Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

An Improved Sparrow Search Algorithm for Deployment Problem in Wireless Sensor Network

Jiazu Xie
Tianjin Normal University
Cuiping Zhang (zcptjnu@163.com)
Tianjin Normal University https://orcid.org/0000-0001-6939-5889

Research

Keywords: wireless sensor network, deployment, sparrow search algorithm, swarm optimization

Posted Date: July 29th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1852768/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

An Improved Sparrow Search Algorithm for Deployment Problem in Wireless Sensor Network

Jiazu Xie[a,b],Cuiping Zhang[a,b*1]

[a]Tianjin Key Laboratory of Wireless Mobile Communications and Power Transmission, Tianjin Normal University, Tianjin 300387, China
[b]College of Electronic and Communication Engineering, Tianjin Normal University, Tianjin 300387, China

Abstract When designing and deploying wireless sensor network (WSN), sensor node deployment is one of the crucial issues for WSN providing high quality of service. Sensor node deployment could also affect network coverage, network connectivity, network lifetime, and energy consumption. This paper attempts to solve sensor node deployment problem in WSN. A mathematical model is constructed by considering coverage maximization, deployment strategy and other constraints. As the problem is non-deterministic polynomial (NP) complete, a new method called sparrow search algorithm (SSA) is used. A novel population structure is proposed to improve SSA to solve the problem. Two populations with central symmetry property are created in initialization. Crossover operator is used to exchange information of the two populations. The computational complexity of the improved SSA algorithm is also discussed. Numerical simulations are performed to solve the deployment problem in WSN. The improved SSA algorithm is tested and compared with other algorithms. The results show that the proposed method is useful to address NP complete sensor node deployment problem in WSN.

Keywords wireless sensor network, deployment, sparrow search algorithm, swarm optimization

1 Introduction

Wireless sensor network (WSN) is comprised of a number of wireless sensors which could collect data and transmit to base station or data center. Some wireless sensors could also have the ability to preprocess the collected data. With the development of microelectronic technology, wireless communication technology, and computing technology, the functionality of wireless sensors has been greatly enhanced [1], [2]. Nowadays, WSN has been used in the Fifth Generation (5G) and Internet of Things (IoT). The commercial usage brings convenience to human lives by connecting logical information world and physical world. These applications, like house condition monitoring, environmental monitoring, medical healthy monitoring, etc., has made WSN a hot research topic in the past decades [3], [4].

WSN generally consists of distributed wireless sensor nodes, sink nodes or base stations, and user terminals [5]. Wireless sensor nodes could collect, preprocess, and transmit sensory data, hence such nodes are the origin of data collection. Sink nodes have stronger data collection ability compared with wireless sensor nodes. Sink nodes could preprocess, analyze, and summarize sensory data, and further send to remote data center. Hence, sink node could also be considered as base stations. Sink nodes could directly communicate with user terminals, or communicate with user terminals through Internet or satellites. User terminals obtain sensor data, manage WSN and arrange tasks to sensor nodes.

Sensor deployment affects coverage and lifetime of WSN [6], [7]. This makes sensor deployment an essential network design criterion in WSN. Given a sensing field, sensor deployment is to rationally place sensors in the field to satisfy predefined requirements. An effective method of sensor deployment is to place sensors in a controllable manner to assure quality of service (QoS).

[1*]Corredponding author: Cuiping Zhang.
Email Addresses: zcptjnu@163.com.

The key problem for deployment is if sensors could detect all events in the field, and achieve efficient data transmission with low energy consumption [8]. Thus, sensor deployment problem is an important topic in WSN.

Swarm intelligence (SI) approaches are suitable to solve network design and deployment problems. Network design and deployment problems generally belong to combinational problems, which are non-deterministic hard (NP hard) or non-deterministic complete (NP complete) [9-11]. Such problems are theoretically difficult to solve, hence, researchers have always paid much attention to such problems. SI approaches utilizes social behaviors of swarm species such as ants, bees, fish flocks, and other particles. SI is a general term, covering particle swarm optimization, artificial bee colony, ant colony optimization, etc. In [12], the authors summarized state-of-the-art SI related methods for mobile WSN. Recently, there are many SI related methods for solving problems in WSN [13-16].

Based on the above discussion of related works, this paper attempts to use a SI approach to solve sensor deployment problem in WSN. The contributions of the paper are listed as follows:

(1) sensor deployment problem is constructed to a mathematical model by considering coverage maximization, deployment strategy and other constraints.

(2) a new method called sparrow search algorithm is improved by a novel population structure. Two populations with central symmetry property are created in initialization. Crossover operator is used to exchange information of the two populations.

(3) different SI approaches are tested to solve the sensor deployment problem. The same simulation configuration is used for a fair comparison of the SI approaches.

The rest of the paper is organized into the following. Related works of sensor deployment are briefly described in Section 2. Then the system model and deployment problem are given in Section 3. In Section 4, the optimization algorithm is given. In Section 5, simulation experiments are conducted to demonstrate the effectiveness of the deployment algorithm. Conclusions are drawn in Section 6.

2 Related Works of Sensor Deployment

Sensor deployment is a valuable research topic in WSN. Many researchers have done in-depth study on the problem, which are summarized in the following. Based on sensor node movement, deployment approaches can be classified into static sensor deployment and dynamic sensor deployment. For static sensor deployment, sensors after deployment do not change the positions. Static sensor deployment can be further classified into random deployment and deterministic deployment [17].

2.1 Random Deployment

In harsh environments like disaster scenes and battle fields, sensor nodes are generally deployed by aircraft or rocket ejection. This causes random deployment of sensor nodes. The distribution function of sensor nodes was theoretically analyzed. Specifically, the fault tolerance in WSN was analyzed under uniform deployment or R-random deployment [18]. Non-uniform deployment was studied in [19], which minimized network overhead and routing strategy. For random deployment, network design may not be optimized especially in harsh environments.

2.2 Deterministic Deployment

Deterministic deployment is to deploy sensor nodes at predefined coordinates in region of interest. It may become impractical and infeasible in disaster scenes and battlefields. The target of deterministic deployment is to compute the positions of sensor nodes to satisfy design objectives and other needs. If position was an important factor for operation of sensors, deterministic deployment is a good choice. Moreover, if sensor was costly like indoor applications, deterministic deployment is also a good choice. In such cases, sensor positions should be computed to satisfy predefined needs such as preferential coverage, network lifetime, network connectivity, etc. Thus, network design should be optimized in deterministic deployment.

Preferential coverage is an important factor in deterministic deployment. For such problem, the objective is to find maximum coverage with minimum cost of sensor nodes. There are three types of preferential coverage model. They are binary coverage model, probabilistic coverage model and evidential coverage model.

For binary coverage model, optimal placement design is the most important so that sensor nodes could completely cover region of interest. In [20], regular triangular lattice placement was used . In a regular deployment, sensor node placement can be done by a star pattern, ring pattern, hexagonal pattern, etc. Sensor deployment on two or three dimensional grids had been shown to be NP complete [9]. If region of interest was large or the number of sensors was unable to fulfill a perfect coverage, grid-based deployment had been widely studied. The related approaches include local search method, ant colony optimization, genetic algorithm, etc.

For probabilistic coverage model, deployment strategy involves probabilistic sensor detection. In such model, a position is assumed to be covered with several sensor nodes with different probability values [21]. In [22], sensor deployment was modeled as an optimization problem. The positions of sensor nodes were determined in region of interest by maximal coverage. The algorithm had high computational complexity and memory complexity. In [23], a jigsaw algorithm was used for sensor deployment at the peripheral of region of interest. The algorithm had medium computational complexity and medium memory complexity.

For evidential coverage model, sensor deployment could be modeled as a nonconvex and binary nonlinear optimization problem. In [24], cost and uniform coverage were considered. The authors used an evidence based (EDBA) algorithm. It was a heuristic method based on evidence and grid-like deployment. The objective of EDBA algorithm was to make sure target points had been covered in region of interest. The number of sensors was constrained to a limited number.

2.3 Dynamic Deployment

For dynamic deployment, node movement schemes include artificial potential field method and virtual force algorithm based on virtual attractive and repulsive forces. Artificial potential field method was to expand network coverage through repulsive force between nodes and obstacles [25]. In [26], self-deployment method was proposed. It assumed sensor nodes moved under virtual force. Energy was assumed to cost more in moving than in data transmission. In [27], self-deployment method is to simulate the molecular balance mechanism. In [28], artificial bee colony algorithm was used to solve dynamic deployment problem in WSN. It made sensor nodes dynamic to keep the balance of the network by maximizing network coverage.

3 System Model of WSN

In this paper, WSN is assumed to be comprised of wireless sensor nodes, sink nodes, and user terminals. An example of WSN is shown in Fig. 1. It is assumed that the sensing field is a two-dimensional (2D) area, denoted by *Area*. There are N nodes being deployed in the area. The radius of wireless sensor nodes is denoted by R_s. The communication radius of sensor nodes is denoted by R_c. The number of user terminals is not considered in the system, while it is assumed that data could be transmitted to user terminals through sink nodes or base stations either by wireless communication or by wired communication.

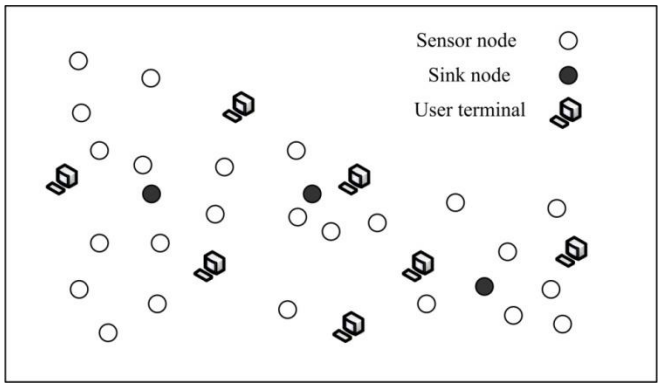


Figure 1. System model of wireless sensor network.

The set of wireless sensor nodes in WSN is given in the following:

$$S_s = \{node_1, node_2, \mathrm{K}\ , node_N\}\ , \tag{1}$$

where a node i in the network is represented by:

$$node_i = (x_i, y_i, R_s)\ , \tag{2}$$

where (x_i, y_i) represents the center of a circle with radius R_s. Based on this assumption, the 2D sensing field is discretized to a set of specific circular particles. The distance between a target and a node is given in the following:

$$d(p, node_i) = \sqrt{(p_{xi} - x_i)^2 + (p_{yi} - y_i)^2}\ , \tag{3}$$

where the coordinates of target p is (p_{xi}, p_{yi}). This is the area coverage model.

In WSN, binary sensing model and probabilistic sensing model are popular node sensing model. For binary sensing model, the probability P that an event is detected by a node is given in the following:

$$P(p, node_i) = \begin{cases} 1, & d(p, node_i) \le R_s \\ 0, & d(p, node_i) > R_s \end{cases}. \tag{4}$$

It can be seen from (4) that the probability is binary based on distance between target and sensor node. This may be too mechanical to simulating real world. Thus, researchers had proposed a more realistic sensing model, i.e., probability sensing model. The energy consumption of a node is related with the distance to target. If the target was located far from its neighboring nodes, the nodes closer to the target would receive stronger energy than farther nodes. This makes the closer nodes have higher probability to sense the signals from the target. In probability sensing model, the probability is given in the following:

$$P(p, node_i) = \begin{cases} 0, & R_s + R_e \le d(p, node_i) \\ e^{-\lambda\alpha}, & R_s - R_e \le d(p, node_i) < R_s + R_e \\ 1, & d(p, node_i) < R_s - R_e \end{cases}, \tag{5}$$

where λ is an attenuation coefficient, R_e is sensing error of node i, and α is given by:

$$\alpha = d(p, node_i) - (R_s - R_e)\,. \tag{6}$$

In the paper, probabilistic sensing model is used in WSN. The objective is to maximize coverage. The constraint is the 2D area, and the limited number of sensor nodes. Mathematically, the system model is given by:

$$\begin{aligned} \max \quad & Cov = \frac{Area\mathbf{I}\left(\mathbf{U}_i^N \varphi_i\right)}{Area} \\ s.t. \quad & 0 \le x_i \le m, \\ & 0 \le y_i \le n, \\ & N \le N_C \end{aligned}, \tag{7}$$

where the *Area* range is [0, *m*] × [0, *n*]. the objective is the ratio of the intersection of node sensing area with the whole *Area* to the *Area*, hence, the value of objective function is between 0 and 1.In the model, the objective is to be maximized to generate the best coverage given N_c sensor nodes.

4 Methods

In the past decades, SI approaches have been applied to many real world problems including wireless communication, wireless networks, and WSN [12-16]. This promotes researchers to think out more and more methods by simulating social behaviors of swarms. Sparrow search algorithm (SSA) was proposed in 2020, which could be classified as SI approaches [29]. The SSA simulates the social behaviors of sparrows. In this algorithm, each sparrow is alike a solution of a given problem. The algorithm iterates to search for food, and at the same time performs anti-predation to survive. During the food search and anti-predation process, the optimization objective is reached.

In SSA, a set of initial positions of the sparrows is created by:

$$X = \left[x_1, x_2, \mathrm{K}\ , x_n\right], \tag{8}$$

where x_i represents a solution of a given problem, which is a column vector, *n* is the number of solutions in initial population, and *X* is a matrix of *n* solutions. Given the mathematical model (7), the fitness value of solutions could be computed and denoted by:

$$Fitness_X = \left[f\left(x_1\right), f\left(x_2\right), \mathrm{K}\ , f\left(x_n\right)\right], \tag{9}$$

where the fitness value of population *X* is denoted by $Fitness_X$. In model (7), the optimization objective is to maximize coverage, thus, higher fitness value means that the associated solution is more probably a good solution for the given problem.

4.1 Sparrow search algorithm

The main procedures of the SSA are shown in Fig. 1. Besides initialization step, the main iteration of the algorithm is comprised of updating the locations of producers, updating the locations of scroungers, and updating the locations of the sparrows at the edge [29]. The main iteration is introduced in the following.

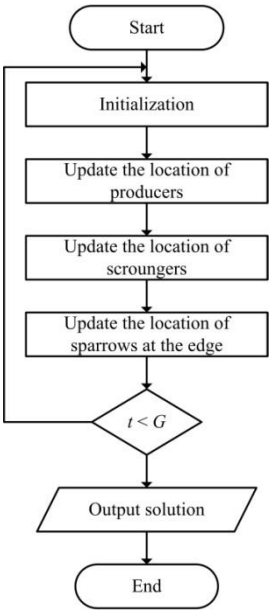


Figure 2. The flow chart of sparrow search algorithm.

In updating the locations of producers, a subset of sparrows flies out searching for food, which is called finders. The finders have higher fitness values than the others in population. The finders have high probability to prioritize to get food, and they look for food for all sparrows in population. Such social behavior makes the finders have to search a larger range than the others. The locations of finders are updated by:

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^{t} \times e^{-\frac{t}{\alpha \times iter}}, & r_1 < ST \\ x_{i,j}^{t} + q \times L, & r_1 \geq ST \end{cases}, \tag{10}$$

where t is the time step, subscript j is the j-th variable of solution x_i, α and r_1 are random numbers between 0 and 1, q is a random number following normal distribution, L is a matrix with element 1. In (10), ST is a number between 0.5 and 1. r_1 and ST are called the warning value and the safety threshold, respectively.

In updating the locations of scroungers, a subset of sparrows flies out to join the search of the finders. Such sparrows are called joiners, which always monitor the behavior of the finders. If they realized that the finders had found better food, they immediately left their locations to compete food with the finders. The joiners'locations are updated by:

$$x_{i,j}^{t+1} = \begin{cases} q \times e^{\frac{x_{w,j}^{t} - x_{i,j}^{t}}{i^2}}, & i > n/2 \\ x_{h,j}^{t} + \left| x_{i,j}^{t} - x_{h,j}^{t} \right| A^{+} L, & i \leq n/2 \end{cases}, \tag{11}$$

where x_h and x_w are the positions having the highest and worst fitness values in population, A^+ is computed by:

$$A^{+} = A^{T} \left(A A^{T} \right)^{-1}, \tag{12}$$

where A is a vector with element randomly assigned 1 or -1.

In updating the locations of the sparrows at the edge, a subset of sparrows flies out to new locations. It is assumed that sparrows are in dangerous environment, which makes some sparrows tend to randomly flying. In general, 10% to 30% sparrows of population are randomly chosen, and then they are updated by:

$$x_{i,j}^{t+1} = \begin{cases} x_{b,j}^{t} + \beta \left| x_{i,j}^{t} - x_{b,j}^{t} \right|, & f\left(x_i\right) > f_g \\ x_{i,j}^{t} + k \left(\dfrac{x_{i,j}^{t} - x_{w,j}^{t}}{f\left(x_i\right) - f_w + \varepsilon} \right), & f\left(x_i\right) \leq f_g \end{cases}, \tag{13}$$

where x_b is the current best solution, f_g and f_w are the current global best fitness value and the current worst fitness value, respectively, β is a random number between 0 and 1, ε is a small constant to avoid divide by zero.

4.2 Improved sparrow search algorithm

This section gives the improved sparrow search algorithm (ISSA). The main procedures of ISSA are shown in Fig. 3. In the ISSA algorithm, initialization is done by Halton sequence method and central symmetry method. It also contains the updating the locations of producers, scroungers and sparrows at the edge, which is similar with the SSA algorithm. The ISSA algorithm has an information exchange step in the last of main iteration.

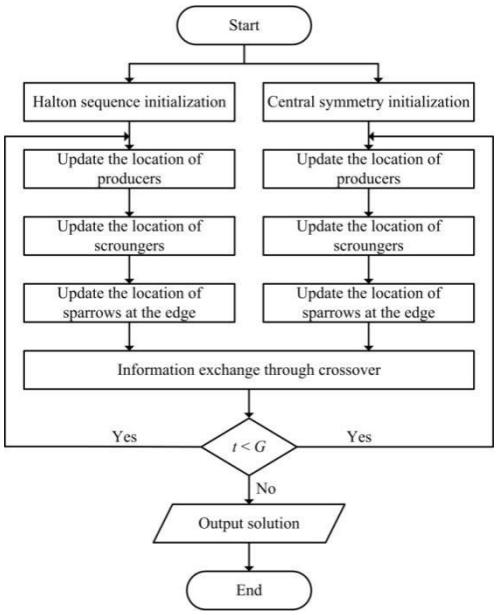
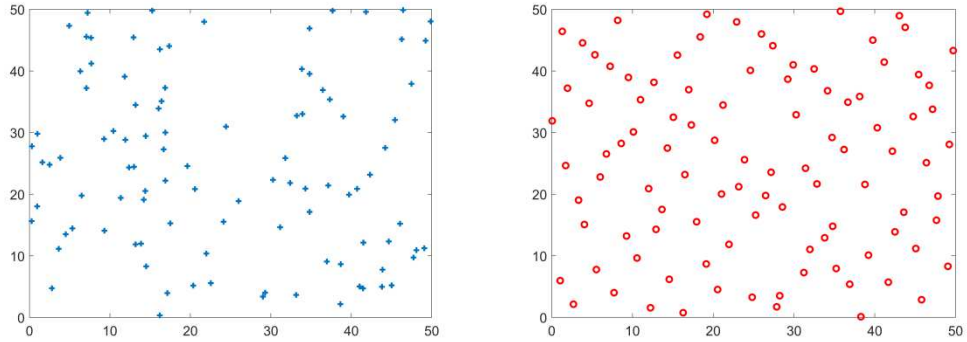


Figure 3. The flow chart of improved sparrow search algorithm.

In initialization, the ISSA algorithm needs to generate two populations. One population is created by Halton sequence method, and then another population is created by central symmetry method. It is well known that initial samples are created by using probability distribution, and uniform distribution is generally used in creating initial population. Probability distribution could affect the performance of SI approaches. When creating a large number of samples, uniform distribution is able to have high ergodicity and diversity; however, population size is generally a small set of samples in SSA algorithm, uniform distribution may have low ergodicity, uneven individual distribution, and unpredictability. Hence, low difference sequence method, i.e., Halton sequence, is used in this paper.

Halton sequence has two properties. One property is that it has more uniform distribution for a small set of samples. The other property is that it has high ergodicity which is able to cover good solution search space [30], [31]. Halton sequence method is a kind of low difference sequence. It uses a prime number as base to continuously create non-repeated and uniform samples. Through division operation, the samples created by Halton sequence method are between 0 and 1.

A comparison of initial population is performed between uniform distribution and Halton sequence method. The results are shown in Fig. 4. The samples created by uniform distribution method are shown in Fig. 4 (a), while the samples created by Halton sequence method are shown in Fig. 4 (b). It can be seen from Fig. 4 (a) that a large gap exists among the samples, and some samples are very close to form a cluster. It can be seen from Fig. 4 (b) that the samples created by Halton sequence method have evenly located into each range in two coordinates. The samples in Fig. 4 (a) have more even distribution and higher ergodic properties compared than the samples in Fig. 4 (b).



(a) uniform distribution method, (b) Halton sequence method,

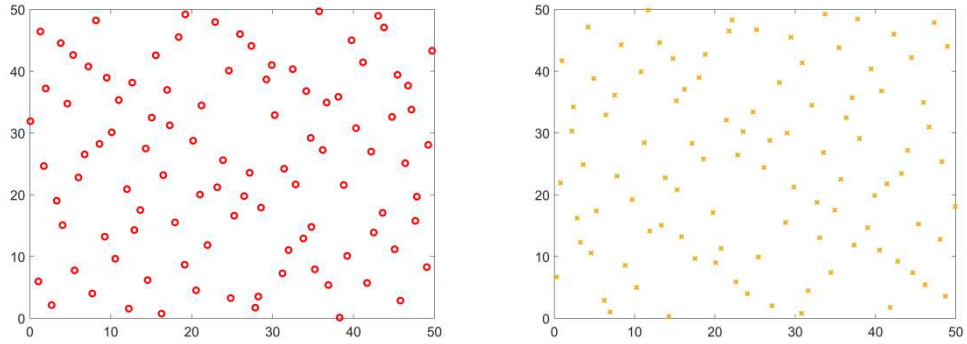
Figure 4. Comparison of initial samples created by uniform distribution method and Halton sequence method.

Based on the comparison between uniform distribution method and Halton sequence method, the initial samples using Halton sequence method can effectively avoid the possibility falling into local optimum. The initial population created by Halton sequence method covers more search space, thus, the ISSA algorithm is able to avoid early convergence to local optimal.

The central symmetry method is used to create another population. Although the Halton sequence method is able to create good initial samples, it does not have central symmetry property. The central symmetry method creates samples based on those created by Halton sequence method. Denote the range of coordinates as [0, m] × [0, m]. The horizontal axis and vertical axis have the same length. That is the sensing field in WSN is a square area. Denote $x_i = (x_{i1}, x_{i2})$. Mathematically, the sample by central symmetry method is:

$$x_i^c = \left(m - x_{i1}, m - x_{i2}\right), \tag{14}$$

where x_i^c is the sample created by central symmetry method.



(a) Halton sequence method, (b) central symmetry method,

Figure 5. initial samples created by Halton sequence method and central symmetry method.

Initial samples created by Halton sequence method and central symmetry method are shown in Fig. 5, where Fig. 5 (a) shows the result by Halton sequence method and Fig. 5 (b) shows the result by central symmetry method. It can be seen from the figure that both method can create evenly distributed samples and cover the full search space. In the ISSA algorithm, two populations are used with one created by Halton sequence method and the other created by central symmetry method.

Besides initialization, the ISSA algorithm contains the updating the locations of producers, scroungers and sparrows at the edge, which is similar with the SSA algorithm. For updating the locations of producers, $r_1 <$ ST means that no predators exist in environment, hence, the finders do global search; on the other hand, $r_1 \geq$ ST means that predators are detected in environment, hence, the finders issue alarm to other sparrows and do local search. For updating the locations of scroungers, $i > n/2$ means that the associated sparrow has

low fitness and should not have food, hence, it has to fly to another location to find food. For the updating of sparrows at the edge, $f(x_i)>f_g$ means that the associated sparrow locates at the edge of search space; on the other hand, $f(x_i) \leq f_g$ means that the associated sparrow locates in the search space. It is aware of danger and needs to fly close to other sparrows.

The ISSA algorithm has two populations to search good solutions; hence, information exchange is useful between the two populations. The ISSA algorithm has a crossover step in the last of main iteration. In this paper, uniform crossover is used which needs two samples to complete the operation. When performing crossover operation, one sample is chosen from the population by Halton sequence method; another sample is chosen from the population by central symmetry method. In this manner, information is exchanged between the two populations. Mathematically, uniform crossover is:

$$x_{i,j}^{t+1} = \begin{cases} x_{hi,j}^{t}, & r_2 \leq p_{cr} \\ x_{ci,j}^{t}, & r_2 > p_{cr} \end{cases}, \tag{15}$$

where x_{hi} and x_{ci} are respectively chosen from Halton sequence population and central symmetry population, r_2 is a random number between 0 and 1, p_{cr} is crossover rate.

During information exchange step, large crossover rate p_{cr} means that the population following Halton sequence method contributes more to the evolution; while small p_{cr} means that the population following central symmetry method contributes more to the evolution. Because central symmetry method depends on Halton sequence method, p_{cr} is set to 0.5 in the paper. This means that both methods have equal contribution to the evolution in ISSA algorithm.

Comparing the computational complexity of the SSA and the ISSA algorithms is in the following. Arithmetic operations are counted for computational complexity. Denote a solution is with length d, i.e., the problem has d variables. Denote n solutions are created in initial population. Denote G iterations are done in an algorithm. In the SSA algorithm, $2d$ operations are required in (10), $6d$ operations are required in (11) and (12), $5d$ operations are required in (13). Hence, the SSA algorithm requires $6d\times n\times G$. That is the computational complexity of the SSA algorithm is $6dnG$. In the ISSA algorithm, $n/2$ solutions are created by Halton sequence method, and $n/2$ solutions are created by central symmetry method. There are n solutions created in initial population, which is the same as the SSA algorithm. In (15), d operations are required. Hence, the ISSA algorithm requires $(6d\times n/2\times G + 6d\times n/2\times G + d\times n/2\times G) = 6.5dnG$. The computational complexity of the ISSA algorithm is slightly larger than that of the SSA algorithm. Overall, the computational complexity of both algorithms is dnG for deployment problem in WSN.

5 Results and Discussion

This section first introduces the simulation setting and evaluation criteria for solving deployment problem in WSN. It then analyzes the results of the ISSA algorithm and discusses the effectiveness of the proposed method.

5.1 Simulation setting and evaluation criteria

To simulate and assess the deployment problem and the proposed method, we simulate WSN in this paper. The assumptions and regulations required in the simulation are given in the following. In WSN, sensor nodes and sink nodes are stationary and not move after deployment in the network. Wireless sensor nodes are assumed to have the same coverage radius. Sink nodesare assumed to have longer coverage radius than wireless sensor nodes. The WSN is assumed to spread in a square area with size 50 m×50 m. A list of network parameters is given in Table 1.

Table 1: Simulation parameters of WSN

Parameter	Value
Square area	50× 50 m
The number of sensor nodes	35
The radius of sensor nodes	5 m
The radius of communication	10 m
Sensing attenuation coefficient	0.1

To assess the solution to deployment problem, coverage rate and connectivity are taken as evaluation criteria. Coverage rate is a proper method to measure sensing quality of a network. Coverage rate is the target of coverage algorithm and the primary criterion for assessing coverage performance in WSN. Mathematically, coverage rate is given by:

$$Rate_{coverage} = \frac{\left(\mathbf{U}_i \varphi_i\right)\mathbf{I}\ \ Area}{Area}, \tag{16}$$

where φ_i is the effective sensing field of sensor node i. The numerator is the intersection of the effective sensing field of all sensor nodes and the whole area. The denominator is the whole area. Hence, coverage rate is a number between 0 and 1, where 1 means that a solution is able to cover all area, while 0 means that a solution is not able to cover any area in the network.

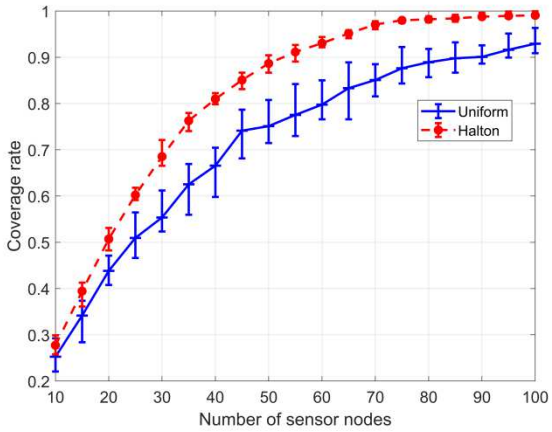


Figure 6. The coverage rate of initial deployment created by uniform distribution method and Halton sequence method.

The coverage rate of initial deployment is shown in Fig. 6. In the figure, the coverage rate of uniform distribution method is shown in plus symbol, while the coverage rate of Halton sequence method is shown in cycle symbol. It can be seen from Fig. 6 that coverage rate increases with the increase of the number of sensor nodes. A gap exists between uniform distribution method and Halton sequence method. From the comparison, initial deployment of Halton sequence method is more even than the deployment of uniform distribution method. Thus, Halton sequence method has better coverage rate than uniform distribution method in initial deployment. Moreover, due to the symmetry property of central symmetry method, central symmetry method has the same coverage rate as Halton sequence method. Hence, the curve of central symmetry method is not shown in Fig. 6.

Besides coverage rate, connectivity is taken to assess the solution quality. As the communication capability of wireless sensor nodes is limited, the collected data has to be transmitted to sink nodes through multi-hop style. Connectivity is to measure if the collected data could be transmitted to sink nodes. Moreover, connectivity also affects the performance of network transmission. Mathematically, connectivity is given by:

$$Connectivity = \frac{\sum_{i=1}^{n} \theta(i)}{n(n-1)}, \tag{17}$$

where $\theta(i)$ is the number of nodes that sensor node i connecting to other sensor nodes in WSN. Hence, the value of $\theta(i)$ is

$$0 \le \theta(i) \le n-1\ . \tag{18}$$

If sensor nodes in WSN are connected in pairs, the number of total connections is $n(n\text{-}1)/2$. In (17), connectivity is defined as the ratio of the number of $\theta(i)$ (i = 1,2,…,n) to the number of total connections.Hence, connectivity is also a number between 0 and 1, where 1 means that the sensor nodes in WSN areconnected, while 0 means that the sensor nodes in WSN are not fully connected.

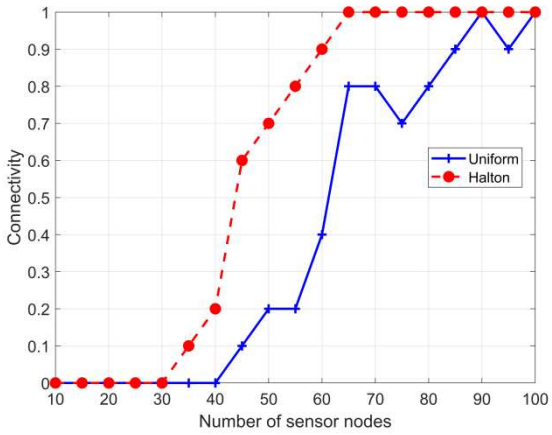


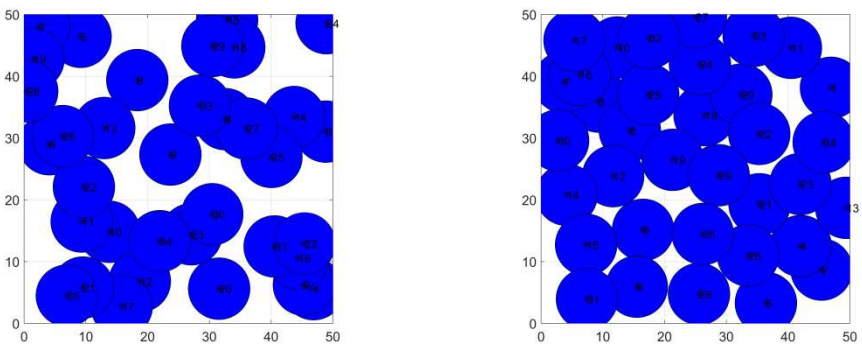
Figure 7. The connectivity of initial deployment created by uniform distribution method and Halton sequence method.

The connectivity of initial deployment is shown in Fig. 7. In the figure, the connectivity of uniform distribution method is shown in plus symbol, while the connectivity of Halton sequence method is shown in cycle symbol. It can be seen from Fig. 7 that connectivity increases with the increase of the number of sensor nodes. When the number of sensor nodes is between 30 and 95, a gap exists between uniform distribution method and Halton sequence method. From the comparison, initial deployment of Halton sequence method has better connectivity than the deployment of uniform distribution method. Moreover, due to the symmetry property of central symmetry method, central symmetry method has the same connectivity as Halton sequence method. Hence, the curve of central symmetry method is not shown in Fig. 7. It can be seen from Fig. 7 that the connectivity of uniform distribution method changes nonlinearly with the change of the number of sensor nodes. On the other hand, the connectivity of Halton sequence method changes linearly with the change of the number of sensor nodes. Thus, Halton sequence method and central symmetry method are better choice than uniform distribution method.

Based on the above analysis, it is known that Halton sequence method and central symmetry method have good performance in initial deployment. The coverage rate and connectivity of WSN is near to 1 when the number of sensor nodes is set to 100. The coverage rate and connectivity is about 0.7 and 0.1, respectively. This means that if cost budget is adequate in deploying WSN, perfect coverage rate and connectivity can be reached; on the other hand, if cost budget is inadequate or limited in deploying WSN, initial deployment is unable to reach satisfactory coverage rate and connectivity. Thus, an optimization algorithm is required to find good deployment scheme.

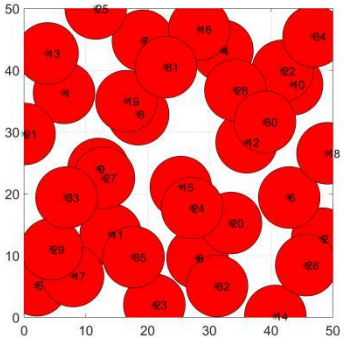
5.2 Network optimization experiment

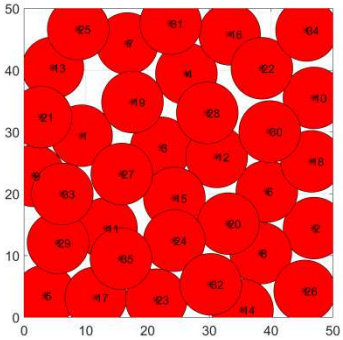
The deployment problem is solved in the following. The SSA and ISSA algorithms are used. The coverage rate of the SSA and the ISSA algorithms is shown in Fig. 8. In Fig. 8 (a) and (b), the initial and final coverage rate of the SSA algorithm is shown. It can be seen from Fig. 8 (a) that a large field is uncovered by the initial deployment solution. After optimization, the coverage in Fig. 8 (b) is much better than the initial deployment.



(a) Initial coverage rate of SSA

(b) Final coverage rate of SSA





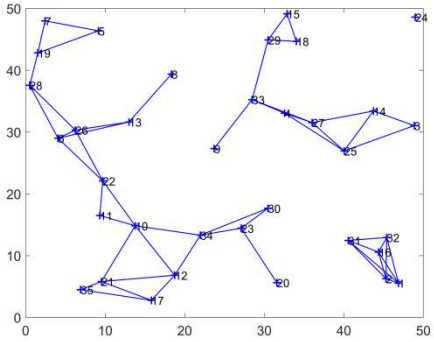
(c) Initial coverage rate of ISSA

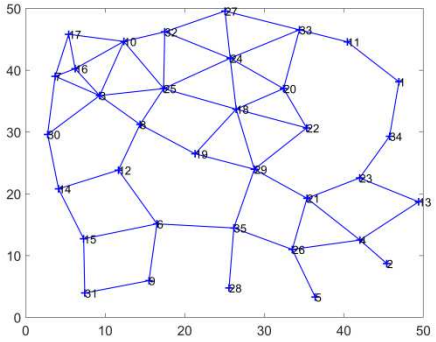
(d) Final coverage rate of ISSA

Figure 8. The coverage rate of the SSA and ISSA algorithms.

The initial and final coverage rate of the ISSA algorithm is shown in Fig. 8 (c) and (d). It can be seen from Fig. 8 (c) that a large field is uncovered by the initial deployment solution. After optimization, the coverage in Fig. 8 (d) is much better than the initial deployment. Moreover, the coverage of Fig. 8 (d) is more uniform than that of Fig. 8 (b).

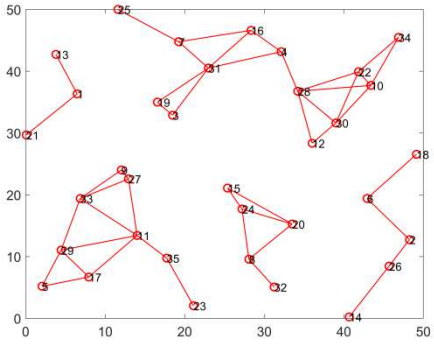
The connectivity of the SSA and the ISSA algorithms is shown in Fig. 9. In Fig. 9 (a) and (b), the initial and final connectivity of the SSA algorithm is shown. It can be seen from Fig. 9 (a) that the network consists of three sub-networks. After optimization, the connectivity in Fig. 9 (b) is much better and all sensor nodes are connected.

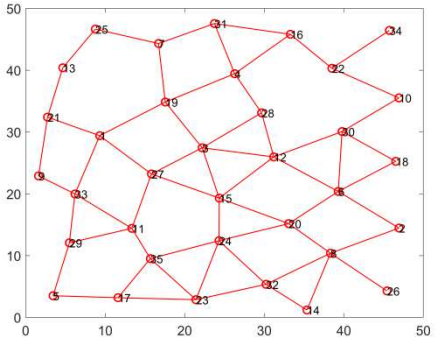




(a) Initial connectivity of SSA,

(b) Final connectivity of SSA,





(c) Initial connectivity of ISSA,

(D) Final connectivity of ISSA,

Figure 9. The connectivity of the SSA and ISSA algorithms.

In Fig. 9 (c) and (d), the initial and final connectivity of the ISSA algorithm is shown. It can be seen from Fig. 9 (c) that the network consists of five sub-networks. After optimization, the connectivity in Fig. 9 (d) is much better and all sensor nodes are connected. Moreover, the connectivity of Fig. 9 (d) is more uniform than that of Fig. 9 (b).

The convergence graph of the SSA and the ISSA algorithms is shown in Fig. 10. In the figure, the solid curve is for the SSA algorithm, while the dashed curve is for the ISSA algorithm. It can be seen from Fig. 10 that the curve of the ISSA algorithm is above that of the SSA algorithm. This means that the ISSA algorithm has better convergence performance than the SSA algorithm.

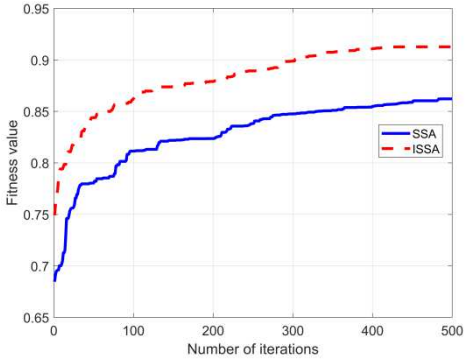


Figure 10. The convergence graph of the SSA and ISSA algorithms.

The comparison of computational complexity is shown in Table 2. In Table 2, the minimum (Min), median (Med), maximum (Max), mean (Mean) and standard deviation (Std) values of running time are given. In the table, the result of the SSA algorithm is shown in the first column. The result of the ISSA algorithm is shown in the second. The last column is also the result of the ISSA algorithm except that it is run in parallel. Due to the ISSA algorithm evolves two populations, it can be run in parallel, while information exchange has to be run in serial. It can be seen from Table 2 that the running time of the ISSA algorithm is longer than the SSA algorithm, while the ISSA algorithm can save running time by executing in parallel.

Table 2: Running time in seconds by the SSA and ISSA algorithms for deployment problem

Algorithm	SSA	ISSA	ISSA in parallel
Min	104.05	136.49	91.87
Med	104.17	137.31	92.48
Max	104.77	137.94	92.92
Mean	104.22	137.15	92.37
Std	0.2036	0.5488	0.3799

5.3 Comparison experiment

To evaluate the performance of the ISSA algorithm, several other algorithms are used to solve the deployment problem. The algorithms include particle swarm optimization (PSO) [32], [33], artificial bee colony (ABC) [34], [35], grey wolf optimizer (GWO) [36] [37], whale optimization algorithm (WOA) [38]. These algorithms are also SI approaches like the SSA and ISSA algorithms. Moreover, they have been used to solve optimization problems in WSN. Hence, they are taken for comparison in the paper. Each algorithm is run 10 times to count the average performance.

The six algorithms are used to solve the deployment problem. The results of coverage rate are shown in Table 3. In the table, the Min, Med, Max, Mean and Std values of coverage rate are given. It can be seen from the second, third, and fifth column of Table 3 that the ISSA algorithm achieves the greatest Min, Med, and Mean value for coverage rate. It can be seen from the fourth column of Table 3 that the GWO algorithm has the greatest Max value, while the ISSA algorithm has the second greatest Max value among the algorithms. Moreover, the ISSA algorithm also has the least Std value among the six algorithms. This means that the proposed algorithm shows good performance for coverage rate measurement.

Table 3: Coverage rate by the SSA and ISSA algorithms for deployment problem

Algorithm	Min	Med	Max	Mean	Std
PSO	0.7609	0.8227	0.8474	0.8197	0.0242
ABC	0.7710	0.7760	0.8046	0.7800	0.0108
GWO	0.7207	0.8885	0.9131	0.8610	0.0746
WOA	0.8201	0.8362	0.8640	0.8367	0.0141
SSA	0.8297	0.8641	0.8873	0.8641	0.0171
ISSA	0.8881	0.9004	0.9127	0.9003	0.0105

The results of connectivity of the six algorithms are shown in Table 4. The format of Table 4 is the same as Table 3. In Table 4, the Min, Med, Max, Mean and Std values of connectivity are given.

Table 4: Connectivity by the SSA and ISSA algorithms for deployment problem

Algorithm	Min	Med	Max	Mean	Std
PSO	0.1378	0.1538	0.1731	0.1543	0.0117
ABC	0.1277	0.1546	0.1815	0.1555	0.0175
GWO	0.1445	0.1555	0.1748	0.1583	0.0098
WOA	0.1378	0.1445	0.1546	0.1444	0.0050
SSA	0.1546	0.1580	0.1664	0.1587	0.0036
ISSA	0.1563	0.1597	0.1630	0.1593	0.0028

It can be seen from the second, third, fifth column of Table 4 that the ISSA algorithm achieves the greatest values for connectivity. It can be seen from the fourth column of Table 4 that the ABC algorithm has the greatest Max value for connectivity. Moreover, the ISSA algorithm also has the least Std value among the six algorithms. This means that the proposed algorithm shows good performance for connectivity measurement.

6 Conclusion

Sensor deployment affects coverage and lifetime of WSN. This makes sensor deployment an essential network design criterion in WSN. An effective method of sensor deployment is to place sensors in a controllable manner to assure high QoS. Thus, sensor deployment problem is an important topic in WSN. This paper uses an improved sparrow search algorithm to solve sensor deployment problem in WSN.

In the paper, sensor deployment problem is constructed to a mathematical model by considering coverage maximization, deployment strategy and other constraints. Then, the proposed ISSA algorithm is introduced to solve the deployment problem. In the ISSA algorithm, two populations with central symmetry property are created in initialization. Crossover operator is used to exchange information of the two populations. Finally, five different SI approaches are tested to solve the sensor deployment problem. The same simulation configuration is used for a fair comparison of the six algorithms.

Through simulation experiments, the results show that the ISSA algorithm has good performance for both coverage rate and connectivity measurements. Although the ISSA algorithm has higher computational complexity than the SSA algorithm, the running time of the ISSA algorithm could be saved by running in parallel. This is feasible because the ISSA algorithm is comprised of two population structure. The sensor deployment problem has been studied and solve in the paper. A good solution can be found by the ISSA algorithm. The deploy method can also be extended to heterogeneous WSN.

Abbreviations

WSN	Wireless sensor network
SSA	Sparrow search algorithm
NP	Non-deterministic polynomial
IoT	Internet of Things
5G	Fifth generation
QoS	Quality of service
SI	Swarm intelligence
EDBA	Evidence based
2D	Two-dimensional
ISSA	Improved sparrow search algorithm
Min	Minimum

Med	Median
Max	Maximum
Std	Standard deviation
PSO	Particle swarm optimization
ABC	Artificial bee colony
GWO	Grey wolf optimizer
WOA	Whale optimization algorithm

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Availability of data and material
Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Competing interests
The authors declare that they have no competing interests.

Funding
Not applicable.

Author contributions
Jiazu Xie developed the algorithm and conducted the experiments. Cuiping Zhang contributed to the theoretical analysis of the algorithm and participated in writing the paper. All authors read and approved the final manuscript.

Acknowledgements
Not applicable.

References

1. Y. Zhao, B. Li, Model and algorithm of innovation performance evaluation for coordination of supply and demand based on wireless sensor network. *EURASIP J. Adv. Signal Process*. 2021, 108 (2021). https://doi.org/10.1186/s13634-021-00809-4.
2. K. Li, W. Ni and F. Dressler, Continuous Maneuver Control and Data Capture Scheduling of Autonomous Drone in Wireless Sensor Networks, *IEEE Transactions on Mobile Computing*, 2022, doi: 10.1109/TMC.2021.3049178.
3. Y.H., Kim, Greedy sensor selection based on QR factorization. *EURASIP J. Adv. Signal Process.* 2021, 117 (2021). https://doi.org/10.1186/s13634-021-00824-5.
4. K. P. Rajput, M. F. Ahmed, N. K. D. Venkategowda, A. K. Jagannatham, G. Sharma and L. Hanzo, Robust Decentralized and Distributed Estimation of a Correlated Parameter Vector in MIMO-OFDM Wireless Sensor Networks, *IEEE Transactions on Communications*, vol. 69, no. 10, pp. 6894-6908, Oct. 2021.
5. Y. Zhu, S. Liang, G. Xue, et al. An efficient multi-objective optimization approach for sensor management via multi-Bernoulli filtering. *EURASIP J. Adv. Signal Process*. 2022, 62 (2022). https://doi.org/10.1186/s13634-022-00881-4
6. O. Dagdeviren and V. K. Akram, The Effect of Random Node Distribution and Transmission Ranges on Connectivity Robustness in Wireless Sensor Networks, *International Symposium on Networks, Computers and Communications (ISNCC)*, Istanbul, Turkey, 2019, pp. 1-5, doi: 10.1109/ISNCC.2019.8909162.
7. D. Wang, B. Xie and D. P. Agrawal, Coverage and lifetime optimization of wireless sensor networks with Gaussian distribution, *IEEE Transactions on Mobile Computing*, vol. 7, no. 12, pp. 1444-1458, Dec. 2008.
8. B. Yener, M. Magdon-Ismail, F. Sivrikaya, Joint problem of power optimal connectivity and coverage in wireless sensor networks, *Wireless Networks*, vol. 13, pp. 537–550, 2007.
9. W.-C. Ke, B.-H. Liu, M.-J. Tsai, The critical-square-grid coverage problem in wireless sensor networks is NP-Complete, *Computer Networks*, vol. 55, no. 9, pp. 2209-2220, Jun. 2011.
10. X. Zhang, H. Qi, X. Zhang, L. Han, Energy-Efficient Resource Allocation and Data Transmission of Cell-Free Internet of Things, *IEEE Internets of Things*, vol. 8, no. 20, pp. 15107-15116, Oct 15, 2021.
11. X. Zhang, X. Lu, X. Zhang, Mobile wireless sensor network lifetime maximization by using evolutionary computing methods, *Ad Hoc Networks*, vol. 101, Article No. 102094, 2020.

12. L. Cao, Y. Cai, Y. Yue, Swarm intelligence-based performance optimization for mobile wireless sensor networks: survey, challenges, and future directions, *IEEE Access*, vol. 7, pp. 161524-161553, 2019.
13. Y. Yoon, and Y. Kim, An efficient genetic algorithm for maximum coverage deployment in wireless sensor networks, *IEEE Transactions on Cybernetics*, vol. 43, no. 5, pp. 1473-1483, Oct. 2013.
14. M. A. Alsheikh, S. Lin, D. Niyato, and H. Tan, Machine learning in wireless sensor networks: algorithms, strategies, and applications, *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 1996-2018, Apr. 2014.
15. X. Zhang, X. Zhang, A binary artificial bee colony algorithm for constructing spanning trees in vehicular ad hoc networks, *Ad Hoc Networks*, vol. 58, pp. 198-204, 2017.
16. J. Xie, B. Zhang, and C. Zhang, A novel relay node placement and energy efficient routing method for heterogeneous wireless sensor networks, *IEEE Access*, vol. 8, pp. 202439-202444, Mar. 2020.
17. R. Priyadarshi, B. Gupta, A. Anurag, Wireless sensor networks deployment: a result oriented analysis, *Wireless Personal Communications*, vol. 113, pp. 843-866, Mar. 2020.
18. M. Ishizuka, and M. Aida, Performance study of node placementin sensor networks, 24th International Conference on DistributedComputing Systems Workshops, Tokyo, Japan, 2004, pp. 598-603.
19. T. Fan, and J. Chen, A New Nonuniform random deployment method to minimize cost for large-scale wireless sensor networks, *IEEE Access*, vol. 8, pp. 198532-198547, 2020.
20. R. Kershner, The number of circles covering a set. American Journal of Mathematics, vol. 61, no. 3, pp. 665-671, Jul. 1939.
21. A. Hossain, P. K. Biswas, S. Chakrabarti, Sensing models and its impact on network coverage in wireless sensor network, *IEEE region 10 colloquium and 3rd international conference on industrial and information systems*, Kharagpur, India, Dec. 2008, pp. 1-5, doi: 10.1109/ICIINFS.2008.4798455.
22. S. S. Dhillon, K. Chakrabarty, S. S. Iyengar, Sensor placement for grid coverage under imprecise detections, *Proceedings of the 5th international conference on information fusion*,Annapolis, MD, USA, Jul. 2002, pp. 1581-1587 vol. 2, doi: 10.1109/ICIF.2002.1021005.
23. S.-C. Huang, H.-Y. Chang, K.-L. Wu, A jigsaw-based sensor placement algorithm for wireless sensor networks, *International Journal of Distributed Sensor Networks*, vol. 9, no. 11, Nov. 2013,
24. M. R. Senouci, A. Mellouk, L. Oukhellou, A. Aissani, Uncertainty-aware sensor network deployment, *IEEE Global Telecommunications Conference (GLOBECOM)*, Houston, TX, USA, 2011, pp. 1-5, doi: 10.1109/GLOCOM.2011.6134363.
25. A. Howard, M. J. Mataric, and G. S. Sukhatme, Mobile sensor network deployment using potential fields: a distributed, scalable solution to the area coverage problem, In: Asama H., Arai T., Fukuda T., Hasegawa T. (eds) Distributed Autonomous Robotic Systems 5. Springer, Tokyo, 2002, pp. 299-308, doi: 10.1007/978-4-431-65941-9_30.
26. Y. Zou, and K. Chakrabarty, Sensor deployment and target localizationin distributed sensor networks, *ACM Transactions on EmbeddedComputing Systems*, vol. 3, no. 1, pp. 61-91. 2004.
27. H. Nojeong, and P. K. Varshney, Energy-efficient deployment of intelligent mobile sensor networks, *IEEE Transactions on Systems,Man, and Cybernetics - Part A: Systems and Humans*, vol. 35, no. 1,pp. 78-92, Jan. 2005.
28. C. Ozturk, D. Karaboga, and B. Gorkemli, Probabilistic dynamic deployment of wireless sensor networks by artificial bee colony algorithm,*Sensors*, vol. 11, no. 6, pp. 6056–6065, 2011.
29. J. Xue, B. Shen, A novel swarm intelligence optimization approach: sparrow search algorithm, Systems Science & Control Engineering, vol. 8, no. 1, pp. 22-34. 2020.
30. L. Kocis, and W. J. Whiten, Computational investigations of low-discrepancy sequences, *ACM Transactions on Mathematical Software*, vol. 23, no. 2, pp. 266-294, 1997.
31. P. Krmer, J Plato, and V Snael, Differential evolution for the optimization of low-discrepancy generalized Halton sequences, *Swarm and Evolutionary Computation*, vol, 54, article number 100649, May 2020.
32. J. Kennedy and R. Eberhart, Particle swarm optimization, International Conference on Neural Networks, Perth, Australia, 1995, pp. 1942-1948, vol.4, doi: 10.1109/ICNN.1995.488968.
33. S. Khezri, K. Faez and A. Osmani, Modified discrete binary PSO based sensor placement in WSN networks, *International Conference on Computational Intelligence and Communication Networks*, Bhopal, India, 2010, pp. 200-204.
34. D. Karaboga, and B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) Algorithm, *Journal of Global Optimization*, vol. 39, pp. 459-471, 2007.
35. C. Ozturk, D. Karaboga, B. Gorkemli, Artificial bee colony algorithm for dynamic deployment of wireless sensor networks, *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 20, no. 2, pp. 252-262. 2012.
36. S. Mirjalili, S. M. Mirjalili , and A. Lewis, Grey Wolf Optimizer, *Advances in Engineering Software*, vol. 69, pp.46–61, MAR. 2014.
37. S. N. Ghorpade, M. Zennaro and B. S. Chaudhari, GWO model for optimal localization of IoT-enabled sensor nodes in smart parking systems, *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 1217-1224, Feb. 2021.
38. S. Mirjalili, A. Lewis, The whale optimization algorithm, *Advances in Engineering Software*, vol. 95, pp. 51-67, 2016.