IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 14, 2021, accepted January 19, 2021, date of publication January 22, 2021, date of current version February 1, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3053594

A Novel Coverage Optimization Strategy for Heterogeneous Wireless Sensor Networks Based on Connectivity and Reliability

LI CAO[1,2], **YINGGAO YUE**[1,3], **YONG CAI**[2], **AND YONG ZHANG**[3]

[1]Oujiang College, Wenzhou University, Wenzhou 325035, China
[2]School of Information Engineering, Southwest University of Science and Technology, Mianyang 621010, China
[3]Computer School, Hubei University of Arts and Science, Xiangyang 441053, China

Corresponding author: Yong Zhang (yzhangedu@yeah.net)

This work was supported in part by the Natural Science Foundation of Hubei Province under Grant 2020CFB304, in part by the Open Fund of Distinctive Disciplines in Hubei University of Arts and Science under Grant XK2020047, in part by the Educational Science Planning Issue of Hubei Province under Grant 2020GA057, in part by the Talent Introduction Project of Oujiang College of Wenzhou University, in part by the Educational Science Planning Subject of Hubei Province under Grant 2020GA057, in part by the Xiangyang Soft Science Research Project under Grant 2020rkx05, and in part by the Counterpart Service for the Construction of Xiangyang Science and Technology Innovation China Innovative Pilot City.

ABSTRACT To overcome the problems of coverage blind areas and coverage redundancy when sensor nodes are deployed randomly in heterogeneous wireless sensor networks (HWSNs). An optimal coverage method for HWSNs based on an improved social spider optimization (SSO) algorithm is proposed, which can reduce the energy consumption and improve the network coverage. First, a mathematical model of HWSN coverage is established, which is a complex combinatorial optimization problem. To improve the global convergence speed of the proposed algorithm, a chaotic initialization method is used to generate the initial population. In addition, the SSO algorithm has a poor convergence speed and search ability, which is enhanced by improving the neighborhood search, global search, and matching radius. In the iterative optimization process, the optimal solution is ultimately obtained by simulating the movement law of the spider colony, i.e., according to the cooperation, mutual attraction, and mating process of female and male spiders. An improved SSO algorithm based on chaos, namely the CSSO algorithm, is proposed to apply to the optimal deployment of sensory nodes in HWSNs. On this basis, the optimization goals are to improve the network coverage and reduce network costs. The optimal deployment plan of nodes is searched via the proposed CSSO algorithm, which effectively prevents coverage blind spots and coverage redundancy in the network.

INDEX TERMS Wireless sensor network, coverage constraint optimization; probabilistic perception model, monarch butterfly algorithm, coverage rate.

I. INTRODUCTION

Wireless sensor networks (WSNs) are composed of many miniature wireless sensor nodes deployed in the monitoring area, which form a multi-hop self-organizing network system via wireless communication [1]. Their purpose is to cooperatively perceive, collect, and process the information of the sensing objects in the network coverage area. WSNs are formed by the self-organization of miniature sensor nodes, and are characterized by low cost, low power consumption, sensing, data processing, storage, and wireless communication capabilities. Via the collaborative work between nodes, a variety of real-time environmental monitoring and sensing information is transmitted to the base station for processing [2].

Heterogeneous wireless sensor networks (HWSNs) are an advanced type of WSN. They are not only compatible with various key techniques of homogeneous networks, but can also use the heterogeneous characteristics of the sensor nodes to improve the network performance. Therefore, combining an optimal coverage algorithm with a heterogeneous network environment can better meet the different needs of practical applications [3]. The network coverage, energy consumption, and network connectivity reliability are the core issues

The associate editor coordinating the review of this manuscript and approving it for publication was Xijun Wang.

of WSNs, and they are closely related. The network coverage determines the ability of HWSNs to monitor the physical world, and reflects the perceived service quality that the network can provide [4].

HWSNs are faced with problems such as limited energy, poor node reliability, and complex deployment environments, which will greatly affect the data collection and transmission of sensor nodes and limit the performance of the network [5], [6]. Thus, it is necessary to introduce an efficient coverage control algorithm to rationally use the limited resources in the WSN, reduce the energy consumption of the sensor nodes, and improve the lifetime and service quality of the network. The network coverage is one of the research hotspots of WSNs, and the coverage of the target area can be maximized by optimizing the deployment location of wireless sensors, and by reliably sensing and monitoring the tracking area.

A. PROBLEM STATEMENT AND MOTIVATE

In the traditional deployment strategy of heterogeneous wireless sensors, to ensure the reliability of HWSNs and the accuracy of the monitoring data, heterogeneous sensor nodes are often deployed in the target area at a higher density. Due to the high density of heterogeneous sensor nodes, a point or an area is often covered by multiple sensors, and coverage redundancy occurs. When these sensors detect an event, they all begin to send data to the base station. These data contain a lot of repeated information, and the limited energy of the sensors will therefore be consumed by redundant data transmission. Moreover, a too-dense node deployment will increase the communication overhead of the network and reduce the performance of HWSNs. Therefore, it is necessary to optimize the topology of the network while ensuring network coverage to minimize redundant coverage.

Coverage control is one of the basic problems of HWSNs, and coverage quality can be used as one of its indicators to measure the service quality of network monitoring functions. The coverage problem of HWSNs reflects the ability of the sensor network to perceive the physical world, and, to a certain extent, determines the network service performance and the network's lifetime.

B. CONTRIBUTION

In this work, a new method of coverage optimization strategy for WSNs based on monarch butterfly algorithm optimized by particle swarm optimization method is proposed. In comparison with the current general selection approaches, the main contributions of our work in this article can be summarized as follows:

1. Characterize the issues of a coverage optimization strategy for HWSNs, and establish a mathematical model of coverage optimization strategy for HWSNs.
2. Present a novel coverage optimization strategy based on monarch butterfly algorithm optimized by particle swarm optimization method.
3. Provide extensive simulation results to demonstrate the use and efficiency of the proposed coverage optimization algorithm.
4. Evaluate the performance of the proposed algorithms by comparing them with the coverage optimization algorithms of the PSO, GWO and the basic MBO algorithm.

The rest of the paper is organized as follows: Section 2 discusses the related work. Section 3 equationtes the problem of the coverage optimization strategy for the WSNs. Section 4 describes the implementation steps of monarch butterfly algorithm optimized by particle swarm optimization and Section 5 presents the applied mathematical models and optimization steps of the coverage optimization strategy for WSNs. Section 6 provides the parameters and simulation results that validate the performance of our algorithm. Section 7 concludes the paper.

II. RELATED WORK

The heterogeneous characteristics of HWSNs are embodied in three aspects, namely node heterogeneity, link heterogeneity, and network protocol heterogeneity [7]–[9]. Among them, node heterogeneity also includes the heterogeneity of perception capabilities, computing capabilities, communication capabilities, and energy, which have the greatest impacts on coverage. There have previously been some studies on the coverage of randomly deployed HWSNs, and determining how to extend the life cycle of the network and improve its connectivity and reliability while maintaining network coverage have become key issues in the research of HWSNs. Scholars both domestically and internationally have conducted research on the coverage of WSNs. In [10], the author proposed the improvised distributed energy-efficient clustering (I-DEEC) algorithm by deploying network nodes in two layers; the I-DEEC algorithm provides blanket coverage by extending the stability period via the reduction of the ratios between the initial energy of different types of nodes. In [11], the author reconstructed the hole model, and proposed a multi-factor collaborative hole repair optimization algorithm (FCH-ROA) for use between HWSN nodes. The algorithm considers the distance between the virtual repair node and the mobile node, the energy consumption during the mobile repair process, and the confidence that the node can be repaired. To improve the regional coverage rate and network lifetime of HWSNs, a sensor node scheduling algorithm for HWSNs was proposed by [12]; the proposed algorithm was found to improve the network lifetime, increase the number of living sensor nodes, and maintain the average node energy consumption at a low level. In [13], the author proposed a multi-objective deployment strategy (MODS), which uses multi-objective evolutionary algorithms to achieve near-optimal solutions for WSN deployment problems. The non-dominated sorting multi-objective flower pollination algorithm (NSMOFPA) was proposed in [14], and was applied to WSN deployment with the goal of optimizing the coverage rate; the experimental results verified that the NSMOFPA exhibited a good optimization effect and could provide a better solution for HWSNs.

The existing coverage algorithms are mainly focused on expanding the coverage of the target area and prolonging the lifetime of the network. By changing the routing of the sensor nodes and the network topology, as well as by introducing mobile sensors, the coverage holes in the target area can be effectively reduced, and the network coverage can be improved. Moreover, the sensors can be dispatched and controlled, and the working time of the sensors can be extended by adjusting their working mode. In areas with dense nodes, some sensors are set to a sleep mode, and sensor nodes are scheduled in a rotating manner to prolonging the lifetime of the network. In the existing research, many scholars have used swarm intelligence algorithms and mobile sensors to solve the network coverage problem of WSNs. The use of swarm intelligence optimization algorithms can yield suitable sensor deployment locations, which can effectively solve the uneven distribution of nodes due to the random deployment of sensors, alleviate the phenomena of network coverage holes or network coverage redundancy, and improve the service quality and lifetime of the network. The introduction of mobile wireless sensors into the WSN, and the use of a scheduling algorithm to plan the movement of mobile sensors, effectively solves the problem of traditional static wireless sensor nodes being unable to update their locations after deployment to prevent network coverage holes. Sangaiah *et al.* [15] proposed the bat algorithm (BA) to select the optimum monitoring sensor node and resulted path to reduce energy consumption, the proposed algorithm reduced the power consumption of the network and increased the lifetime of the network. Alia and Al-Ajouri [16] proposed a harmony search (HS) based deployment algorithm that can locate the optimal number of sensor nodes as well as their optimal locations for maximizing the network coverage and minimizing the network cost. By using truncated octahedrons to stack the 3-D environment seamlessly, the coverage enhancement and energy optimization problems are transformed into a task-assignment problem of moving nodes to truncated octahedrons, and an energy-efficient coverage enhancement strategy based on the vampire bat optimizer (VBO) is proposed to solve the above problem [17].

In [18], the author presented a node optimization coverage method under a link model in the passive monitoring system of a three-dimensional WSN; the particle swarm optimization (PSO) algorithm, which integrates the concept of simulated annealing, was proposed, and the proposed method was found to improve the network coverage, converge quickly, and reduce the network energy consumption. Massive sensor nodes are randomly deployed and remain static after deployment, which will usually cause coverage redundancy or coverage holes. To effectively deploy sensors to cover a whole area, In [19], the author presented a novel node deployment algorithm based on mobile sensors. Hashim *et al.* [20] proposed an enhanced deployment algorithm based on the artificial bee colony (ABC) algorithm. The ABC-based deployment was found to be guaranteed to extend the lifetime by optimizing the network parameters and constraining the total number of deployed relays. To maximize the network coverage and minimize the energy consumption to ultimately ensure the quality of service, a WSN coverage optimization method based on an improved artificial fish swarm algorithm was proposed in [2‘]. Moreover, Wang *et al.* [22] proposed a WSN coverage optimization model based on an improved whale swarm algorithm to monitor the field of interest and obtain valid data; the concept of reverse learning was introduced into the original whale swarm optimization algorithm to optimize the initial distribution of the population, which was found to enhance the node search capability and speed up the global search.

The existing research on the coverage of WSNs has been primarily based on homogeneous networks, i.e., the performance of the nodes participating in the network is considered the same in terms of the power supply, sensing radius, cost, computing power, lifetime, and mobility. In actual applications, however, some heterogeneous nodes with unique performance may be deployed for special "hot spots," and their parameter performance is better than that of ordinary nodes in all aspects. In a WSN, the deployment of an appropriate number of heterogeneous nodes can not only improve the service quality of the entire WSN, but can also effectively prolong the lifetime of the network. This article investigates an improved social spider optimization (SSO) algorithm based on chaos, namely the CSSO algorithm, that is applied to the optimal deployment of sensor nodes in HWSNs. On this basis, the optimization goal is to improve network coverage and reduce network costs. The proposed algorithm is used to search for the optimal deployment plan of sensor nodes, and is found to effectively prevent coverage blind spots and coverage redundancy in the network.

III. MATHEMATICAL MODEL

Assuming that the monitoring area is a limited two-dimensional plane, an appropriate number of sensor nodes is placed in the area to achieve the complete coverage of the area. In practical applications, the complete coverage of the monitoring area does not need to be achieved, and the deployment of a large number of nodes will impose unnecessary costs. Generally, only incomplete area coverage and a limited coverage rate are required for a specific area. Under the minimum cost, an appropriate number of nodes is deployed to achieve the coverage control of the network. Alternatively, under a certain cost mechanism, a limited number of nodes are deployed to achieve the optimal network coverage. An HWSN refers to a WSN with a small number of heterogeneous nodes. The heterogeneous nodes investigated in this article are different from ordinary nodes primarily in terms of their perception radius and cost. For the deployment of nodes in a two-dimensional monitoring area, the following assumptions are made:

1) The monitoring area of the sensor network is a two-dimensional plane, i.e., all the sensor nodes are on this plane;
2) Each node adopts a probabilistic perception model, the coverage area is a circular area centered on the node's position, and the radius of the circle is its perception radius r;
3) The sensing radii of sensor nodes are heterogeneous, i.e., sensor nodes have different sensing radii, and the sensing model of the sensor nodes is binary;
4) A node in the sensor network is static, and is connected to the sensor network.

In this research, the probability-aware model is used to calculate the coverage rate of the network. Each sensor node in the HWSN takes itself as the sensing coverage center, and has a circular area with a fixed communication radius. Therefore, it is difficult for all the sensor nodes to solve the total coverage of the monitoring area via mathematical equations. To simplify the coverage problem in WSNs, the area to be monitored can be discretized into $m \times n$ pixels. Assuming that x pixels are covered by WSN, the coverage can be expressed as $x / (m \times n)$.

Suppose that the measurement radius r of each sensor node in a WSN is the same as the communication radius r_s, and the coverage area of each sensor node is a circular area with radius r. In this work, it is assumed that the measured area of the sensor network is a two-dimensional plane M, which is discretized into $m \times n$ pixels. There are N sensor nodes in the WSNs. The set of sensor nodes in the measured area is $G = \{g_1, g_2, \ldots, g_N\}$, and the position of the i-th sensor node g_i is (x_i, y_i). Assuming that the coordinates of the pixel H are (x_H, y_H), then the distance between the pixel and the sensor node of g_i is as follows.

$$d(g_i, H) = \sqrt{(x_H - x_i)^2 + (y_H - y_i)^2} \tag{1}$$

Using a two-dimensional perception model, the probability of the sensor node g_i sensing pixel H is:

$$p(g_i, H) = \begin{cases} 1, & d(g_i, H) \le r \\ 0, & d(g_i, H) > r \end{cases} \tag{2}$$

Assuming that any one sensor node can be sensed by multiple sensor nodes at the same time, the joint probability that the sensor node at pixel H is sensed by the node set G of wireless sensor network is:

$$p(G, H) = 1 - \prod_{g_i \in G} [1 - p(g_i, H)] \tag{3}$$

The coverage rate λ of all the sensor nodes to be detected is:

$$\lambda = \frac{\sum\limits_{H \in m \times n} p(G, H)}{m \times n} \tag{4}$$

In addition, assuming that the network nodes work efficiently θ is

$$\theta = \frac{S_2}{S_1} \tag{5}$$

In equation (5), the parameter S_1 is the total number of sensor nodes, and the parameter S_2 is the number of effective working sensor nodes.

Taking into account the energy balance of the network, the definition of energy balance coefficient η is introduced, specifically, the parameter E_i represents the remaining energy of the node i, and the parameter k represents the number of the active nodes.

$$\eta = \frac{Max(E_i) - Min(E_i)}{\frac{1}{k} \sum\limits_{i=1}^{k} E_i} \tag{6}$$

The parameter η reflects the equilibrium degree of the network energy consumption. The larger the value, the more uneven the energy consumption. On the contrary, the more uniform the energy consumption.

Since the coverage of WSNs is optimized to integrate the number of the working nodes, the coverage rate and the energy balance, on the basis of making the network coverage rate meet the actual application requirements, as many redundant nodes as possible go to sleep state, thereby saving the energy consumption. Therefore, the coverage optimization objective mathematical model f of WSNs is

$$f = \omega_1 \times \lambda + \omega_2 \times \theta + \omega_3 \times (1 - \eta) \tag{7}$$

In the equation (7), the parameters ω_1, ω_2 and ω_3 are the weight coefficients, $\omega_1 + \omega_2 + \omega_3 = 1$. Wherein, $\omega_1 = 0.5$, $\omega_2 = 0.25$, $\omega_3 = 0.25$.

The optimization goal of the network coverage model of HWSNs is the maximum value of the coverage function in equation (7).

IV. CHAOTIC SOCIAL SPIDER OPTIMIZATION ALGORITHM

A. SOCIAL SPIDER OPTIMIZATION ALGORITHM

The SSO algorithm is a novel biologically-inspired optimization algorithm that simulates the collaborative behavior of social spiders [23]. The search space is analogous to a spider web, and two types of spiders, namely female and male, are generated to search for individuals in the space according to different search criteria, which effectively prevents individuals from gathering around the dominant group and falling into the local optimum. New individuals are generated by information exchange via marriage and mating behavior, and the goal of finding the best individual is ultimately realized [24]. In the SSO algorithm, the total number of spiders is set, male and female spiders are randomly allocated. According to the iterative movement of the female spiders, the male spiders move closer to the female spiders, or the middle individual of the male spiders moves closer and searches for female spiders within its range and match with one of them. The generation of new spider individuals continues to iterate, thereby realizing the iteration of the entire spider group, and making the group continuously move closer toward the optimal solution; finally, the value of the optimal solution is obtained [25]. The basic steps of the SSO algorithm are as follows.

1) SPIDER POPULATION PARAMETER SETTING

Assuming the total number of spiders N, the number of female spiders N_f and the number of male spiders N_m can be calculated by equation (8).

$$\begin{cases} N_f = floor[(0.9 - rand \times 0.5) \times N] \\ N_m = N - N_f \end{cases} \tag{8}$$

In the equation (8), the parameter *rand* is a random number between [0,1], and the parameter *floor* represents the integer part.

2) POPULATION INITIALIZATION

Assuming that the solution space has D dimensions, the spider population consists of male subpopulations $SM = \{SM_1, SM_2, \cdots, SM_{Nm}\}$ and female subpopulations $SF = \{SF_1, SF_2, \cdots, SF_{Nf}\}$. Initialize all the spider positions in a random manner, as shown in equation (9):

$$\begin{cases} SM_{ij} = lb_j + rand \times (ub_j - lb_j) \\ SF_{ij} = lb_j + rand \times (ub_j - lb_j) \end{cases} \tag{9}$$

In equation (9), the parameter ub_j and the parameter lb_j respectively represent the upper and lower bounds of the solution space in the dimension $j (j \in \{1, 2, 3, \cdots, D\})$. According to the positions of all spiders, we can calculate the mating radius r.

$$r = \frac{\sum_{j=1}^{D} SP_j^{\max} - SP_j^{\min}}{2D} \tag{10}$$

In the equation (10), the parameter SP represents the entire spider population, consisting of all the male and female spiders $SP = [SF, SM]$, and the parameters $SP_j^{\max}$ and $SP_j^{\min}$ represent the maximum and minimum values in the j-th dimension of the spider population, respectively.

3) ITERATION OF INDIVIDUAL POSITIONS OF FEMALE SPIDERS

Female spiders transmit information through vibrations, and the SSO algorithm simulates vibrations to attract or repel other individuals. In the minimum optimization problem, the individual FS_i weight value is ω_i, and its vibration perception ability $Vib_{c,i}$ of the individual FS_c is calculated as follows [26]:

$$\omega_i = \frac{J(FS_i) - worst(J)}{best(J) - worst(J)} \tag{11}$$

$$Vib_{c,i} = \omega_i e^{-d_{ci}^2} \tag{12}$$

In equation (11), the parameter $J(FS_i)$ represents the value of the objective function of the spider FS_i, the parameter $worst(J)$ represents the maximum value of the objective function in the population, $best(J)$ represents the minimum value of the objective function in the population, and the parameter d_{ci} represents the distance between the FS_i and FS_c of the individual spider [27].

The female spider individual iteration is carried out by equation (13).

$$SF_i^{T+1} = \begin{cases} SF_i^T + \alpha Vib_{c,i}(SP_c - SF_i^T) + \beta Vib_{c,i}(SF_b - SF_i^T) \\ \quad +\delta(rand - 0.5), \quad rand \le PF \\ SF_i^T - \alpha Vib_{c,i}(SP_c - SF_i^T) - \beta Vib_{c,i}(SF_b - SF_i^T) \\ \quad +\delta(rand - 0.5), \quad else \end{cases} \tag{13}$$

In equation (13), the parameters α, β, δ and *rand* are random numbers between [0, 1], and the parameter T represents the number of iterations. The parameter SP_c is the spider individual closest to the individual SF_i and higher than its weight. The parameter SF_b is the highest weighted individual in the female subpopulation.

4) ITERATION OF INDIVIDUAL POSITIONS OF THE MALE SPIDERS

There are two types of spiders in male subpopulations, one is the male-dominated population, and the other is the male non-dominated population. Among them, the dominant spider has the ability to attract female spiders close to it, while the non-dominant male spider tends to approach the center of the dominant spider. The behavior of two types of male spiders is simulated by equation (14).

$$SM_i^{T+1} = \begin{cases} SM_i^T + \gamma\left(\frac{\sum_{k=1}^{N_m} SM_k^T \times \omega_k}{\sum_{k=1}^{N_m} \omega_k}\right), \quad \omega_k \le \omega_{mid} \\ SM_i^T + \gamma Vib_{fi}(SF_f - SM_i^T) + \delta(rand - 0.5), \\ \qquad else \end{cases} \tag{14}$$

In equation (14), the parameter SF_f represents the female spider closest to SM_i, the parameters γ and δ are random numbers between [0, 1], and the parameter ω_{mid} represents the median weight of the male spider population [28].

5) MARRIAGE BEHAVIOR

After all the individual iterations are completed, the male-dominant individual will engage in mating behavior. All female spiders within the mating radius r of the male-dominant individual distribute the mating probability Pm according to the weight ω, as shown in equation (15):

$$P_m = \frac{\omega FS_i}{\sum \omega} \tag{15}$$

According to Eq. (15), the probability of all female individuals mating is obtained, and the mating objects are selected according to the probability in each dimension in the manner of roulette. The value of the new individual in this dimension is the value of the selected female spider in this dimension. After obtaining the new individual, its fitness value is calculated; if it is better than the fitness value of the parent, the inferior individual is replaced. Finally, it is judged whether the iteration termination condition is reached. If the condition is met, the optimal individual is output; otherwise, the next iteration is entered [29].

B. CHAOTIC SOCIAL SPIDER OPTIMIZATION ALGORITHM

The standard SSO algorithm has a slower convergence speed and can easily fall into a local optimum, primarily due to the complexity of the calculation process; for each new individual, learning from the neighborhood optimal solution and learning from the global optimal solution must both be completed. In the initial stage, the individual spiders are relatively scattered, and the value of the neighborhood search is not exceptional; thus, the global search capabilities should be emphasized. In the later stage, the global search capability should be gradually reduced. Regarding the mating behavior, the radius r should be kept unchanged in the early stage of mating, while the mating behavior of individual spiders should be expanded. In the later stage, the search radius r should be reduced to search for individual spiders in a small area to avoid repetition. To improve the global convergence speed of the SSO algorithm, the initial population is generated by the chaotic initialization method. In addition, the SSO algorithm has a poor convergence speed and search ability. By improving the neighborhood search, global search, and matching radius, a better convergence speed and search ability can be achieved. In the iterative optimization of the algorithm, the optimal solution is ultimately obtained by simulating the movement of the spider colony, i.e., according to the cooperation, mutual attraction, and mating process of female and male spiders. The maximum value of the objective function of WSNs coverage optimization is solved, and the convergence of the improved SSO algorithm is accelerated. In the subsequent search process, the diversity of the population will be increased, the coverage rate of the HWSN will be improved, and the distribution position of all the sensor nodes in the area to be tested will be determined after optimized deployment.

1) POPULATION INITIALIZATION METHOD BASED ON CHAOS

Chaos has the characteristics of randomness, ergodicity and regularity, and has been widely used in swarm intelligence optimization algorithms such as particle swarm algorithm, evolutionary algorithm and artificial bee colony algorithm to improve the search efficiency of the algorithm. In order to make the initial population individuals use the information of the solution space as much as possible, this article uses Skew Tent mapping to generate chaotic sequences for population initialization. The mathematical model is

$$\begin{cases} x_{k+1} = x_k/\varphi & 0 < x_k < \varphi \\ x_{k+1} = (1 - x_k)/(1 - \varphi), & \varphi < x_k < 1 \end{cases} \tag{16}$$

When the parameter $\varphi \in (0, 1)$ and the parameter $x \in [0, 1]$, the parameters in equation (16) are in a chaotic state.

2) IMPROVED NEIGHBORHOOD SEARCH

When appropriate, the hybrid algorithm incorporating the neighborhood search operator can improve the local search ability, so as not to fall into the local optimum and improve the search accuracy. The steps are as follows.

At the beginning, when the spider individual i is in the optimal position for successive iterations and there is no change, perform learning from the optimal solution in the neighborhood, let the spider individual i search in the neighborhood, prevent falling into the local optimal, and improve the search ability of the spider individual.

The judgment condition is

$$J(FS_{i+1}) = J(FS_i) \tag{17}$$

In equation (17), the parameter $J(FS_i)$ is the fitness value of the spider individual i.

3) GLOBAL SEARCH IMPROVEMENTS

In the initial stage of the search, the global search should be the main focus, and in the later stage of the search, the global search should be gradually reduced. Introduce the global search factor σ.

$$\sigma = \sigma_{min} + (\sigma_{max} - \sigma_{min})\sqrt{\frac{T_{max} - T}{T_{max}}} \tag{18}$$

In equation (18), the parameter σ_{max} is the maximum parameter factor, the parameter σ_{min} is the minimum parameter factor, the parameter T is the number of iterations, and T_{max} is the maximum number of iterations. Generally, $\sigma_{max} = 0.9$ and $\sigma_{min} = 0.4$.

4) IMPROVED MATCH RADIUS R_r

For the initial stage of the search, the distance between female individuals is relatively large, and the radius R_r should be maintained to promote marriage between individuals and replace the worst individual. In the later stage of mating, female individuals are more concentrated, so the mating radius r should be reduced and some poor individuals should be eliminated.

$$R_r = R_r \times (1 - a \times e^{b(1-\frac{T}{T_{\max}})}) \times rand \tag{19}$$

In equation (19), the parameter a is 1.1, the parameter b is 0.9, the parameter T is the number of iterations, and T_{max} is the maximum number of iterations. The parameter $rand$ is a random number between (0, 1).

The flow chart of sensor coverage optimization of heterogeneous wireless sensor network based on CSSO algorithm is shown in Figure 1.

V. APPLICATION OF IMPROVED SOCIAL SPIDER ALGORITHM IN COVERAGE OPTIMIZATION OF HWSNS

The improved SSO algorithm is used to solve the problem of sensor node deployment optimization in HWSNs. First, the HWSN system is initialized and the SSO algorithm is initialized. After calculating and evaluating the network coverage, the location and speed of each individual spider are updated according to the coverage. The best individual of the spiders is screened out, and the information of each individual spider, including the location and coverage of the

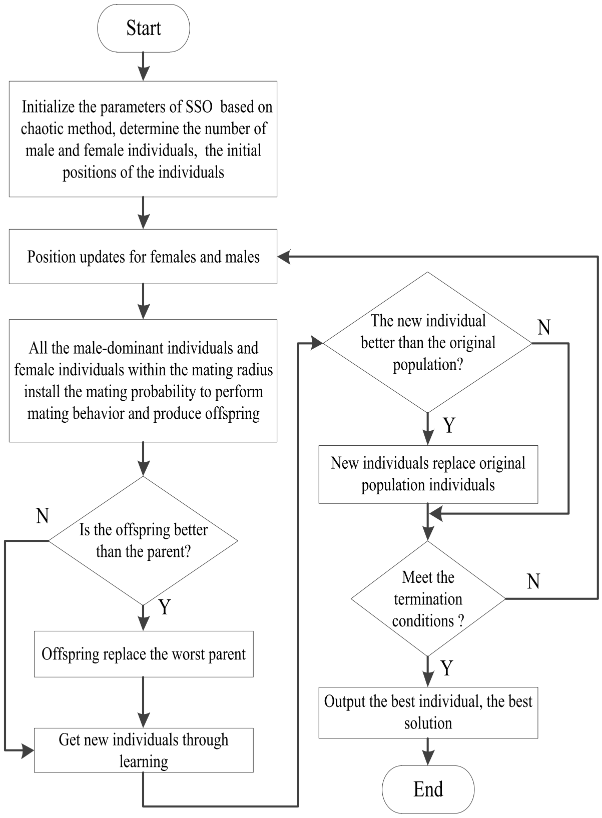

FIGURE 1. Flow chart of the proposed CSSO algorithm.

solution space, is updated via the group's learning of the best individual. To improve the global convergence speed of the SSO algorithm, the initial population is generated by the chaotic initialization method. The improved social spider algorithm is characterized by accelerated convergence and the increase of the diversity of the population in the subsequent search process, and improves the coverage of HWSNs. An optimal coverage strategy for HWSNs based on the SSO algorithm with chaotic optimization (CSSO) is proposed in this article. The maximum value of the objective function of the WSN coverage optimization is solved, the optimized coverage rate is output, and the distribution position of all sensor nodes in the area to be tested is determined after optimal deployment.

The specific experimental steps are as follows:

1) The coverage model parameters of the HWSNs sensor nodes are set, the initial sensor node position is generated, and the corresponding network coverage rate is calculated according to the objective function. For initialization, the overall initial spider population N, the number of iterations T_{max}, and the numbers of female and male spiders, namely N_f and N_m, respectively, are set. The probability factor is PF, and the number of subgroup divisions is E_{max};

2) *Fitness calculation*. Each individual spider is evaluated based on the location of the spider. The individual optimal coverage rate P_{best} and the group optimal coverage rate P_g are initialized for each individual spider;

3) Individuals in the female and male spider subgroups are arranged in descending order of fitness, and are divided into multiple groups;

4) *The renewal method of female spiders*. The female spiders mainly use vibration to attract or repel other individuals;

5) *The renewal method of male spiders*. Male individuals are arranged in descending order of the weight value;

6) The leader and non-leader individuals in the population are updated. Simultaneously, the male individuals who meet the mating conditions in the male population are updated, as are the leader individuals in the population;

7) The average best position of the population is calculated, the position of each spider operator is updated, and the coverage rate of the WSN after the update of the individual position of each spider is calculated according to the objective function f;

8) The coverage rate of each individual spider after the position is updated is compared with the coverage rate corresponding to the individual optimal value P_{best}, if the former is larger, P_{best} is updated;

9) A retrospective iterative update is performed, the spider population and optimal coverage are updated, and the optimal solution is output;

10) If the loop does not reach the preset maximum number of iterations, the process returns to step 2; otherwise, the algorithm ends, and the optimal solution is output.

Table 1 presents the pseudo-code of the implementation steps of coverage optimization for HWSNs based on the CSSO algorithm.

The coverage problem of HWSNs based on the improved social spider algorithm was optimized to improve the optimization efficiency and accuracy of the algorithm. The proposed CSSO algorithm was then simulated and verified by unimodal functions, multimodal functions, and network coverage experiments to prove the effectiveness and correctness of the improved algorithm. The proposed CSSO algorithm can reasonably allocate the resources of the entire WSN, reduce node redundancy, increase coverage, improve the monitoring quality and service quality of the network, reduce the energy consumption, and prolong the lifetime of the network.

VI. COMPARISON AND ANALYSIS OF ALGORITHM SIMULATION

A. SIMULATION ENVIRONMENT SETTINGS

To reflect the superior performance of the proposed CSSO algorithm, accurately test and improve the performance of the SSO algorithm, and ensure the fairness of the compared algorithm conditions, consistent parameters were set when simulating various algorithms. The proposed CSSO algorithm was compared with the PSO algorithm, monarch butterfly optimization (MBO), and basic social spider optimization (SSO) algorithms, and its performance was tested with unimodal and multimodal functions. Numerous experiments on the coverage performance of HWSNs were conducted to verify the superior performance of the proposed CSSO algorithm.

TABLE 1. Implementation steps of HWSNs coverage optimization based on CSSO algorithm.

Step	Cover optimization steps of HWSNs based on the CSSO algorithm
(1)	Set the coverage model parameters of the sensor nodes of HWSNs, generate the initial sensor node positions, and calculate the corresponding network coverage according to the objective function. Set the overall number of the initial spider population N, the initial iteration number t=1, the maximum iteration number T_{max}, the number of female and male spiders are N_f and N_m, respectively, the probability factor is PF, and the number of subgroup divisions is E_{max}.
(2)	Fitness calculation. Evaluate each individual spider based on its location.
(3)	While $t < T_{max}$ Sort all spider individuals according to their coverage fitness function, and update their positions according to equations (13) and (14). All the male-dominant individuals and female individuals within the mating radius install the mating probability to perform mating behavior and produce offspring. If the offspring is better than the parent, the offspring will replace the worst parent individual and get a new individual. If the new individual is better than the original population individual, the new individual will replace the original population individual.
(4)	Compare the coverage rate of each individual spider after the position is updated with the coverage rate corresponding to the individual optimal value P_{best}. If the former is larger, update P_{best}.
(5)	Output an optimal solution of the coverage of HWSNs.
(6)	Determine whether the termination conditions are met. If not satisfied, go to Step 3. If satisfied, stop. end while
(7)	Output the global optimal solution , end.

Regarding the simulation parameters for the investigation of the coverage of HWSNs, each algorithm was run 50 times to obtain an average value to reflect its coverage performance. It was assumed that 50 sensor nodes were randomly deployed in a monitoring area of 100×100 m^2 in size, and that the initial energy of the sensor node was 1 J. The heterogeneity of HWSNs considered in this article is mainly reflected in the sensing range of nodes. The sensing radius of each sensor node was set as a random number within the interval [5,15]. In the simulation environment set in this article, E_s is the sensing energy consumption of the sensor node, E_{N-R} is the energy consumption of the received data, received data energy consumption, E_{N-T} is the energy consumption of the transmitted data, F_a is the average transmission delay, and R_{gad} is the average delivery rate. $\varepsilon_s = 60 \times 10^{-9}$J/bit, $\varepsilon_r = 135 \times 10^{-9}$J/bit,, ξ_{mp} is a multipath transmission parameter, $\xi_{mp} = 0.0013 \times 10^{-12}$J/bit/m^4, ξ_{fs} is the ordinary space transmission parameter, $\xi_{fs} = 10 \times 10^{-12}$J/bit/m^2. $E_{elect} = 45 \times 10^{-9}$J/bit, the parameter l is the length of the transmitted packet data, $l = 4000$bit, the initial energy of the normal sensor node is 1J. The initial energy of the heterogeneous sensor node is 5J. The sensing range of common sensor node is 10m, and that of the heterogeneous sensor node is 15m. To compare and analyze the performance of the algorithms, consistent simulation conditions were used to randomly generate the initial location of the sensor node in the monitored area. The running environment of the experiment was an Intel Core TM i7-8700 3.2 GHz CPU with a 16 GB memory, Windows 10 operating system, and MATLAB R2017a simulation software.

The parameters of the PSO algorithm were set as follows: the learning factors were $C_1 = 2$ and $C_2 = 2$, the inertia weight factors were $\omega_1 = 0.9$ and $\omega_2 = 0.4$, the number of particles in the population of N_P was 50, and the maximum number of iterations T_{max} was set as 50. The parameters of the MBO algorithm were set as follows: the maximum step size S_{max} was 1.0, the butterfly adjustment rate BAR was 5/12, the migration period per was 1.2, the migration ratio p was 5/12, the butterfly population of N_m was 50, and the maximum number of iterations T_{max} was 50. Therefore, the numbers of monarch butterfly individuals in Land 1 and Land 2 were 21 and 29, respectively. The parameter settings of the CSSO algorithm were set as follows: the number of spider populations N_C was 50, the maximum number of iterations T_{max} was 50, and the probability factor PF was 0.5. The global search factor is σ, the parameter σ_{max} is the maximum parameter factor, the value of which was 0.9, and the parameter σ_{min} is the minimum parameter factor, the value of which was 0.1. The parameter settings of the whale optimization algorithm (WOA) were set in [22].

B. FUNCTION OBJECTIVE OPTIMIZATION

To verify that the improved CSSO algorithm exhibits better performance in terms of convergence and optimization speed, comparative experiments were conducted based on 10 test functions, as detailed in Table 2. For the selected 10 function optimization problems, if the algorithm was run for the maximum number of iterations, the algorithm was terminated. The three algorithms were independently employed to conduct 200 experiments for each optimization problem to avoid the influence of randomness on the experimental results, and the average of the 200 running results was recorded as the final value. Table 3 shows the experimental results of the four algorithms CSSO, SSO, PSO, and MBO after running 200 times independently on the multiple test functions.

TABLE 2. The basic information of the test function.

Function	Equation	Dimension	Bounds	Optimum
Sphere	$F01=\sum_{i=1}^{d} x_i^2$	30	[-100,100]	0
Schwefel's problem 2.22	$F02=\sum_{i=1}^{d} \lvert x_i \rvert + \prod_{i=1}^{d} \lvert x_i \rvert$	30	[-10,100]	0
Schwefel's problem 12	$F03=\sum_{i=1}^{d}\left(\sum_{j=1}^{i} x_j\right)^2$	30	[-100,100]	0
Schwefel's problem 2.21	$F04=\max\{\lvert x_i \rvert, 1 \le i \le d\}$	30	[-100,100]	0
Rosenbrock	$F05=\sum_{i=1}^{d-1}\left[100\left(x_{i+1}-x_i^2\right)^2+\left(x_i-1\right)^2\right]$	30	[-10,100]	0
Step	$F06=\sum_{i=1}^{d}\left(\lvert x_i+0.5 \rvert\right)^2$	30	[-100,100]	0
Quartic	$F07=\sum_{i=1}^{d} i x_i^4 + rand(0,1)$	30	[-1.28,1.28]	0
Alpine	$F08=\sum_{i=1}^{d}\lvert x_i \sin(x_i)+0.1x_i \rvert$	30	[-10,100]	0
Rastrigin	$F09=10d+\sum_{i=1}^{d}\left[x_i^2-10\cos(2\pi x_i)\right]$	30	[-5.12,5.12]	0
Ackley	$F10=-20\exp\left(-0.2\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right)-\exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(2\pi x_i)\right)+20+\exp(1)$	30	[-32,32]	0

TABLE 3. The function test results.

Function	CSSO			SSO			PSO			MBO		
	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std
F1	2.16E+02	4.26E+02	1.99E+02	5.15E+03	5.16E+03	4.88E+00	2.42E+02	5.57E+02	3.50E+00	3.30E+04	4.52E+04	5.75E+03
F2	1.04E+01	1.37E+01	2.29E+00	3.87E+01	2.26E+05	4.32E+03	5.50E+00	7.43E+00	1.94E+00	2.73E+03	8.81E+09	5.09E+10
F3	9.63E+02	1.56E+03	1.14E+03	1.10E+04	1.10E+04	1.34E+01	3.37E+03	3.56E+03	2.03E+02	4.80E+04	1.41E+06	1.26E+06
F4	9.04E+00	1.22E+01	3.94E+00	2.58E+01	2.59E+01	3.89E-02	1.03E+01	1.07E+01	4.65E-01	4.76E+01	4.80E+01	3.04E-01
F5	1.77E+03	5.04E+03	4.61E+03	8.98E+05	9.02E+05	1.87E+03	1.01E+04	3.83E+05	1.30E+06	1.31E+06	1.92E+06	2.99E+05
F6	6.36E+01	1.52E+02	2.56E+00	2.80E+03	2.80E+03	7.64E+01	2.07E+02	2.11E+02	2.20E+00	8.13E+03	9.32E+03	6.39E+02
F7	1.01E+00	4.14E+00	2.91E+00	1.38E+00	1.88E+00	2.88E-01	3.38E+00	2.48E+02	4.64E+02	1.72E+01	6.00E+01	2.67E+01
F8	2.78E+00	5.56E+00	1.57E+00	1.40E+01	1.45E+01	1.51E-01	3.93E+00	7.06E+00	3.32E+00	4.26E+00	6.07E+00	1.35E+00
F9	1.28E+02	1.69E+02	2.25E+01	1.37E+02	1.47E+02	1.99E-01	1.41E+02	2.38E+02	6.46E+01	1.55E+02	1.84E+02	1.77E+01
F10	3.70E+00	4.28E+00	1.85E-01	1.15E+01	1.15E+01	4.22E-03	3.88E+00	4.19E+00	4.24E-01	4.49E+00	4.84E+00	2.64E-01

Table 3 lists the optimal values, average values, and standard deviations of the CSSO, SSO, PSO, and MBO algorithms after independently being run 200 times. For a selected test function, the CSSO algorithm exhibited the strongest optimization performance, which was significantly better than those of the MBO, PSO, and SSO algorithms. For functions F1, F2, F3, F5, F7, F9, and F10, the optimal and average values of the CSSO algorithm were the best, the SSO exhibited the second-best performance, the PSO algorithm exhibited poor performance, and the MBO algorithm exhibited the worst performance. For function F6, the accuracy of the CSSO algorithm was improved by 2 as compared to the basic SSO algorithm, and the overall stability of the algorithm was better. For functions F4 and F8, the optimal values of the CSSO algorithm and the MBO algorithm were close to 0. However, in terms of stability, the optimization

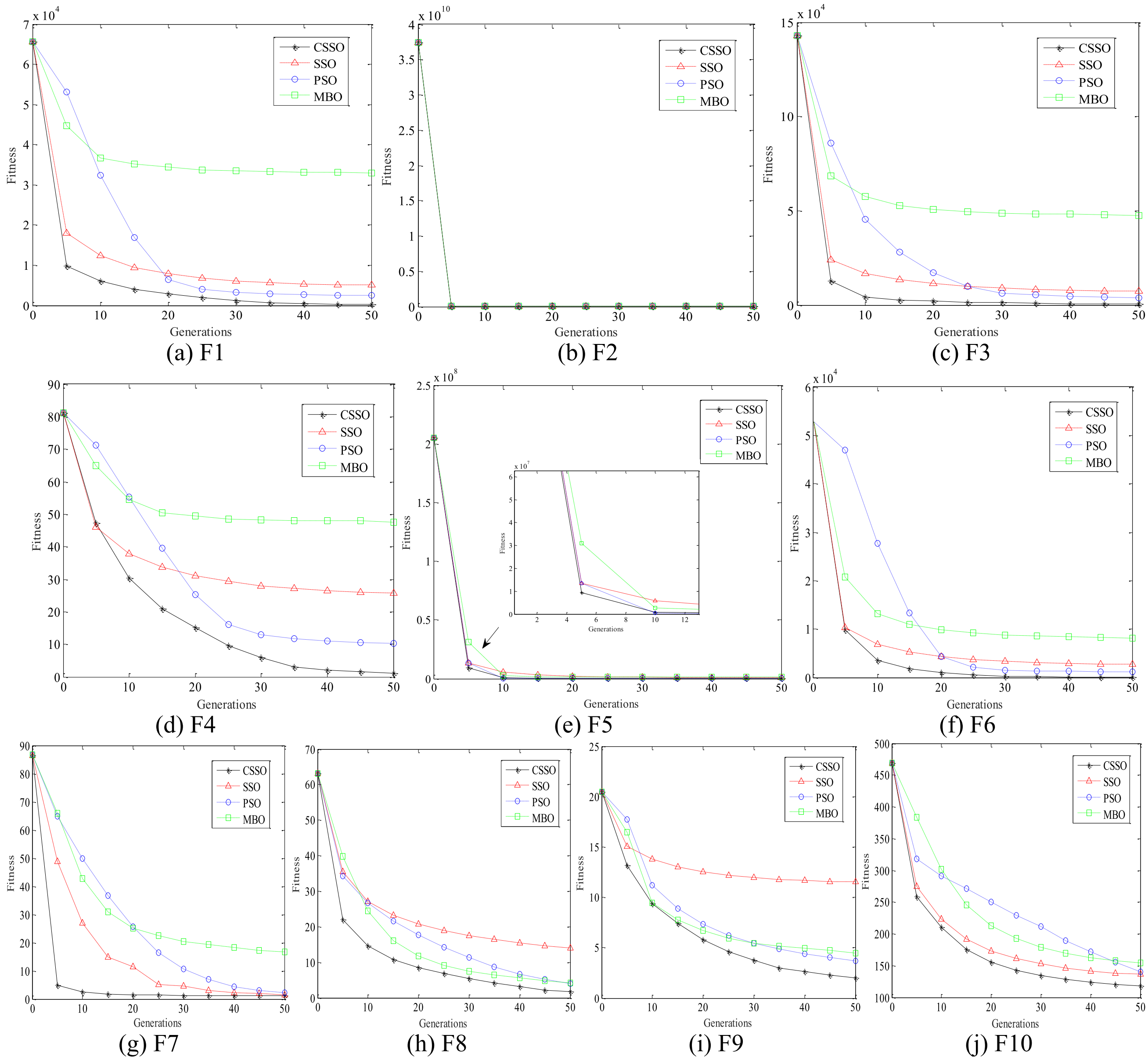

FIGURE 2. Comparison of convergence curves for function optimization.

performance of CSSO was far better than those of MBO and SSO, while the optimization effect of PSO was relatively poor. It is evident that the performance of the improved algorithm displayed obvious competitive advantages. To visually demonstrate the optimization performance of the CSSO algorithm, the convergence iteration curves of 10 benchmark test functions are presented in Figure 2.

From the 10 test function optimization convergence curves, it is evident that the improved CSSO algorithm was better than the PSO, MBO, and basic SSO algorithms in terms of the convergence speed. Functions F1, F3, F4, F5, F6, F7, and F8 are unimodal functions, which can more easily reach the optimal value; thus, they are often used to test the convergence ability of algorithms. From the function convergence curves, it can intuitively be seen that the improved CSSO algorithm achieved higher convergence accuracy and overcame the problem of low accuracy in the basic SSO algorithm. Moreover, there were multiple inflection points in the iterative process, which proves that the improved algorithm easily jumped out of the local optimum and converged to the optimum value at a faster speed. Compared with the basic SSO algorithm, the improved CSSO algorithm displayed improved optimization accuracy. For the multimodal functions F9 and F10, the improved CSSO algorithm had a faster convergence rate than the PSO, MBO, and basic SSO algorithms, and the optimal solution found was the best, as was the performance. In conclusion, it can be seen from the convergence graph that the convergence of the

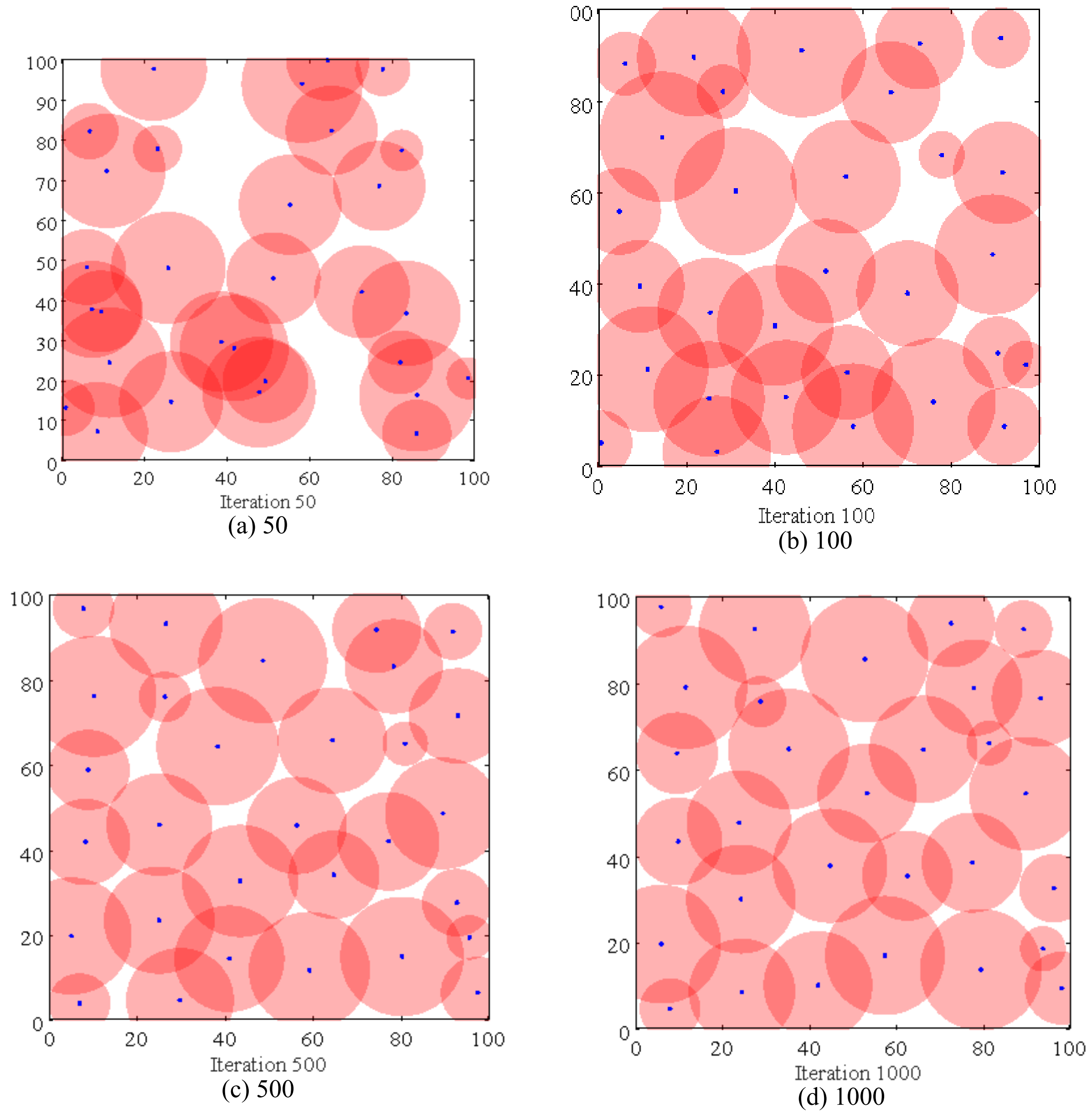

FIGURE 3. **Coverage effect of PSO.**

CSSO algorithm was better than those of the PSO, MBO, and basic SSO algorithms.

C. SIMULATION COMPARISON AND ANALYSIS

1) COMPARISON OF THE ALGORITHM COVERAGE EFFECTS

To verify the coverage effect of the proposed CSSO algorithm, the optimal simulated coverage results of the PSO, MBO, basic SSO, and proposed CSSO algorithms at different iteration times are respectively shown in Figures 3-6. In the simulation diagrams, a square is used to indicate the area of 100 × 100 m^2 to be detected by the HWSN, the symbol "■" is used to indicate the location of a single heterogeneous sensor node, and a circle is used to indicate the range in the monitoring area that the heterogeneous sensor node can cover. The perception radius of each heterogeneous sensor node was set as a random number within the interval [5,15].

As can be seen from Figures 3-6, as the number of iterations increased, the coverage of the HWSNs of the four algorithms all increased. After 1000 iterations, the coverage of the HWSNs of the PSO algorithm was poor, that of the MBO algorithm was average, that of the SSO algorithm was better, and that of the CSSO algorithm was the best. After 100 iterations, the network coverage effects of the four algorithms had overlapping areas. However, after 1000 iterations, the proposed CSSO algorithm had the least overlapping areas of the coverage of different nodes, whereas the

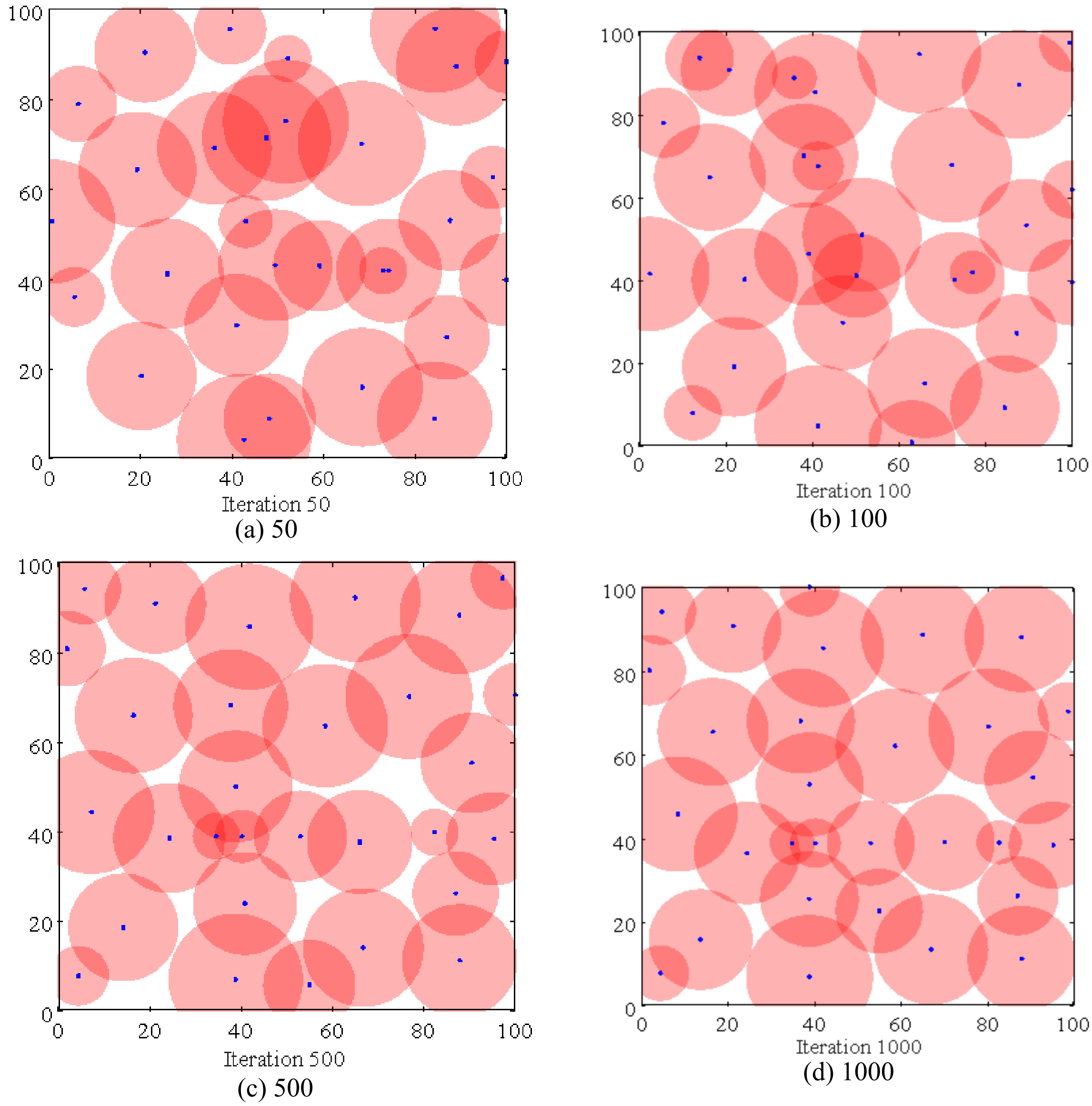

FIGURE 4. Coverage effect of MBO.

PSO algorithm had the most. The coverage area of the nodes of the MBO algorithm was more overlapped, while that of the nodes of the SSO algorithm was less overlapped. The proposed CSSO algorithm therefore improved the coverage effect of the nodes, while also improving the working efficiency of the nodes and reducing the cost of the network.

2) COMPARISON OF NETWORK COVERAGE UNDER DIFFERENT NUMBER OF NODES

To increase the experimental effect, the coverage effect of the network was increased under different numbers of nodes. Figures 8-10 respectively present the network coverage effects of 30, 40, and 50 sensor nodes in the case of 1000 iterations. In addition, the coverage rates of networks with 30, 40, and 50 nodes were compared, as shown in Figure 11.

As can be seen from Figures 8-10, as the number of iterations increased, the network coverage of the four algorithms gradually increased. The network coverage rates of the four algorithms after 100 iterations are provided as a reference. The network coverage rate of 30 nodes was greater than 0.65, that of 40 nodes was greater than 0.68, and that of 50 nodes was greater than 0.72. After 1000 iterations were completed, the network coverage of 30 nodes was greater than 0.85, that of 40 nodes was greater than 0.58, and that of 50 nodes was greater than 0.92. In addition, it can be seen from Figures 8-11 that regardless of whether the number of nodes was 30, 40, or 50, the network coverage of the MBO algorithm was the lowest, that of the PSO algorithm was the second-lowest, that of the SSO algorithm was high, and that of the proposed CSSO algorithm was the highest. The proposed CSSO

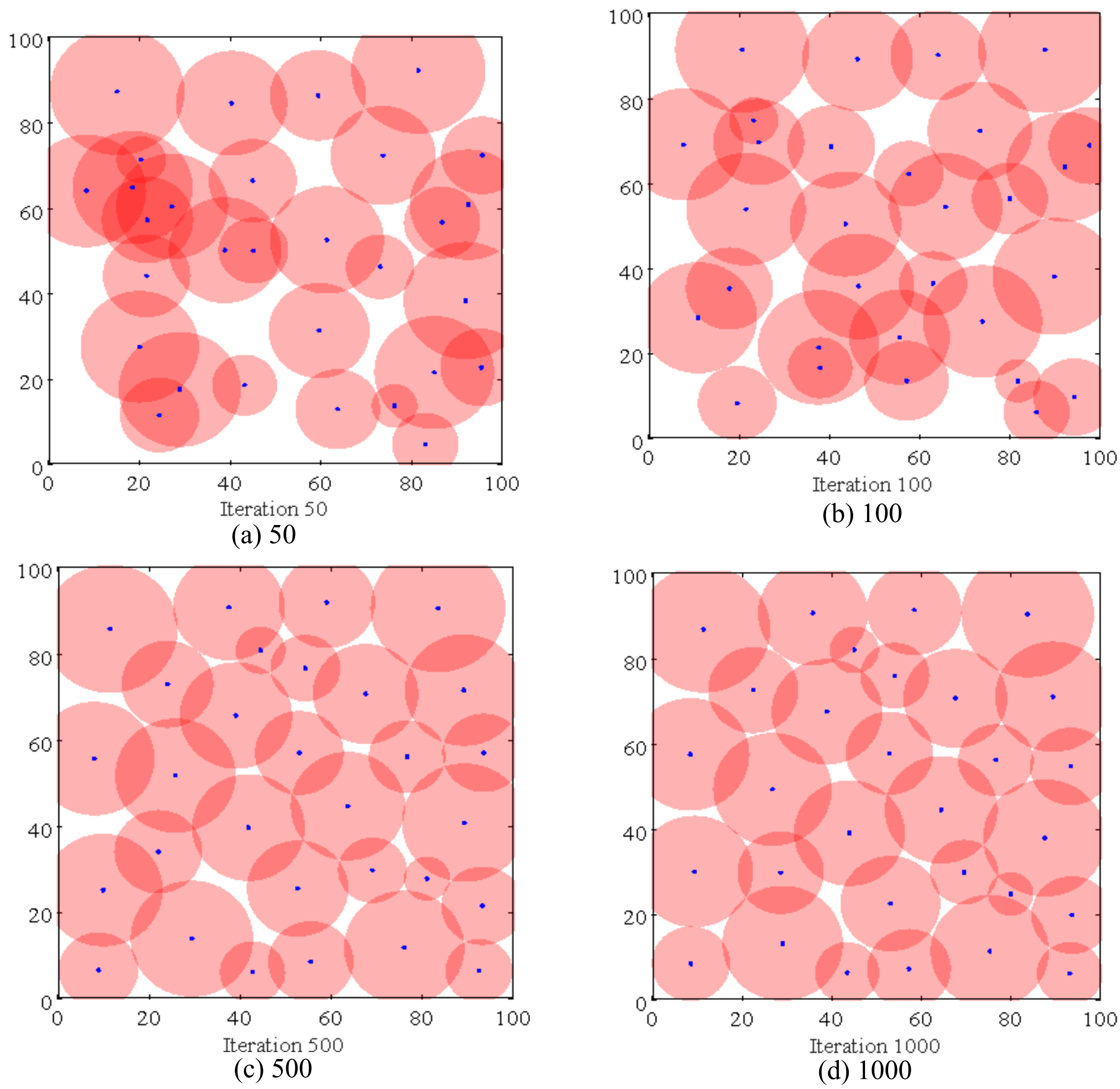

FIGURE 5. **Coverage effect of WOA.**

algorithm improved the coverage effect of the nodes, while also improving the working efficiency of the nodes and reducing the cost of the network. Taking the number of sensor nodes shown in Figure 11 as an example, the network coverage rate of the MBO algorithm was 87.8%, that of the PSO algorithm was 89.1%, that of the SSO algorithm was 92.9%, and that of the proposed CSSO algorithm was 93.9%. The main reason why the CSSO algorithm proposed in this article is superior to the other three algorithms is that the CSSO algorithm proposed in this article has a faster speed of finding the optimal solution, jumping out of the local optimal solution, and finding the optimal solution better than the other three algorithms with higher accuracy.

3) COMPARISON OF NETWORK CONNECTIVITY

For HWSNs, continuous motion discretization is generally used to calculate the connectivity of the network, i.e., the network structure remains unchanged in a short period of time. For a network at a certain moment, the calculation of the connectivity rate of an HWSN is generally determined by the traversal method of the sensor nodes. Assuming that a sensor node is used as a reference, the nodes connected from its one-hop, two-hop, and three-hop steps are sequentially searched until the number of nodes connected with the initial sensor node no longer increases. The mathematical calculation equation of the connection rate N_{con} is as follows.

$$N_{con} = \frac{N_l}{n} \tag{20}$$

The parameter N_l is the number of adjacent nodes within the communication range of the node, and the parameter n is the total number of nodes in the entire sensor network. The comparison of the network connectivity of the four network coverage methods is presented in Figure 12.

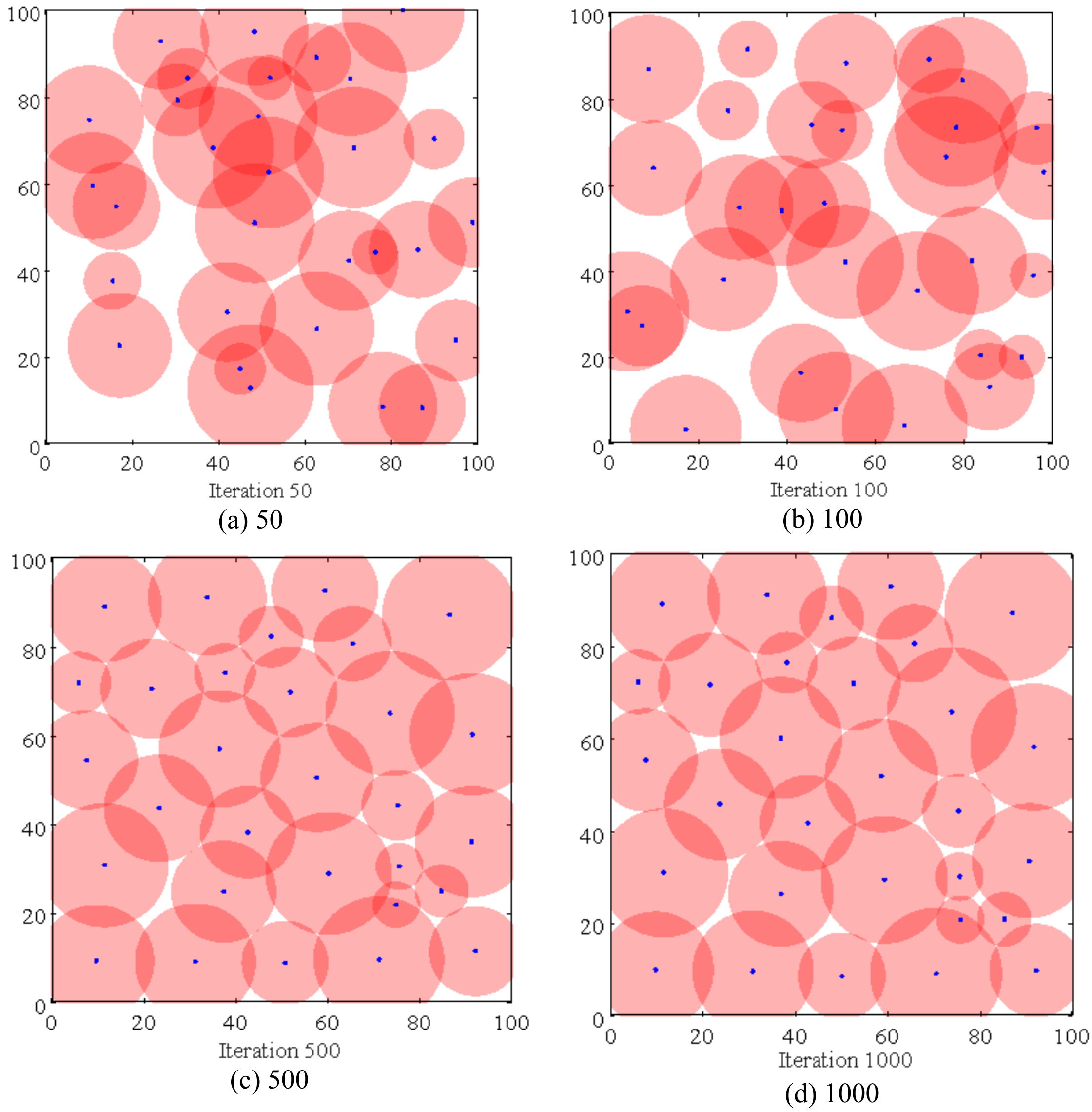

FIGURE 6. **Coverage effect of SSO.**

It can be seen from Figure 12 that as the number of simulation iterations increased, the network connectivity of the four algorithms gradually decreased. The MBO algorithm exhibited the lowest network connectivity rate; its decline was large, and the average value was 0.41. The network connectivity rate of the PSO algorithm was the second-lowest; it also had a large decline, and the average value was 0.47. The network connectivity rate of the SSO algorithm was relatively high; its decline was low, and the average value was 0.65. Finally, the proposed CSSO algorithm achieved the highest network connectivity performance and the lowest decline, and the average value was 0.78. Overall, the proposed CSSO algorithm achieved the best network connectivity performance.

4) COMPARISON OF NETWORK RELIABILITY

The reliability of the network is one of the important indicators of the coverage performance of HWSNs. The network reliability R_{net} is composed of the network connectivity reliability Q_1, network connectivity rate Q_2, and network capacity Q_3. The mathematical equation is as follows [30].

$$R_{net} = 0.17Q_1 + 0.5Q_2 + 0.33Q_3 \tag{21}$$

The network connectivity reliability Q_1 refers to the reliability of end-to-end node interconnection. Generally, the reliability matrix is calculated based on the distance between the sensor nodes of an HWSN. Then, the average node connectivity reliability value after 100 rounds is determined by referring to the obtained reliability matrix and random edge reliability matrix samples, and according to a Monte Carlo analysis. The parameter Q_2 is the connectivity rate of the network. The network capacity Q_3 is the network survival probability, which is generally considered the ratio of the current network node survival nodes to the total number of nodes in the network. The comparison of the network reliability of the four algorithms is presented in Figure 13.

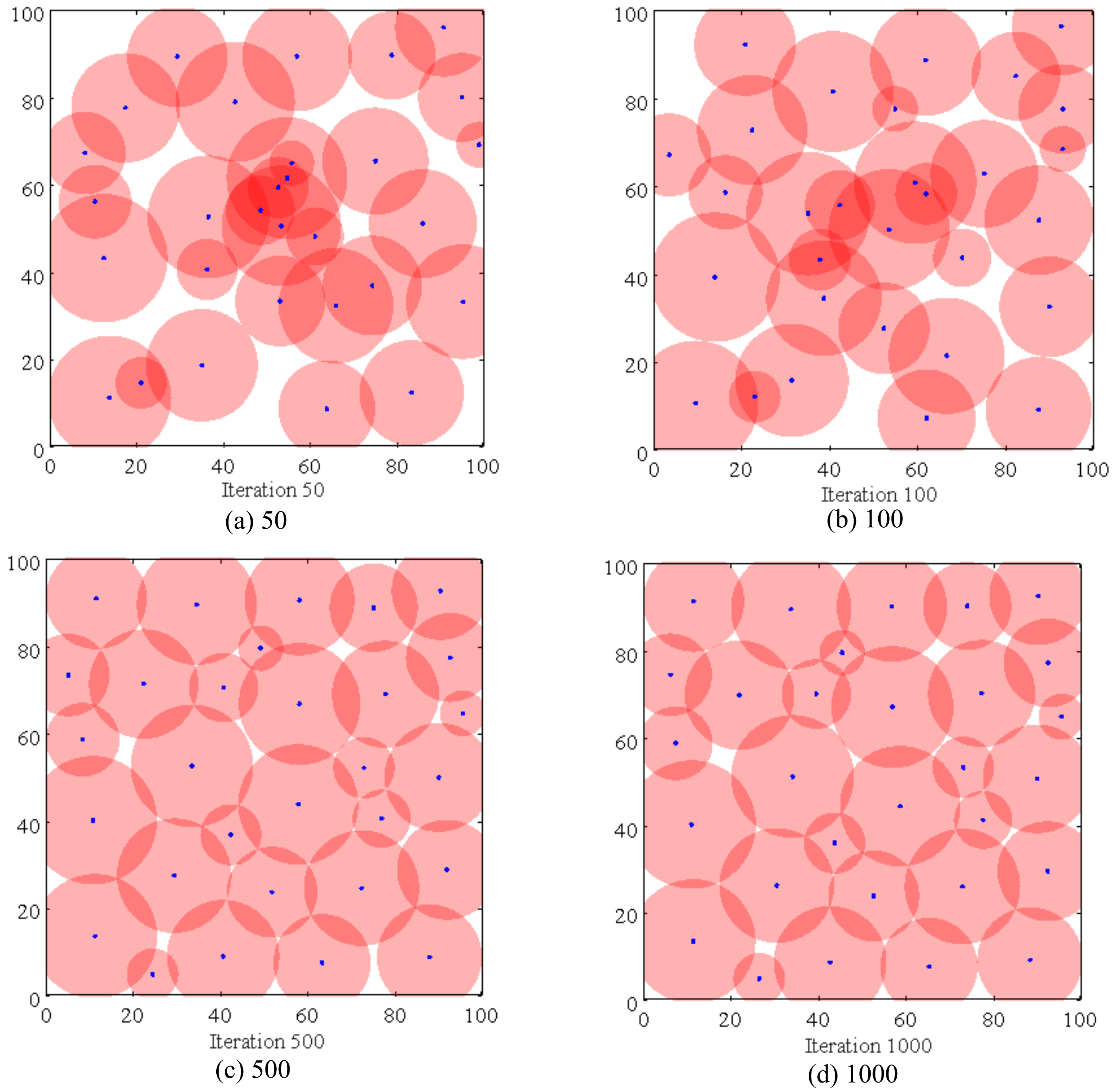

FIGURE 7. **Coverage effect of CSSO under different number of nodes.**

It can be seen from Figure 13 that with the increase in the number of iterations, the network reliability of the four algorithms gradually decreased. The network reliability of the MBO algorithm had the largest decline, and the average value was 0.52. The network reliability of the PSO algorithm had a large decline, and the average value was 0.63. The network reliability of the SSO algorithm presented a slow decline in the first 60 rounds, and a larger decline after 60 rounds; this is mainly because the energy consumption of the network was greater after 60 iterations. Moreover, the average reliability of the network was 0.66. The network reliability of the proposed CSSO algorithm exhibited the lowest decline, and the network achieved the highest average reliability of 0.82. It is therefore evident that the proposed CSSO algorithm exhibited the highest network reliability. The reason why the CSSO algorithm proposed in this article has better network connectivity and reliability performance than the other three algorithms is that the CSSO proposed in this article has higher algorithm optimization accuracy in the process of network connectivity and reliability optimization. In the optimal coverage optimization problem of node deployment, as the number of simulation polling increases, the comprehensive consideration of network coverage, work efficiency and load balance of our proposed algorithm is stronger than the performance of the other three algorithms.

5) COMPARISON OF THE THREE-DIMENSIONAL NETWORK ENERGY CONSUMPTION OF THE FOUR ALGORITHMS

The problem of network energy consumption has always been a popular topic and concern of HWSN research, and also restricts the research progress of network coverage. By reasonably deploying nodes in the network, coupled with

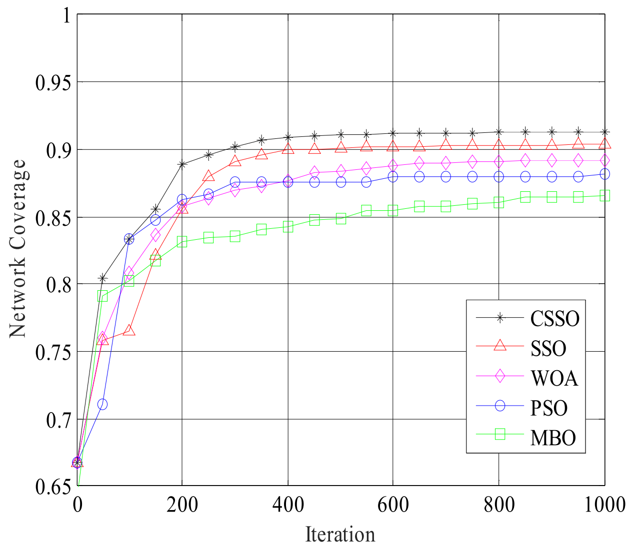

FIGURE 8. Coverage effect of 30 nodes.

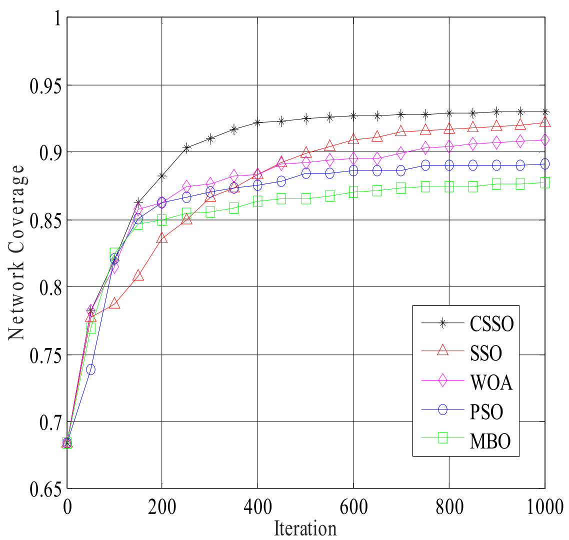

FIGURE 9. Coverage effect of 40 nodes.

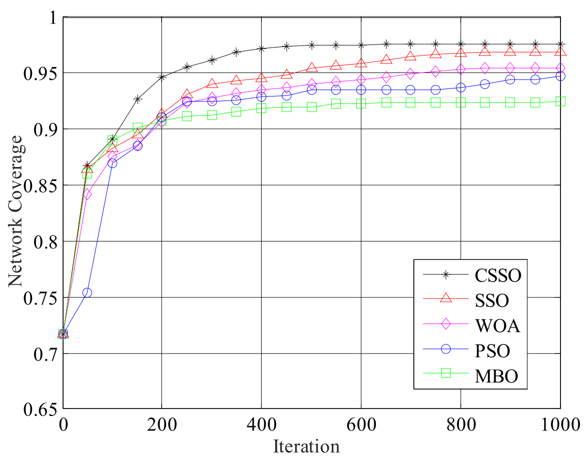

FIGURE 10. Coverage effect of 50 nodes.

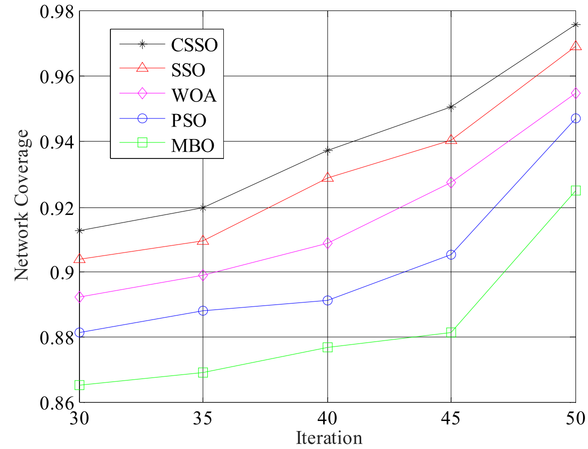

FIGURE 11. Comparison of coverage.

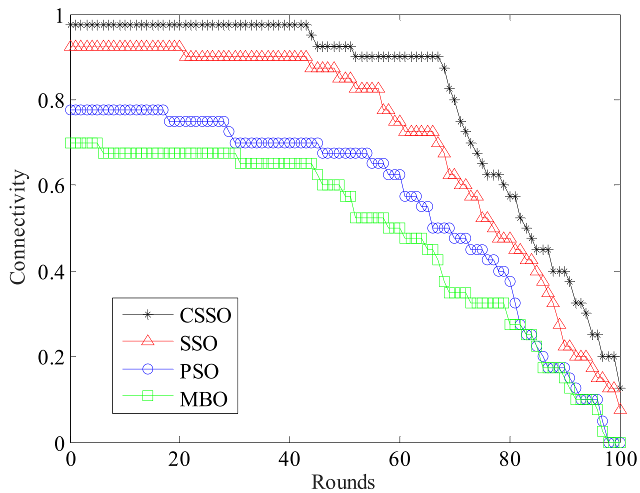

FIGURE 12. Comparison of network connectivity.

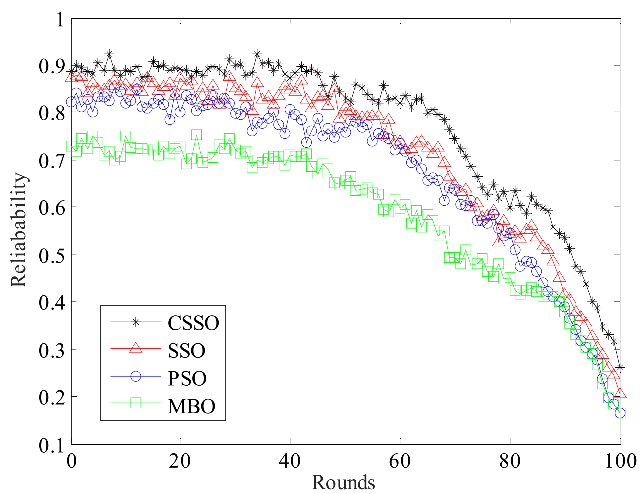

FIGURE 13. Comparison of network reliability.

node sleep/wakeup control, power control, etc., the network energy utilization rate can be maximized, and the realization of energy balance can contribute to the achievement of the optimal coverage performance of the network. The comparison of the three-dimensional network energy consumption

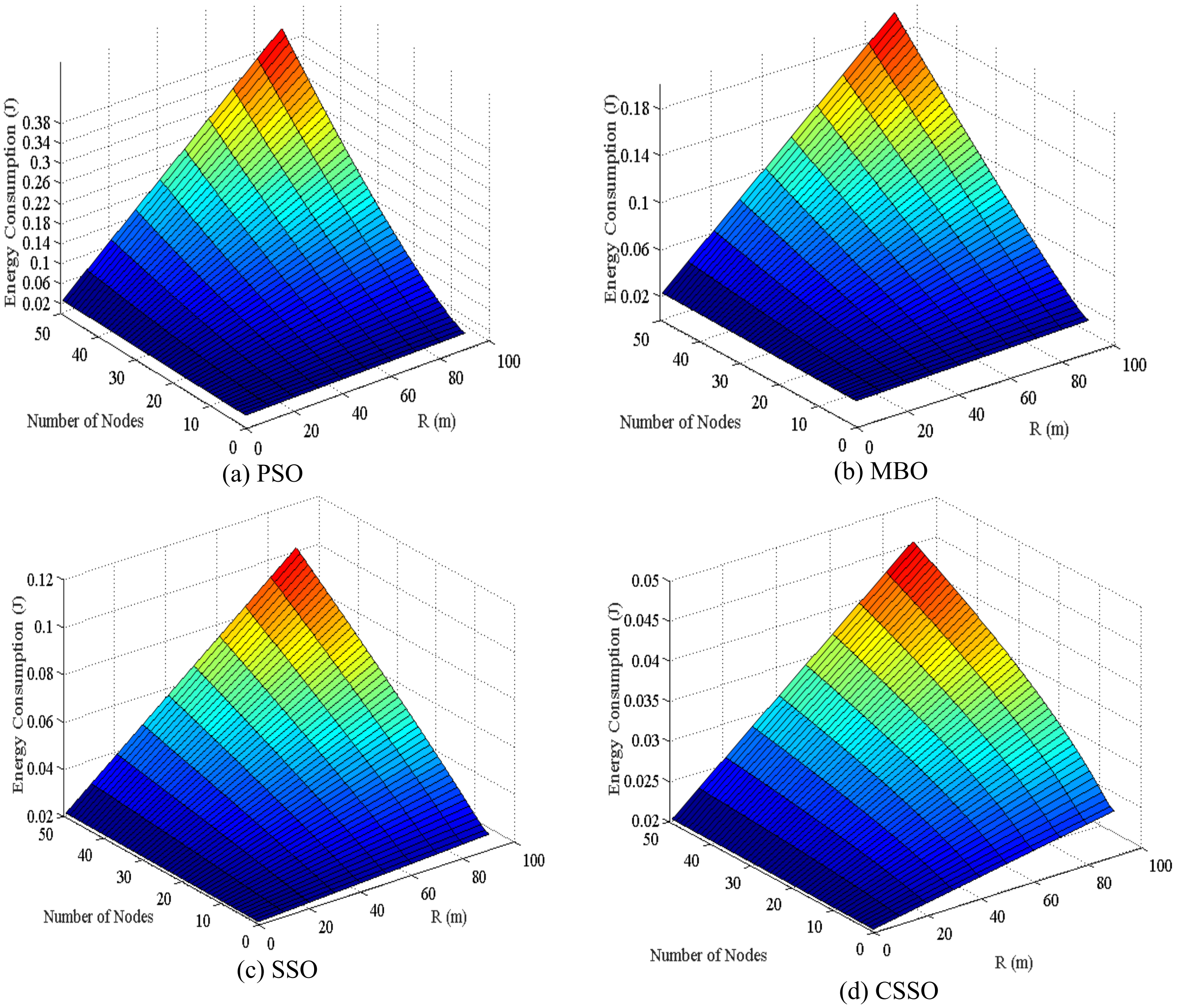

FIGURE 14. Comparison of the energy consumption of the four algorithms.

of the PSO, MBO, SSO, and CSSO algorithms is presented in Figure 14.

It can be seen from Figure 14 that under the same number of sensor nodes and communication radius, the energy consumption of the four algorithms was relatively low. As the number of nodes and communication radius increased, the energy consumption of the networks gradually increased. The network energy consumption of the PSO algorithm was the highest; most of the nodes exhibited high energy consumption (as high as 0.38 J), and the energy consumption was unbalanced. The network energy consumption of the MBO algorithm was relatively high, as was the energy consumption of most nodes (as high as 0.18 J), and the energy consumption was relatively balanced. The network energy consumption of the basic SSO algorithm was relatively low, and the maximum energy consumption of the network nodes reached 0.12J. The proposed CSSO algorithm achieved the lowest network energy consumption, and the energy consumption of the nodes was also relatively balanced, with the highest energy consumption of 0.02 J. Overall, the proposed CSSO algorithm exhibited the lowest average energy consumption.

6) COMPARISON OF THE SIMULATION TIME OF THE ALGORITHMS

The simulation time of an algorithm reflects the calculation speed of the algorithm, and is also one of the important indicators of network performance. The comparison of the simulation time of the four algorithms in the cases of 30, 40, and 50 sensor nodes is shown in Figure 15.

It can be seen from Figure 15 that when the number of sensor nodes was 30, the network time consumption of the four algorithms was relatively short, and the PSO algorithm took the shortest time, namely 14.4 s. The proposed CSSO algorithm took 16.2 s, the SSO algorithm took 16.4 s, and the

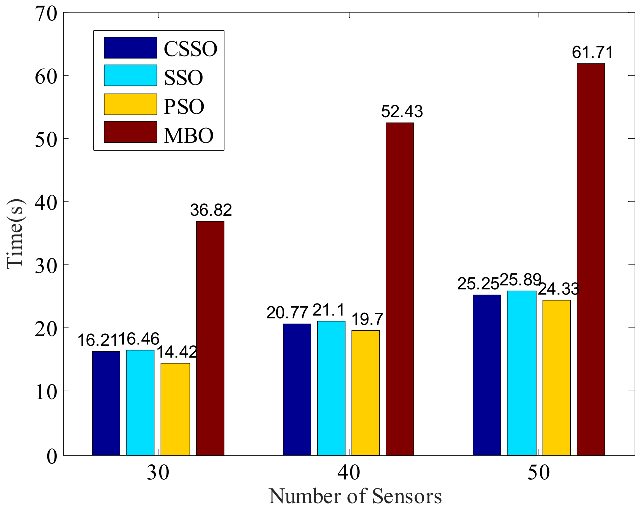

FIGURE 15. Comparison of the simulation time.

MBO algorithm took the longest time, namely 36.8 s. When there were 50 sensor nodes, the network time consumption of the four algorithms was relatively long, and the PSO algorithm took the shortest time, namely 24.3 s. The proposed CSSO algorithm took 25.2 s, the SSO algorithm took 25.9 s, and the MBO algorithm took the longest time, namely 61.7 s. From the comparison of the simulation time consumption of the four algorithms, it is evident that the simulation of the proposed CSSO algorithm did not take long and can fully meet the needs of the actual application environment.

Based on the simulation results, it is evident that due to the introduction of the chaos mechanism in the proposed CSSO algorithm, the diversity of the SSO algorithm was enhanced, and the global optimization capability and convergence speed of the algorithm were improved. The CSSO algorithm was found to exhibit excellent performance in the optimization of the deployment of heterogeneous wireless sensor nodes, and achieved the best coverage effect. Therefore, the proposed CSSO algorithm has strong adaptability and a fast optimization speed, and the application of this algorithm to the coverage optimization problem of HWSNs can significantly improve the network coverage rate, thereby reducing coverage blind spots.

VII. CONCLUSION

This article investigated the optimal coverage of the sensor nodes in HWSNs. With the goals of reducing network costs and improving network coverage and reliability, an optimal coverage method for HWSNs based on an improved SSO algorithm was proposed. To improve the global convergence speed of the algorithm, a chaotic initialization method is used to generate the initial population. In addition, the original SSO algorithm has a poor convergence speed and search ability. By improving the neighborhood search, global search, and matching radius, a better convergence speed and search ability can be achieved. In the iterative optimization of the algorithm, the optimal solution is ultimately obtained by simulating the movement of a spider colony, i.e., according to the cooperation, mutual attraction, and mating process of male and female spiders. The simulation results demonstrate that the proposed improved CSSO algorithm has a strong ability to jump out of the local optimum and quickly converge to the optimal solution. On this basis, with the optimization goals of improving network coverage and reducing network costs, the optimal deployment plan of nodes is searched via intelligent optimization, thereby effectively preventing coverage blind spots and coverage redundancy in the network.

Although the proposed CSSO algorithm was found to achieve the improved coverage performance and efficiency of HWSNs, some sensor nodes in some regions are too clustered during the application process. In future research, HWSNs should be covered more evenly, and the area where nodes gather should be reduced. Moreover, in the future, the coverage of mobile WSNs should be considered, and their optimal coverage should be investigated.

REFERENCES

[1] Y. Sun, L. Zhang, G. Feng, B. Yang, B. Cao, and M. A. Imran, "Blockchain-enabled wireless Internet of Things: Performance analysis and optimal communication node deployment," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5791–5802, Jun. 2019.

[2] Y.-G. Yue and P. He, "A comprehensive survey on the reliability of mobile wireless sensor networks: Taxonomy, challenges, and future directions," *Inf. Fusion*, vol. 44, pp. 188–204, Nov. 2018.

[3] S. Chakraborty, N. K. Goyal, S. Mahapatra, and S. Soh, "A Monte-Carlo Markov chain approach for coverage-area reliability of mobile wireless sensor networks with multistate nodes," *Rel. Eng. Syst. Saf.*, vol. 193, Jan. 2020, Art. no. 106662.

[4] X. Fu, H. Yao, and Y. Yang, "Modeling and analyzing cascading dynamics of the clustered wireless sensor network," *Rel. Eng. Syst. Saf.*, vol. 186, pp. 1–10, Jun. 2019.

[5] S. Verma, N. Sood, and A. K. Sharma, "Genetic algorithm-based optimized cluster head selection for single and multiple data sinks in heterogeneous wireless sensor network," *Appl. Soft Comput.*, vol. 85, Dec. 2019, Art. no. 105788.

[6] R. Priyadarshi, P. Rawat, and V. Nath, "Energy dependent cluster formation in heterogeneous wireless sensor network," *Microsyst. Technol.*, vol. 25, no. 6, pp. 2313–2321, Jun. 2019.

[7] I. Ullah, S. Qian, Z. Deng, and J.-H. Lee, "Extended Kalman filter-based localization algorithm by edge computing in wireless sensor networks," *Digit. Commun. Netw.*, early access, Aug. 16, 2020, doi: 10.1016/j.dcan.2020.08.002.

[8] S. Yu, J. Liu, J. Wang, and I. Ullah, "Adaptive double-threshold cooperative spectrum sensing algorithm based on history energy detection," *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–12, Jun. 2020.

[9] I. Ullah, J. Chen, X. Su, C. Esposito, and C. Choi, "Localization and detection of targets in underwater wireless sensor using distance and angle based algorithms," *IEEE Access*, vol. 7, pp. 45693–45704, 2019.

[10] V. Nehra, A. K. Sharma, and R. K. Tripathi, "I-DEEC: Improved DEEC for blanket coverage in heterogeneous wireless sensor networks," *J. Ambient Intell. Humanized Comput.*, vol. 11, no. 9, pp. 3687–3698, Sep. 2020.

[11] P. Gou, G. Mao, F. Zhang, and X. Jia, "Reconstruction of coverage hole model and cooperative repair optimization algorithm in heterogeneous wireless sensor networks," *Comput. Commun.*, vol. 153, pp. 614–625, Mar. 2020.

[12] Z. Wang, Y. Chen, and B. Liu, "A sensor node scheduling algorithm for heterogeneous wireless sensor networks," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 1, 2019.

[13] M. A. Benatia, M. Sahnoun, D. Baudry, A. Louis, A. El-Hami, and B. Mazari, "Multi-objective WSN deployment using genetic algorithms under cost, coverage, and connectivity constraints," *Wireless Pers. Commun.*, vol. 94, no. 4, pp. 2739–2768, Jun. 2017.

[14] Z. Wang, H. Xie, D. He, and S. Chan, "Wireless sensor network deployment optimization based on two flower pollination algorithms," *IEEE Access*, vol. 7, pp. 180590–180608, 2019.
[15] A. K. Sangaiah, M. Sadeghilalimi, A. A. R. Hosseinabadi, and W. Zhang, "Energy consumption in point-coverage wireless sensor networks via bat algorithm," *IEEE Access*, vol. 7, pp. 180258–180269, 2019.
[16] O. M. Alia and A. Al-Ajouri, "Maximizing wireless sensor network coverage with minimum cost using harmony search algorithm," *IEEE Sensors J.*, vol. 17, no. 3, pp. 882–896, Feb. 2017.
[17] X.-Q. Zhao, Y.-P. Cui, C.-Y. Gao, Z. Guo, and Q. Gao, "Energy-efficient coverage enhancement strategy for 3-D wireless sensor networks based on a vampire bat optimizer," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 325–338, Jan. 2020.
[18] Z. Hao, N. Qu, and X. Dang, "Node optimization coverage method under link model in passive monitoring system of three-dimensional wireless sensor network," *Int. J. Distrib. Sensor Netw.*, vol. 15, no. 8, 2019.
[19] J. Wang, C. Ju, H.-J. Kim, R. S. Sherratt, and S. Lee, "A mobile assisted coverage hole patching scheme based on particle swarm optimization for WSNs," *Cluster Comput.*, vol. 22, no. 1, pp. 1787–1795, Jan. 2019.
[20] H. A. Hashim, B. O. Ayinde, and M. A. Abido, "Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm," *J. Netw. Comput. Appl.*, vol. 64, pp. 239–248, Apr. 2016.
[21] Y. Feng, S. Zhao, and H. Liu, "Analysis of network coverage optimization based on feedback K-means clustering and artificial fish swarm algorithm," *IEEE Access*, vol. 8, pp. 42864–42876, 2020.
[22] L. Wang, W. Wu, J. Qi, and Z. Jia, "Wireless sensor network coverage optimization based on whale group algorithm," *Comput. Sci. Inf. Syst.*, vol. 15, no. 3, pp. 569–583, 2018.
[23] E. Cuevas, M. Cienfuegos, D. Zaldívar, and M. Pérez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider," *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6374–6384, Nov. 2013.
[24] T. T. Nguyen, "A high performance social spider optimization algorithm for optimal power flow solution with single objective optimization," *Energy*, vol. 171, pp. 218–240, Mar. 2019.
[25] G. Zhang and K. Xing, "Memetic social spider optimization algorithm for scheduling two-stage assembly flowshop in a distributed environment," *Comput. Ind. Eng.*, vol. 125, pp. 423–433, Nov. 2018.
[26] M. A. E. Aziz and A. E. Hassanien, "An improved social spider optimization algorithm based on rough sets for solving minimum number attribute reduction problem," *Neural Comput. Appl.*, vol. 30, no. 8, pp. 2441–2452, Oct. 2018.
[27] C. E. Klein, E. H. V. Segundo, V. C. Mariani, and L. D. S. Coelho, "Modified social-spider optimization algorithm applied to electromagnetic optimization," *IEEE Trans. Magn.*, vol. 52, no. 3, pp. 1–4, Mar. 2016.
[28] A. Y. Husodo, G. Jati, A. Octavian, and W. Jatmiko, "Enhanced social spider optimization algorithm for increasing performance of multiple pursuer drones in neutralizing attacks from multiple evader drones," *IEEE Access*, vol. 8, pp. 22145–22161, 2020.
[29] A. Fathy, K. Kaaniche, and T. M. Alanazi, "Recent approach based social spider optimizer for optimal sizing of hybrid PV/wind/battery/diesel integrated microgrid in Aljouf region," *IEEE Access*, vol. 8, pp. 57630–57645, 2020.
[30] Y. Lyu and P. Yin, "Internet of Things transmission and network reliability in complex environment," *Comput. Commun.*, vol. 150, pp. 757–763, Jan. 2020.

LI CAO received the B.S. degree in electrical automation from the North University of China, in 2009, the M.S. degree in pattern recognition and intelligent systems from the Sichuan University of Science and Engineering, in 2013, and the Ph.D. degree in control science and engineering from the Southwest University of Science and Technology, China, in 2020. Since 2020, she has been with the School of Mechanical and Electrical Engineering, Oujiang College, Wenzhou University. Her research interests include wireless sensor networks, signal processing, and the Internet of Things.

YINGGAO YUE received the B.S. degree in automation from the Wuhan Institute of Technology, in 2009, the M.S. degree in pattern recognition and intelligent systems from the Sichuan University of Science and Engineering, in 2012, and the Ph.D. degree in instrument science and technology from Southeast University, China, in 2016. Since 2020, he has been with the School of Mechanical and Electrical Engineering, Oujiang College, Wenzhou University, where he is currently an Associate Professor. His research interests include wireless communication, sensor networks, and the Internet of Things.

YONG CAI received the Ph.D. degree in electrical system control and information control from Southwest Jiaotong University, China, in 2006. Since 2005, he has been the Dean and a Professor with the School of Manufacturing Science and Engineering, Southwest University of Science and Technology. His research interests include mechanical engineering, rapid manufacturing, control information technology, and signal processing.

YONG ZHANG received the B.S. degree in electronic information engineering from the Huazhong University of Science and Technology, in 2003, the M.S. degree in communication and information system from the Wuhan Research Institute of Posts and Telecommunications, in 2007, and the Ph.D. degree in communication and information system from the Wuhan University of Technology, Wuhan, China, in 2017. Since 2018, he has been with the Hubei University of Arts and Science, where he is currently a Lecturer. His research interests include wireless networks, evolutionary computing, and mobile edge computing.