Measurement: Sensors 39 (2025) 101883

Contents lists available at ScienceDirect

Measurement: Sensors

journal homepage: www.sciencedirect.com/journal/measurement-sensors

Distributed data acquisition optimization algorithm for wireless sensor networks

Youxian Zhang [a,*], Zhen Nie [a], Hongxu Zhang [b]

[a] School of Electrical Engineering & Automation, Henan Institute of Technology, Xinxiang, 453003, China
[b] Henan Kelong Battery Materiall Co., Ltd., Xinxiang, 453003, China

ARTICLE INFO

Keywords:
Wireless sensor network
Distributed
Data collection
Sparrow search algorithm
Virtual grid

ABSTRACT

With the rapid development of applications such as the Internet of Things and intelligent transportation, wireless sensor networks play an important role in data collection and environmental monitoring. However, wireless sensor networks face low efficiency and high energy consumption in distributed data collection and node configuration. In this context, a sensor node configuration optimization algorithm based on an improved sparrow search algorithm by introducing reverse elite selection, dynamic perturbation, and dynamic warning update strategies is proposed. Secondly, a virtual grid partitioning strategy is designed, and a distributed data collection and transmission optimization algorithm is proposed. The node configuration algorithm achieved the most uniform distribution of nodes in simulation testing and almost achieved complete region coverage. Under 30 % node failure, its network coverage rate was 83.5 %. When the packet size was 1000 kb, the data transmission rate and average communication delay of the data collection algorithm were 4.2 Mbps and 42 ms, respectively. Compared with existing algorithms, the proposed scheme performs well in coverage retention, energy consumption reduction, and fault recovery capability, and can meet the efficient and reliable distributed data collection needs of wireless sensor networks in complex environments.

1. Introduction

Wireless Sensor Network (WSN), as a core technology in modern intelligent systems, is widely used in various fields such as environmental monitoring, intelligent transportation, military reconnaissance, etc [1]. With the rapid development of IoT technology, WSN plays a key role in data collection and real-time monitoring [2]. However, WSN often faces problems such as limited node energy, limited communication range, and low data transmission efficiency in large-scale networks during actual deployment. Especially in complex environments and high dynamic changes, traditional centralized data collection methods are difficult to meet the requirements of efficient transmission and low energy consumption [3,4]. Therefore, distributed data collection optimization algorithms have gradually become a hot research direction. In recent years, significant progress has been made in intelligent optimization algorithms in Wireless Sensor (WS) node configuration and data acquisition optimization [5]. Kou G et al. proposed an optimization strategy based on an improved Sparrow Search Algorithm (SSA) to address the insufficient coverage and positioning error control in WSN nodes. The deviation correction factor and the minimum mean square error criterion were introduced, while utilizing an improved population initialization process. The results showed that the optimized positioning average error was 0.72 m, which was 77.71 % lower than the traditional DV-Hop algorithm [6]. Wajgi D W et al. proposed a clustering-based node deployment optimization strategy for node localization in WSN and wireless multimedia sensor networks. WS nodes were grouped into independent subgroups to reduce communication costs and improve network throughput through distributed computing and parallel processing [7]. Jaiswal K et al. proposed a node scheduling and deployment method based on firefly optimization algorithm to maximize the lifespan and robustness of WSN node deployment in the Internet of Things. A fitness function was designed, taking into account factors such as node distance, survivability, coverage, and connectivity. The algorithm was statistically analyzed through simulation results [8]. Boualem A et al. proposed a hybrid model based on fuzzy possibility theory. Uncertainty theory was used to schedule and plan the active/passive states of sensor nodes, and optimize node activation strategies through possibility information fusion. The results showed that the model had significant

This article is part of a special issue entitled: Smart sensing Agriculture published in Measurement: Sensors.

* Corresponding author.
E-mail address: zhangyouxian1020@163.com (Y. Zhang).

https://doi.org/10.1016/j.measen.2025.101883
Received 3 December 2024; Received in revised form 11 April 2025; Accepted 13 May 2025
Available online 14 May 2025

advantages in regional coverage issues, maintaining a coverage rate of 99.99 %–90.00 % over a long period of time [9].

In order to further improve network performance, scholars have also proposed various data collection optimization strategies, combined with node deployment and path planning, to achieve efficient and low-energy data transmission solutions. Xuan C proposed a health sensor data acquisition system based on symmetric encryption algorithm to address the security issues of data collection in WSN. A data acquisition circuit for MSP430 module was designed, and MapReduce model was used for data acquisition. At the same time, symmetric encryption algorithm was introduced to achieve privacy protection function [10]. Srinivas M et al. proposed a mobile sink scheduling method based on clustering and trajectory optimization to address the complex scheduling problem of mobile receivers in WSN. The optimal mobile receiver set was conducted through hierarchical clustering and geometric modeling to reduce node energy consumption and data transmission delay. The research results indicated that this method could determine the optimal number of mobile sinks [11]. Xiaoxiang S et al. proposed a missing data recovery model based on compressed sensing to address the data loss and difficult recovery in WSN. The measurement matrix and sparse representation matrix were designed, and an improved fast matching tracking algorithm was introduced to eliminate the dependence on prior information. The results indicated that this method could effectively recover lost data [12]. Pravija Raj P V et al. proposed a trajectory planning method for unmanned aerial vehicles based on improved genetic algorithms. A data collection scheme was designed for the collaboration between unmanned aerial vehicles and ground WS nodes. The trajectory path in a three-dimensional environment was optimized. The results showed that using this method extended the network lifetime by about 11 %, and reduced path length and flight time by 42 % and 35 %, respectively [13].

In order to further improve the data collection and node scheduling strategies, some studies have conducted useful explorations from the perspective of cross-domain applications. Choudhary et al. proposed a hybrid data collection scheme combining grey wolf optimization and whale optimization to address the problems of high node mobility and difficult path planning in mobile-assisted underwater wireless sensor networks. This method optimizes the data path scheduling strategy of the receiver by introducing factors such as buffer occupancy, node energy, and transmission delay [14]. Kaur et al. proposed a hybrid deep learning model that integrates convolutional neural networks, support vector machines, and attention mechanisms. This method can automatically classify a variety of plant leaf diseases with an accuracy of 98.72 % [15]. Gupta et al. proposed a cross-device, lightweight federated multimodal classification system to achieve early diagnosis and privacy protection of depression. The system builds a model based on electroencephalogram and audio signals and achieves efficient classification under the premise of data privacy [16]. Fan et al. proposed a clustering and routing algorithm to address the rapid changes in large-scale WSNs in the Internet of Things. The algorithm dynamically adapts to changes in node energy consumption and distribution according to the clustering radius to address the challenges of changes in node energy consumption and distribution in large-scale sensor networks [17]. Gulganwa et al. proposed an energy-saving and secure weighted clustering algorithm based on machine learning to address the issues of energy efficiency and security in WSNs. This method combines energy efficiency optimization with a machine learning-based intrusion detection system to detect and classify malicious nodes by creating network clusters and collecting traffic samples on base stations [18].

In summary, WSN still faces problems such as frequent node failures, high energy consumption, and increased complexity in path planning during node configuration optimization and data collection in complex environments. In view of this, a node configuration strategy based on Enhanced Adaptive Sparrow Optimization Algorithm (EASOA) and a Virtual Grid-based Distributed Data Collection and Transmission Optimization Algorithm (VG-DDCOTA) are proposed in this study.

The innovation of the research lies in that, unlike existing studies that only introduce a single perturbation mechanism or local update strategy into traditional algorithms, EASOA comprehensively introduces three mechanisms: reverse elite selection, brightness-driven perturbation, and dynamic warning update, which enhances the adaptability of the algorithm in dynamic scenarios while improving global search capabilities and convergence efficiency. Compared with common data collection methods based on clustering or fixed area division, VG-DDCOTA significantly improves the efficiency and robustness of data transmission by constructing a fine-grained virtual grid structure and combining node state feedback to achieve adaptive intra-network transmission and inter-network scheduling. The above two improvements not only build a complete optimization framework from node layout to data scheduling, but also reflect obvious innovative value in terms of algorithm integration and practicality.

2. Methods and materials

2.1. Optimization design of WS node configuration based on improved SSA

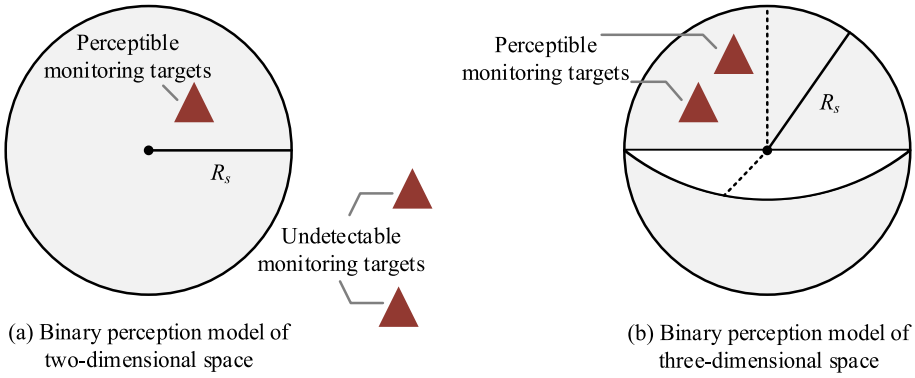
In WSN, the reasonable configuration of nodes is crucial for the efficiency of data collection and network performance [19]. Due to the limited energy and computing power of sensor nodes, it is necessary to optimize node configuration while ensuring efficient data collection and low energy consumption. Firstly, in the field of WSN configuration optimization, the classic binary perception model is shown in Fig. 1 [20, 21].

Fig. 1 (a) and 1 (b) are perception models of two-dimensional and three-dimensional planes, respectively, revealing the possible monitoring blind spots in the network. In Fig. 1 (a), the node is able to cover and detect the monitoring target within its perception radius R_s. In Fig. 1 (b), its coverage area is a sphere with the same perceived radius. In the model, the monitoring area is divided into perceivable and imperceptible areas. When the target is located outside the perception radius, the node cannot be detected. The research mainly focuses on the configuration problem of WS nodes in a two-dimensional plane. Further optimization of node configuration should consider more complex node layout and coverage to improve data collection efficiency. The optimized coverage range is shown in Fig. 2.

Fig. 2 (a) and 2 (b) respectively show the probability coverage structure and hexagonal coverage structure. The probability coverage model in Fig. 2 (a) defines regions with different perception probabilities through a layered circular structure. The layers with gradually decreasing perception radii make the target object have different perception probabilities within the coverage area of the node, which is suitable for perception needs with uneven distribution. Fig. 2 (b) uses a hexagonal geometric distribution to make the coverage area of WS nodes more compact and uniform, reducing blind spots and facilitating communication path design between nodes. Assuming node p_j in monitoring area P, its perception probability $B\left(p_j\right)$ is represented by the superposition probability of multiple WS nodes s_i, as shown in equation (1).

$$B\left(p_j\right) = 1 - \prod_{i=1}^{n}\left(1 - B\left(s_i, p_j\right)\right) \tag{1}$$

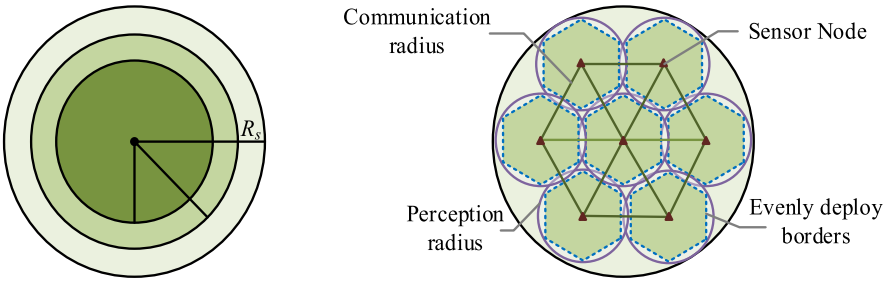
In equation (1), $B\left(s_i, p_j\right)$ B represents the perception probability of the sensor node s_i to the monitoring point p_j, which is determined according to the distance between the two and the perception radius. The closer the point is to the node, the higher the perception probability. Monitoring points are preset discrete sampling points within the monitoring area, which are used to evaluate the network coverage performance and should be effectively sensed by at least one sensor node in an ideal state. n is the total number of WS nodes covering that point. Subsequently, the total coverage of the monitoring area is expressed as equation (2).



(a) Binary perception model of two-dimensional space

(b) Binary perception model of three-dimensional space

Fig. 1. Binary sensing model of sensor node.



(a) Probabilistic covering structure

(b) Hexagonal covering structure

Fig. 2. Coverage range diagram.

$$R_{cover} = \frac{\sum B\left(p_j\right)}{m} \tag{2}$$

In equation (2), R_{cover} represents the total coverage rate. m is the total number of monitoring points.

After calculating the coverage probability, it is particularly important to optimize the WS node configuration to improve the coverage. In order to better improve the coverage and reduce blind spots, it is necessary to select an optimization algorithm with efficient global search capabilities and stable local development capabilities. Compared with particle swarm optimization and genetic algorithms, the sparrow search algorithm has stronger global search capabilities and faster convergence speed, can more effectively jump out of the local optimum, and can effectively deal with high-dimensional nonlinear problems in WSN node configuration. Therefore, this study selects SSA as the solution algorithm for WS node configuration.

In addition, to improve the search accuracy of the algorithm while accelerating the convergence speed and avoiding getting stuck in local optima, reverse elite selection, brightness driven perturbation, and dynamic warning update strategies are introduced to enhance its global search ability and convergence performance in complex optimization problems. Firstly, the reverse elite selection strategy is based on the reverse generation of new solutions by elite individuals. The reverse elite position is shown in equation (3).

$$x' = x_{\max} + x_{\min} - x \tag{3}$$

In equation (3), x and x' respectively represent the current elite individual position and the generated reverse elite position. Here, the position refers to the coordinate vector of the sensor node in the solution space. In a two-dimensional deployment scenario, the node position x represents a coordinate point on the plane (x_1, x_2). $x_{\max}$ and $x_{\min}$ represent the maximum and minimum positions in the current population, respectively. By generating reverse solutions to increase population diversity and expand search space, it is more likely to escape from local optima. On this basis, the fitness value of each reverse elite individual is calculated and the individual with better fitness is selected for population update, as shown in equation (4).

$$x_{new} = \begin{cases} x', & if \quad f(x') < f(x) \\ x, & otherwise \end{cases} \tag{4}$$

In equation (4), $f(x)$ and $f(x')$ represent the fitness values of the current elite x and the reverse elite x', respectively. x_{new} is the optimized new individual. Secondly, the brightness driven perturbation strategy adjusts individual positions through distance and brightness. The perturbation formula based on brightness attraction is shown in equation (5).

$$x_i = x_i + \beta \cdot e^{-\gamma \cdot d_{ij}^2} \cdot \left(x_j - x_i\right) + \alpha \cdot \theta \tag{5}$$

In equation (5), $d_{ij} = \left\| x_i - x_j \right\|$ represents the distance between individuals i and j. β and γ represent attraction and attenuation coefficients, respectively. α and θ control the disturbance intensity. When x_i approaches x_j, x_i is moved to a more optimal position through brightness driving. To enhance individuals' adaptability to different goals, an adjustment formula for the dynamic attraction coefficient β is added, as shown in equation (6).

$$\beta = \beta_{initial} \times \left(1 - \frac{k}{K_{\max}}\right) \tag{6}$$

In equation (6), $\beta_{initial}$ is the initial attraction coefficient. k and $K_{\max}$ represent the current iteration count and maximum iteration count, respectively.

Finally, to enhance the robustness of the algorithm and its ability to adapt to dynamic environments, a dynamic warning update strategy is introduced. This strategy simulates the behavior of individuals "escaping from potential risks" and makes guided jumps based on their positional relationship with the current optimal solution. Specifically,

the direction vector between the current individual and the optimal individual is calculated and multiplied by a random factor to obtain the update amplitude, thereby generating a new position. In each iteration, the warning individuals dynamically update their positions based on environmental changes to address potential risks, and the update calculation is shown in equation (7).

$$x_{new} = x + \delta \cdot (r \cdot x_{best} - x) \tag{7}$$

In equation (7), x, x_{best}, and x_{new} represent the current warning individual, optimal solution, and updated position, respectively. δ is the warning response coefficient. r is a random number that follows a uniform distribution. When the population searches for areas close to the edge, dynamic warning updates help individuals actively stay away from possible local optimal traps. Among them, the optimal trap refers to the local optimal point in the solution space, where the algorithm may converge prematurely and miss the global optimal solution.

On the basis of applying the above strategies to improve the optimization effect, the overall optimization goal of the WS node configuration problem is mathematically modeled to clarify the optimization direction and constraints of the algorithm. The study defines the node deployment optimization problem as a multi-objective weighted model, which comprehensively considers factors such as network coverage, node distribution uniformity, and network energy consumption. The optimization objective function is shown in formula (8).

$$\max F = \omega_1 R_{cover} - \omega_2 D_{var} - \omega_3 E_{total} \tag{8}$$

In formula (8), D_{var} represents the spatial distribution variance of all nodes in the deployment area, which is used to measure whether the nodes are evenly distributed. E_{total} is the total energy consumption of all sensor nodes in the network. ω_1, ω_2, and ω_3 are the weight coefficients of the three performance indicators, and the sum of the three is equal to 1. The node position is represented by a two-dimensional coordinate vector x_i, and its average coordinate is $\overline{x}$.

Based on the above objective function, the optimization process must also meet the following constraints: First, the position x_i of all nodes must be located within the set deployment area A. Second, to avoid excessive concentration between nodes, the distance between any two nodes must not be less than the set minimum spacing $d_{\min}$. Finally, to ensure effective coverage, the probability of being perceived $B\left(p_j\right)$ of each monitoring point should not be less than the set threshold θ. Therefore, based on the above calculations, the WS node configuration process based on EASOA is shown in Fig. 3.

As shown in Fig. 3, the EASOA node configuration algorithm first randomly initializes node positions and objective function values, and applies a reverse elite selection strategy to increase population diversity. Secondly, the fitness of individuals is calculated and ranked accordingly. If the fitness exceeds the warning value, proceed to the evacuation step towards the safe zone. The safe zone refers to the central area in the solution space that is less likely to fall into the local optimum. Algorithm individuals tend to this area during the alarm or escape stage to maintain population diversity. On the contrary, extensive searches are conducted and the discoverer's status is updated. Subsequently, brightness driven perturbations are introduced for position perturbations targeting the discoverers and joiners. Brightness driven perturbation is a perturbation strategy that simulates the distance and attraction mechanism between individuals. Low-fitness individuals move closer to high-fitness individuals to achieve adaptive adjustment of the search direction. If the individual is close to the edge, it will be updated to avoid local optima. Otherwise, it will be updated according to the edge approach strategy. Finally, whether the iteration count has reached the maximum value is determined. If reaches the maximum value, the result is output. Otherwise, continue to execute until the optimal WS node configuration result is obtained.

To more clearly understand the execution process of the EASOA node configuration optimization algorithm, its pseudo code is shown in Fig. 4.

As shown in Fig. 4, EASOA achieves an effective balance between global search and local refinement by introducing mechanisms such as reverse elite selection, brightness-driven perturbation, and dynamic warning updates, providing an optimized node deployment foundation for the subsequent data collection stage.

2.2. Design of distributed data acquisition optimization algorithm based on improved GAF

After completing the reasonable configuration of WS nodes, the optimization of node layout only provides a basic guarantee for improving network performance. In practical applications, how to efficiently organize data collection and transmission when the node configuration is fixed is a key issue to further improve the overall efficiency of wireless sensor networks [22]. Although node layout provides the foundation for data transmission, there is still room for improvement in path planning and communication optimization for distributed data collection in large-scale WSN [23,24]. Therefore, based on the Gramian Angular Field (GAF) method, a VG-DDCOTA distributed data acquisition algorithm is proposed. Virtual grids simplify data transmission path planning and reduce redundant communication between nodes by reasonably dividing monitoring areas [25]. This method is based on the optimized node positions and improves the network's energy efficiency and path planning performance during data transmission through

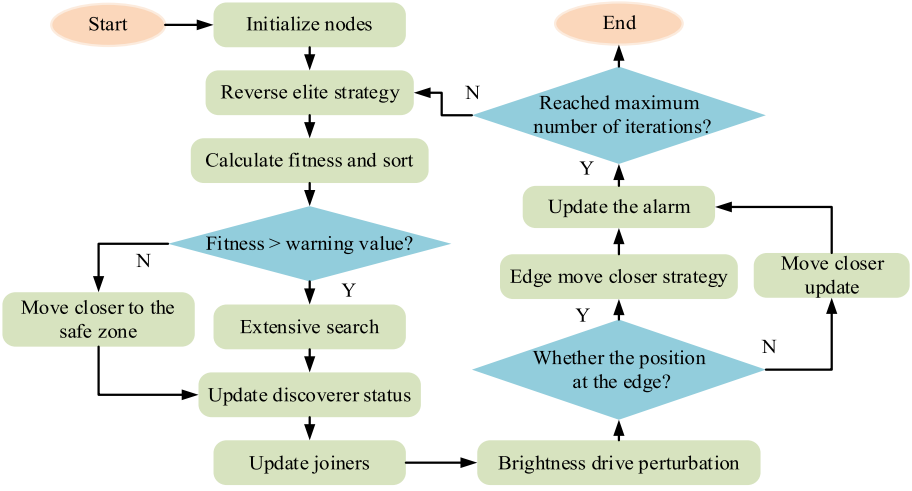


Fig. 3. EASOA flow.

Algorithm 1: EASOA-based WS Node Configuration Optimization

Input:
- N: Number of WS nodes
- MaxIter: Maximum number of iterations
- D: Deployment area boundaries
- w1, w2, w3: Weights for coverage, uniformity, and energy consumption

Output:
- Optimal node positions

```
Initialize node positions randomly within area D
Evaluate fitness of each node using multi-objective function
For each iteration k from 1 to MaxIter do
    Apply Reverse Elite Selection to increase population diversity
    Update individuals using brightness-driven perturbation
    Adjust attraction coefficient β according to Eq. (6)
    For low-performing individuals
        If located near boundary then
            Apply edge move strategy
        Else
            Move towards safe zone
    End if
    Apply dynamic warning update strategy
    Evaluate fitness and select best individuals
End for
Return best node positions
```

Fig. 4. EASOA pseudo code diagram.

virtual grid division and scheduling mechanisms, thereby achieving a systematic improvement from "structural optimization" to "communication optimization". Firstly, the schematic diagram of virtual grid partitioning is shown in Fig. 5.

Fig. 5 (a) is a schematic diagram of the virtual grid division of the monitoring area. By dividing the area into grid cells of size $L \times L$, each node within the grid only communicates with neighboring nodes within the same grid, reducing long-distance transmission energy consumption. Fig. 5 (b) shows the communication radius of nodes to ensure smooth transmission between adjacent grids. The communication radius of nodes satisfies the condition that R_s is less than $\sqrt{2}L$ to cover the edge positions of adjacent grids. If the conditions are met, the grid cells will be further refined into $m \times m$ basic coverage cells. This refinement step helps identify and determine the coverage unit to which each node belongs, thereby achieving complete coverage within the grid, as shown in equation (9).

$$\left(\frac{l}{m}\right)^2 + \left(\frac{l}{m}\right)^2 \le R_s^2 \tag{9}$$

In equation (9), m represents the number of cell subdivisions. l is the edge length of the grid cell. This ensures that each basic coverage unit is within the perception range of the node, achieving complete coverage of the entire virtual grid, and improving the coverage performance and transmission efficiency of data collection. Subsequently, the algorithm enters the distributed data collection, which is divided into two steps: intra cell Data Delivery (DD) and inter cell forwarding Scheduling Update (SU). The working sequence diagram is shown in Fig. 6.

As shown in Fig. 6, the VG-DDCOTA algorithm mainly includes the steps of node initialization, virtual grid division, route construction, data collection and scheduling. First, the node initialization configuration is completed according to the optimization results of EASOA, and the monitoring area is divided into several grid units. Subsequently, local routes are constructed within each grid to complete the data transmission between nodes. The aggregation node then forwards the data to the upper node according to the scheduling strategy to achieve cross-grid data fusion. Finally, data integration is performed through the central node, and abnormal nodes are dynamically adjusted through the status monitoring mechanism to ensure stable operation of the system.

Firstly, in the DD step, the amount of in chain data push D_{DD} can be calculated based on the number of in chain nodes N_{DD} and the data volume of each node, as shown in equation (10).

$$D_{DD} = \sum_{i=1}^{N_{DD}} d_i \tag{10}$$

In equation (10), d_i represents the amount of data generated by the i-th node. All nodes within the grid will transmit the data to the aggregation node, completing the first stage of data transmission. In the SU step, after completing the intra cell data transmission, the aggregation node needs to transfer the data to the upper layer aggregation node. This stage involves inter grid data forwarding, which is limited by the communication radius R_t and the transmission distance between nodes. The aggregation nodes of each grid select the upper level nodes within the communication distance for data forwarding. The schematic diagram of the upstream path is shown in Fig. 7.

As shown in Fig. 7, the underlying nodes $N_1, N_2, \ldots, N_k$ first transmit data to the aggregation node N_m within the cell. Subsequently, aggregation node N_n selects upper layer nodes for further data forwarding based on signal strength and communication distance. By constructing this hierarchical uplink structure, data can gradually converge from lower level nodes to upper level central nodes, achieving efficient data uplink transmission. Therefore, the total delay T_{SU} of inter cell data transmission is represented as the sum of the delays of forwarding from all underlying nodes to upper layer nodes, as shown in equation (11).

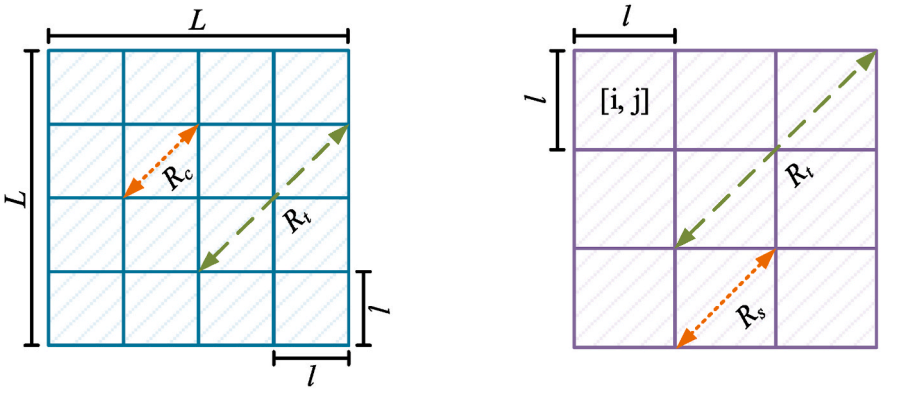


Fig. 5. Virtual grid division diagram.

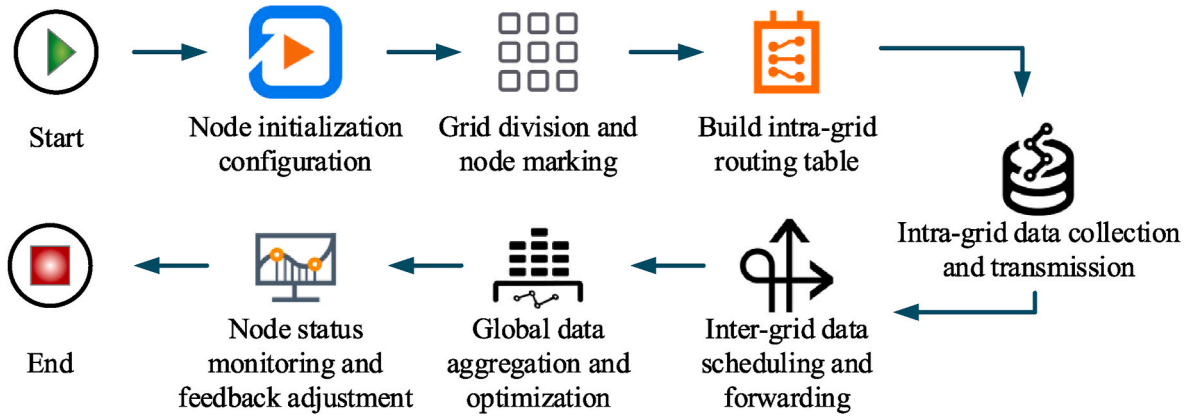


Fig. 6. Distributed data collection timing diagram.

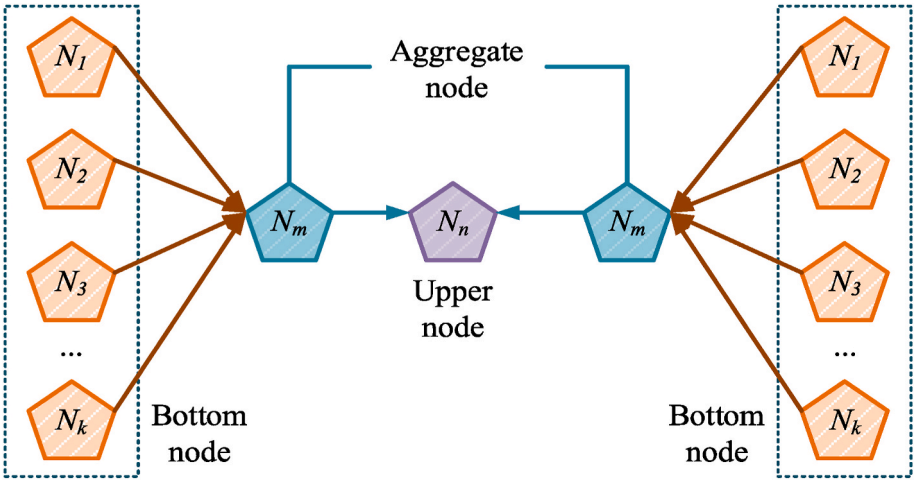


Fig. 7. Inter-grid data forwarding uplink diagram.

$$T_{SU} = \sum_{j=1}^{k} \frac{D_j}{R_j} \tag{11}$$

In equation (11), D_j represents the data volume of the j-th underlying node. R_j is the transmission rate from the bottom node N_j to the aggregation node N_m. Subsequently, the total energy consumption of inter cell data transmission E_{SU} is calculated using equation (12).

$$E_{SU} = \sum_{j=1}^{k} P_{send} \times T_j + P_{receive} \times T_j j \tag{12}$$

In equation (12), P_{send} and $P_{receive}$ represent the transmission and reception power of the node, respectively. T_j is the transmission delay of the j-th node. Finally, the inter cell data transmission efficiency is shown in equation (13).

$$\eta_{SU} = \frac{\sum_{j=1}^{k} D_j}{T_{SU}} \tag{13}$$

In equation (13), η_{SU} represents the inter cell data transmission efficiency. By evaluating the efficiency of the upstream data forwarding path, the amount of data that can be transmitted under specific transmission delays can be reflected, which can effectively measure the performance of the upstream path design. The pseudo code of VG-DDCTOA is shown in Fig. 8.

As shown in Fig. 8, the VG-DDCOTA distributed data acquisition and transmission algorithm first divides virtual grids within the monitoring

Algorithm 2: VG-DDCOTA Distributed Data Collection Algorithm

Input:
- Optimized node positions from EASOA
- L: Grid size
- Rs: Communication radius
- M: Number of grid subdivisions

Output:
- Completed data transmission to sink

1: Divide monitoring area into virtual grids of size L×L
2: Mark nodes according to grid cell location
3: For each grid cell do
4: Build intra-grid routing table
5: Perform intra-grid data delivery (DD)
6: Calculate data volume D_DD using Eq. (9)
7: End for
8: For each aggregation node do
9: Select upper-layer node within Rs for inter-grid forwarding
10: Calculate transmission delay T_SU using Eq. (10)
11: Compute energy consumption E_SU using Eq. (11)
12: Evaluate efficiency η_SU using Eq. (12)
13: End for
14: Aggregate data at central node
15: Monitor node status and adjust routing adaptively
16: Repeat until data collection is complete

Fig. 8. VG-DDCOTA pseudo code.

area according to the optimized node configuration results, and determines the aggregation nodes for each grid. During the data collection phase, each node within the grid sequentially transmits data to the aggregation node to complete the intra grid DD. Next, the aggregation node selects the optimal upper layer node based on distance and signal strength, performs inter cell SU, and ensures efficient uplink transmission of data. The algorithm runs in a loop until all data transmission is completed, in order to minimize energy consumption and achieve efficient data collection.

3. Results

3.1. Performance evaluation of EASOA node configuration optimization algorithm

To verify the performance of the proposed algorithm, this paper conducted simulation experiments in the Windows 10 operating system environment. The experimental platform is configured with an Intel Core i7-10700 processor, 16 GB memory, and is implemented using Python 3.9 programming language. The wireless sensor network simulation platform is built based on a self-built model. The node perception model, energy consumption model, and data transmission mechanism can all be configured. The perception radius of the sensor node is set to 10 m, the communication radius is 15 m, and the deployment area is a 50 × 50 m two-dimensional plane. The node deployment data used in the study comes from the public experimental platform provided by FIT IoT-LAB [26]. The platform is led by INRIA in France and has a large-scale real node deployment environment. It supports node status collection, task configuration, and network simulation. It is widely used in wireless sensor networks and Internet of Things algorithm verification. The node location and network structure in a typical scenario are selected as the initial input to simulate WSN configuration environments of different scales and distributions.

Standard SSA, Grey Wolf Optimizer (GWO) and Whale Optimization Algorithm (WOA) were selected as comparison algorithms. These algorithms have a broad application base and are representative in wireless sensor network optimization. SSA is a swarm intelligence algorithm commonly used in node positioning and path planning problems in recent years. GWO has strong global search capabilities and is suitable for dealing with multi-objective optimization problems. WOA shows good stability in multi-modal function optimization and convergence speed. The experimental parameter settings of all algorithms are the same, the population size is 50, the maximum number of iterations is 500, and the other parameter settings refer to their original settings. The parameters in the EASOA algorithm are set as follows: the brightness drive perturbation coefficient is 0.5, the reverse elite selection rate is 0.2, and the warning update coefficient is 0.3. All experiments were repeated 10 times under the same initial conditions, and the average results were taken for performance comparison and evaluation.

First, the classic Schwefel's Problem 1.2, High Conditioned Elliptic, Ackley, and Griewank were used as benchmark functions to evaluate the convergence effect and global search performance of the algorithm. The Schwefel function has multiple local extreme points and is suitable for examining the ability of an algorithm to escape from local optimality. The Ackley function has steep edges and a flat center and is often used to evaluate the global search ability of an algorithm in a complex search space. The Griewank function contains a large number of periodic local extreme points and is suitable for testing the convergence accuracy and stability of the algorithm. These three types of functions together construct a variety of optimization scenarios, which can more comprehensively verify the adaptability of the algorithm in global search, local refinement, and complex function space.

Fig. 9(a)–(d) show the convergence performance of each algorithm on the benchmark function. In Fig. 9 (a), although the final fitness value of EASOA is not the lowest, it decreases rapidly in the early stage of iteration, and the convergence curve is relatively stable in the later stage, showing good early search ability and late convergence stability. This feature is due to the introduction of the reverse elite selection strategy, which expands the search space in the early stage, improves the diversity of solutions and the ability to jump out of the local optimum. However, due to the lack of adaptive perturbations, SSA had a slower convergence speed in the later stage. In Fig. 9 (b), EASOA had strong

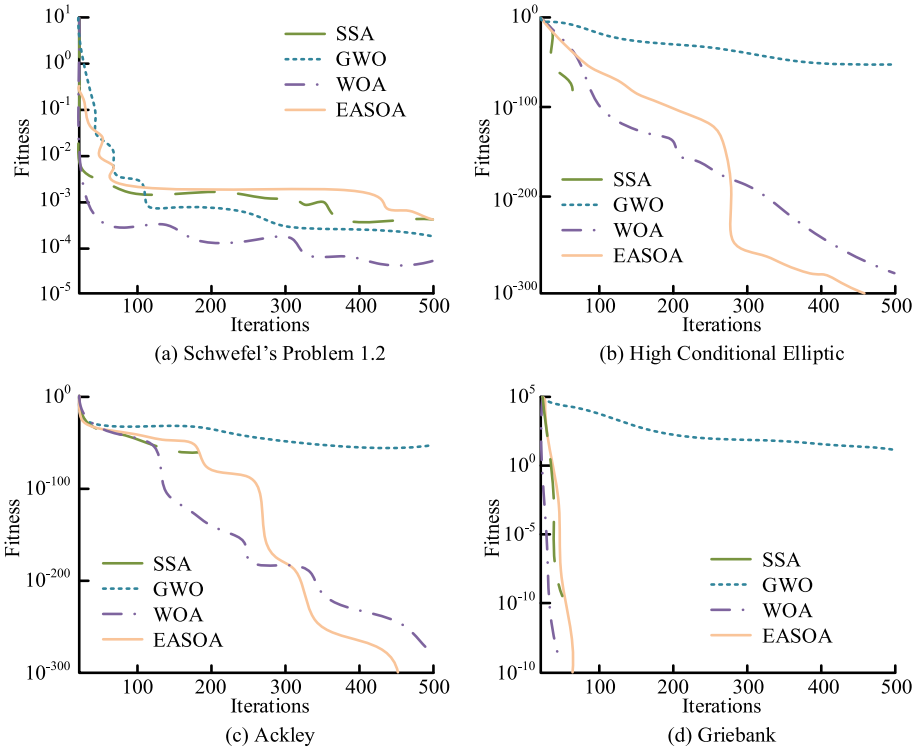


Fig. 9. Comparison of algorithm convergence curves.

anti-interference ability and quickly approached the optimal solution. In Fig. 9 (c), EASOA exhibited a smooth and stable convergence curve, attributed to its brightness driven perturbation strategy, which could effectively escape local optima in multi-modal function environments. In Fig. 9 (d), EASOA had a much faster convergence speed than other algorithms. By adaptively adjusting the exploration and development ratio, it effectively avoided getting stuck in local optima in complex search spaces. EASOA had demonstrated excellent global search and local development capabilities in different types of optimization problems.

Subsequently, a flat area of 50 × 50 m was selected as the experimental site, and 20 WS nodes were deployed with a node perception radius of 10 m. WS nodes were equipped with low-power wireless communication modules and battery power, arranged according to the optimized node positions of various algorithms, and manually calibrated to ensure that the actual sensing radius was consistent with the theoretical value. During the experiment, the optimization effect of the algorithm was verified by monitoring the node position and perception range. The WS node configuration results are shown in Fig. 10.

Fig. 10 (a)–(d) show the configuration results of each WS node. In Fig. 10 (a), the node distribution of SSA was relatively random, with uneven coverage and insufficient coverage in some areas. In contrast, the coverage effect of GWO in Fig. 10 (b) was better, with uniform node distribution and reduced overlap in coverage. However, there were still over-development issues in some areas, resulting in dense distribution of nodes. In Fig. 10 (c), the configuration effect of WOA was between SSA and GWO, with some deficiencies in the local development stage, resulting in uneven distribution of some nodes and slightly uneven coverage. The EASOA in Fig. 10 (d) performed the best, with evenly distributed nodes and complete coverage. Thanks to its introduced reverse elite selection, brightness driven perturbation, and dynamic warning update improvement strategies, the algorithm could adaptively adjust the search direction, effectively balancing global exploration and local development, and avoiding overlapping and insufficient coverage. The uniform distribution of nodes shown in Fig. 10 (d) can indeed be used as an idealized goal. However, under the combined effects of complex constraints, overlapping node perception ranges, and fault tolerance mechanisms, it is still practical to rely on heuristic algorithms for efficient and automated configuration. EASOA can achieve comprehensive optimization of maximizing coverage and minimizing redundancy under the trade-off of multiple objective functions, and is suitable for actual application scenarios such as irregular deployment areas and changes in node density.

Finally, different node failure scenarios are simulated to test the network robustness and coverage retention under different node failure rates. The network coverage retention rate indicates the percentage of the monitoring area that can still be covered by the remaining nodes after a certain percentage of nodes fail. The recovery time refers to the time required from the occurrence of a node failure to the restoration of stable network coverage. The node replacement rate indicates the percentage of nodes that are replaced or redeployed during the recovery process. The calculation time is the running time of the algorithm in a complete optimization process. The above indicators are obtained by averaging multiple experiments in a simulation environment. The coverage retention rate is calculated by the effective perception ratio of the monitoring points. The node replacement rate and recovery time are based on the response process of the algorithm to the failed nodes. Node failure adopts random failure mode, that is, the failed nodes are randomly selected from all nodes under a set ratio to simulate general non-centralized failure scenarios. The results are shown in Table 1.

As shown in Table 1, under different failure rates, EASOA significantly outperformed other comparative algorithms in terms of network coverage retention and fault recovery time. In the extreme case of 30 % node failure, EASOA maintained 83.5 % network coverage with a recovery time of 3.7 s, demonstrating excellent robustness and fast recovery capability. In contrast, SSA, GWO, and WOA had lower coverage retention rates and longer recovery time, and performed poorly under high failure rates. The reason is the dynamic warning update strategy and reverse elite selection strategy introduced by EASOA, which enables it to quickly adjust configuration after node failure, reduce node replacement rate, and improve network stability.

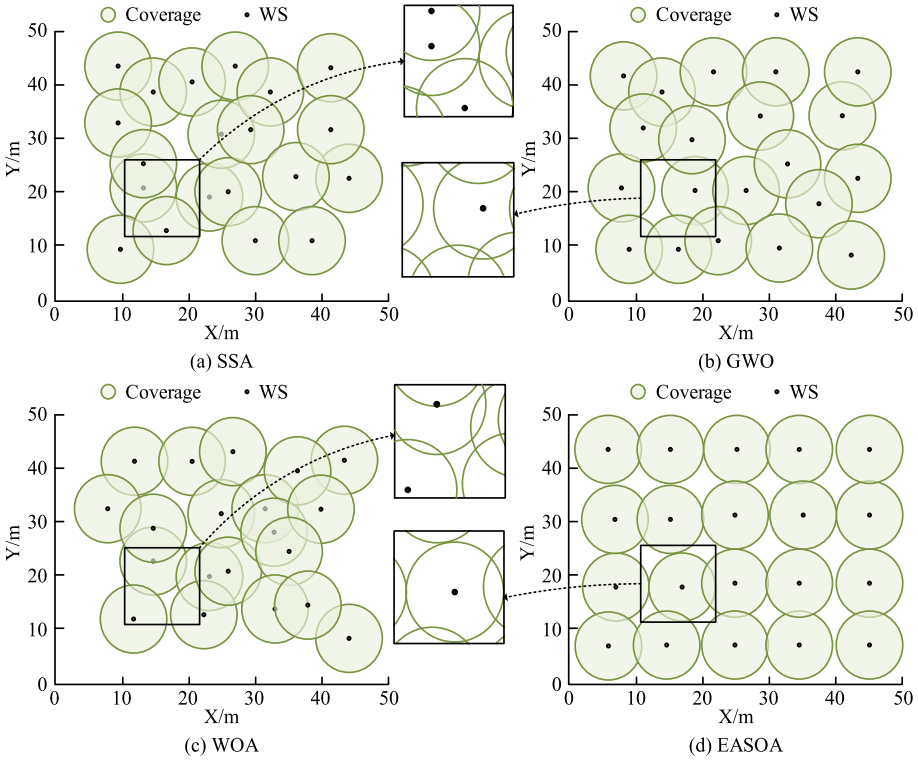


Fig. 10. WS node configuration test.

Table 1
Simulation test results under different node failure rates.

WS failure rate	Algorithm	Network coverage retention/ %	Recovery time/s	Node replacement rate/%	Computation time/s
10 %	SSA	82.3	3.8	15.3	2.5
	GWO	85.5	3.2	12.4	2.4
	WOA	84.1	3.5	13.5	2.3
	EASOA	91.2	2.7	9.8	2.2
20 %	SSA	76.6	4.2	20.3	2.8
	GWO	78.8	3.9	18.8	2.7
	WOA	77.5	4.1	19.2	2.6
	EASOA	87.5	3.2	12.5	2.4
30 %	SSA	69.5	5.1	25.0	3.2
	GWO	72.2	4.8	22.5	2.9
	WOA	70.3	5.0	24.7	2.8
	EASOA	83.5	3.7	15.8	2.5

3.2. Simulation testing of VG-DDCOTA distributed data acquisition and transmission model

In the same experimental environment, the Improved Power-Efficient GAthering in Sensor Information System (I-PEGASIS), Multi-path Cooperative Routing Protocol (MPC-ROUTING), and Minimum Spanning Tree-based Data Routing Protocol (MST-DRP) are selected as the comparison models. The OpenSLAM dataset, which contains approximately 40,000 pieces of sensor data for real-time positioning and map construction, test the algorithm's data collection performance in dynamic environments. The experiment uses sensor nodes of different scales to simulate a large-scale WSN environment. The data collection test results of each algorithm under different scales of WS nodes are shown in Fig. 11.

Fig. 11 (a) and (b) show the network coverage and average energy consumption of I-PEGASIS, MPC-ROUTING, MST-DRP, and VG-DDCOTA models under different WS nodes, respectively. In Fig. 11 (a), when the number of nodes was 600, the coverage rates of each algorithm were 98.4 %, 94.6 %, 92.5 %, and 91.3 %, respectively. In Fig. 11 (b), when the number of nodes was 600, the average energy consumption of each algorithm's nodes was 0.60 J, 0.55 J, 0.58 J, and 0.47 J, respectively. The VG-DDCTOA virtual grid structure and adaptive scheduling strategy can effectively reduce overlapping coverage and redundant communication between nodes. Compared with the algorithm, due to the frequent overlapping sensing areas and communication redundancy between nodes, the node energy consumption is high, which in turn accelerates the node failure process and shortens the effective operation time of the network. VG-DDCOTA's intelligent node configuration and transmission path optimization make the load of each node more balanced, avoiding the rapid consumption of energy caused by the frequent operation of local nodes. This load balancing strategy further reduces the energy consumption of nodes and effectively prolongs the survival time of nodes in the network, so that the network can still maintain high energy efficiency when more nodes exist, extending the life cycle of the entire network.

Subsequently, different packet sizes are selected for the experiment, covering various scenarios from small sensor data to large-scale multimedia data. The data transmission efficiency and communication experimental test results are shown in Fig. 12.

Fig. 12(a) and (b) show the data transmission rate and average communication delay results under different packet sizes, respectively. The data transmission rate is used to evaluate the transmission efficiency per unit time, while the average communication delay measures the time delay from data generation to successful reception. In Fig. 12 (a), when the packet size was 1000 KB, the data transmission rate of VG-DDCOTA reached 4.2 Mbps, while I-PEGASIS, MPC-ROUTING, and MST-DRP were 3.5 Mbps, 3.8 Mbps, and 3.4 Mbps, respectively. In Fig. 12 (b), the average communication delay of VG-DDCTOA was 42 ms, significantly lower than the comparison model. VG-DDCOTA had higher data transmission rates and lower latency in large packet transmission scenarios, and its efficient data scheduling strategy effectively reduced data redundancy and congestion.

Subsequently, the study attempts to simulate complex application environments with different network loads, and evaluates the comprehensive performance of the model using Load Balancing Coefficient (LBC), Path Redundancy (PR), and Congestion Control Efficiency (CCE) indicators. The results are shown in Table 2.

As shown in the table, VG-DDCOTA exhibited superior load allocation capability under all load conditions, especially in low load scenarios where LBC was 94.5 %. Even in high load scenarios, LBC remained at 90.2 %, reflecting good adaptability. In addition, the PR of VG-DDCTOA reached 78.2 %, 75.3 %, and 73.1 % under low, medium, and high load conditions, respectively, which was significantly better than other comparison algorithms. Finally, VG-DDCOTA maintained a high efficiency of 89.8 % under high load conditions, effectively managing data flows and reducing network congestion issues in high traffic environments. Overall, VG-DDCOTA has demonstrated significant advantages in load balancing, path redundancy, and congestion control, especially in high load environments, while traditional models have exposed their shortcomings in adaptability and traffic control.

Finally, in order to fully verify the large-scale adaptability of the proposed algorithm in complex environments, simulation experiments were conducted with a scale of 500–2000 nodes. A random node failure model with a non-fixed ratio was introduced in the experiment, and the nodes gradually failed during the simulation operation. At the same time, multiple irregular obstacle areas were set to limit the

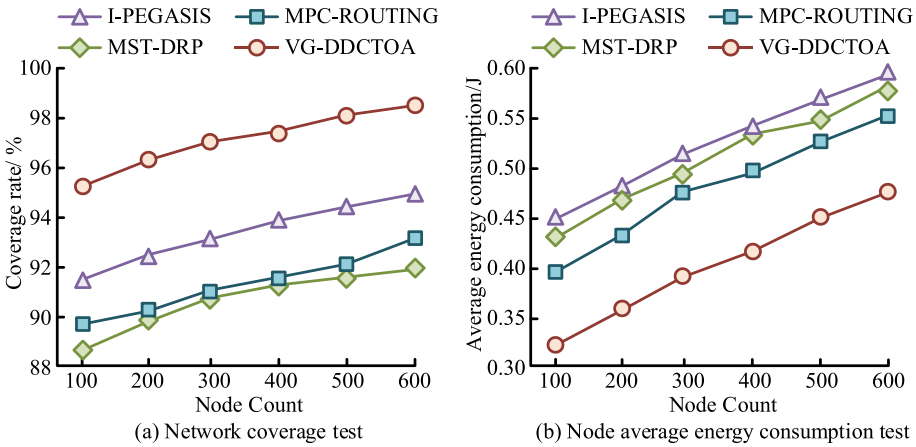


Fig. 11. Network coverage and node average energy consumption test.

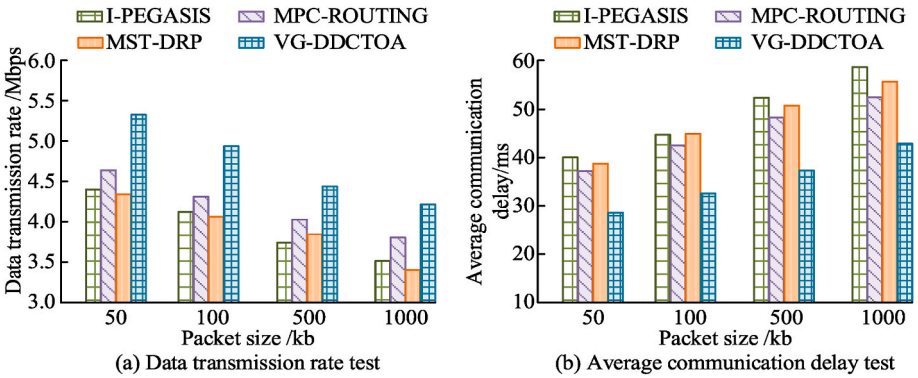


Fig. 12. Data transmission rate and communication delay test.

Table 2
Test results under different network loads.

Network load (%)	Metric	I-PEGASIS	MPC-ROUTING	MST-DRP	VG-DDCTOA
30 % (Low load)	LBC/%	87.3	89.6	86.1	94.5
	PR/%	63.5	70.4	61.8	78.2
	CCE/%	84.2	87.1	82.5	93.0
60 % (Medium load)	LBC/%	85.5	88.3	84.2	92.8
	PR/%	60.2	67.8	58.7	75.3
	CCE/%	82.5	85.7	80.3	91.5
90 % (High load)	LBC/%	82.7	85.8	80.9	90.2
	PR/%	58.3	65.2	55.3	73.1
	CCE/%	80.1	83.6	78.2	89.8

communication path, and Gaussian interference was added to simulate channel instability. To improve the reliability of the results, all indicators were averaged from 30 independent experiments and statistically significant. The results are shown in Table 3.

As can be seen from the table, as the node scale expands, VG-DDCOTA always maintains a high coverage rate, still reaching 84.6 % in the 2000-node scenario, which is significantly better than other algorithms. At the same time, its average energy consumption is controlled at 0.51 J, and the calculation time is only 5.8 s, with a relatively small increase. This shows that VG-DDCOTA still has good optimization efficiency and resource management capabilities when facing the increase in network scale and environmental complexity, showing strong engineering adaptability and deployability.

Table 3
Performance test results in large-scale node networks.

Node Count	Algorithm	Coverage Retention (%)	Avg. Energy Consumption (J)	Computation Time (s)
500	I-PEGASIS	84.1 ± 1.9	0.50 ± 0.02	3.1 ± 0.1
	MPC-ROUTING	85.5 ± 1.6	0.48 ± 0.01	2.9 ± 0.1
	MST-DRP	86.3 ± 1.5	0.46 ± 0.01	2.8 ± 0.1
	VG-DDCOTA	91.8 ± 1.1*	0.42 ± 0.01*	2.6 ± 0.1
1000	I-PEGASIS	77.3 ± 2.4	0.58 ± 0.03	4.8 ± 0.2
	MPC-ROUTING	79.6 ± 2.1	0.56 ± 0.02	4.3 ± 0.2
	MST-DRP	80.7 ± 2.0	0.54 ± 0.02	4.0 ± 0.2
	VG-DDCOTA	88.9 ± 1.5*	0.47 ± 0.01*	3.5 ± 0.1
2000	I-PEGASIS	70.2 ± 3.0	0.65 ± 0.03	8.2 ± 0.3
	MPC-ROUTING	72.5 ± 2.7	0.63 ± 0.02	7.4 ± 0.3
	MST-DRP	74.1 ± 2.4	0.60 ± 0.02	6.9 ± 0.2
	VG-DDCOTA	84.6 ± 2.0*	0.51 ± 0.02*	5.8 ± 0.2

Note: All experiments were run 30 times under dynamic node failure and environmental interference scenarios, and the results are mean ± standard deviation; "*" indicates that the result is significantly better than all the compared algorithms (t-test, $p < 0.05$).

4. Conclusion

Aiming at the optimization problems of node configuration and distributed data collection in WSN, a WS node configuration algorithm based on EASOA and a VG-DDCOTA for virtual grids were proposed. The performance test results showed that it exhibited excellent global search and local development capabilities in both single peak and multi-peak benchmark test functions. In the WS node configuration test, the WS nodes configured by EASOA had the most uniform distribution and complete coverage. In the extreme case of 30 % node failure, the network coverage rate was maintained at 83.5 %, with a recovery time of 3.7 s and a node replacement rate of 15.8 %. In the VG-DDCOTA data acquisition model test, when the number of nodes was 600, the network coverage and average energy consumption of the nodes were 91.3 % and 0.47 J, respectively. When the packet size was 1000 KB, the data transmission rate was 4.2 Mbps and the average communication delay was 42 ms. In 90 % of high load scenarios, its LBC was 90.2 %, PR was 73.1 %, and CCE was 89.8 %. The experimental results show that EASOA significantly improves network coverage, reduces energy consumption, and performs well in high failure rate scenarios.

Although the proposed method shows good performance in node configuration and data collection tasks in simulation environment, there are still some limitations in practical applications. First, the current model relies on centralized data aggregation and global network information, which is difficult to implement in large-scale or resource-constrained wireless sensor networks. Most sensor nodes are limited by energy consumption, computing power and communication bandwidth, and it is difficult to support the data transmission and processing overhead required for centralized optimization. Second, in actual adversarial or harsh environments, data transmission may face security risks such as interception, interference or node failure, affecting the stability and reliability of the network. The current model has not yet considered adaptive behavior under targeted attacks or dynamic threats. Future research can explore lightweight and distributed optimization mechanisms to enable nodes to achieve self-organization and self-optimization without the need for global information, and introduce artificial intelligence methods such as reinforcement learning to enhance the real-time decision-making and robustness of the system in complex scenarios.

CRediT authorship contribution statement

Youxian Zhang: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Zhen Nie:** Conceptualization, Data curation, Resources, Software, Supervision, Validation, Visualization, Writing – original draft, Writing – review & editing. **Hongxu Zhang:** Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Writing – original draft, Writing – review & editing, All authors have reviewed and agreed to publish.

Fundings

The research is supported by: The Scientific and Technological Project in Henan Province (Project No. 232102240100).

Declaration of competing interest

The author declares that there is no conflict of interest in this paper.

References

[1] P.P. Groumpos, A critical historic overview of artificial intelligence: issues, challenges, opportunities, and threats, Artif. Intell. Appl. 1 (4) (2023) 197–213.
[2] H. Han, Z. Jing, Anomaly detection in wireless sensor networks based on improved GM model, Tehnički vjesnik 30 (4) (2023) 1265–1273.
[3] V. Chandrasekar, A. Bashar, T.S. Kumar, B.A. Vani, R. Santhosh, Hybrid deep learning approach for improved network connectivity in wireless sensor networks, Wirel. Pers. Commun. 128 (4) (2023) 2473–2488.
[4] S.K. Rajendran, G. Nagarajan, Network lifetime enhancement of wireless sensor networks using EFRP protocol, Wirel. Pers. Commun. 123 (2) (2022) 1769–1787.
[5] T. Fang, Y. Yang, Distributed communication protocol in wireless sensor network based on Internet of Things technology, Wirel. Pers. Commun. 126 (3) (2022) 2361–2377.
[6] G. Kou, G. Wei, Improved sparrow search algorithm optimized DV-hop for wireless sensor network coverage, IEEE Access 11 (2) (2023) 62348–62357.
[7] D.W. Wajgi, J.V. Tembhurne, Localization in wireless sensor networks and wireless multimedia sensor networks using clustering techniques, Multimed. Tool. Appl. 83 (3) (2024) 6829–6879.
[8] K. Jaiswal, V. Anand, ESND-FA: an energy-efficient scheduled based node deployment approach using firefly algorithm for target coverage in wireless sensor networks, Int. J. Wireless Inf. Network 31 (2) (2024) 121–141.
[9] A. Boualem, C. De Runz, M. Ayaida, et al., A fuzzy/possibility approach for area coverage in wireless sensor networks, Soft Comput. 27 (14) (2023) 9367–9382.
[10] C. Xuan, Design of wireless sensor network data acquisition system via health sensor based on symmetric encryption algorithm, J. Test. Eval. 51 (1) (2023) 278–290.
[11] M. Srinivas, T. Amgoth, Data acquisition in large-scale wireless sensor networks using multiple mobile sinks: a hierarchical clustering approach, Wirel. Netw. 28 (2) (2022) 603–619.
[12] S. Xiaoxiang, G. Yan, L. Ning, R. Bing, A practical approach for missing wireless sensor networks data recovery, China Communications 21 (5) (2024) 202–217.
[13] P.V. Pravija Raj, A.M. Khedr, Z. Al Aghbari, EDGO: UAV-based effective data gathering scheme for wireless sensor networks with obstacles, Wirel. Netw. 28 (6) (2022) 2499–2518.
[14] M. Choudhary, N. Goyal, D. Gupta, B. Sharma, N. Sharma, An oceanographic data collection scheme using hybrid optimization for leakage detection during oil mining in mobility assisted UWSN, Multimed. Tool. Appl. 83 (7) (2024) 1–9.
[15] P. Kaur, A.M. Mishra, N. Goyal, S.K. Gupta, A. Shankar, W. Viriyasitavat, A novel hybrid CNN methodology for automated leaf disease detection and classification, Expert Syst. 41 (7) (2024) e13543.
[16] C. Gupta, V. Khullar, N. Goyal, K. Saini, R. Baniwal, S. Kumar, R. Rastogi, Cross-silo, privacy-preserving, and lightweight federated multimodal system for the identification of major depressive disorder using audio and electroencephalogram, Diagnostics 14 (1) (2023) 43.
[17] B. Fan, Y. Xin, A clustering and routing algorithm for fast changes of large-scale WSN in IoT, IEEE Internet Things J. 11 (3) (2023) 5036–5049.
[18] P. Gulganwa, S. Jain, EES-WCA: energy efficient and secure weighted clustering for WSN using machine learning approach, Int. J. Inf. Technol. 14 (1) (2022) 135–144.
[19] S. Das, S. Bhowmik, C. Giri, Cross-layer MAC protocol for semantic wireless sensor network, Wirel. Pers. Commun. 120 (4) (2021) 3135–3151.
[20] R. Maivizhi, P. Yogesh, Q-learning based routing for in-network aggregation in wireless sensor networks, Wirel. Netw. 27 (3) (2021) 2231–2250.
[21] G.P.N. Hakim, M. Hadi Habaebi, E.A.A. Elsheikh, F.M. Suliman, M.R. Lslam, S.H. B. Yusoff, Levenberg Marquardt artificial neural network model for self-organising networks implementation in wireless sensor network, IET Wirel. Sens. Syst. 14 (5) (2024) 195–208.
[22] J. Xiao, F. Gao, P. Li, X. Ji, Data acquisition mechanism of wireless sensor network pavement monitoring system based on hybrid compressive sensing, Wirel. Pers. Commun. 121 (3) (2021) 1707–1724.
[23] V.O. Romanov, I.B. Galelyuka, V.M. Hrusha, O.V. Voronenko, O.V. Kovyrova, H. V. Antonova, A.V. Kedych, Wireless sensor networks for digital agriculture, environmental protection, and healthcare, Cybern. Syst. Anal. 59 (6) (2023) 1023–1030.
[24] X. Ding, W. Feng, An anomaly detection method based on feature mining for wireless sensor networks, Int. J. Sens. Netw. 36 (3) (2021) 167–173.
[25] S. Boyineni, K. Kavitha, M. Sreenivasulu, Cat swarm optimisation-based mobile sinks scheduling in large-scale wireless sensor networks, Int. J. Commun. Network. Distr. Syst. 29 (1) (2023) 47–70.
[26] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pissard-Gibollet, F. Saint-Marcel, G. Schreiner, J. Vandaele, T. Watteyne, FIT IoT-LAB: A Large Scale Open Experimental IoT Testbed, IEEE World Forum on Internet of Things (WF-IoT), 2015, pp. 459–464, 2015.