Application of Adaptive SSA in Low Power Cluster Routing Matlab Simulation Education

Xiao-Ling Guo, Xing-Hua Sun, Rui Wang, Bing-Qing Han, Xin-Yu Yang

Abstract—**The low-power cluster routing protocol is a significant research focus in the WSN course. The utilization of Matlab for digital modeling and teaching simulation significantly enhances the understanding of this particular subject matter. This article introduces a low-power cluster routing algorithm based on the adaptive SSA and applies it to the simulation teaching process. Firstly, the SSA is enhanced by sine and cosine functions, with the addition of an adaptive adjustment factor. Subsequently, Levy flight is incorporated into SSA to improve its ability to escape from localized extremes. Additionally, a standard normal distribution random number is introduced to increase individual diversity within the algorithm's population. Furthermore, this proposed adaptive algorithm ASSA undergoes testing for convergence accuracy and speed using standard test functions. Finally, the proposed algorithm ASSA is utilized in cluster routing algorithms and simulation teaching. This article utilizes MATLAB simulation analysis tool for modeling and visually demonstrating processes such as cluster head election, clustering, data communication, and node death in clustering algorithms. Simultaneously, it illustrates how the ASSA algorithm effectively reduces energy consumption in each round and prolongs system stability running time to students. This simulation teaching process enables students to gain a deeper and more intuitive understanding of important features such as low power consumption of nodes, system lifetime, and energy utilization; thereby enhancing teaching effectiveness.**

Index Terms—**SSA, Low power, Cluster routing, Matlab simulation teaching**

I. Introduction

Numerous studies and applications have been conducted on routing algorithms in low power sensor networks. However, with the continuous development of IoT, many scholars are persistently deepening their research on routing algorithms in order to further enhance and optimize algorithm performance and reliability. Their ultimate goal is to make significant contributions to the rapid advancement of IoT. This is because low power sensor routing plays a crucial role in overall network performance, directly impacting transmission efficiency and energy consumption [1].

In Ref. [2], an adaptive double cluster head energy-saving routing program is proposed above non-uniform partitioning. The algorithm initially divides the network into several uneven regions based on spatial factors, and then selects the main cluster head for each partition. Additionally, the algorithm also chooses secondary cluster heads in the vicinity of the BS to handle data routing. The utilization of non-uniform regions and dual cluster heads strategies leads to a more proportional distribution of energy usage. Sowndeswari et al. [3] conducted a comparative analysis of five distinct energy-saving clustering algorithms based on ABC optimization, namely EMABC, CGTABC, EC-ABC, MeABC, and RMABC. These algorithms were then utilized to form clusters, with cluster heads and relay nodes being dynamically selected. The findings revealed that EMABC demonstrated outstanding performance in maintaining low power consumption. Wu et al. [4] propose an efficient double cluster head approach called DCK-LEACH. Firstly, they utilize Canopy and K-means for clumping to achieve as uniform clustering as possible. Next, DCK-LEACH exploits a hierarchical structure to minimize the load on head nodes. In addition, it selects the main cluster head based on surplus energy and range, then chooses the secondary cluster head based on remaining energy and position relative to the BS. The results indicate that the algorithm can significantly extend the lifespan of sensors in both homogeneous and heterogeneous networks. Koyuncu et al. [5] proposed an energy-efficient clustering algorithm DEC based on multi-layer random probability protocols. They also introduced the concept of a two intermediate node structure. The algorithm fully utilizes distributed networks and presents a mathematical model for cluster head selection based on probability. This approach has been shown to improve the survival time of sensor nodes in the network. In Ref. [6], a path selection strategy is proposed based on a clustering mechanism and particle swarm optimization algorithm. This strategy involves monitoring, tracking, and predicting the remaining energy of adjacent sensor nodes. Additionally, it includes increasing data weights, introducing global optimal factors, and optimizing routing using energy prediction scheduling algorithms to ensure the power supply of sensors by balancing node scheduling. Hilal et al. [7] proposed the ECHO-BAT algorithm, which is based on the metaheuristic echo localization. The algorithm begins by clustering sensor nodes and then utilizes the BAT algorithm to identify and tentatively determine temporary cluster heads and entropy values. Subsequently, it searches for high-energy nodes to

Manuscript received April 20, 2024; revised October 16, 2024.

This work is supported in part by the Doctoral Research Fund of Hebei North University (BSJJ202322), in part by the Medical Science Research Project of Hebei Province (No. 20200488), and in part by the Research Project on Educational and Teaching Reform of Hebei North University (JG2024044).

Xiao-Ling Guo is a lecturer of Hebei North University, Zhangjiakou 075000, China. (e-mail: 175666832@qq.com).

Xing-Hua Sun is a professor of Hebei North University, Zhangjiakou 075000, China. (corresponding author to provide phone:189-313-11869; fax:189-313-11869; e-mail:1030704295@qq.com).

Rui Wang is a lecturer of Hebei North University, Zhangjiakou 075000, China. (e-mail: 33403408@qq.com).

Bing-Qing Han is a postgraduate student of Hebei North University, Zhangjiakou 075000, China. (e-mail: 2559403868@qq.com).

Xin-Yu Yang is a postgraduate student of Hebei North University, Zhangjiakou 075000, China. (e-mail: 990820203@qq.com).

replace temporary cluster heads. Finally, it applies Dijkstra's algorithm to find the best route for sending and transmitting data from each cluster. Manikandan et al. [8] propose the Memory Adaptive Hill Climbing method, which fully leverages the advantages of Memetic and Hill Climbing. This method effectively prevents premature convergence and efficiently clusters. It utilizes Memetic to determine the best cluster head and Hill Climbing to determine the shortest path. Anuradha et al. [9] propose a novel unequal clustering model (SGOBUC) based on Seagull Optimization (SGO). The model utilizes the SGO algorithm to simulate the migration and attack behavior of seagulls, collecting information such as node ID, layer ID, and RSSI. Subsequently, it employs RE, BS, ND, NC, and LQ to construct uneven clustering. Mehra et al. [10] proposed a cluster head selection algorithm based on fuzzy equilibrium cost. This algorithm takes surplus energy, node degree, and range to aggregation node as inputs. By evaluating the fuzzy cost, the qualification index of each node is calculated, and the cluster head role is ultimately determined. Venkatesan et al. [11] propose a novel clustering and routing energy-saving strategy for heterogeneous networks. The Minkowski distance is utilized to calculate routing, while a ranking strategy is employed to select cluster heads. Additionally, DD-TDMA scheduling is implemented to balance the load. This study has also been validated in the context of smart cities application. Saini et al. [12] introduces the concept of Virtual Grid Dynamic Path Adjustment, which involves dividing regions into equally sized blocks and then using the GA algorithm for clustering and optimal path selection. The data is transmitted among cluster heads and ultimately forwarded to the aggregation node. The study also evaluates its effectiveness in heterogeneous networks with varying numbers of nodes and areas. Chang et al. [13] propose an allocation cluster head selection scheme based on node distribution density. This approach effectively utilizes spatial position parameters and local distribution features to calculate each node's own distribution density. Cluster heads are then selected through a specific merging strategy, allowing for quick clustering segmentation and cluster head selection. In Ref. [14], a software-defined multi-hop WSN architecture is proposed. This model encompasses neighbor detection, neighbor advertising, system organization, and package gathering functions. It calculates the overall energy consumption of SD-WSN across all processes and presents the advantages of SD-WSN in terms of cost control, node survival, and lifecycle management perspectives. Jovith et al. [15] proposed the EETTC-MRP technology, which involves three main stages. Firstly, preliminary cluster head selection is completed using Type II fuzzy logic. Subsequently, the Quantum Group Teaching Optimization Algorithm is employed to derive the best cluster head. Finally, the Political Optimizer is utilized to establish multi-hop routing between cluster heads. Saoud et al. [16] propose a solution for determining the optimal cluster head set using the firefly optimization algorithm. This process fully considers topology architecture, energy hierarchy, and flow mode. It also explores methods for selecting the best cluster heads and provides a simple derivation. Additionally, it illustrates the relationship between cluster head number and energy drain. In Ref. [17], the algorithm described is a dual cluster head heterogeneous network clustering routing algorithm. It makes use of quantum optimization algorithms and PSO to minimize power consumption. The algorithm not only ensures an even distribution of cluster heads by designing reasonable objective functions, but also establishes weights between cluster heads and aggregation nodes. By constructing a directed connected graph, it utilizes a minimum spanning tree for inter-cluster communication. SSA is a novel search strategy that emulates the predatory behavior of sparrows, and it has been further developed and utilized in numerous academic works [18]-[20]. Sun et al. [21] proposed an enhanced SSA algorithm based on the t-distribution and applied it to WSN clustering. The algorithm initially utilizes the t-distribution to update discoverer positions, thereby improving search efficiency.

The main contribution of our article is to enhance the SSA algorithm through various strategies and apply the improved algorithm to low-power cluster routing MATLAB simulation teaching. The simulation method demonstrates to students the routing processes, such as cluster head election, clustering, and data communication. Through data comparison and analysis, it helps students comprehend energy utilization efficiency and network lifetime.

II. Improved Algorithm

A. Sparrow Search Algorithm

In the context of SSA, there are three key roles: the discoverer, the follower, and the scout. The discoverer is responsible for locating food sources, while the follower retrieves the food once it has been found. The scout's primary duty is to monitor the area where food is located and provide early warnings in case of potential predator threats. If a warning signal surpasses a predetermined threshold, the entire population will relocate to a new foraging location under the guidance of the discoverer. The position of the discoverer is determined by Eq. (1).

$$X_i^{t+1}=\begin{cases} X_i^t \bullet \exp(\dfrac{-i}{\alpha \bullet T_{\max}}), R_2 < ST \\ X_i^t + Q \bullet L, R_2 \geq ST \end{cases} \tag{1}$$

$\alpha \in [0,1]$. $T_{\max}$ is maximum iteration. L is a vector of $1*d$. Q follows a normal distribution. $R_2 \in [0,1]$ and $ST \in [0.5,1]$ is respectively warning and safety value.

Follower's position is renewed by Eq. (2).

$$X_i^{t+1}=\begin{cases} Q \bullet \exp(\dfrac{X_{worst}^t - X_i^t}{i^2}), i > \dfrac{n}{2} \\ X_p^{t+1} + \left| X_i^t - X_p^{t+1} \right| \bullet A^+ \bullet L, i \leq \dfrac{n}{2} \end{cases} \tag{2}$$

X_{worst}^t is the global worst sparrow position. X_p^{t+1} represents the optimal sparrow position for discoverers in $t+1$ generation. n is population number. A is a matrix of $1*d$, with each element randomly assigned a value of 1 or -1. And A meets the criteria of $A^+ = A^T \bullet (AA^T)^{-1}$

Scout's position is renewed by Eq. (3).

$$X_i^{t+1}=\begin{cases} X_{best}^t + \beta \bullet \left| X_i^t - X_{best}^t \right|, f_i \neq f_g \\ X_i^t + K \bullet (\dfrac{\left| X_i^t - X_{worst}^t \right|}{(f_i - f_w) + \varepsilon}), f_i = f_g \end{cases} \tag{3}$$

X_{best}^t is the global optimal position. β follows a standard normal distribution, f_i, f_g and f_w is respectively the current, global optimal, and the worst fitness value. $K \in [-1,1]$.

B. Adaptive Sparrow Search Algorithm

Utilizing the sine and cosine functions with volatility and periodicity for iterative optimization not only exhibits strong local search capability, but also demonstrates robust global search ability. The discoverer formula in Eq. (4) is updated by incorporating the sine and cosine functions.

$$X_i^{t+1} = \begin{cases} X_i^t \bullet \exp\left(\frac{-t}{\alpha \bullet T_{\max}}\right), R_2 < ST \\ X_i^t + r_1 \sin r_2 \left| r_3 X_{best}^t - X_i^t \right|, R_2 \ge ST \cap r_4 < 0.5 \\ X_i^t + r_1 \cos r_2 \left| r_3 X_{best}^t - X_i^t \right|, R_2 \ge ST \cap r_4 \ge 0.5 \end{cases} \quad (4)$$

$r_1 = a(1 - \frac{t}{T_{\max}})$, $a = 2$, r_1 is a adaptive adjustment factor. $r_2 \in [0, 2\pi]$, $r_3 \in [0,2]$, $r_4 \in [0,1]$.

To prevent the algorithm from becoming trapped in local optima, a Levy flight strategy is employed for followers to enhance their capacity for global search exploration. $r \in [0,1]$, $r' \in [0,1]$, and $\xi = 1.5$. The improved formula is shown in Eq. (5)-(8).

$$Levy = 0.01 \times \frac{r \times \sigma}{|r'|^{(1/\xi)}} \quad (5)$$

$$\sigma = \left(\frac{\Gamma(1+\xi) \times \sin(\pi\xi/2)}{\Gamma((1+\xi)/2) \times \xi \times 2^{((\xi-1)/2)}} \right)^{(1/\xi)} \quad (6)$$

$$\Gamma(x) = (x-1)! \quad (7)$$

$$X_i^{t+1} = \begin{cases} Levy * \left| \frac{X_{best}^t - X_i^t}{i^2} \right|, i > \frac{n}{2} \\ X_p^{t+1} + \left| X_i^t - X_p^{t+1} \right| \bullet A^+ \bullet L, i \le \frac{n}{2} \end{cases} \quad (8)$$

For scouts, use standard normal distribution random number β to update the position when $f_i = f_g$. The improved formula is shown in Eq. (9).

$$X_i^{t+1} = \begin{cases} X_i^t + \beta \bullet \left| X_i^t - X_{best}^t \right|, f_i \ne f_g \\ X_i^t + \beta \bullet \left| X_{worst}^t - X_i^t \right|, f_i = f_g \end{cases} \quad (9)$$

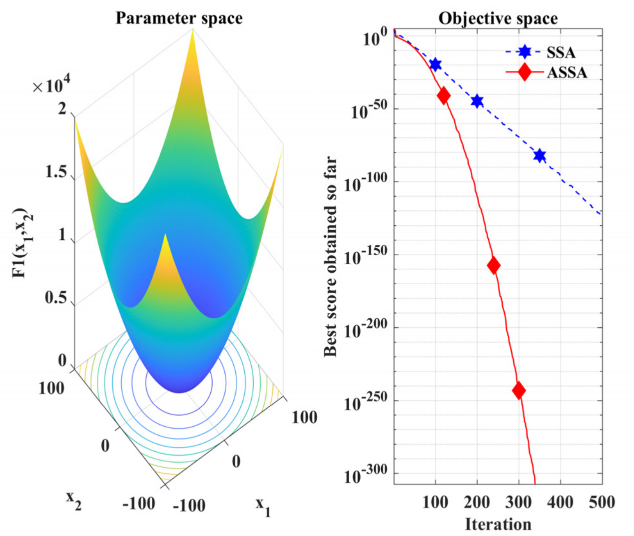
C. Test Function Validation

To assess feasibility and effectiveness, a comparison is made between ASSA and SSA in finding the best value. One unimodal test function, two multimodal test functions, and one fixed dimensional multimodal test function are selected for analysis in Table I. To ensure fairness, the population is set to 15 and the maximum iteration is set to 500. Thirty optimization experiments are conducted to contrast the capabilities of the two algorithms.

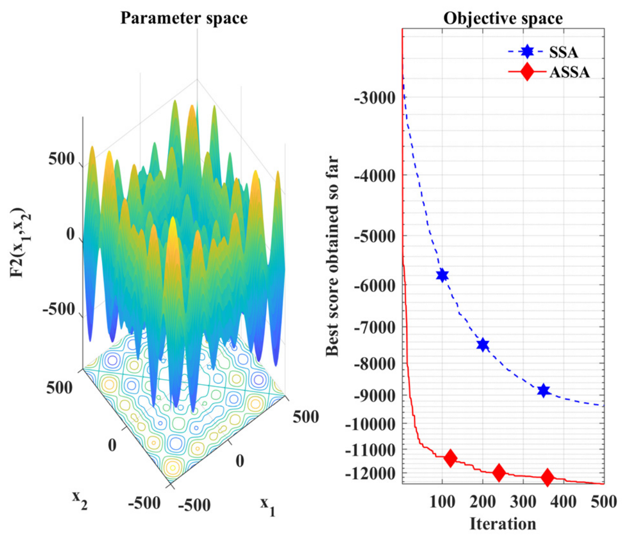
Fig. 1(a)-(d) illustrates the convergence curves of SSA and ASSA applied to four test functions. It is evident from these graphs that the convergence speed and accuracy of the ASSA method are notably superior.

Table I
Test function and parameter

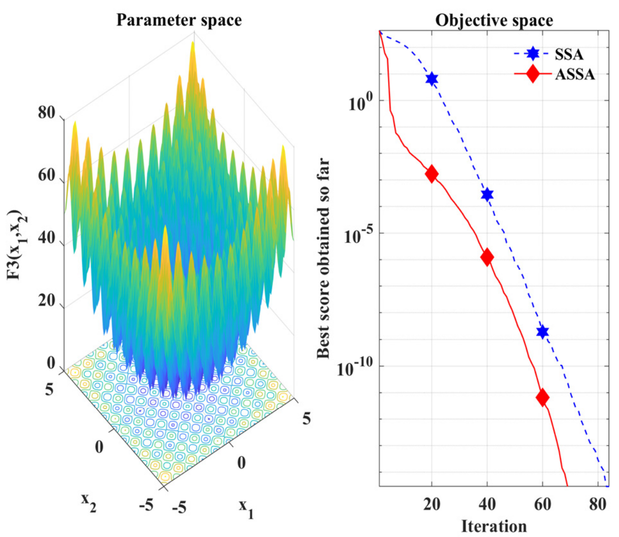
Test Function
$F_1(x) = \sum_{i=1}^{n} x_i^2$ d=30, [-100,100], 0
$F_2(x) = \sum_{i=1}^{n} -x_i \sin(\sqrt{\lvert x_i \rvert})$ d=30, [-500,500], -418.98*30
$F_3(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$ d=30, [-5.12,5.12], 0
$F_4(x) = -\sum_{i=1}^{4} c_i \exp(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2)$ d=6, [0,1], -3.32



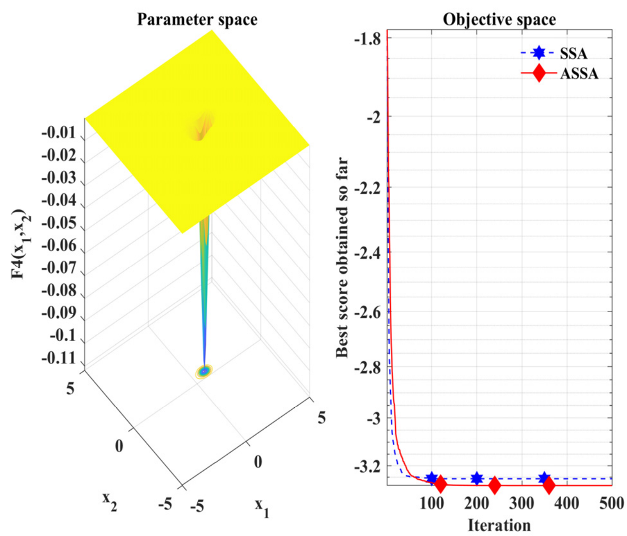
(a)



(b)



(c)



(d)

Fig. 1. The convergence curve of test function. (a) shows the convergence curve of $F_1(x)$. (b) shows the convergence curve of $F_2(x)$. (c) shows the convergence curve of $F_3(x)$. (d) shows the convergence curve of $F_4(x)$.

III. ASSA CLUSTER ROUTING APPLICATION

Applying the ASSA algorithm for cluster head election and routing communication results in the development of an algorithm known as LEACH-ASSA.

A. *The Optimal Number of Cluster Head*

In LEACH-ASSA routing protocol, the cluster head is first chosen and then the cluster is built. The optimal number of selected cluster heads plays a crucial role as it directly impacts energy usage and network lifetime. The number of cluster heads is related to the total number of nodes, the area of the region, and the distance to BS. Therefore, the optimal number of cluster heads is determined by Eq. (10).

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \bullet \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{mp}}} \bullet \frac{M}{d_{toBS}^2} \tag{10}$$

B. *Initial Population*

In each round, the cluster head accepts packets from member nodes, performs local fusion, and then forwards them to the BS. Member nodes only need to transfer packets to the cluster head. As a result, member nodes consume less energy, while cluster heads lose more capacity. If the remaining energy of some node is low, it is no longer suitable as a cluster head. Considering the remaining energy of surviving nodes, the LEACH-ASSA algorithm defines nodes with remaining energy higher than the average value of all surviving nodes as high-energy nodes. It then selects cluster heads for each round from high-energy nodes according to Eq. (11), electing p_n times to form a population.

$$E_i > \frac{1}{N_{alive}} \sum_{i=1}^{N_{alive}} E_i \tag{11}$$

C. *Fitness Function*

Equation (12) serves as the fitness function in the LEACH-ASSA algorithm for cluster head election. It encompasses three influencing factors.

$$fitness = f_1 + f_2 + f_3 \tag{12}$$

f_1 calculates the sum of squared distances from all member to their respective cluster heads. A shorter distance results in lower energy consumption for data transmission. f_1 is represented as follows in Eq. (13).

$$f_1 = \sum_{i=1}^{N-K_{opt}} (d_{itoCH}^2 \varepsilon_{amp}) \tag{13}$$

f_2 calculates the sum of the squared distances from all cluster head to the BS position. A shorter distance results in lower energy consumption for the cluster head node. f_2 is represented as follows in Eq. (14).

$$f_2 = \sum_{i=1}^{K_{opt}} (d_{itoBS}^2 \varepsilon_{amp}) \tag{14}$$

f_3 calculates the number of member C within a cluster and aims to ensure that member within each cluster is as evenly distributed as possible. f_3 is represented in Eq. (15).

$$f_3 = \sum_{i=1}^{K_{opt}} \left| \frac{N - K_{opt}}{K_{opt}} - n_i \right| \tag{15}$$

IV. SIMULATION TEACHING AND ANALYSIS

A. *Simulation Deploy*

The algorithm parameters are presented in Table II. Subsequently, we utilized Matlab to program and simulate the algorithm for educational purposes. Throughout the teaching process, Matlab digital modeling offers a visualization of the entire cluster head election, clustering, and data communication processes. This enables students to gain a deep understanding of fundamental concepts such as survival nodes, death nodes, and base stations while also vividly experiencing the data communication process among cluster heads, member nodes, and BS. Such an approach is highly beneficial for comprehending cluster routing algorithms. Furthermore, this article compares the performance of LEACH, LEACH-C, and LEACH-ASSA in terms of lifetime and energy utilization which significantly enhances students' learning experience and the effectiveness of education and teaching.

Table II
Simulation parameter

Parameter	Value	Description
$M*M$	200*200	Region size
N	200	The total sensor number
BS	(0,0)	Base station position
p	0.05	Cluster head ratio
E_0	0.5J	Initial energy
l	4000bit	Data length
p_n	15	Population number
$T_{\max}$	20	Max iteration number
E_{elec}	$50nJ/bit$	Unit data transmission energy consumption
ε_{fs}	$10pJ/bit/m^2$	Free space coefficient
ε_{mp}	$0.0013pJ/bit/m^4$	Multipath attenuation coefficient
d_0	$d_0=\sqrt{\varepsilon_{fs}/\varepsilon_{mp}}\approx 87$	Coefficient ratio
E_{DA}	$5nJ/bit$	Unit data fusion energy consumption

B. Simulation Teaching Process

Fig. 2 illustrates the outcome of cluster head election using the LEACH-ASSA algorithm in the 550th round. The depicted region size is 200 * 200, with the central pentagram representing the position of the base station and hollow circles symbolizing randomly deployed sensor nodes, totaling 200 in number. The purple solid circle indicates the cluster head elected in the 550th round. A total of 10 cluster heads are elected in each round before any occurrence of dead nodes.

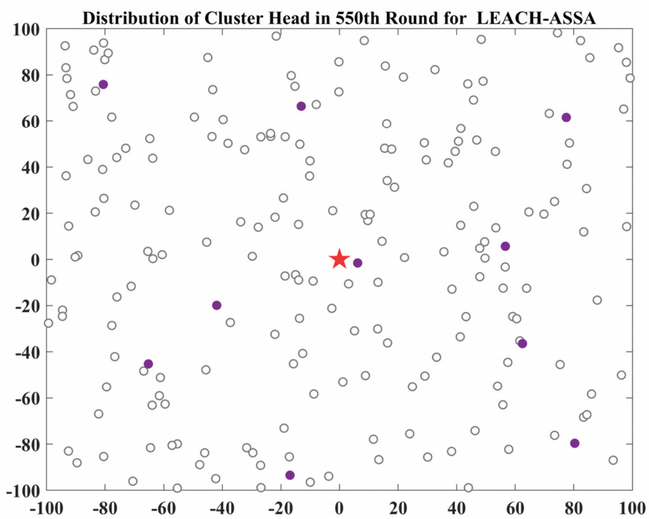


Fig. 2. Cluster head election in 550th round

During the cluster head election stage, high-energy nodes are filtered out first using formula (11), as they are more likely to become cluster heads due to their higher remaining energy levels, thus avoiding premature depletion of low-energy nodes. Subsequently, formula (12) for fitness function is employed to select an optimal set of cluster heads for this round based on minimizing energy consumption during data communication. The advantage of this approach lies in its ability not only to select a group of cluster heads with minimal energy consumption for each round but also to balance energy consumption .

Fig. 3 depicts the node clustering diagram for the 550th round. After the selection of the cluster head node, other sensor nodes will choose the closest cluster head node to form clusters. It is evident that the cluster structure is predominantly uniform, with cluster heads evenly distributed throughout the survey region. Member nodes surround their respective cluster heads in an even manner, and members within each cluster exhibit similar characteristics. This indicates that our designed fitness function has effectively fulfilled its role. A stable cluster structure is advantageous for balancing energy consumption among nodes. In this routing communication, cluster members are tasked with data collection and direct transmission to their respective cluster heads. The diagram clearly depicts the cluster structure and intra-cluster communication.

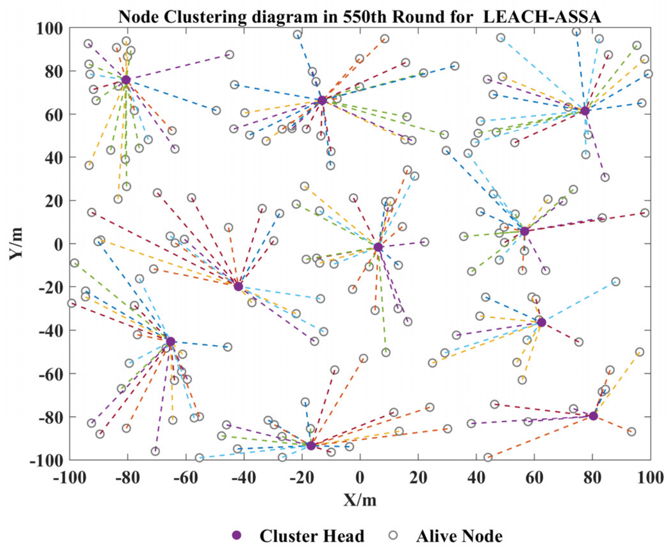


Fig. 3. Cluster structure in 550th round

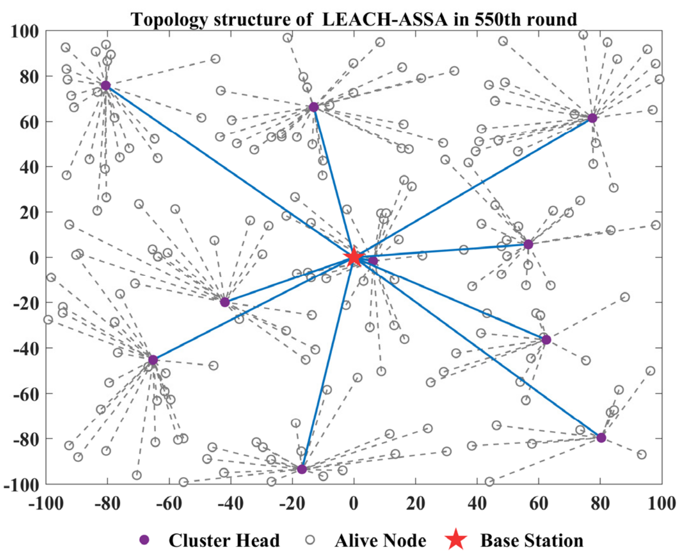


Fig. 4. Data communication structure diagram

Fig. 4 illustrates the communication process from cluster heads to BS. The LEACH-ASSA algorithm utilizes a direct one-hop communication method. Upon receiving data from member nodes, the cluster head initially conducts local data fusion and subsequently transmits the data to the BS. The figure vividly depicts the path selection and data forwarding process of the clustering routing algorithm for students, aiding in their comprehensive understanding of fundamental concepts such as survival nodes, member nodes, cluster head nodes, data fusion, routing paths, and data forwarding.

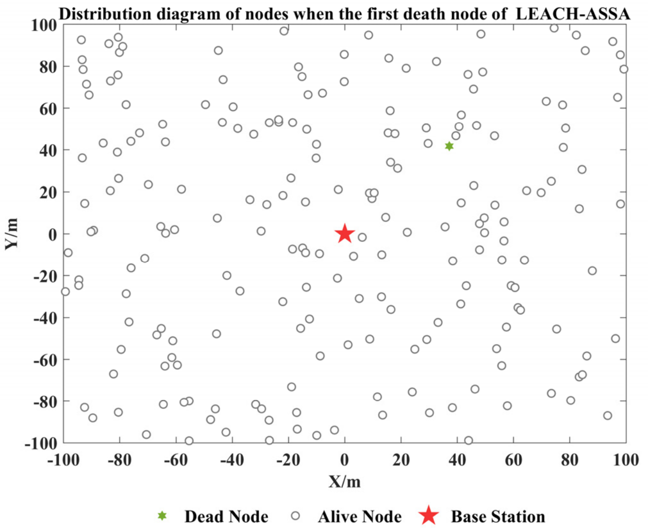


Fig. 5. Distribution map with 1st death node

Fig. 5 depicts a schematic diagram illustrating the first dead node that appears in the LEACH-ASSA algorithm. The green hexagonal star symbolizes the initial node depletion of energy. The presence of a dead node signifies a blind spot in sensor monitoring and data collection tasks, leading to decreased precision in these activities within its vicinity. Therefore, the emergence of the first dead node plays a crucial role in exploring relevant applications of IoT.

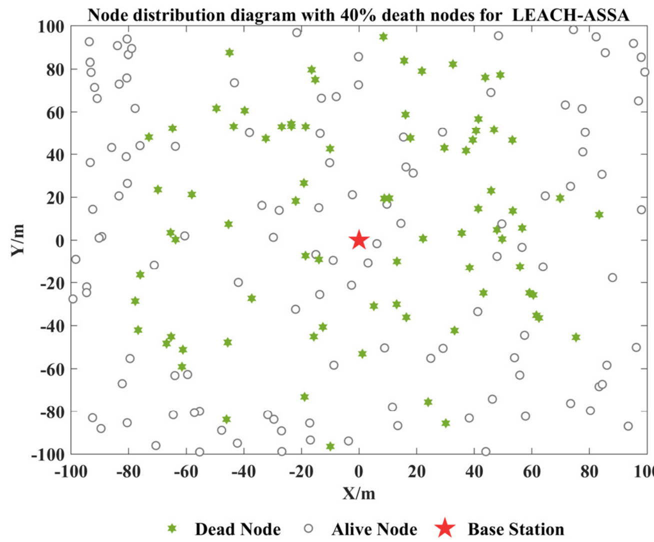


Fig. 6. Distribution map with 40% death nodes .

Fig. 6 presents a schematic diagram illustrating the presence of 40% dead nodes. In the LEACH-ASSA algorithm, the program terminates when 80% of nodes perish, indicating that half of the dead nodes have already manifested in the network at this juncture. The distribution of deceased nodes in the region appears to be relatively uniform, suggesting that sensor nodes have similar probabilities of being chosen as cluster heads and consequently exhibit comparable energy consumption patterns.

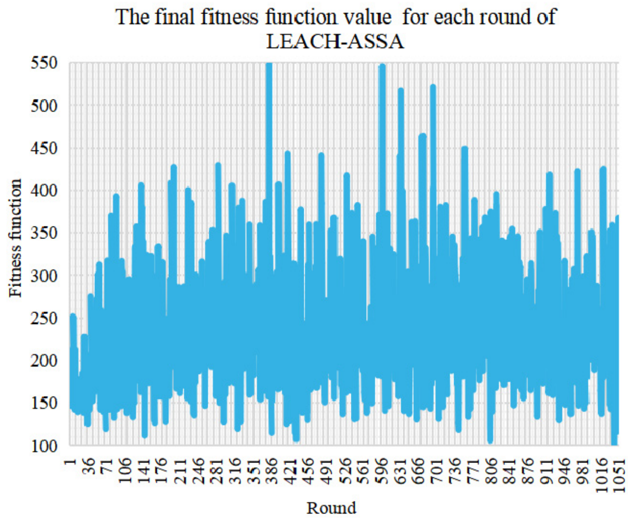
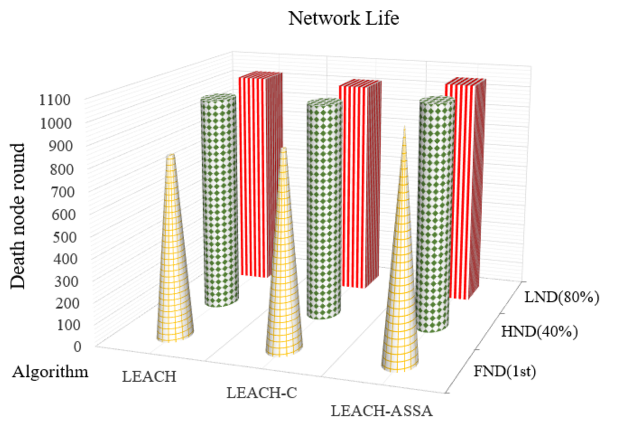


Fig. 7. Fitness function value in each round

Fig. 7 illustrates the fluctuation of fitness function values for each round of the LEACH-ASSA algorithm. The graph indicates that the fluctuation range of the fitness function value is predominantly between 150 and 300, demonstrating a relatively small and stable fluctuation range. Although there are occasional individual high values, reaching around 550; these occurrences are rare and represent small probability events suggesting that cluster head election scheme in each round remains highly effective at minimizing energy consumption while ensuring consistency across rounds.

C. Network Lifetime

Fig. 8 illustrates comparison graphs of the network lifetimes of three algorithms.



	LEACH	LEACH-C	LEACH-ASSA
FND(1st)	829	897	1029
HND(40%)	998	1007	1047
LND(80%)	1027	1014	1051

Fig. 8. Comparison chart of cumulative death nodes

Lifetime is defined as the round at which the first death node appears (FND). A later appearance of a death node indicates better overall operation of the network, while no death nodes indicate stable network performance. The appearance of a death node signifies a monitoring blind spot in the network. Additionally, this algorithm defines the round at which 80% of nodes in the network have died as program ending (LND), and the round at which half of nodes (40%) have died as HND.

Fig. 8 illustrates the occurrence of the first dead node in the LEACH algorithm at round 829, while LEACH-C terminates at round 897. In contrast, LEACH-ASSA emerges in the 1029th round, which is 200 rounds later than LEACH. It is noteworthy that the lifetime of LEACH-ASSA is the longest. LEACH algorithm concludes at round 1027, while for LEACH-C it ends at round 1014 and for LEACH-ASSA it finishes at round 1051. It is worth noting that the termination times of these three algorithms are almost identical, with a difference not exceeding 30 rounds. This phenomenon can be attributed to uneven energy consumption between nodes in the first two algorithms, resulting in some nodes having a longer survival time and consequently prolonging the overall program duration. From a vertical perspective, the first dead node in the LEACH algorithm occurs at round 829, and the program concludes at round 1027. It takes approximately 200 rounds from the appearance of the first dead node to the conclusion of the program, indicating a relatively lengthy process. Similarly, in the case of LEACH-C, the process from round 897 to round 1014 spans about 120 rounds, also considered relatively long. On the other hand, utilizing the LEACH-ASSA algorithm results in a process ranging from round 1029 to round 1051, which only takes approximately 20 rounds. This rapid progression suggests that energy consumption in LEACH-ASSA is essentially uniform.

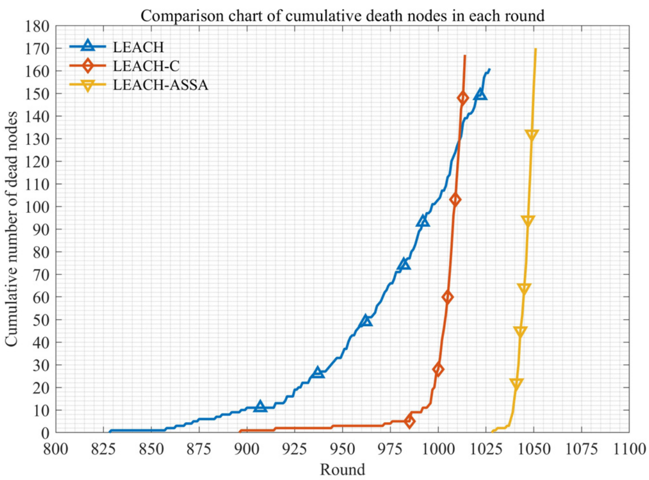


Fig. 9. Comparison chart of network life

Fig. 9 illustrates a comparison graph of the lifetime, vividly depicting the entire process of changes in the algorithm program's lifetime. The graph clearly shows that both LEACH and LEACH-C exhibit prolonged periods of decline, while LEACH-ASSA demonstrates a nearly constant trend. This indicates that all node deaths occur within a relatively short time, suggesting an ideal effect in terms of uniform energy consumption and supporting the rationality of the fitness function. Additionally, this diagram facilitates a more vivid and profound understanding of basic concepts and principles such as lifetime and energy utilization for students.

D. Lifetime with Different Cluster Head

Table III and Fig. 10 present a comparison of the network lifetime of three algorithms across varying cluster head ratios.

Table III
Lifetime with different cluster head

Cluster Head	LEACH	LEACH-C	LEACH-ASSA
$p = 0.03$ (6)	507	602	976
$p = 0.04$ (8)	704	780	1009
$p = 0.05$ (10)	829	897	1029
$p = 0.06$ (12)	884	1012	1050
$p = 0.07$ (14)	894	1042	1059
$p = 0.08$ (16)	889	1055	1069
$p = 0.09$ (18)	916	1062	1079
$p = 0.10$ (20)	911	1073	1086
$p = 0.11$ (22)	884	1081	1090
$p = 0.12$ (24)	872	1083	1093
$p = 0.13$ (26)	860	1087	1083
$p = 0.14$ (28)	821	1086	1093
$p = 0.15$ (30)	821	1060	1094

As the ratio increases, the lifetime of LEACH initially rises and then declines, indicating the presence of an optimal ratio in the LEACH algorithm. In a 200*200 area, it is observed that approximately 16-20 cluster heads are optimal for maximizing lifetime. For LEACH-C, the lifetime also increases until it stabilizes at 22 cluster heads; beyond 30 cluster heads, there is a decline in lifetime. The initial ratio for LEACH-ASSA is found to be 0.05; decreasing this ratio results in decreased lifetime for LEACH-ASSA as well. Conversely, an increase in the ratio leads to only a slight increase in lifetime not exceeding 50 rounds. These findings suggest that the initial ratio discussed in this article is quite reasonable.

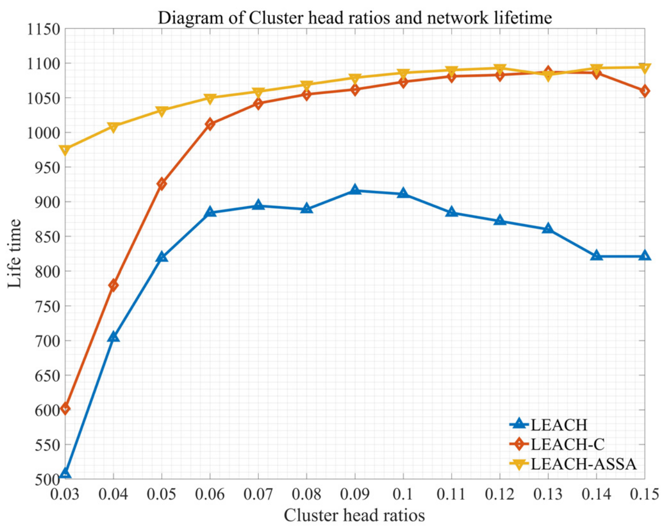


Fig. 10. Lifetime with different cluster head

E. Average Energy Usage

Fig. 11 illustrates the remaining energy of nodes that have survived in the region. Each node initially possesses an energy level of 0.5 joules for all three algorithms, and there are a total of 200 nodes in the region. The overall energy carried by sensor nodes in the region is 100 joules, which remains constant across the algorithms. In the case of the LEACH algorithm, when the first node reaches depletion, there is still a significant amount of energy remaining in the region. Similarly, with the LEACH-C algorithm, there is also a relatively large amount of remaining energy when the first node becomes inactive. However, with the LEACH-ASSA algorithm, the remaining energy ratio at the point of its first dead node is already relatively low. Based on Fig. 8, it is evident that the lifetime of LEACH-ASSA is longest, reaching 1029 rounds. This indicates that the LEACH-ASSA algorithm program is relatively stable and capable of running for an extended period of time. In contrast to this finding, it can be observed that both LEACH and LEACH-C have shorter lifespans implying non-uniform energy consumption patterns. Additionally noticeable observation shows that lines representing performance for LEACH-ASSA are relatively flat while those for LEACH show gradual decrease and those for LEACH-C exhibit significant turning points indicating optimal performance with 80% dead nodes occurring at relatively low energy levels.

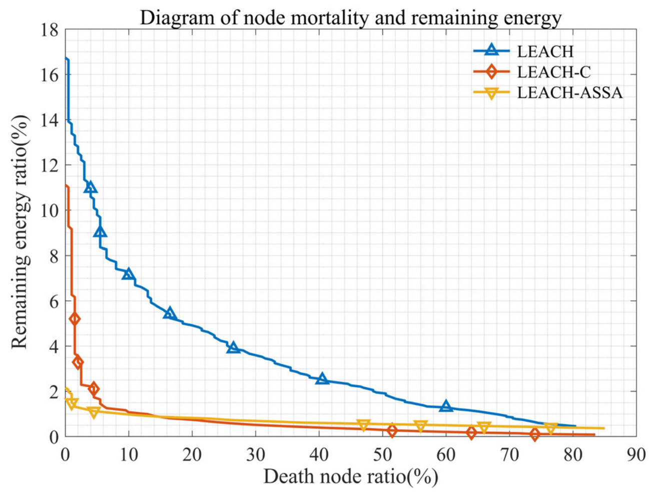


Fig. 11. Comparison chart of remaining energy

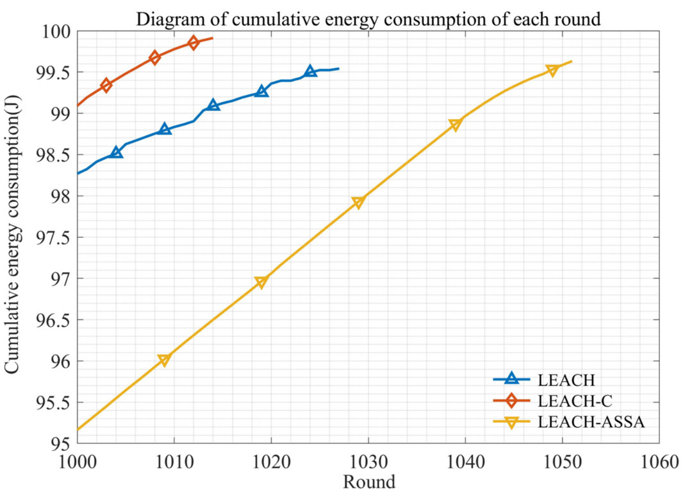


Fig. 12. Cumulative energy consumption for each round

Fig.12 illustrates a comparison of cumulative energy consumption for each round among these three algorithms. For all three algorithms, there is a significant difference in cumulative energy consumption after running for 1000 rounds. At this stage, the LEACH-C algorithm exhibits the highest cumulative energy consumption, exceeding 99 joules. It is followed by the LEACH algorithm, which also surpasses 98 joules, and then the LEACH-ASSA algorithm with the lowest energy consumption at slightly over 95 joules. As the network operates further, it can be observed that the LEACH-C algorithm consumes nearly 100 joules of energy in 1014 rounds, leading to all nodes in the network depleting their energy. The LEACH algorithm consumes over 99.5 joules of energy in 1027 rounds while the LEACH-ASSA algorithm consumes approximately 96.5 and 97.5 joules in respective rounds, maintaining network operation until round 1051. It is evident that compared to other two algorithms, the LEACH-ASSA algorithm consumes less energy per round.

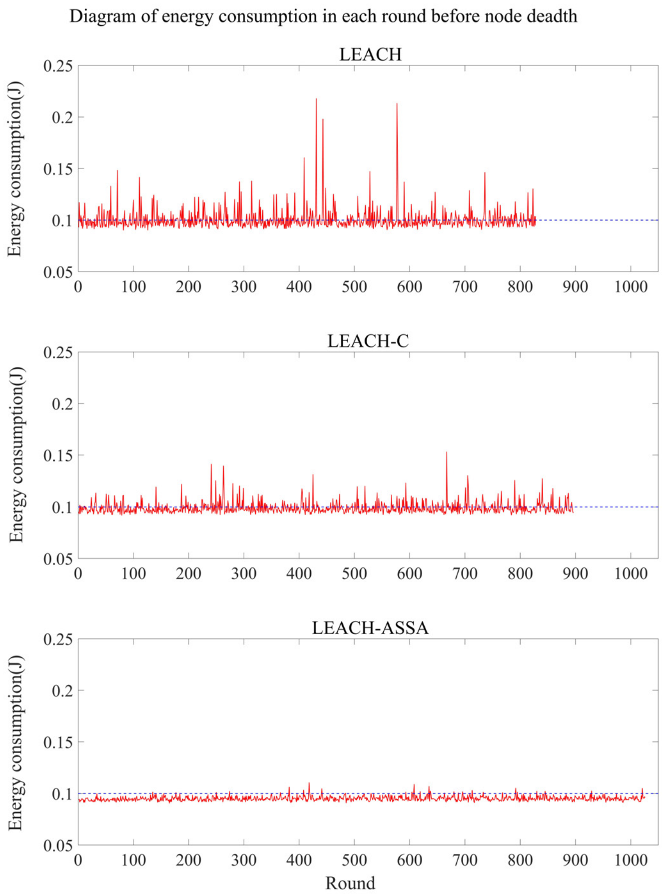


Fig. 13. Comparison chart of energy consumption in each round before node dead

Fig. 13 illustrates that the energy consumption of each round of the LEACH algorithm fluctuates around 0.1 joules, with a large upward fluctuation amplitude and a high frequency of abnormally high energy consumption occurrences. High peak energy consumption occurs multiple times and exceeds 0.2 joules. The energy consumption of each round for the LEACH-C algorithm also fluctuates around 0.1 joules, with relatively small fluctuation amplitude and fewer instances of abnormally high energy consumption, reaching a peak at 0.15 joules. In contrast, the energy consumption of each round of the LEACH-ASSA algorithm fluctuates below the 0.1 joule line, with extremely small fluctuation amplitude and no abnormally high energy consumption.

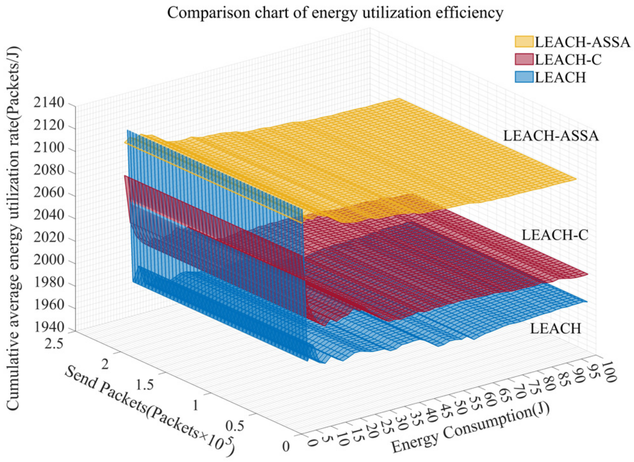


Fig. 14. Comparison chart of energy utilization efficiency

Fig. 14 illustrates a comparison of energy utilization rates for each round of three algorithms. In this context, average energy usage is defined as the number of sent packets per joule. As depicted in Fig. 13, the energy utilization plane of the LEACH-ASSA algorithm demonstrates the highest level and tends to be parallel. This suggests that the LEACH-ASSA algorithm has the highest energy utilization rate and maintains a relatively balanced performance in each round. The energy utilization efficiency of LEACH-C falls within the middle range, while LEACH exhibits the lowest energy utilization efficiency. In comparison to the LEACH-ASSA algorithm, both LEACH-C and LEACH show some degree of fluctuation in their planes, indicating poor balance in energy utilization.

V. Conclusion

This article introduces a low-power cluster routing algorithm based on the adaptive SSA algorithm and applies it to MATLAB simulation for educational research purposes. The original SSA algorithm is first improved by incorporating adaptive adjustment factors using sine and cosine functions. Furthermore, Levy flight is integrated to enhance its ability to escape from local extremes, while standard normal distribution random numbers are added to increase individual diversity. This enhanced algorithm is then utilized in simulating teaching research on cluster routing communication.

The simulation visually demonstrates the processes of cluster head election, clustering, data communication, and node death in clustering algorithms. This visual teaching process enables students to develop a deeper and more intuitive understanding of important characteristics such as low power consumption, network lifetime, and IoT energy utilization, thereby enhancing their cognition and improving teaching effectiveness. The next step will involve investigating the performance of the LEACH-ASSA algorithm in heterogeneous network deployment.

References

[1] C. Nakas, D. Kandris, and G. Visvardis, “Energy Efficient Routing in Wireless Sensor Networks: A Comprehensive Survey,” *Algorithms.*, vol. 13, no. 3, pp. 72, 2020.

[2] C. L. Tang, “A clustering algorithm based on nonuniform partition for WSNs,” *Open Physics.*, vol. 18, no. 1, pp. 1154-1160, 2020.

[3] S. Sowndeswari and E. Kavitha, “A comparative study on energy efficient clustering based on metaheuristic algorithms for WSN,” *International Journal of Advanced Technology and Engineering Exploration.*, vol. 9, no. 86, pp. 111-126, 2022.

[4] M. Wu, Z. L. Li, J. Chen, Q. S. Min, and T. Lu, “A Dual Cluster-Head Energy-Efficient Routing Algorithm Based on Canopy Optimization and K-Means for WSN,” *Sensors.*, vol. 22, no. 24, pp. 9731, 2022.

[5] H. Koyuncu, G. S. Tomar, and D. Sharma, “A New Energy Efficient Multitier Deterministic Energy-Efficient Clustering Routing Protocol for Wireless Sensor Networks,” *Symmetry.*, vol. 12, no. 5, pp. 837, 2020.

[6] T. T. Zhang, “An intelligent routing algorithm for energy prediction of 6G-powered wireless sensor networks,” *Alexandria Engineering Journal.*, vol. 76, pp. 35-49, 2023.

[7] A. M. Hilal, S. B. H. Hassine, J. S. Alzahrani, M. Alajmi, F. N. Al-Wesabi, M. A. Duhayyim, I. Yaseen, and A. Motwakel, “Echo Location Based Bat Algorithm for Energy Efficient WSN Routing,” *Computers, Materials & Continua.*, vol. 71, no. 3, pp. 6351-6364, 2022.

[8] M. Manikandan, S. Sakthivel and V. Vivekanandhan, “Efficient Clustering Using Memetic Adaptive Hill Climbing Algorithm in WSN,” *Intelligent Automation & Soft Computing.*, vol. 35, no. 3, pp. 3169-3185, 2023.

[9] D. Anuradha, R. Srinivasan, T. Ch. Anil Kumar, J. Faritha Banu, A. K. S. Pundir and D. Vijendra Babu, “Energy Aware Seagull Optimization-Based Unequal Clustering Technique in WSN Communication,” *Intelligent Automation & Soft Computing.*, vol. 32, no. 3, pp. 1325-1341, 2022.

[10] P. S. Mehra, M. N. Doja, and B. Alam, “Fuzzy based enhanced cluster head selection (FBECS) for WSN,” *Journal of King Saud University-Science.*, vol. 32, no. 1, pp. 390-401, 2020.

[11] V. K. Venkatesan, I. Izonin, J. Periyasamy, A. Indirajithu, A. Batyuk, and M. T. Ramakrishna, “Incorporation of Energy Efficient Computational Strategies for Clustering and Routing in Heterogeneous Networks of Smart City,” *Energies.*, vol. 15, no. 20, pp. 7524, 2022.

[12] A. Saini, A. Kansal, and N. S. Randhawa, “Minimization of Energy Consumption in WSN using Hybrid WECRA Approach,” *Procedia Computer Science.*, vol. 155, pp. 803-808, 2019.

[13] L. Y. Chang, F. Li, X. Z. Niu, and J. H. Zhu, “On an improved clustering algorithm based on node density for WSN routing protocol,” *Cluster Computing.*, vol. 25, pp. 3005-3017, 2022.

[14] F. F. Jurado-Lasso, K. Clarke and A. Nirmalathas, “Performance analysis of software-defined multi-hop wireless sensor networks,” *IEEE Systems Journal.*, vol. 14, no. 8, pp. 4653-4662, 2020.

[15] A. A. Jovith, M. Mathapati, M. Sundarrajan, N. Gnanasankaran, S. Kadry, M. N. Meqdad and S.M. Aslam, “Two-Tier Clustering with Routing Protocol for IoT Assisted WSN,” *Computers, Materials & Continua.*, vol. 71, no. 2, pp. 3375-3392, 2022.

[16] B. Saoud, I. Shayea, M. H. Azmi, and A. A. El-Saleh, “New scheme of WSN routing to ensure data communication between sensor nodes based on energy warning,” *Alexandria Engineering Journal.*, vol. 80, pp. 397-407, 2023.

[17] J. Yang, X. L. Liu, and Q. Xu, “A new clustering routing protocol for heterogeneous WSNs,” *Transducer and Microsystem Technologies.*, vol. 39, no. 4, pp. 121-128, 2020.

[18] S. Q. Yan, W. D. Liu, X. Q. Li, P. Yang, F. X. Wu, and Z. Yan, “Comparative Study and Improvement Analysis of Sparrow Search Algorithm,” *Wireless Communications and Mobile Computing.*, vol. 2022, Article ID 4882521, 2022.

[19] J. Ma, Z. Y. Hao, and W. J. Sun, “Enhancing sparrow search algorithm via multi-strategies for continuous optimization problems,” *Information Processing and Management.*, vol. 59, no. 2, pp. 102854, 2022.

[20] N. Zhou, S. L. Zhang , and C. Zhang, “Multi Strategy Improved Sparrow Search Algorithm Based on Rough Data Reasoning,” *Journal of University of Electronic Science and Technology of China.*, vol .51, no. 5, pp. 743-753, 2022.

[21] Q. C. Sun, Z. L. Zhu, and F. Q. Zhang, “Optimizing WSN Clustering Protocol Based on Improved SSA,” *Radio Communications Technology.*, vol. 48, no.1, pp. 132-139, 2022.