms of FND and HNA.

5.4. FND and HNA. Secondly, FND and HNA in each algorithm are analyzed, as shown in Figure 12.

From Figure 12, it is clear that the ECOR algorithm has achieved better performance than all other algorithm in terms of FND and HNA. Compared with I-LEACH, FND of the FCCR algorithm is improved by nearly 13%. The reason is that the FCCR algorithm has used fuzzy comprehensive evaluation to select CHs, which makes CHs more evenly distributed and balances energy consumption between nodes.

Compared with the I-LEACH algorithm and FCCR algorithm, the FND of the ECOR algorithm is increased by 18% and 4%, respectively. This is mainly due to the following reasons: CH distribution of the ECOR algorithm is reasonable. The time slot of nodes is adjusted within the intracluster communication in order to optimize the time slot allocation. When the nodes that do not need to transmit data go to sleep, the energy consumption is reduced.

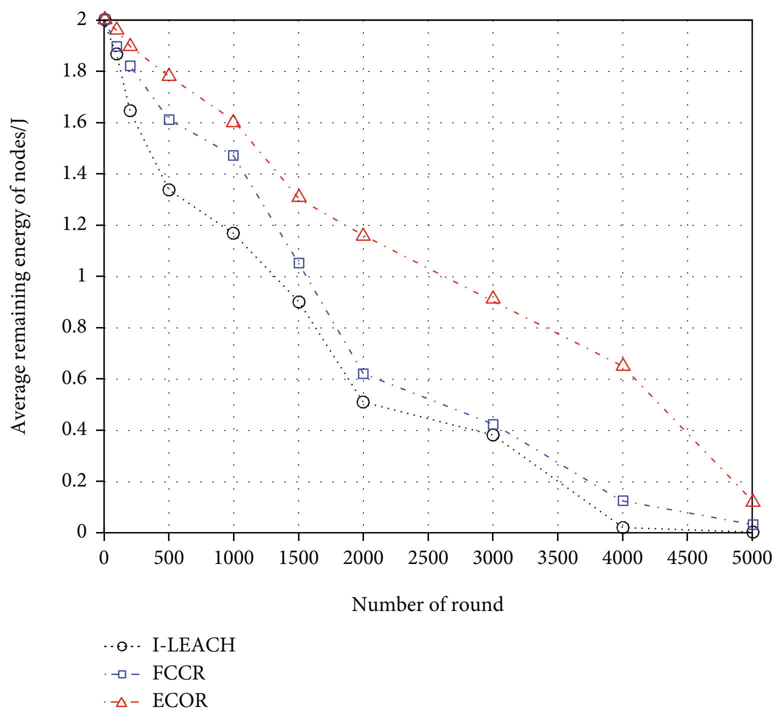

Figure 13: Performance analysis in terms of average remaining energy.

Figure 12 also represents the HNA of the I-LEACH, FCCR, and ECOR algorithm. The HNA of the I-LEACH algorithm is the lowest. The HNA of the ECOR algorithm is the highest. Compared with the I-LEACH and FCCR algorithm, the HNA of the ECOR algorithm is improved by about 32% and 21%, respectively. These results show that the ECOR algorithm effectively reduces energy consumption and improves the energy efficiency by optimizing CH selection.

5.5. Average Remaining Energy of Nodes. Figure 13 displays the average remaining energy of the I-LEACH, FCCR, and ECOR algorithm. From Figure 13, it is clear that the average remaining energy of ECOR is higher than the I-LEACH algorithm and FCCR algorithm. The reason is that the ECOR algorithm makes CH distribution more even, and the transmission path between CHs is shorten. In consequence, energy consumption is reduced.

The proposed ECOR routing has capitalized on two complementary techniques for reducing energy consumption. More precisely, in intracluster communication, the duration of time slot is dynamically adjusted according to the state of nodes. In intercluster communication, the Dijkstra algorithm is used to construct the shortest path between CHs.

In addition, in order to analyze the contribution of these two complementary technologies to the proposed routing, the performance of two technologies is analyzed separately.

Therefore, let us think about two cases in the ECOR algorithm. Case 1:the ECOR algorithm does not adopt the shortest path between CHs, that is, ECOR without the Dijkstra algorithm. Case 2: the ECOR algorithm does not adopt

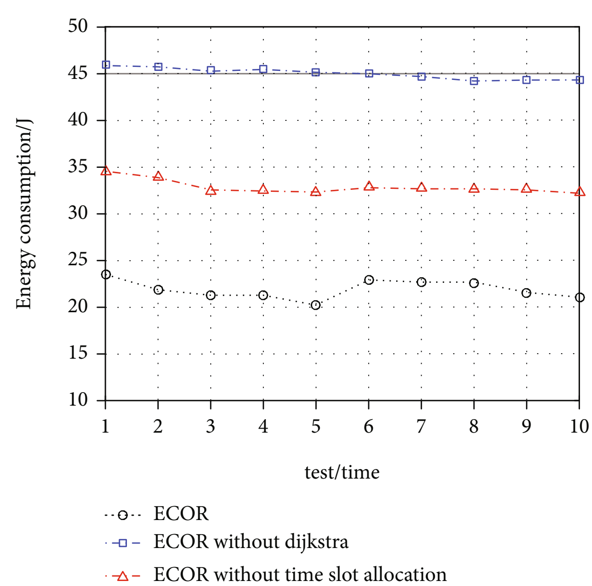

Figure 14: Consumed Energy of ECOR, ECOR without Dijkstra, and ECOR without time slot allocation.

duration of time slot allocation, that is, ECOR without time slot allocation.

Energy consumption of ECOR, ECOR without Dijkstra, and ECOR without time slot allocation is shown in Figure 14 when they performed 1000 rounds. From Figure 14, we can obviously see that energy consumption of ECOR routing is the lowest in line with expectations. ECOR has used the two complementary technologies to reduce energy consumption of nodes.

As is clearly shown in Figure 14, compared with ECOR without Dijkstra, ECOR without time slot allocation consumes less energy. The result shows that the shortest path strategy constructed by the Dijkstra algorithm is more effective in reducing energy consumption compared with time slot allocation. The main reason is that power dissipation of transmission unit in sensors is much bigger than the sum of other units, to shorten the path of data transmission can greatly reduce energy consumption of nodes.

6. Conclusion

In order to reduce energy consumption and optimize CH selection, the ECOR algorithm was proposed in this paper. In ECOR, CHs are selected by using the centroid of hexagonal network topology. The ECOR algorithm makes the distribution of CHs more even. In intracluster communication, the duration of time slot is dynamically allocated according to the state of node. In intercluster communication, the Dijkstra algorithm is used to construct the shortest path among CHs, in order to shorten the path of transmitting data.

Simulation results show that the ECOR algorithm outperforms the I-LEACH algorithm in terms of distribution of CHs and energy consumption. The aim of the ECOR algorithm is to reduce energy consumption of node and make CHs' distribution even. In the future, other performance of the ECOR algorithm will be analyzed, including transmission data delay and packet transmission success rate.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partly supported by the Zhejiang Province Soft Science Foundation of China (2021C35129) and Huzhou Natural Science Foundation (2021YZ20).

References

[1] S. K. Yang, Y. M. Shiue, Z. Y. Su, I. H. Liu, and C. G. Liu, "An authentication information exchange scheme in WSN for IoT applications," *IEEE Access*, vol. 8, pp. 9728–9738, 2020.

[2] A. Albakri, L. Harn, and S. Song, "Hierarchical key management scheme with probabilistic security in a wireless sensor network (WSN)," *Security and Communication Networks*, vol. 2019, Article ID 3950129, 11 pages, 2019.

[3] B. Mohammed, R. Mehdi, A. Mokaddem et al., "LE-OLSR protocol performance evaluation in various energy conditionsof mobile ad-hoc and sensor wireless networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1391–1398, 2020.

[4] X. Li, B. Keegan, F. Mtenzi, T. Weise, and M. Tan, "Energy-efficient load balancing ant based routing algorithm for wireless sensor networks," *IEEE Access*, vol. 7, pp. 113182–113196, 2019.

[5] A. E. Fawzy, A. Amer, M. Shokair, and W. Saad, "Proposed intermittent cluster head selection scheme for efficient energy consumption in WSNs," in *Radio Science Conference*, pp. 275–283, Alexandria, Egypt, 2017.

[6] M. Zareei, C. Vargas-Rosales, M. H. Anisi et al., "Enhancing the performance of energy harvesting sensor networks for environmental monitoring Applications," *Energies*, vol. 12, no. 14, p. 2794, 2019.

[7] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *33rd Annual Hawaii International Conference on System Sciences*, pp. 1–10, Maui, HI, USA, 2000.

[8] J. Wang, Y. Gao, W. Liu, W. Wu, and S. J. Lim, "An asynchronous clustering and Mobile data gathering schema based on timer mechanism in wireless sensor Networks," *Computers, Materials and Continua*, vol. 58, no. 3, pp. 711–725, 2019.

[9] J. Wang, Y. Gao, X. Yin, F. Li, and H. J. Kim, "An enhanced PEGASIS algorithm with mobile sink support for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2018, 9 pages, 2018.

[10] W. Osamy, A. Ahmed, and A. Salim, "CSOCA:Chicken Swarm optimization based clustering algorithm for wireless sensor networks," *IEEE Access*, vol. 8, pp. 60676–60688, 2020.

[11] P. Kuila and P. K. Jana, "A novel differential evolution based clustering algorithm for wireless sensor networks," *Applied Soft Computing Journal*, vol. 25, pp. 414–425, 2014.

[12] T. M. Behera, S. K. Mohapatra, U. C. Samal, M. S. Khan, M. Daneshmand, and A. H. Gandomi, "Residual energy-based cluster-head selection in WSNs for IoT application," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5132–5139, 2019.

[13] A. Panchal, L. Singh, and R. K. Singh, "RCH-LEACH: Residual energy based cluster head selection in LEACH for wireless sensor networks," in *2020 International Conference On Electrical And Electronics Engineering (ICE3-2020)*, pp. 321–325, Gorakhpur, India, 2020.

[14] Y. Feng, H. E. Kang, H. Yang, Z. Qiu, and Y. Liu, "The research and optimization of data gathering protocol for wireless sensor network," *Chinese Journal of Sensors and Actuators*, vol. 27, no. 3, pp. 355–360, 2014.

[15] P. S. Sreedharan and D. J. Pete, "Spatial correlation based clustering with node energy based multi-hop routing scheme for wireless sensor networks," *Tehnički Glasnik*, vol. 15, no. 1, pp. 25–36, 2021.

[16] S. Zhang, Q. Yang, and Y. B. Feng, "LEACH improved method based on energy consumption balance," *Computer Applications and Software*, vol. 37, no. 1, pp. 133–138, 2020.

[17] B. F. Solaiman and A. F. Sheta, "Energy optimization in wireless sensor networks using a hybrid K-means PSO clustering algorithm," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 24, no. 4, pp. 2679–2695, 2015.

[18] S. M. M. H. Daneshvar, P. A. A. Mohajer, and S. M. Mazinani, "Energy-efficient routing in WSN: a centralized cluster-based approach via grey wolf optimizer," *IEEE Access*, vol. 7, pp. 170019–170031, 2019.

[19] H. L. Jia, X. L. Fan, and Y. F. Lu, "Energy-efficient wireless sensor networks routing protocol based on clustering,"

Transducer and Microsystem Technologies, vol. 37, no. 8, pp. 116–119, 2018.

[20] L. Jian, "Bi Xiao-dong. Optimized energy efficient routing algorithm based on improved firefly clustering for heterogeneous WSNs," *Chinese Journal of Sensors and Actuators*, vol. 31, no. 10, pp. 1584–1591, 2018.

[21] L. Qing, Z. Jiang, Z. Jing, and W. Jian, "WSN energy optimization on BIRCH based dual head clustering," *Computer Engineering and Science*, vol. 42, no. 6, pp. 974–985, 2020.

[22] X. Li-wen, Q. Li-juan, and C. Jie, "A K-means clustering routing algorithm based on modified VANETs," *Computer Technology and Development*, vol. 28, no. 3, pp. 15–19, 2018.

[23] Z. Liang, L. Zhigao, and X. Zhili, "Modified cluster-head selection algorithm for WSNs based on LEACH," *Journal of Electronic Measurement and Instrumentation*, vol. 33, no. 12, pp. 86–94, 2019.

[24] S. Arjunan, S. Pothula, and D. Ponnurangam, "F5N-based unequal clustering protocol (F5NUCP) for wireless sensor networks," *International Journal of Communication Systems*, vol. 31, no. 17, pp. 1–14, 2018.

[25] Z. Wei, "Dynamic time slot allocation MAC protocol for clustered wireless sensor," *Networks*, vol. 30, no. 9, pp. 126–129, 2017.

[26] G. P. Zhang, Z. Fen, M. Gang, and J. X. Dong, "An energy-balanced WSNs routing optimization algorithm based on AGNES clustering," *Computer Engineering and Science*, vol. 42, no. 4, pp. 620–628, 2020.

[27] J. Liu, Y. Li, and H. Qi, "The improvement and simulation of LEACH routing protocol in wireless sensor network based on NS2," *Application of Electronic Technique*, vol. 38, no. 2, pp. 21–23, 2012.

[28] R. Zhang and S. Qin, "LEACH-based clustering routing protocol for sensor networks," *Computer Engineering and Design*, vol. 33, no. 4, pp. 1333–1336, 2012.