IECE

IECE Transactions on Internet of Things
http://dx.doi.org/10.62762/TIOT.2023.564457

RESEARCH ARTICLE

Check for updates

Adaptive Binary Particle Swarm Optimization for WSN Node Optimal Deployment Algorithm

Yujiang Li[1,*] **and Jinghua Cao**[1]

[1] School of Mathematics and Statistics, Lingnan Normal University, Zhanjiang 524048, China

Abstract

In order to optimize the deployment of wireless sensor network nodes, and avoid network energy consumption increase due to node redundancy and uneven coverage, the multi-objective mathematical optimization problem of area coverage is transformed into a function problem. Aiming at network coverage rate, node dormancy rate and network coverage uniformity, the idea of genetic algorithm mutation is introduced based on the discrete binary particle swarm optimization and the global optimal speed is mutated to avoid the algorithm falling into the local optimal solution. In order to further improve the optimization ability of the algorithm, the adaptive learning factor and inertia weight are introduced to obtain the optimal deployment algorithm of wireless sensor network nodes. The experimental results show that the algorithm can reduce the number of active nodes efficiently, improve coverage uniformity, reduce network energy consumption and prolong network lifetime under the premise that the coverage rate is greater than 90%, and compared with an algorithm called coverage configuration protocol, an algorithm called finding the minimum working sets in wireless sensor networks, and an algorithm called binary particle swarm optimization-g in literature, the number of active nodes in this algorithm is reduced by about 36%, 30% and 23% respectively.

Academic Editor:
Jinchao Chen

Submitted: 01 April 2023
Accepted: 17 April 2023
Published: 17 April 2023

Vol. 1, **No.** 1, 2023.
10.62762/TIOT.2023.564457

***Corresponding author:**
✉ Yujiang Li
spring888999@163.com

Keywords: Uniform coverage, Discrete binary particle swarm optimization algorithm, Wireless sensor network, Optimal deployment.

Citation
Li, Y., & Cao, J. (2023). Adaptive Binary Particle Swarm Optimization for WSN Node Optimal Deployment Algorithm. IECE Transactions on Internet of Things, 1(1), 1–8.

1 Introduction

In the related research of wireless sensor networks, how to reduce redundancy and save energy consumption has always been an important issue in this field. In order to reduce energy consumption, the redundant nodes can be reduced through sleep scheduling, so as to reduce the number of working nodes in a certain time and achieve energy saving [1–5]. Wang et al. [6] proposed a coverage algorithm for finding the minimum working point set. The number of working nodes and the coverage were considered in the algorithm, but the optimization ability of the algorithm was limited, and the number of dormant nodes did not achieve good results. Aiming at optimizing the area coverage, Liu et al. [7] studied the network coverage problem based on distributed cuckoo algorithm, which improved the calculation speed to a certain extent. Yu et al. [8] used the extrapolated artificial bee colony algorithm to optimize the deployment of nodes with preset and randomly superimposed distribution, and finally obtained the sensor node set with the largest coverage rate. Fish swarm algorithm was used to optimize node deployment by Zhou et al. [9], aiming to maximize network coverage and minimize network working nodes. However, those algorithms didn't consider the impact of area coverage uniformity on network performance. The impact of balanced coverage and energy uniformity on the network life cycle are described, and an area coverage algorithm with coverage equalization is proposed in literature [10]. In this paper, discrete binary particle swarm optimization algorithm is used, the escape factor is introduced, the local search ability of the algorithm is improved, the learning factor and inertia weight are improved in an adaptive way, and the area coverage of wireless sensor networks is studied by considering the network coverage rate, node dormancy rate and network coverage uniformity. Finally, an optimal deployment algorithm is obtained, and its effectiveness is proved by simulation experiments.

2 Improved Discrete Binary Particle Swarm Optimization Algorithm

Discrete binary particle swarm optimization algorithm is mainly proposed to solve discrete combinatorial optimization problems, such as biological information, knapsack problem, economic planning, graphics and images , wireless sensor network optimization[11–15], etc., and shows great advantages in solving multi-objective optimization problems. Discrete binary particle swarm optimization [16] has a strong global search ability, but in the process of particle approaching the optimal particle, the mutation probability becomes larger and larger, the population diversity becomes stronger and stronger, and the global search ability is enhanced. However, it lacks the local search ability and can not converge to the global optimal solution. Moreover, with the increasing randomness of iterative search, the algorithm lacks the ability of local search.

2.1 Basic Discrete Binary Particle Swarm Optimization

The discrete binary particle swarm optimization algorithm adopts 0-1 binary coding mode, assuming that the nodes with N members are scheduled for sleep, and each node is numbered sequentially, which is $1, 2, \ldots, N$. A particle represents a feasible solution in the sleep scheduling problem, and the position of the particle at a certain time is expressed as an n-dimensional 0-1 permutation string, for example, the position of particle i at time t can be expressed as ${X_i}^t = \left(x_{i1}^t, x_{i2}^t, \ldots, x_{iN}^t\right)$, where $x_{ik}^t(k = 1, 2, \ldots, N)$ represents that particle i, represents the node bit of the k th node, 1 represents that the node is in working state, and 0 represents that the node is in sleeping state. The updating formula of velocity vector is as follows:

$$\begin{aligned} V_{iQ}(k+1) = &\omega V_{iQ}(k) + c_1\gamma_1\left(p_{hi} - X_{iQ}(k)\right) \\ &+ c_2\gamma_2\left(p_g - X_{iQ}(k)\right) \end{aligned} \tag{1}$$

Where, c_1 and c_2 are the weight coefficients of the historical optimal value of the particle tracking itself and the optimal value of the particle tracking group respectively, which are usually set as 2, and γ_1, γ_2 are random numbers uniformly distributed in the interval [0,1]. ω is the inertia weight. The updating formula of position vector is as follows:

$$x_{(i,r)} = \begin{cases} 1 & g > \text{ rand} \\ 0 & g < \text{ rand} \end{cases} \tag{2}$$

$$g = \frac{1}{1 + \exp\left(-v_{id}\right)} \tag{3}$$

Where g is the probability that the position x_{ir} is taken as 1.

2.2 Variation Method of Velocity

In order to avoid the algorithm falling into the local optimum, based on the mutation idea of genetic algorithm, the escape operator is added to the speed update formula [17], which makes the algorithm easier to jump out of the local optimum and prevents the algorithm from premature convergence.

$$\begin{aligned} V_{iQ}(k+1) = &\omega V_{iQ}(k) + c_1\gamma_1\left(p_{hi} - X_{iQ}(k)\right) \\ &+ c_2\gamma_2\left(p_g - X_{iQ}(k)\right) \\ &+ \rho\gamma_3\left(Random - X_{iQ}(k)\right) \end{aligned} \tag{4}$$

Where, $Random$ is the random position of the solution space, ρ is the curiosity coefficient of the unknown space, generally 2, and γ_3 is a random number.

2.3 Adaptive Learning Factor and Inertia Weight

The learning factor and inertia weight of the coverage algorithm based on the traditional particle swarm algorithm are generally fixed values, which can not make the local or global search capability fully play in each stage, and it is difficult to obtain the optimal node set. In the process of finding the optimal node set, the previous iteration focuses on finding the nodes that can be dormant globally to reduce the number of active nodes. In the later iteration, the local uniformity optimization is mainly carried out. The cognitive learning factor c_1 and social learning factor c_2 are set to change dynamically [18], so that c_1 changes from small to large, and c_2 changes from large to small.

$$\begin{cases} c_1 = 0.5 + 2(k/stopf)^2 \\ c_2 = 2.5 - 2(k/stopf)^2 \end{cases} \tag{5}$$

The inertia weight is dynamically adjusted by cosine to increase the global search time at the beginning of iteration and the local search time at the end of iteration.

$$\omega_k = \omega_{\min} + (\omega_{\max} - \omega_{\min}) \times \cos(k/stopf) \tag{6}$$

The value range of ω is generally [0.4,0.9], and stopf is the maximum number of iterations.

3 Node Optimal-Deployment Based on Improved Algorithm

3.1 Node Perception Model

Suppose that N sensor nodes with the same attributes are randomly deployed in a two-dimensional monitoring interval, and the set of sensor nodes is $S = \{s_i \mid i = 1, 2, \ldots, N\}$, and each node has the same performance. The sensing radius r_s is R and the communication radius r_c is $2R$. The position coordinates of node S_i are (x_i, y_i), and the coordinates of any detection point S_j are (x_j, y_j). The distance between node S_i and node S_j can be expressed as:

$$d(s_i, s_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \tag{7}$$

The sensing area of a node is a circular closed area with node coordinates as the center and radius R. If the sensor perception model uses Boolean (0-1) perception model, the probability model of node detection is as follows:

$$p_{(S_i,o,R)} = \begin{cases} 0 & D_{(S_i,o)} > R \\ 1 & D_{(S_i,o)} < R \end{cases} \tag{8}$$

3.2 Regional Coverage Model

Suppose there are $m \times n$ pixel points in the monitoring area A, and the cooperative sensing model is used for the sensing of a certain pixel. Let P_{all-j} denote the set of sensor nodes of all the detected pixel S_j, assuming that there are L. such sensor nodes, then the sensing index at pixel point S_j is the sum of the probability of S_j of the L sensor nodes, namely:

$$I(s_{all-j}, s_j) = \sum_{all-j=1}^{L} p(s_{all-j}, s_j) \tag{9}$$

When the induction index of the grid point is greater than or equal to a certain threshold θ, that is, when $I(s_{all-j}, s_j) \geq \theta$, the grid point S_j is covered. Use num to record the number of grid points that meet the coverage requirements.

$$Num = \begin{cases} 1 & I(s_{all-j}, s_i) \geq \theta \\ 0 & I(s_{all-j}, s_i) < \theta \end{cases} \tag{10}$$

3.3 Determination of Objective Function

Under the condition of ensuring the coverage rate of wireless sensor network, in order to make the redundant nodes in the monitoring area sleep to the maximum extent, reduce the overlapping coverage area of working nodes, and achieve the goal of maximum coverage and uniform network coverage, the area coverage rate, node sleep rate, and area coverage uniformity are selected as the fitness function indexes of the algorithm in this paper.

Definition 1 The area coverage rate $R_{\text{area}}(S)$ of node set S is the ratio of the total number of covered pixels of node S to the total number of pixels of monitoring area A, that is:

$$R_{\text{area}}(S) = \frac{\sum_{x_j=1}^{m} \sum_{x_i=1}^{n} Num}{m \times n} \tag{11}$$

Definition 2 Node dormancy rate R_{sleep} is the ratio of the number of dormant nodes after coverage control to the total number of sensor nodes in the network:

$$R_{\text{sleep}} = \frac{N - N_w}{N} \tag{12}$$

Where N is the total number of sensor nodes in the network; N_w is the number of working sensor nodes in the network.

Definition 3 The regional coverage uniformity is expressed by the standard deviation of the distance between nodes. The smaller the standard deviation is, the better the coverage uniformity is. Coverage uniformity reflects the balance of network energy consumption and load. The better the balance of networ coverage the better the of nodes effectively covered. The smaller the redundancy distance between nodes, the more balanced the energy consumption and load between nodes.

$$u = \frac{1}{N} \sum_{i=1}^{N} U_i \tag{13}$$

$$U_i = \sqrt{\frac{1}{K_i}\sum_{j=1}^{K_i}\left(d\left(s_i, s_j\right) - M_i\right)^2} \tag{14}$$

Where u is the coverage uniformity index; U_i is the standard deviation of distance between node i and neighbor nodes; K_i is the number of neighbor nodes of the ith node; M_i represents the average distance between the ith node and all nodes that intersect its sensing range. The neighbor nodes of node i are all the nodes in the working node set whose distance to node i is less than the sensing radius R.

In order to ensure the same change trend of the three index parameters and the target value, the utility function is used to normalize the target function, and the results are as followed:

$$f_1(X) = R_{\text{area}}(S) \tag{15}$$

$$f_2(X) = R_{\text{sleep}} \tag{16}$$

Since u represents the average value of the sum of the standard deviations of the distance between N nodes i and neighboring nodes, it tends to the minimum value in the process of searching for the optimal value. In order to make the optimization trend of coverage uniformity index consistent with coverage rate and sleep rate index introduced to represent the maximum value u in each cycle, and a function f_3 as shown in formula (16) is constructed. When f_3 is larger, the coverage uniformity is better, on the contrary, when f_3 is smaller, the coverage uniformity is worse.

$$f_3(X) = \frac{u_{\max} - u}{u_{\max}} \tag{17}$$

Finally, the objective function of the optimization algorithm is obtained:

$$\max f(X) = \lambda_1 f_1(X) + \lambda_2 f_2(X) + \lambda_3 f_3(X) \tag{18}$$

Among them, λ_1, λ_2 and λ_3 are weight parameters, which satisfy $\lambda_1 + \lambda_2 + \lambda_3 = 1$. Function f_1 represents network coverage index, that is, the ratio of sensor node coverage area to the the area of monitoring area; The function f_2 represents the node sleep rate, that is, the ratio of sleep nodes to the total number of nodes in the network area; the function f_3 represents the normalized index of coverage uniformity, and the value of $u_{\max}$ is related to the number of nodes and the distribution of nodes.

3.4 Algorithm Flow

(1) Establish the area and lay the nodes. Generate N randomly distributed nodes, nodes will monitor the full coverage of the area, get node location information.

(2) Initialize population information. Initialize the whole population position X, the number of rows of X is the number of population, each row is a particle, and each particle is an N-bit binary number.

(3) Initialize the speed, the maximum speed is 7, the minimum speed is -7.

(4) Calculate the objective function. According to the objective function formula, the coverage rate, dormancy rate and area coverage uniformity index are obtained, and the fitness value of each particle, the local optimal solution and the global optimal solution of each particle are obtained.

(5) The number of iterations is initialized to $k = 1$, and the program enters the main loop.

(6) The particle velocity and position are updated according to equations (1) and (2), and the velocity variable, learning factor and inertia weight are reinitialized according to adaptive variogram (4), (5) and (6).

(7) Update particle fitness value, local optimal solution and global optimal solution of each particle.

(8) Judge whether k is equal to the maximum number of iterations $stopf$, if not, then $k = k + 1$, return to step (6), if equal to $stopf$, then output the global optimal function value and node location information, and the algorithm is over.

4 Simulation Experiment Analysis

After the accuracy of the improved algorithm is analyzed and its feasibility is verified, the parameter setting of the algorithm applied to the optimal deployment of sensor nodes is discussed, and the coverage performance is analyzed. Assuming that the nodes are randomly distributed in the monitoring area and ensure full coverage, the sensing radius of sensor nodes is 10 meters, and the communication radius is 20 meters, which satisfies the relationship that the communication radius is twice the sensing radius, and the simulation parameters are ω=1, $C_1 = C_2 = 2$, and the population size is 40.The coverage of sensors with different area and number of nodes is simulated. Set the monitoring area S={$50 \times 50\,\mathrm{m}$, $100 \times 100\,\mathrm{m}$, $150 \times 150\,\mathrm{m}$}, the node number N={100, 200, 300, 350, 400, 500, 600, 700, 800, 900}, and the simulation results are executed for 30 times and the average value is taken. A large number of calculations show that the algorithm can converge quickly and sleep redundant nodes to a large extent while ensuring coverage.

4.1 The Influence of Weight Parameters

In this paper, coverage rate, sleep rate and coverage uniformity are constructed into objective functions by means of weight parameters, so that the coverage problem of wireless sensor networks is transformed into a mathematical optimization problem. In order to discuss the influence of weight parameters on coverage control, eight combinations are selected to optimize the coverage control problem. Take the area S=50 $\times$ 50 m, the number of sensor nodes N= 100, 300, the number of iterations is 1000, and the results shown in Table 1 are obtained.

It can be found from Table 1 that the weight parameter has a direct impact on the corresponding fitness function index. When λ_1 is large, the coverage of node deployment is relatively large; When λ_2 is large and node size is N=100, the number of dormant nodes after coverage control will increase. When N=300 and λ_2=0.6, the dormancy rate of nodes will reach the maximum; However, the change trend of regional coverage uniformity index with λ_3 is not obvious. Secondly, the relationship between coverage uniformity and coverage rate, dormancy rate is analyzed horizontally. Under the condition of ensuring a certain coverage rate, after coverage control, when there are more dormant nodes, the coverage uniformity index is higher, and the active nodes can cover more evenly. On the contrary, the coverage redundancy is higher. Based on the above analysis, the weight parameters can be selected as (0.3, 0.6, 0.1).

4.2 Experimental Analysis of Different Network Scale

Set the weight parameters as (0.3, 0.6, 0.1) through the above analysis, the coverage performance is analyzed when the monitoring area S={50 $\times$ 50 m, 100 $\times$ 100 m, 150 $\times$ 150 m} and the number of nodes is N=100, 200, 300, 400, 500, 600, 700, 800, 900. It can be seen from Table 2 that in the same size area, the number of active nodes after coverage control is similar, when the area S={50 $\times$ 50 m, 100 $\times$ 100 m, 150 $\times$ 150 m}, the number of active nodes is between 13-20, 45-57, 99-105, the regional coverage rate is more than 90%, and the coverage uniformity index is more than 0.75.

In order to discuss the relationship between the active node and the area as well as the perceived radius, the sensor radius of different sensor nodes, namely, r_s={8, 10, 13, 15}, is used for experimental analysis.

The relative area S' is taken as:

$$S' = r_s^2/S \tag{19}$$

Figure 1 shows the relationship between the number of active nodes and the relative area S'. The power function can be constructed by drawing the scatter diagram and fitting the trend line according to the scatter trend:

$$y = \kappa_1 x^{-\kappa_2} \tag{20}$$

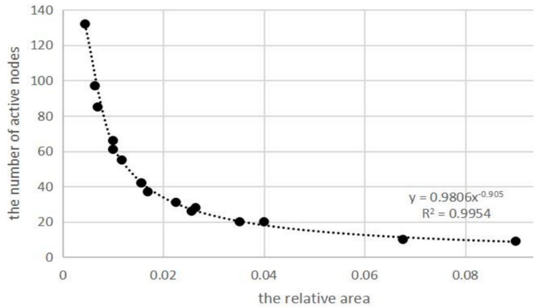

Figure 1. The relationship between the number of active nodes and the relative area of the region.

Among them, k_1 and k_2 can be obtained through experiments. According to the formula, the number of active nodes after coverage control under a certain relative area can be approximately calculated. The number of active nodes obtained indicates that in the coverage problem of wireless sensor networks, the set of active nodes less than or equal to the number calculated by equation (20) can always be obtained through optimized coverage control.

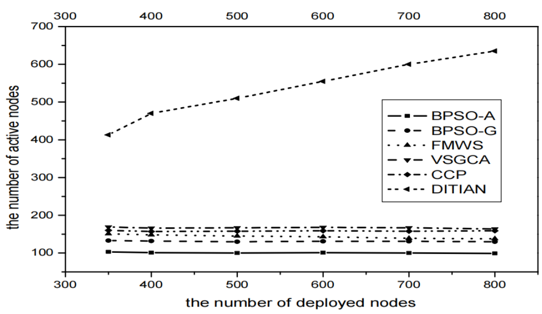

Figure 2. The relationship between the number of active nodes and the total number of nodes.

4.3 Comparison with Other Algorithms

In order to further verify the performance of the improved algorithm, DITIAN algorithm, CCP algorithm, VSGCA algorithm, FMWS algorithm and BPSO-G algorithm [19] are selected for comparative

Table 1. Coverage Performance under Different Weight Parameters.

Weight parameter setting (λ_1,λ_2 ,λ_3)	N=100/300		
	Coverage/%	Number of active nodes	Uniformity index
(0.7,0.2,0.1)	100/100	17/28	0.78/0.60
(0.45,0.5,0.05)	100/100	16/27	0.74/0.52
(0.45,0.05,0.5)	100/100	33/47	0.65/0.70
(0.4,0.3,0.3)	96/100	18/33	0.65/0.67
(1/3,1/3,1/3)	98/100	16/37	0.78/0.69
(0.3,0.6,0.1)	97/98	13/15	0.81/0.83
(0.2,0.7,0.1)	98/99	15/18	0.78/0.76
(0.05,0.5,0.45)	95/99	15/31	0.76/0.67

Table 2. Coverage performance under different network sizes.

Number of nodes N	S={50×50 m, 100×100 m, 150×150 m}		
	Coverage/%	Number of active nodes	Uniformity index
400	100/95/91	18/48/101	0.80/0.75/0.81
500	99/97/92	16/55/100	0.80/0.91/0.82
600	100/97/93	16/53/101	0.84/0.81/0.80
700	100/98/94	17/56/100	0.81/0.78/0.80
800	100/98/91	20/55/99	0.84/0.84/0.83
900	100/99/95	20/57/105	0.87/0.77/0.76

experiments with the improved BPSO-A algorithm in this paper. The parameter setting of monitoring area and sensor node is the same as that in reference [6] of active nodes calculated by DITIAN algorithm increases approximately linearly with the increase of the total number of deployed nodes N. The number of active nodes obtained by other algorithms remains within a certain range. Compared with CCP, FMWS and BPSO-G algorithm, BPSO-A algorithm reduces by about 36%, 30% and 23% respectively, as shown in Table 3. However, we can also see that BPSO-A algorithm sacrifices coverage rate to some extent in order to get less active nodes and more uniform node distribution. Figure 3 compares the area coverage rate controlled by BPSO-A algorithm and BPSO-G algorithm. Of course, this sacrifice is desirable under certain coverage rate.

Table 3. Comparison of the number of active nodes obtained by four algorithms.

algorithm	Total number of deployed nodes					
	350	400	500	600	700	800
CCP	160	157	158	159	158	159
FMWS	151	148	145	143	139	138
BPSO-G	133	132	130	131	131	130
BPSO-A	103	101	100	101	100	99

5 Conclusion

This paper proposes a deployment algorithm for wireless sensor networks, which aims at network coverage, node dormancy rate and area coverage uniformity. In the coverage control of the algorithm, sleep most nodes, reasonably select the distributed working nodes, and form a group of optimal node distribution set while ensuring the network coverage, which lays the foundation for network transmission and communication. The algorithm is mainly based on discrete binary particle swarm optimization algorithm, using escape factor to mutate the global optimal speed, and using adaptive learning factor and inertia weight, so that the algorithm can quickly converge to the optimal solution. After discussing the reliability and stability of the improved algorithm, the weight parameters of fitness function are studied, and the final weight parameters of the algorithm are (0.3, 0.6, 0.1). In this paper, through a large number of simulation experiments, we can get an approximate formula, which describes the relationship between the number of active nodes and the relative area after coverage

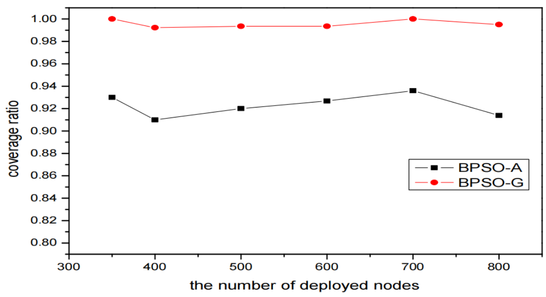

Figure 3. Relationship between coverage and total number of nodes.

control. We can use this formula to approximate the number of active nodes after coverage control in a certain relative area.

Compared with other algorithms, it can be seen that this algorithm can effectively reduce the number of active nodes, improve the uniformity of coverage area and reduce network energy consumption under the condition that the area coverage is greater than 90%.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] Zhu, C., Zheng, C., Shu, L., & Han, G. (2012). A survey on coverage and connectivity issues in wireless sensor networks. *Journal of Network and Computer Applications*, 35(2), 619-632. [CrossRef]

[2] Renold, A. P., & Chandrakala, S. (2016). Survey on state scheduling-based topology control in unattended wireless sensor networks. *Computers & Electrical Engineering*, 56, 334-349. [CrossRef]

[3] Zhang, B., Tong, E., Hao, J., Niu, W., & Li, G. (2016). Energy efficient sleep schedule with service coverage guarantee in wireless sensor networks. *Journal of Network and Systems Management*, 24, 834-858. [CrossRef]

[4] Singh, B., & Lobiyal, D. K. (2013). Traffic-aware density-based sleep scheduling and energy modeling for two dimensional Gaussian distributed wireless sensor network. *Wireless personal communications*, 70, 1373-1396. [CrossRef]

[5] Li, J., Chen, Z., Cheng, L., & Liu, X. (2022). Energy data generation with wasserstein deep convolutional generative adversarial networks. *Energy*, 257, 124694. [CrossRef]

[6] Wang, A., Liu, Y., Zhang, J., & Liu, Y. (2016). Coverage algorithm for finding the minimum working sets in WSNs. *Journal of Xidian University*, 43(04), 141-146.

[7] Liu, X., Zhang, X., Hu, T., & Zhu, Q. (2018). Deployment optimization of wireless sensor network based on parallelized cuckoo search algorithm. *Application Research of Computers*, 35(7), 2063-2065.

[8] Yu, W., Li, X., Yang, H., Huang, B. (2017). Extrapolation artificial bee colony algorithm research on deployment optimization in wireless sensor network. *Instrument Technique and Sensor*, 6,158-160.

[9] Zhou, L., Yang, K., Zhou, P. (2010). Optimal coverage configuration based on artificial fish swarm algorithm in WSNs. *Application Research of Computers*, 6, 2276-2279.

[10] Qin, N., Chen, J., Ding, Z. (2015). Balanced rate area coverage algorithm. *Chinese Journal of Sensors and Actuators*, 28(4),578-584.

[11] Liu, Q., Cheng, L., Jia, A. L., & Liu, C. (2021). Deep reinforcement learning for communication flow control in wireless mesh networks. *IEEE Network*, 35(2), 112-119. [CrossRef]

[12] Huang, Y., Cheng, L., Xue, L., Liu, C., Li, Y., Li, J., & Ward, T. (2021). Deep adversarial imitation reinforcement learning for QoS-aware cloud job scheduling. *IEEE Systems Journal*, 16(3), 4232-4242. [CrossRef]

[13] Cheng, L., Wang, Y., Liu, Q., Epema, D. H., Liu, C., Mao, Y., & Murphy, J. (2021). Network-aware locality scheduling for distributed data operators in data centers. *IEEE Transactions on Parallel and Distributed Systems*, 32(6), 1494-1510. [CrossRef]

[14] Cheng, F., Huang, Y., Tanpure, B., Sawalani, P., Cheng, L., & Liu, C. (2022). Cost-aware job scheduling for cloud instances using deep reinforcement learning. *Cluster Computing*, 1-13. [CrossRef]

[15] Li, J., Tong, X., Liu, J., & Cheng, L. (2023). An Efficient Federated Learning System for Network Intrusion Detection. *IEEE Systems Journal*. [CrossRef]

[16] Li, Y., Pan, B. (2018). Research of WSN regional coverage based on adaptive mutation binary particle swarm optimization. Journal of Sichuan University of Science & *Engineering* (*Natural Science Edition*), 31(01), 20-24.

[17] Wang, Y., Qiu, F. Y., & Guo, H. D. (2019). Adaptive inertia weight binary particle swarm optimization algorithm with mutation operator. *Journal of Chinese Computer Systems*, 40(04), 733-737.

[18] Wu, X., Zhang, C., Zhang, R., & Sun, Y. (2019). Clustering routing protocol based on improved PSO algorithm in WSN. *Journal on Communications*, 40(12), 114-123.

[19] Li, Y., & Pan, B. (2018). Research of WSN Regional Coverage Based on Adaptive Mutation Binary Particle Swarm Optimization. *Journal of Sichuan University of Science & Engineering* (*Natural Science Edition*), 31(01), 20-24.

Yujiang Li is a M.S. and a lecturer working on computer sceince and technology in Lingnan Normal University. Her research interests include WSN, big data,distributed simulation. more than 10 papers published.

Jinghua Cao is a PhD and an associate professor working on computer sceince and technology in Lingnan Normal University, also a member of computer society. Her research interests include computer application,modeling and control, big data, artificial intelligence, computer teaching, more than 17 papers published and 7 book published.