International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 1739-1747

https://doi.org/10.70517/ijhsa463136

Energy optimization in smart sensor networks: application of particle swarm optimization algorithms to the deployment of electronic information sensing nodes

Liang Wang[1,*]
[1] School of Electrical Engineering, Hunan Mechanical and Electrical Vocational and Technical College, Changsha, Hunan, 410151, China
Corresponding authors: (e-mail: wangliang123144@163.com).

Abstract Smart sensor networks have a wide range of applications in the fields of environmental monitoring and industrial automation, but their energy efficiency and node coverage optimization problems need to be solved. This paper proposes a dual-strategy improved particle swarm optimization algorithm (CMKPSO) for solving the deployment optimization problem of electronic sensing nodes. Combining the Boolean sensing model and probabilistic sensing model, the coverage probability calculation framework is constructed. The cross-variance and adaptive parameter improvement strategies are designed to enhance the global search capability of the algorithm and accelerate the convergence. Simulation experiments show that the evaluation function value of CMKPSO is finally stabilized at 0.94, which is 13%, 9%, and 4% higher than that of PSO, VF, and EABC. The CMKPSO algorithm reduces the average queue captains to less than 15, so that most of the queue captains are concentrated in less than 10, and significantly reduces the packet loss rate of the network and the node load pressure.

Index Terms sensor network, coverage optimization, node deployment, cross-variance, CMKPSO algorithm

I. Introduction

With the continuous progress and application of science and technology, smart sensor network has become an indispensable part of modern society [1], [2]. It can help us understand and monitor various complex environments and devices more accurately, thus realizing more efficient and intelligent management and control [3], [4]. However, one of the significant problems of smart sensor networks is that it requires a large amount of energy to support its normal operation and work [5], [6]. Therefore, how to perform energy management and optimization in smart sensor networks has become an important problem to be solved in modern technology [7].

For sensor networks, energy management is an extremely important issue [8], [9]. Because the basic characteristics of sensor nodes are small size, limited energy, and need to work for a long time [10]. Therefore, it is crucial to solve the energy problem and effectively utilize the energy of each sensor node in a sensor network [11], [12]. If the energy in the sensor network can be managed reasonably and effectively, it can not only improve the operational efficiency of the system, but also increase the lifetime of the sensor network, which in turn extends its service life and reduces energy consumption, as well as reduce the management cost and maintenance difficulty [13]-[16]. And energy optimization is an important issue in the design of smart sensor networks [17]. Sensor nodes are numerous and widely distributed, so the management and utilization of energy must be efficient and durable [18]. During the deployment and operation of sensor nodes, there are problems of uneven energy consumption and energy wastage, so energy optimization needs to be studied to improve the energy utilization efficiency of smart grid sensor networks [19]-[22]. As an optimization algorithm designed to simulate the behavior of bird flock foraging, particle swarm optimization algorithm has gradually attracted attention in this field, which is able to achieve energy optimization by simulating the search and migration strategy of particles in the solution space based on group collaboration and information sharing during bird flock foraging [23]-[26].

In this paper, the construction method of sensing node coverage model is firstly described and the sensing node deployment problem is proposed. Based on the Boolean sensing and probabilistic sensing model, the joint optimization objective of area weight coverage and area coverage is designed. Improve the cross-variation mechanism of genetic algorithm and optimize the control parameters relying on the adaptive dynamic adjustment of inertia weights and learning factors. Construct node deployment optimization model based on CMKPSO algorithm with coverage rate as the objective function. Design simulation experiments to verify the performance level of CMKPSO algorithm and the advantages in network load balancing.

II. Improved particle swarm based optimization algorithm for sensing node coverage

The main objectives of the IoT sensing layer are environment sensing and information transfer. Among them, the sensing nodes have communication and data processing capabilities, so they can accomplish monitoring tasks in unsupervised environments. The coverage of sensing nodes is an important indicator of the service quality of IoT sensing layer, which directly determines the information acquisition and data transmission ability of the network, and the optimization of node coverage is important for improving the reliability of the network. At present, the coverage optimization strategy is mainly divided into two categories, one is to plan the scheduling of nodes after random deployment, so as to ensure the overall coverage of the network. The other category is to optimize the deployment location of sensing nodes to maximize the network coverage. In practical environments, the random deployment strategy usually deploys a large number of nodes to avoid coverage blind zones, and at the same time will generate coverage redundancy resulting in network resource waste. Therefore, in environments where the coverage area is small or the geographic information is known, rationalizing the deployment location of nodes can save the deployment cost and reduce the waste of resources. For this reason, this paper proposes a fault-tolerant optimization strategy of node coverage for the case of deterministic deployment of sensing nodes.

II. A. System Model and Problem Description

II. A. 1) Coverage model

Assume that the sensing nodes are distributed in the region $L \times L$ with a sensing radius of R, and the set of nodes is defined as $N = \{N_1, N_2, \cdots N_n\}$, and accordingly, the coordinate information of each node is (x_n, y_n). The perception range of a node is a circular range centered on the coordinates of the node with R as the radius. The perception probability of a target within this range is 1 and vice versa is 0. Thus, the perception probability of a node N_n for any point $q(a,b)$ within the region of interest is defined as shown in the following equation:

$$p = \begin{cases} 1, & d_{N_n q} < R \\ 0, & otherwise \end{cases} \tag{1}$$

where $d_{N_n q}$ is the Euclidean distance from the sensing node N_n to the target point q.

$$d_{N_n q} = \sqrt{(x_n - a)^2 + (y_n - b)^2} \tag{2}$$

II. A. 2) Description of the problem

In order to solve the problem of uneven distribution of sensing nodes in deterministic deployment, which leads to excessive redundant coverage area, it is necessary to calculate the deployment position of sensing nodes so as to optimize the coverage of sensing nodes. The deployed sensing nodes no longer move and each sensing node has the same sensing range, so the total coverage of the monitoring area remains unchanged. Due to the existence of duplicate coverage areas among the sensing nodes, the total coverage rate of the sensing nodes is difficult to be calculated by the formula. To facilitate the calculation of the total coverage rate, the square monitoring area is divided into $L \times L$ grids of equal size and area 1. The node coverage rate is the ratio of the number of grids covered by nodes to the area of the square area, as shown in equation (3):

$$cov = \frac{\sum_{a=1}^{L}\sum_{b=1}^{L} p}{L * L} \tag{3}$$

where cov is the total coverage of the sensing node, (a,b) is an arbitrary point in the monitoring area, and p is the probability that the point is covered by the sensing node.

The coverage calculation process is described as follows: the probability of sensing a grid by a sensing node is calculated using Eq. (1) and Eq. (2); then, the total coverage of the monitoring area by a sensing node is calculated using Eq. (3). On this basis, the total coverage cov of the sensing nodes is used as the objective function of the improved particle swarm algorithm proposed in this paper to solve the coverage optimization problem. The location of the sensing nodes is optimized using the proposed algorithm in this paper, so as to maximize the coverage of the nodes on the monitoring area.

II. B.Node Deployment Techniques

II. B. 1) Node-aware modeling

To facilitate the study of the node deployment problem, a node perception model is first established to determine how the nodes are perceived. There are two types of perception models currently used: the Boolean perception model and the probabilistic perception model.

(1) Boolean perception model

Boolean models are also called 0-1 models. Denoted by the coordinates $s(X_s, Y_s)$ is the position of a node in a two-dimensional plane. It is assumed that each node in the model has the same circular sensing range, centered on the node and with sensing radius R. The sensor node is able to collect data information within the sensing range, while the region outside the sensing range is set as the sensing blind zone of the node, which cannot be covered by the node.

Under this model, the probability that an event located at coordinates $p(X_p, Y_p)$ in the heliometric region is sensed by the node $s(X_s, Y_s)$ is as follows:

$$P(s,p)=\begin{cases}1, d(s,p)\le R_s\\ 0, otherwise\end{cases} \tag{4}$$

(2) Probabilistic perception model

The Boolean perception model has only two results for the perception probability of each node in the target area, one is perceived within the perception range of a node, and one is failed to perceive outside the perception range of a node. In the actual process, the sensor nodes will be affected by the surrounding environment as well as the problems such as the weakening of the transmission signal during the transmission and reception of the signal, which leads to the inaccuracy of the data. Therefore, a perception probability model is proposed after considering the actual situation. In the perception probability model, the probability that an event located at coordinates $p(X_p, Y_p)$ in the target area is perceived by the node $s(X_s, Y_s)$ is as follows:

$$P(s,p)=\lambda / d(s,p)^{\alpha} \tag{5}$$

where the parameter λ denotes a constant, the parameter α denotes the sensing ability of the sensor node s, i.e., the strength of the signal reception attenuation when the distance increases, and $d(s,p)$ denotes the distance from the node s to the location of the event p. From the formula, the probability of perception is 1 when the distance is 0, and the probability of being able to perceive the event p decreases as the distance increases.

II. B. 2) Node Deployment Classification

Categorized by the way nodes are deployed in their locations, node deployment can be divided into two main categories: deterministic node deployment and random node deployment.

(1) Deterministic node deployment

Deterministic deployment is manual deployment, i.e., all sensor nodes have been planned for their location before deployment. It is suitable for environments where the area is small, the application environment is more stable and easy to deploy manually, such as hospitals, homes and other environments. With the goal of full coverage of the event area, the network in the region is optimized to determine the location of each node to be installed in the region.

(2) Random node deployment

For application scenarios affected by factors such as large regions and harsh environments, such as deserts, forests, and complex discrete production process workshops, it is impractical to use deterministic deployment. Random deployment, on the other hand, can solve the above unfavorable factors. Since the initial deployment location of the nodes in the random deployment scheme is not restricted, it is possible to use aircraft broadcasting and other ways to deliver the sensor node's to the designated area. Random coverage can be divided into two categories: random node coverage and dynamic coverage. Random node coverage refers to the delivery of sensor nodes to the event monitoring area in a randomized way, which is characterized by the irregular distribution of nodes. The disadvantage is that a large number of sensor nodes need to be deployed under the premise of ensuring full coverage of the event area, resulting in a large number of node redundancies. Dynamic coverage is to utilize sensor nodes with certain mobility to automatically form a network through relevant node deployment algorithms to achieve full coverage of the event area.

II. C.Particle swarm optimization algorithm with dual strategy improvement

II. C. 1) Cross-variant improvement strategies

In this section, the cross-mutation mechanism of the genetic optimization algorithm is introduced into the particle swarm optimization algorithm in three steps, and the operation flow of particle cross-mutation is shown in Figure 1.

Assume that the population size of the algorithm is N, the fitness function is $fit(x)$, and the average fitness value of all particles in the current population is $fav(t)$ at the t th iteration. Then for the particle $x(t,i)$ whose fitness value is larger than the current average fitness value, the first step is to randomly select a particle $x(t,k)$ from the population, and the second step is to produce a new individual $x(t,ii)$ by hybridizing $x(t,k)$ and $x(t,i)$. Hybridization variation operation is shown in Eqn. (6), where rand (0, 1) is a random number of 0-1. N (0, 1) obeys normal distribution, and e is Euler's constant. The third step is to judge according to the fitness value, if the new individual $x(t,ii)$ is better than $x(t,i)$, the new individual $x(t,ii)$ is retained and used to replace $x(t,i)$, and vice versa, it is not retained as shown in Equation (7).

$$\begin{cases} x(t,ii) = \dfrac{R1^{*} x(t,i) + R2^{*} x(t,k)}{R1+R2} \quad if\,(fit(x(t,i)) > fav(t)) \\ where \quad R1 = \dfrac{rand(0,1)}{2}; \quad R2 = e^{N(0,1)} \end{cases} \tag{6}$$

$$x(t,i) = \begin{cases} x(t,ii) & if\,(fit(x(t,ii)) < fit(x(t,i))) \\ x(t,i) & others \end{cases} \tag{7}$$

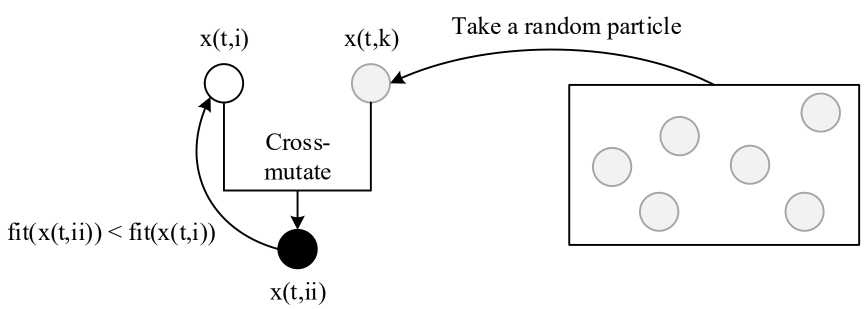

Figure 1: Process of particle cross variation

II. C. 2) Adaptive Inertia Weights and Learning Factors

Particle swarm optimization algorithm in the iterative process, all the particles through the individual optimal and the current global optimal tracking, the position and speed of the particles will be dynamically adjusted. PSO algorithm through the fitness value of the objective function to evaluate the advantages and disadvantages of the particles as well as the degree of evolution. In this paper, we take solving the minimum value of the objective function as an example (solving the maximum value can be taken as the inverse of the same reason), assuming that the current number of iterations is t times, $f_v(t,i)$ is the current fitness value of the ith particle, $f_{\min}(t)$ is the current minimum fitness value of all the particles, $f_{\max}(t)$ is the current maximum of all the particles fitness value. We define the concept of particle individual evolution factor $K(t,i)$, then the mathematical expression of the evolution factor $K(t,i)$ of particle i at the tth iteration is shown in Equation (8):

$$K(t,i) = \frac{f_v(t,i) - f_{\min}(t)}{f_{\max}(t) - f_{\min}(t)} \tag{8}$$

Obviously $K(t,i)$ takes values in the range of [0, 1], and when $f_v(t,i) = f_{\min}(t)$, $K(t,i)$ is 0, which means that the current particle i is the strongest evolutionary degree at the tth iteration. When $f_v(t,i) = f_{\max}(t)$, $K(t,i)$ is 1, i.e., it means that the current particle i is the weakest in evolution at the tth iteration.

Assuming that the maximum number of iterations is T, it is not difficult to find that the evolutionary factor of the particle is to judge the evolutionary degree of each particle at the tth iteration through Eq. (8), and different inertia weight improvement strategies and learning factor improvement strategies can be formulated according to the evolutionary factor of the particle. In this paper, the evolution factor is introduced into the inertia weights and learning factor, while the linear variation is changed to sinusoidal transformation in terms of inertia weights. The mathematical expression of the inertia weight model of particle i at the tth iteration is shown in Equation (9):

$$w(t,i) = K(t,i) * w_{end} + (w_{start} - w_{end}) * \sin\left[\frac{\pi}{2} * (1 - \frac{t}{T})\right] \tag{9}$$

where T is the maximum number of iterations of the algorithm, w_{start} is the initial inertia weight with a value of 0.9, w_{end} is the inertia weight at the end with a value of 0.4, and $K(t,i)$ is the evolutionary factor for the tth iteration of particle i.

In terms of learning factor, this section does the improvement strategy as shown in Eq. (10):

$$\begin{cases} c_1 = 2-0+0.5*\dfrac{1-K(t,i)}{1-\dfrac{t}{T}} \\ c_2 = 2-0-0.5*\dfrac{1-K(t,i)}{1-\dfrac{t}{T}} \end{cases} \tag{10}$$

Then its velocity and position update equations are shown in equation (11) and equation (12).

$$V_i(t+1) = w(t,i)^* V_i(t) + c_1 r_1 (Pbest_i(t) - X_i(t)) + c_2 r_2 (Gbest(t) - X_i(t)) \tag{11}$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{12}$$

In summary, this paper proposes the CMKPSO algorithm.

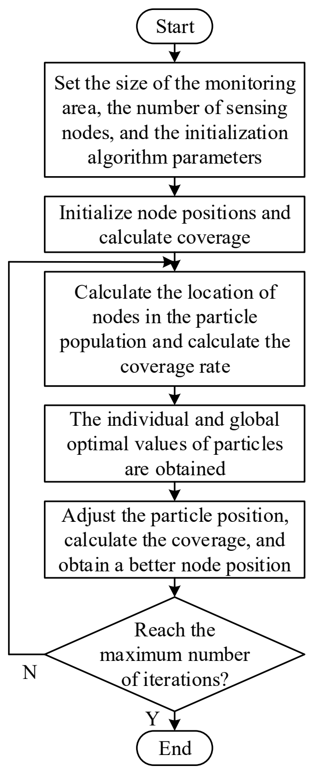

Figure 2: Flowchart of the algorithm

II. C. 3) Description of the Coverage Optimization Algorithm

The particle swarm algorithm is prone to premature maturation of the population during the search process, which leads to uneven distribution of nodes and high duplicate coverage when used to solve the node coverage problem. For the previous paper, an improved particle swarm algorithm CMKPSO is proposed, which improves the search strategy of the particles. Based on this, a node coverage optimization algorithm is proposed in this section, and its flow is shown in Fig. 2.

The algorithm flow description is specified as follows:

Step 1: Set the size of the monitoring area and the number of sensing nodes to initialize the algorithm parameters;

Step 2: Correspond the coordinates of nodes to particles and initialize the position coordinates, and calculate the coverage of the initialized population;

Step 3: The algorithm enters into iteration, calculates the position of the node after each iteration, and calculates the current coverage rate;

Step 4: obtaining the individual optimal value and the global optimal value based on the current coverage rate and the particle position;

Step 5: Adjusting the particle positions using an improved search strategy and calculating the coverage rate to arrive at a more optimal node position;

Step 6: determine whether the number of iterations is satisfied, if not, continue to execute step 3, otherwise output the optimal node coordinates as well as the adaptation value.

III. Experimental analysis of CMKPSO-based simulation for coverage optimization of sensing nodes

III. A. Performance validation

In order to further analyze the performance of the proposed algorithm, this paper simulates and compares the standard particle swarm optimization (PSO) algorithm, the virtual force (VF) algorithm, the extrapolated artificial bee colony (EABC) algorithm, and the CMKPSO algorithm proposed in this paper, and the results of the performance comparison are shown in Figure 3. The results in Fig. 3 show that the evaluation function value of this paper's algorithm is finally stabilized at 0.94 after 60 iterations. in terms of coverage, this paper's algorithm improves 13% compared to PSO algorithm, 9% compared to VF algorithm, and 4% compared to EABC algorithm. Meanwhile, from the point of view of the convergence speed of the algorithm, the convergence speed of this paper's algorithm is the fastest among all the algorithms. PSO algorithm relies only on the self-cognition and social cognition of the particles to guide the evolution of the particles, which leads to the slow iteration of the algorithm and is difficult to obtain the global optimal results; VF algorithm iterates quickly but the coverage effect is not ideal; and the EABC algorithm is a typical improvement to the swarm intelligence algorithm, which speeds up iteration and improves the optimality finding effect by extrapolating equations, but not sufficiently mining The EABC algorithm is a typical improvement of the swarm intelligence algorithm, which speeds up the iteration and improves the optimization effect by extrapolating the equations, but it does not fully explore the relevant characteristics in the model, which leads to the effect is not as good as that of this paper.

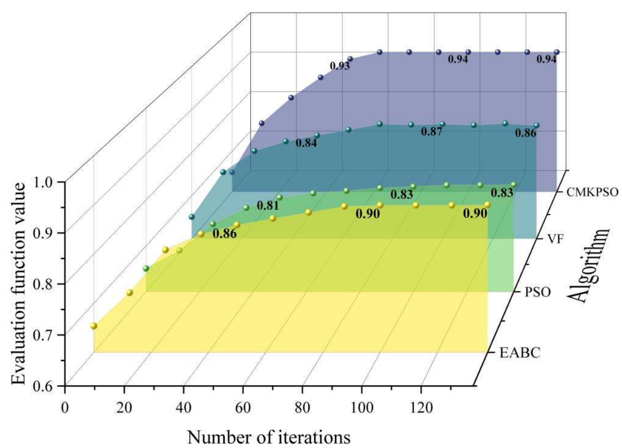

Figure 3: Performance comparison of four algorithms

The coverage of each algorithm after running with different number of sensing nodes under a 100m×50m area is shown in Table 1 and Figure 4. Where, f1 represents the area weight coverage and f2 represents the area coverage. Obviously, as the number of nodes increases, the coverage of the area also increases. Combining Table 1 and Figure 4, it can be seen that the CMKPSO algorithm in this paper outperforms all other three algorithms in terms of area coverage, especially in terms of focus area coverage, this paper is able to maintain a higher focus area coverage by adaptively and dynamically adjusting the inertia weights and the learning factor, and when the number of nodes is smaller, this advantage is more obvious. For example, when N=15, the regional weight coverage of CMKPSO algorithm is about 35% higher than that of EABC algorithm, VF algorithm, and PSO algorithm on average, and the other 3 algorithms show a certain degree of randomness in their coverage of the focus region due to the lack of differentiation of regional importance.

Table 1: Comparison of coverage under different number of nodes (%)

Number of nodes	CMKPSO		EABC		PSO		VF	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
15	82.5	68.3	59.3	58.2	62.9	52.5	59.5	58.4
30	89.3	76.2	62.5	67.9	68.4	60.4	66.3	62.5
45	90.7	79.6	65.5	69.2	70.1	63.5	68.4	65.8
60	92.9	82.5	73.7	74.1	74.8	67.7	72.8	71.2
75	94.8	91.4	84.6	81.7	80.2	75.2	80.4	77.9
90	97.2	93.3	91.3	88.5	87.6	82.5	89.7	84.4

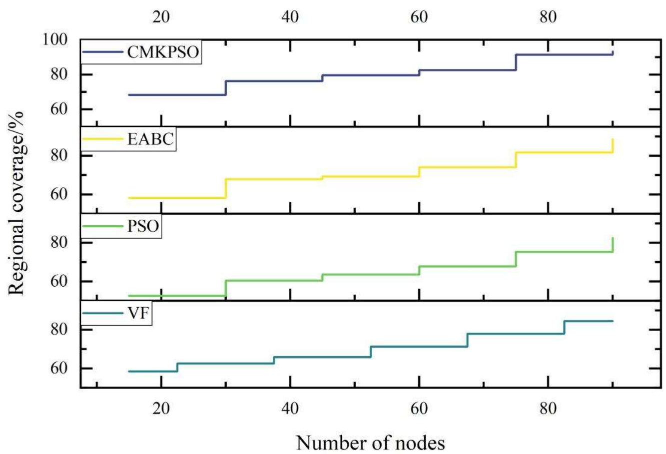

Figure 4: Comparison of the coverage degree of each algorithm

III. B. Analysis of simulation results

When the simulation system enters the steady state, the packet loss rate also tends to be constant. The CMKPSO algorithm is used to fine-tune the number of service desks of the nodes with higher burdens, and the number of queues at steady state before and after adjustment is shown in Fig. 5. After adjustment, the average captains of all nodes are within 15. While before adjustment, the maximum average captain of the nodes is close to the cache maximum value of 20. Therefore, after adjustment, the nodes have greater compatibility with network congestion.

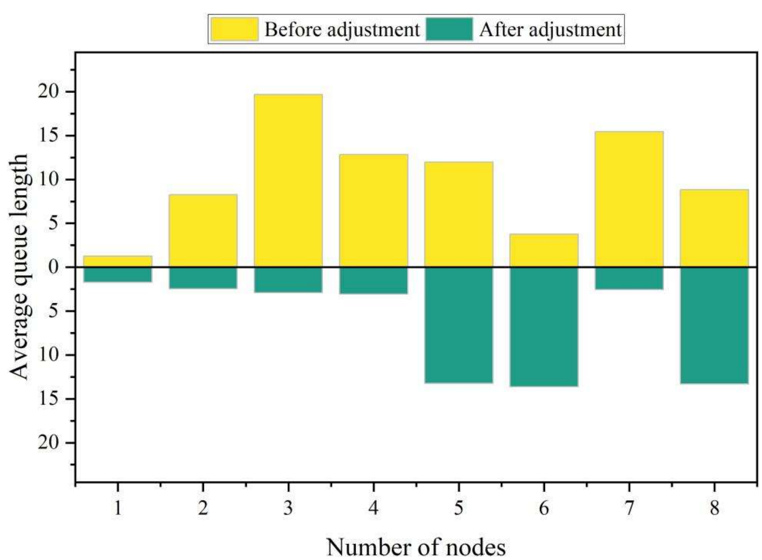

Figure 5: Comparison of average cohort lengths

Before and after the adjustment, the steady state probability of the nodes also changed greatly, and the comparison of the steady state probability is shown in Figure 6. Before the adjustment, the captain of node 1 has been in an increasing state, and the steady state probability shows an increasing trend. This indicates that the node data queuing continues to rise, and when it exceeds the maximum value of the node packet cache, severe packet loss occurs and blocks the network. When the system is adjusted to steady state, the steady state probability shows a decreasing trend as the captain increases. When the captain exceeds 15, its steady state probability tends to 0, which indicates that when the system reaches steady state, this node will not cause cache underrun. Therefore, blocking and packet loss will not occur at this node.

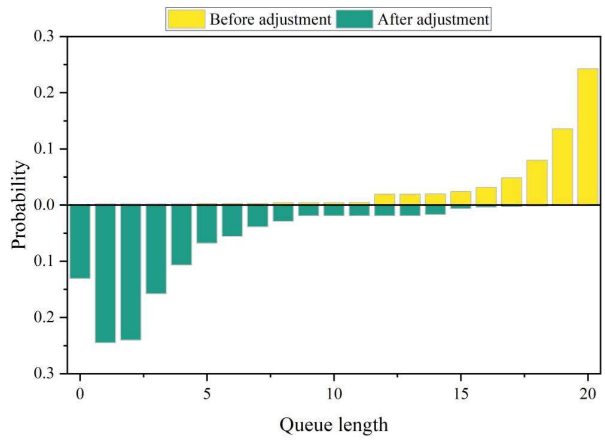

Figure 6: Comparison of steady-state probabilities at node 1

After the adjustment, the node queuing situation also changes greatly, and its before and after comparison results are shown in Fig. 7. From the queuing situation of node 1 before the adjustment using CMKPSO algorithm, it can be seen that the packet captain is almost close to the maximum value of the node cache length. Therefore, system blocking due to insufficient node cache may occur. From the queuing situation after the CMKPSO algorithm adjustment, it can be seen that most of the queuing captains of the nodes are concentrated in less than 10, and basically there is no blocking caused by insufficient packet queuing cache.

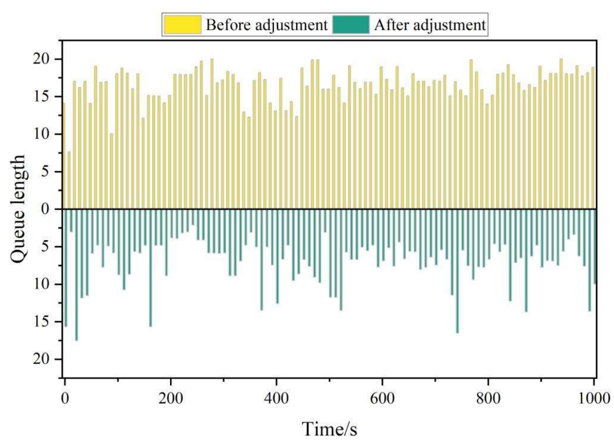

Figure 7: Comparison of queuing situation at node 1

IV. Conclusion

In this paper, for the problem of energy optimization and node coverage in smart sensor networks, a deployment optimization method based on the dual-strategy improved particle swarm algorithm (CMKPSO) is proposed, and the research conclusions are as follows:

(1) The performance is significantly improved: the CMKPSO evaluation function value is finally stabilized at 0.94, which is 13% higher than the PSO algorithm, 9% higher than the VF algorithm, and 4% higher than the EABC algorithm. Meanwhile, the convergence speed of the algorithm is the best among all algorithms. When there are fewer nodes (e.g., N=15), the coverage of key areas is improved by 35% on average compared with the comparison algorithm, which verifies the adaptability of the algorithm to sparse deployment.

(2) Network load optimization: Based on CMKPSO algorithm, the number of node service stations is adjusted to reduce the average captain to less than 15, and most of the queuing captains are concentrated in less than 10, and the steady-state probability tends to be close to zero, which reduces the risk of packet loss and improves the network stability.

References

[1] Chang, F. C., & Huang, H. C. (2016). A Survey on Intelligent Sensor Network and Its Applications. J. Netw. Intell., 1(1), 1-15.

[2] Verma, S., Zeadally, S., Kaur, S., & Sharma, A. K. (2021). Intelligent and secure clustering in wireless sensor network (WSN)-based intelligent transportation systems. IEEE Transactions on Intelligent Transportation Systems, 23(8), 13473-13481.

[3] Shanmugham, S. R., & Paramasivam, S. (2018). Survey on power analysis attacks and its impact on intelligent sensor networks. IET Wireless Sensor Systems, 8(6), 295-304.

[4] Sofi, A., Regita, J. J., Rane, B., & Lau, H. H. (2022). Structural health monitoring using wireless smart sensor network–An overview. Mechanical Systems and Signal Processing, 163, 108113.

[5] Khalifeh, A., Darabkh, K. A., Khasawneh, A. M., Alqaisieh, I., Salameh, M., AlAbdala, A., ... & Rajendiran, K. (2021). Wireless sensor networks for smart cities: Network design, implementation and performance evaluation. Electronics, 10(2), 218.

[6] Su, B., & Wang, S. (2020). An agent-based distributed real-time optimal control strategy for building HVAC systems for applications in the context of future IoT-based smart sensor networks. Applied Energy, 274, 115322.

[7] Fadel, E., Gungor, V. C., Nassef, L., Akkari, N., Malik, M. A., Almasri, S., & Akyildiz, I. F. (2015). A survey on wireless sensor networks for smart grid. Computer Communications, 71, 22-33.

[8] Babayo, A. A., Anisi, M. H., & Ali, I. (2017). A review on energy management schemes in energy harvesting wireless sensor networks. Renewable and Sustainable Energy Reviews, 76, 1176-1184.

[9] Singh, J., Kaur, R., & Singh, D. (2020). A survey and taxonomy on energy management schemes in wireless sensor networks. Journal of Systems Architecture, 111, 101782.

[10] Anjum, S. S., Noor, R. M., Anisi, M. H., Ahmedy, I. B., Othman, F., Alam, M., & Khan, M. K. (2017). Energy management in RFID-sensor networks: Taxonomy and challenges. IEEE Internet of Things Journal, 6(1), 250-266.

[11] Thomas, D., Shankaran, R., Sheng, Q. Z., Orgun, M. A., Hitchens, M., Masud, M., ... & Piran, M. J. (2020). QoS-aware energy management and node scheduling schemes for sensor network-based surveillance applications. IEEE Access, 9, 3065-3096.

[12] Naji, N., Abid, M. R., Benhaddou, D., & Krami, N. (2020). Context-aware wireless sensor networks for smart building energy management system. Information, 11(11), 530.

[13] Li, J., Lv, J., Zhao, P., Sun, Y., Yuan, H., & Xu, H. (2023). Research and application of energy-efficient management approach for wireless sensor networks. sensors, 23(3), 1567.

[14] Mohapatra, H., Debnath, S., & Rath, A. K. (2019). Energy management in wireless sensor network through EB-LEACH. International journal of research and analytical reviews (IJRAR), 56-61.

[15] Bouabdallah, F., Zidi, C., & Boutaba, R. (2017). Joint routing and energy management in underwater acoustic sensor networks. IEEE Transactions on Network and Service Management, 14(2), 456-471.

[16] Dehwah, A. H., Shamma, J. S., & Claudel, C. G. (2017). A distributed routing scheme for energy management in solar powered sensor networks. Ad Hoc Networks, 67, 11-23.

[17] Ekpenyong, M. E., Asuquo, D. E., & Umoren, I. J. (2019). Evolutionary optimisation of energy-efficient communication in wireless sensor networks. International Journal of Wireless Information Networks, 26(4), 344-366.

[18] Singh, S. P., & Sharma, S. C. (2018). An improved cluster-based routing algorithm for energy optimisation in wireless sensor networks. International Journal of Wireless and Mobile Computing, 14(1), 82-89.

[19] Verma, N., Nimesh, M., & Rani, R. (2015). Energy-efficient sensor optimisation in wireless sensor networks. International Journal of Wireless and Mobile Computing, 9(4), 355-362.

[20] Hua, M., Wang, Y., Zhang, Z., Li, C., Huang, Y., & Yang, L. (2019). Energy-efficient optimisation for UAV-aided wireless sensor networks. IET Communications, 13(8), 972-980.

[21] Khedkar, M., & Asutkar, G. M. (2018, November). Energy optimisation in wireless sensor network for video data transmission. In 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN) (pp. 20-24). IEEE.

[22] Diety, G. L., Ali, K. E., Asseu, O., Zehero, B. B., & Hamouda, S. (2017). Energy Optimisation in Wireless Sensor Network. Engineering, 9(10), 880-889.

[23] Khabiri, M., & Ghaffari, A. (2018). Energy-aware clustering-based routing in wireless sensor networks using cuckoo optimization algorithm. Wireless Personal Communications, 98, 2473-2495.

[24] Razooqi, Y., Al-Asfoor, M., & Abed, M. H. (2024). Optimise Energy Consumption of Wireless Sensor Networks by using modified Ant Colony Optimization. Acta Technica Jaurinensis, 17(3), 111-117

[25] Nayak, P., & Reddy, C. P. (2020). Bio-inspired routing protocol for wireless sensor network to minimise the energy consumption. IET Wireless Sensor Systems, 10(5), 229-235.

[26] Omeke, K. G., Mollel, M. S., Zhang, L., Abbasi, Q. H., & Imran, M. A. (2020, August). Energy optimisation through path selection for underwater wireless sensor networks. In 2020 International Conference on UK-China Emerging Technologies (UCET) (pp. 1-4). IEEE.