Mathematical Biosciences and Engineering

MBE, 20(2): 3191–3215.
DOI: 10.3934/mbe.2023150
Received: 08 October 2022
Revised: 21 November 2022
Accepted: 25 November 2022
Published: 02 December 2022

http://www.aimspress.com/journal/MBE

Research article

An efficient coverage method for SEMWSNs based on adaptive chaotic Gaussian variant snake optimization algorithm

Xiang Liu[1], Min Tian[1,*] Jie Zhou[2] and Jinyan Liang[1]

[1] College of mechanical and electrical engineering, Shihezi University, Shihezi 832000, China
[2] College of information science and technology, Shihezi University, Shihezi 832000, China

* **Correspondence:** Email: tm_inf@shzu.edu.cn; Tel: +8613319937952.

Abstract: Soil element monitoring wireless sensor networks (SEMWSNs) are widely used in soil element monitoring agricultural activities. SEMWSNs monitor changes in soil elemental content during agriculture products growing through nodes. Based on the feedback from the nodes, farmers adjust irrigation and fertilization strategies on time, thus promoting the economic growth of crops. The critical issue in SEMWSNs coverage studies is to achieve maximum coverage of the entire monitoring field by adopting a smaller number of sensor nodes. In this study, a unique adaptive chaotic Gaussian variant snake optimization algorithm (ACGSOA) is proposed for solving the above problem, which also has the advantages of solid robustness, low algorithmic complexity, and fast convergence. A new chaotic operator is proposed in this paper to optimize the position parameters of individuals, enhancing the convergence speed of the algorithm. Moreover, an adaptive Gaussian variant operator is also designed in this paper to effectively avoid SEMWSNs from falling into local optima during the deployment process. Simulation experiments are designed to compare ACGSOA with other widely used metaheuristics, namely snake optimizer (SO), whale optimization algorithm (WOA), artificial bee colony algorithm (ABC), and fruit fly optimization algorithm (FOA). The simulation results show that the performance of ACGSOA has been dramatically improved. On the one hand, ACGSOA outperforms other methods in terms of convergence speed, and on the other hand, the coverage rate is improved by 7.20%, 7.32%, 7.96%, and 11.03% compared with SO, WOA, ABC, and FOA, respectively.

Keywords: coverage optimization; snake optimizer; chaotic operator; Gaussian variant; power consumption

1. Introduction

With the comprehensive promotion of agricultural automation technology and the rapid development of IoT technology, soil element monitoring wireless sensor networks (SEMWSNs) have become essential tools for monitoring the agricultural production environment [1–5]. SEMWSNs use the agricultural internet of things (AIoT) concept, characterized by low cost, flexible structure, and self-organizing networks [6]. SEMWSNs can accurately collect and transmit data on the content of various elements in the soil [7]. By deploying SEMWSNs in natural production environments, farmers can make more accurate agricultural production decisions based on the soil info collected by the sensor nodes. SEMWSNs coverage is one of the crucial research directions in agriculture, seeking solutions to maximize the range of the destination field by changing the distribution position of wireless sensor nodes [8–11]. SEMWSNs are widely used in agricultural production activities but face finite power, lower node dependability, and complex application surroundings. These problems significantly impact data collection and transmission at the nodes, limiting the network's overall performance [12,13]. Avoiding too many blind spots in the monitoring range when sensor nodes are inappropriately deployed is one of the essential aspects of research on SEMWSNs [14]. The primary approach taken in extensive farmlands to reduce economic costs is the unplanned distribution of sensor nodes by Unmanned Aerial Vehicles. However, the random deployment of sensor nodes is often haphazardly dispersed, resulting in coverage gaps or repetition in parts of the farmland [15].

Wireless sensor coverage optimization problem can be described as an N-P hard problem, and researchers have done much research on metaheuristic algorithms for solving N-P hard problems. For example, Tanweer et al. proposed a new particle swarm optimization algorithm (PSO) to find the optimal strategy [16]. The inspiration for the ant colony optimization algorithm (ACO) is the foraging behavior of an actual ant colony. This behavior is exploited in artificial ant colonies to search for discrete and continuous optimization problems [17]. Heidari et al. proposed the harris hawk optimization (HHO) algorithm [18] and the hunger game search (HGS) algorithm [19], respectively, based on the predatory behavior of animals. Wang et al. proposed the monarch butterfly optimization (MBO) algorithm inspired by the migratory behavior of American monarch butterflies [20]. Zheng et al. proposed an improved wild horse optimizer (IWHO) for the shortcomings of WHO to optimize the problem of low exploitation capability and stagnation in local optimization [21]. Although these metaheuristics algorithms perform well, they suffer from low availability and evolutionary stagnation in local optimization. Applying them to the coverage problem of SEMWSNs, there are still more coverage gaps and redundancy in the coverage area. The main focus of this paper is to optimize the deployment location of wireless sensor nodes. ACGSOA effectively reduces the monitoring blind spots and maximizes the savings of sensor node resources by using a smaller number of sensor nodes and finally finds a solution that maximizes the coverage of the monitoring field.

This paper proposes a novel coverage optimization pattern for SEMWSNs, and a unique adaptive chaotic Gaussian variant snake optimization algorithm (ACGSOA) is designed. The algorithm uses multiple optimization operators to accelerate the convergence speed and augment the global exploration capability of the algorithm. Relevant simulation experiments show that the approach proposed in this study effectively improves the network coverage rate.

The main contributions of this paper are as follows.

1) This paper proposes a unique adaptive chaotic Gaussian variant snake optimization algorithm. The algorithm combines the advantages of the chaotic operator and adaptive Gaussian variant

operator, which has low time complexity and high efficiency and can perform the SEMWSNs coverage task excellently.

2) Two new operators, the chaotic operator and the adaptive Gaussian variant operator, are designed. The chaotic operator is used to optimize the sensor node position parameters, speeding up the algorithm's convergence. The adaptive Gaussian variant operator not only dramatically enhances the global search capability of ACGSOA but also effectively prevents ACGSOA from falling into local optimum.

3) A new mathematical model for coverage optimization of SEMWSNs is developed. The model considers and optimizes several essential factors affecting the performance of SEMWSNs, including the node sensing distance, the number of nodes, and the coverage area. In addition, the model proposes a new objective function to balance the relationship between these factors and serves as an evaluation criterion for adequate coverage.

4) New relevant simulation experiments are designed to compare ACGSOA with WOA, ABC, and FOA according to different constraints. The results show that ACGSOA has a remarkable improvement in terms of convergence speed, coverage results, and energy saving.

The remainder of the paper is organized as follows: Section 2 presents relevant research on the coverage optimization problem for SEMWSNs. Section 3 displays the coverage optimization pattern for SEMWSNs. Section 4 proposes ACGSOA to deal with the coverage optimization problem for SEMWSNs. Subsequently, there are simulation experiments and discussions of the results in Section 5. The conclusions section is given in Section 6.

2. Related work

Last decades, the coverage of wireless sensors in agriculture had attracted increasing attention. In different application scenarios, the coverage constraints to be considered are not precisely the same, so different deployment schemes arise. Traditional approaches to sensor node deployment are large-scale deployments of static nodes, too many of which can cause data redundancy [22]. Traditional wireless sensor deployment schemes typically use virtual force and computational geometry algorithms. Virtual force and computational geometry algorithms show exclusive advantages in fixed node distribution. However, the main objective of the coverage strategies currently used is to maximize network coverage using a small number of nodes in combination with dynamic scheduling methods [23]. Moreover, with the increasing demand for applications in harsh environments, the long-established haphazard distribution methods are no longer desirable for the coverage optimization of SEMWSNs [24]. Metaheuristic algorithms have been widely used in wireless sensor deployment due to their advantages, such as fewer parameters, easy implementation, and good search performance.

Bionic algorithms based on PSO, ACO, and genetic algorithm (GA), as the first proposed metaheuristic algorithms, received much favor from researchers once they were proposed. Saha et al. [25] proposed an adaptive virtual anchor node based on an improved shortest path algorithm with PSO technique. The method improves the accuracy by reducing the localization error for unknown nodes. However, the method needs to improve the problem that PSO converges slowly and tends to fall into local optimum. Hanh et al. [26] modified local exploration, initialization, and representation of individuals, to elevate the field coverage rate based on GA. Hanh used field integration as an adaptation function to enhance the dependability of the simulation, but the computational complexity of the method is very high. Lee et al. [27] proposed an ant colony-based scheduling algorithm (ACB-

SA) to solve the energy coverage problem. This algorithm can extend the sensor lifetime, but the effect is not significant. In addition, the slow convergence speed of ACO, the tendency to fall into local optimum, and the inapplicability for optimization problems with continuous solution space still need to be improved. Although this strategy effectively speeds up the algorithm's convergence, it increases its time complexity and running time.

Research on metaheuristic algorithms has never stopped. In recent years researchers have proposed much faster and better metaheuristic algorithms, which are also effectively used in various industries. Two teams, Hussien and Sharma, have improved the algorithm's performance by combining the chaotic operator with the harris hawk optimizer (HHO) [28–30] and applied it to wireless sensor networks (WSNs) with good results. Meanwhile, the Hussien team explored and combined the optimization operator [31,32] with the metaheuristic algorithms to enhance the utility of COOT optimization algorithm [33], the remora optimization algorithm (ROA) [34], the water-cycle algorithm (WCA) [35], and the aquila optimizer (AO) [36]. Although the above optimization algorithms have relatively good results, these algorithms still have problems such as slow convergence and relatively poor solution quality in dealing with such high dimensional problems as SEMWSNs coverage. Strumberge et al. applied the monarch butterfly optimization (MBO) algorithm to the task scheduling and WSNs location optimization problems [37], using a multi-stage localization approach to enhance the search capability of the algorithm. At the same time, the team also investigated the moth search algorithm (MSA), optimized the original algorithm, and used it to solve the WSNs location problem [38]. However, several of the above algorithms have certain advantages, but they can easily fall into suboptimal solutions with small solution spaces and poor solution robustness when solving high-dimensional problems. Funda et al. [39] applied the hunger games search (HGS) optimization algorithm to a practical engineering problem. The team combined chaotic operators with HGS, which effectively accelerated the convergence speed of HGS algorithm. However, the algorithm is prone to suboptimal solutions, and the algorithm's robustness is not good. Jesline et al. [40] optimized the butterfly optimization algorithm (BOA) and applied it to the field of WSNs, effectively reducing the overall energy consumption of WSNs and extending the sensor lifetime. However, the algorithm tends to fall into local optima and the complexity of the algorithms increases. Deepa et al. [41] propose a new algorithm called LWOA based on the levy flight mechanism and whale optimization algorithm (WOA) to solve the problem that the randomly deployed sensor node positions are easily trapped in local optima. LWOA significantly improves the search capability of WOA, but the convergence speed is not significantly improved.

To address the problems of slow convergence speed, easy fall into local optimum, poor solution quality, and high algorithm complexity in the abovementioned studies. It is necessary to propose an algorithm with fast convergence speed, strong search capability, robustness, and low algorithm complexity in solving high-dimensional problems. In this study, to maximize cost and wireless sensor resources savings, this paper proposes a unique coverage model for SEMWSNs that considers the number of nodes, coverage area, and power consumption during deployment. In addition, inspired by the snake optimizer (SO) [42], this paper proposes a novel adaptive chaotic Gaussian variant snake optimization algorithm (ACGSOA). ACGSOA has a faster convergence speed, higher coverage quality, and better ability to jump out of the local optimum compared with other metaheuristics and can effectively reduce network power consumption.

3. System model

3.1. Coverage strategy

The coverage problem of SEMWSNs is related to the deployment location and model of the nodes. Generally speaking, depending on the sensing direction, node sensing models can be divided into omnidirectional and directed sensing models. Regarding sensing characteristics, the node sensing models can be classified into Boolean and probabilistic sensing models. Standard SEMWSNs coverage control methods can be divided into target coverage, fence coverage, and area coverage. Target coverage means that the SEMWSNs complete monitoring of a specified number of stationary or moving targets in the monitoring area. Fence coverage refers to the ability of the SEMWSNs to monitor a moving target as it traverses the sensor node deployment area. Depending on the model, fence coverage can be divided into "exposed traversal" and "worst and best coverage". Area coverage refers to the ability of SEMWSNs to complete monitoring coverage of a specified target area. It is usually required that any point in the area is covered by at least one sensor node.

According to the qualities of SEMWSNs, this paper selects an area coverage approach to monitor changes in the content of various elements in soil in agricultural fields. This paper intends to maximize coverage through fewer sensor nodes so that the entire sensor network can monitor soil info within the target area at all times. The area coverage model used in this paper is shown in Figure 1.

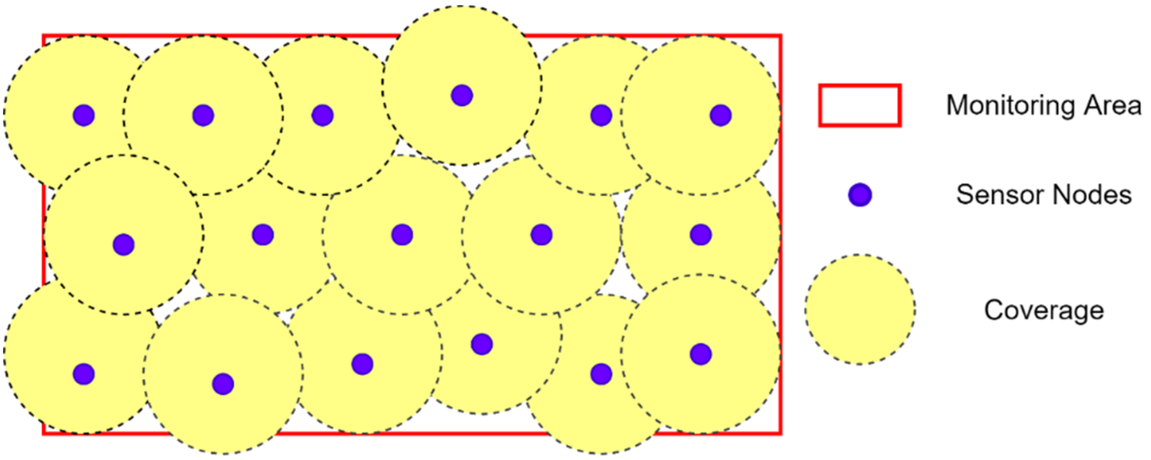


Figure 1. SEMWSNs area coverage model.

For ease of application, the wireless sensor coverage model is further simplified in this paper. Nodes are admitted to placing anywhere within the monitoring region, assuming that the region is two-dimensional.

1) Nodes are placed in the target area. Each node is considered a mass point.

2) The wireless sensor sensing way uses the Boolean sensing model.

3) The sensing distance of each node is ρ, which is taken as 5.

The coverage problem of SEMWSNs can be thought of as randomly placing nodes within the target area and using a population intelligence algorithm to adjust the node positions to achieve maximum coverage ultimately.

3.2. Sensing model

The agricultural soil element monitoring system uses the efficiency of data transmission between wireless sensor nodes as an essential indicator in selecting the transmission way. This paper assumes that the target region $\boldsymbol{A}$ of SEMWSNs is two-dimensional, considered a square field consisting of $W \times L$ pixel blocks. $\boldsymbol{N}$ isomorphic nodes are deployed in the target region. Each node has the same parameter information, radius of communication R_c, and sensing radius ρ. To assure the stability of the SEMWSNs, the set of nodes can be represented as $T = \{t_1, t_2, t_3, \cdots, t_N\}$, where the position of the monitoring node is $p_i = (x_i, y_i)$, and the coordinates of pixel m are assumed to be $m = (x, y)$. Equation (1) represents the distance between the monitoring node and the pixel point.

$$d(p_i, m) = \sqrt{(x - x_i)^2 + (y - y_i)^2} \tag{1}$$

This paper uses the Boolean model as the wireless sensor sensing way. The Boolean model considers that when the length between the monitored pixel and the sensing node is less than or equal to the sensing radius ρ, the target is considered to be sensed with probability 1. In comparison, the target is not sensed when the distance between the monitored target and the sensing node is greater than the sensing radius ρ. The specific formulas of the Boolean model are shown in Eq (2) and Figure 2.

$$p_{cov}(t_i, m) = \begin{cases} 1, & d(t_i, m) \le \rho \\ 0, & d(t_i, m) > \rho \end{cases} \tag{2}$$

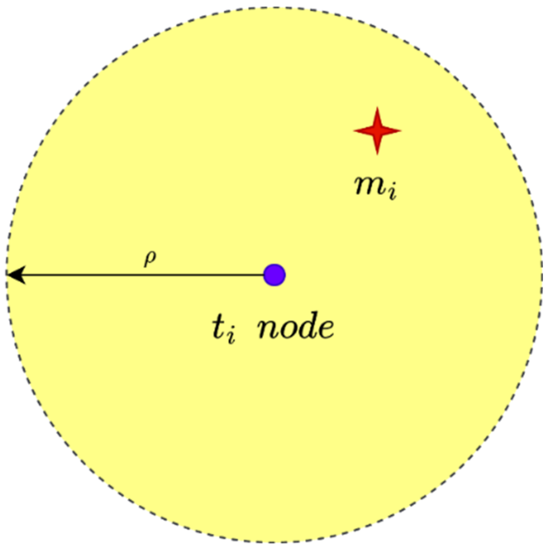


Figure 2. Boolean sensing model.

3.3. Evaluation metrics

Coverage algorithms can be evaluated in a variety of ways. Generally, they can be evaluated in the following three ways.

1) Coverage rate: The primary function of SEMWSNs is to complete the coverage of the monitored area or the monitored target, so coverage capability is an essential criterion for evaluating the coverage protocol or algorithm of SEMWSNs. All nodes are detected as mutually independent

events, and the comprehensive sensing probability of all nodes for a particular target is shown in Eq (3),

$$p_{cov}(T,m) = 1 - \prod_{c_i \in C}\left(1 - p_{cov}(t_i,m)\right) \quad (3)$$

where $\boldsymbol{T}$ denotes all nodes in the target region; in area coverage problems, the coverage rate is usually defined as the ratio of the entire field covered by the SEMWSNs to the field as the whole of the monitored field. It is described as shown in Eq (4),

$$S_{cov}(T) = \frac{\sum P_{cov}(T)}{W \times L} \quad (4)$$

where $\boldsymbol{L}$ is the target region's length, $\boldsymbol{W}$ is the target region's width, and $P_{cov}(T)$ is the sum of the sensing probabilities of all sensor nodes in the target region.

2) Coverage efficiency: Coverage efficiency is usually used to evaluate the utilization of a sensor node within a coverage area. In area coverage, the coverage efficiency is described as the ratio of the range of all nodes in the monitoring area to the sum of the scope of multiple nodes. In the case where the monitored area is completely covered, the higher the coverage efficiency, the smaller the size repeatedly monitored by the node, and the more efficient the utilization of the wireless sensor node. The specific mathematical representation is shown in Eq (5).

$$r_{cov} = \frac{\cup P_{cov}(T)}{\sum P_{cov}(T)} \quad (5)$$

The relation between coverage rate and coverage efficiency is shown in Eq (6).

$$r_{cov} = \frac{S_{cov}(T) \times L \times W}{N \times \pi \times \rho^2} \quad (6)$$

3) Network connectivity: SEMWSNs are self-organizing into networks where all nodes must upload data to the gateway nodes via wireless multi-hop. If the sensor nodes cannot access the network, the nodes cannot transmit the sensed data to the user. At the same time, nodes need to maintain communication with surrounding nodes to collaborate on tasks such as sensing and data fusion, so maintaining network connectivity is critical to the coverage of SEMWSNs.

3.4. Coverage objective function

The coverage problem of SEMWSNs can be studied from two aspects: how to ensure the adequate coverage of nodes over the target area to accurately transmit and collect soil data of the target area. The other is to improve the coverage scope of the SEMWSNs with a reasonable spatial resource allocation to ensure high node resource utilization.

This paper uses the magnitude of coverage to represent the level of coverage of the node over the monitored area. The final strategy in this paper is to constantly monitor the target area with maximum coverage while ensuring that the nodes use fewer nodes at high coverage and that the nodes can collect and transmit soil element data for an extended period. The SEMWSNs coverage mathematical model solution function is shown in Eq (7).

$$f(x) = Max\left(S_{cov}(T)\right) \quad (7)$$

3.5. Power consumption analysis of SEMWSNs

In SEMWSNs, the three components of inter-node communication, data sensing, and data collection and processing are the main components of power consumption. In the SEMWSNs coverage model, this paper simplifies the SEMWSNs by considering only the energy required by the communication module, the transmission power consumption, and the receiving power consumption of each sensor node. Equations (8) and (9) represent the power consumption required by a node to receive and transmit data, respectively.

$$E_t(k,d) = k \times (E_{elec} + \varepsilon_{amp} \times d^n) \tag{8}$$

$$E_r(k) = k \times E_{elec} \tag{9}$$

In Eq (8), $E_t(k,d)$ denotes the energy consumed by a node to send $\boldsymbol{k}$ bit of data to another node, $\boldsymbol{d}$ denotes the distance between two nodes, E_{elec} denotes the electronic energy parameter, ε_{amp} denotes the power amplification factor, $\boldsymbol{n}$ denotes the quality factor of the communication environment where the sensor is located, the worse the communication environment, the larger the value of $\boldsymbol{n}$. The value of n takes the range of 2 to 4. In Eq (9), $E_r(k)$ represents the energy required for a sensor node to receive $\boldsymbol{k}$ bits of data.

The energy required for a sensor node to receive and send data constitutes the total energy $\boldsymbol{E}$ consumed by a sensor node, as shown in Eq (10).

$$E = E_t + E_r \tag{10}$$

Assuming the presence of $\boldsymbol{N}$ nodes in the target region, the total energy consumed by all sensors when communicating can be calculated using Eq (11), where E_{sum} is the total system power consumption.

$$E_{sum} = \sum_{n=1}^{N} E_n \tag{11}$$

4. ACGSOA for improved coverage rate of SEMWSNs

This paper proposes an adaptive chaotic Gaussian variant snake optimization algorithm (ACGSOA) for optimizing the coverage of SEMWSN in the target region. This paper designs a new chaotic operator to make the initial positions of the resulting wireless sensor nodes more uniform and improve the convergence speed of the algorithm. Also, to address the problem of quickly falling into local optima, an adaptive Gaussian variant operator is designed in this paper to complement the exploration capability of the algorithm. Using ACGSOA, maximum coverage of the target area using a smaller number of sensor nodes is achieved, and the overall network power consumption is reduced.

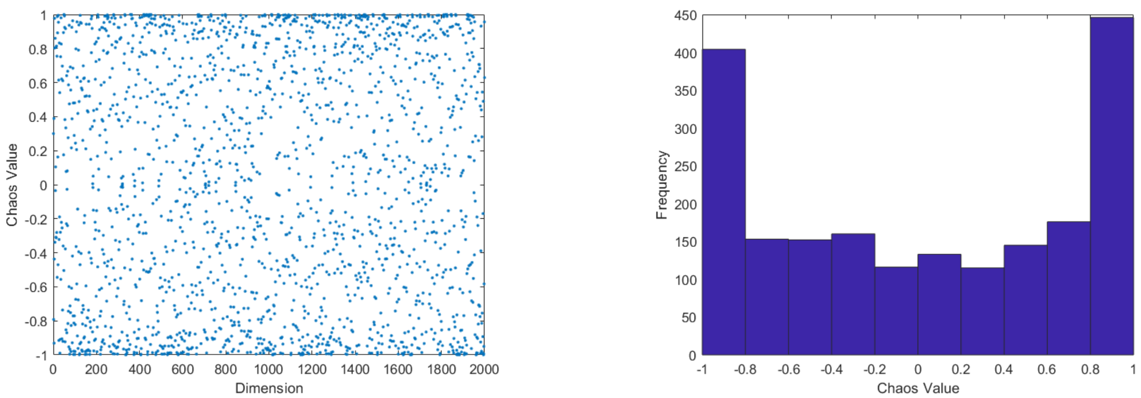
4.1. Snake optimizer

Snake optimizer [42] (SO) is an optimization algorithm proposed by Professors Hashim, F. A. and Hussien, A. G. in 2022, whose algorithm is inspired by the foraging and reproductive behavior and patterns of snakes. SO first generates randomly distributed populations to be able to start the optimization process. SO is mainly influenced by temperature and the amount of food. When the amount of food is more significant than a set threshold, SO searches for food and updates its position relative to the food by selecting random locations, and this phase is the exploration phase. The development phase of SO is relatively complex and is divided into a near-food mode, a combat mode, and a mating mode; all three modes are affected by both temperature and the amount of food.

This paper proposes ACGSOA by combining the novel chaotic operator with the new Gaussian variational operator based on SO, so the relevant mathematical principles of SO and ACGSOA are explained in detail in the subsequent subsections. In ACGSOA, a series of unique behaviors of the snake symbolizes finding the optimal solution for the coverage of SEMWWSNs. ACGSOA process is divided into Chaotic initialization for nodes of SEMWSNs, grouping and adaptive Gaussian variant of ACGSOA's population, coverage rate exploration, and coverage rate development.

4.2. Chaotic initialization for nodes of SEMWSNs

The first step in applying ACGSOA to SEMWSNs coverage optimization problem is determining the initialized wireless sensor node locations. To deploy the initial nodes as uniformly as possible, a novel chaotic initialization strategy is proposed in this paper, which helps to speed up the algorithm's convergence. The chaotic mapping distribution and histogram are shown in Figure 3(a),(b), respectively. According to Figure 3(a), it can be seen that the chaotic operator proposed in this paper is more uniformly distributed.



(a) Chaotic mapping distribution (b) Chaotic Mapping Histogram

Figure 3. Chaotic distribution.

The expression for chaotic sequence generation is shown in Eq (12),

$$z(n+1) = 4z(n)^3 - 3z(n) \quad \{n \in N^*\} \tag{12}$$

where $z(n) \in [-1,1]$. The chaotic sequence is mapped into the space of values of the optimization variables, and the chaotic properties are used to search for the node's initial position. The specific steps are:

1) For $\boldsymbol{M}$ individuals in a $\boldsymbol{D}$-dimensional space, a $\boldsymbol{D}$-dimensional vector $Z = \{z_1, z_2, \cdots, y_d\}$ is randomly generated as the first individual, where $z_i \in [-1,1], 1 \leq i \leq d$.

2) Use Eq (12) for $\boldsymbol{M}$-1 iterations of $\boldsymbol{Y}$ to produce the remaining $\boldsymbol{M}$-1 individuals.

3) The resulting chaotic variables are mapped into the search space of the solution according to Eq (13).

$$x_{id} = x_L + \frac{(1+z_{id})\times(x_u - x_l)}{2} \tag{13}$$

where x_{id} is the location of individual ***i*** in the ***d*-th** dimension; z_{id} is the ***d*-th** dimensional value of individual ***i*** generated by Eq (12); and x_u and x_l are the maximum and minimum of the values taken for x_{id}, respectively.

4.3. Grouping and adaptive Gaussian variation of ACGSOA's population

Applying ACGSOA to SEMWSNs, the individual's fitness in ACGSOA represents the coverage rate of the SEMWSNs. It is necessary to calculate each individual's fitness to find the best individual's current position before proceeding with other operations of ACGSOA. The formula for calculating the fitness is shown in Eq (7). In ACGSOA, populations are divided into two groups, and current fitness values are calculated for the individual in each group. This paper assumes that the number of male individuals is fifty percent and the number of female individuals is fifty percent. Equations (14) and (15) are used to divide the populations,

$$N_m = \left\lfloor \frac{N}{2} \right\rfloor \tag{14}$$

$$N_f = N - N_m \tag{15}$$

where ***N*** is the population size of SEMWSNs, N_m and N_f are the number of males and females, respectively.

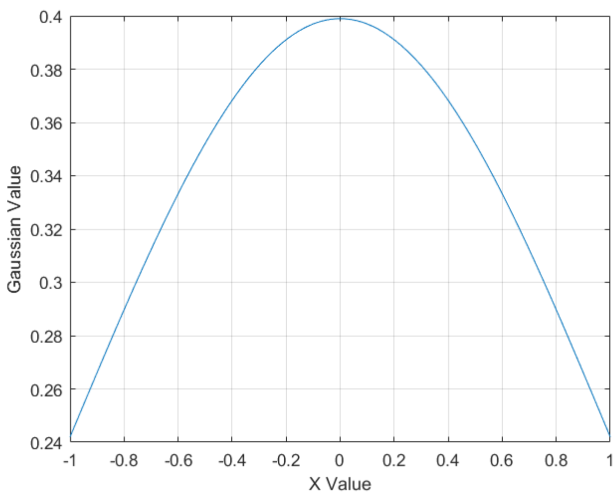


Figure 4. Gaussian distributions.

To avoid ACGSOA from falling into local optimum, after a certain number of individuals are randomly selected, the adaptive Gaussian variant operator is designed in this paper to perform various operations on them. The mutated individuals can be calculated using the following Eqs (16) and (17),

$$V_i = X_i + e \times \left(X_{B_m} - X_i\right) \; i \in [1, N_m] \tag{16}$$

$$V_i = X_i + e \times \left(X_{B_f} - X_i\right) \ i \in [N_m + 1, N] \tag{17}$$

where ***e*** denotes a Gaussian distribution with a variance of 1 and a mean of 0, the Gaussian distribution is shown in Figure 4. X_{Bm} is the most adapted individual in the male population, and X_{Bf} is the most adapted individual in the female population. X_i denotes the individual to be mutated.

4.4. Coverage rate exploration

During the deployment of SEMWSNs, the size of the food in ACGSOA represents the current individual best fitness, which is the best coverage of SEMWSNs, and the location of the food is the present global best fitness location. The population search for the food location is equivalent to the search for the SEMWSNs' best coverage solution. ACGSOA chooses the viable strategies for the coverage problem by the fitness value size, calculated after the individuals move to a new position in each iteration. In the SEMWSNs coverage optimization problem, if the food quantity is less than a set threshold of 0.25, individuals will search for food and update their location by choosing any random location. The formula for defining the number of food items is given in Eq (18). The location update method is shown in Eqs (19) and (21),

$$Q = c_1 \times exp\left(-\frac{g}{G}\right) \tag{18}$$

where ***g*** is the present number of iterations; ***G*** is the maximum number of iterations. c_1 is a constant and is taken as 0.5,

$$X_{i,m}(g+1) = c_2 \times A_m \times \left(X(Xmin_{max})_{rand,m}(g)_{min}\right) \tag{19}$$

where $X_{i,m}$ is the location of the male individual; $X_{rand,m}$ is the location of the randomly selected male individual; $rand \in [0,1]$; A_m is the ability of the male individual to find food, and the food finding ability is calculated as shown in Eq (20),

$$A_m = exp\left(-\frac{f_{rand,m}}{f_{i,m}}\right) \tag{20}$$

where $f_{rand,m}$ is the $X_{rand,m}$ fitness value for the location of a randomly selected male, and $f_{i,m}$ is the $X_{i,m}$ fitness value for the location of a male. c_2 is a constant, taken as 0.5,

$$X_{i,f}(g+1) = c_2 \times A_f \times \left(X(Xmin_{max})_{rand,f}(g+1)_{min}\right) \tag{21}$$

where $X_{i,f}$ is the location of female individuals; $X_{rand,f}$ is the location of randomly selected females; A_f is the ability of females to find food, and the food-finding ability is calculated as shown in Eq (22),

$$A_f = exp\left(-\frac{f_{rand,f}}{f_{i,f}}\right) \tag{22}$$

where $f_{rand,f}$ is the $X_{rand,f}$ fitness value for the location of a randomly selected female individual, and $f_{i,f}$ is the $X_{i,m}$ fitness value for the location of a female individual.

4.5. Coverage rate development

In the SEMWSNs coverage optimization problem, the particular behavior of ACGSOA individuals helps to find a better coverage solution. There are two types of individual behavior, one in which individuals fight each other to update position and the other in which mating between males and females takes place to update position. In ACGSOA, when the amount of food is greater than 0.25, and the temperature is greater than 0.6, individuals will only update their position continuously to find the best fitness. The temperature definition formula is shown in Eq (23). Position updating is shown in Eq (24),

$$Temp = exp\left(-\frac{g}{G}\right) \tag{23}$$

$$X_{i,j}(g+1) = X_{food} \pm c_3 \times Temp \times rand\left(X_{food} - X_{i,j}(g)\right) \tag{24}$$

where $X_{i,j}$ is the individual location, X_food is the best location of the individual, c_3 is a constant, which is taken as 2.

If the amount of food is greater than 0.25 and the temperature is less than 0.6, the individual will be in mating or fight pattern.

1) Fight pattern: Individuals update their position parameters by fighting with each other, with position updates as shown in Eqs (25) and (27),

$$X_{i,m}(g+1) = c_3 \times rand \times F_m \times \left(\left(Q \times X_{best,f}\right) - X_{i,m}(g)\right) + X_{i,m}(g) \tag{25}$$

where $X_{i,m}$ is the location of the ***i-th*** male; $X_{best,f}$ is the best position of the female; and F_m is the male fighting power, as shown in Eq (26),

$$F_m = exp\left(-\frac{f_{best,f}}{f_i}\right) \tag{26}$$

$$X_{i,f}(g+1) = c_3 \times rand \times F_f \times \left(\left(Q \times X_{best,m}\right) - X_{i,f}(g)\right) + X_{i,f}(g) \tag{27}$$

where $X_{i,f}$ is the position of the ***i-th*** female; $X_{best,m}$ is the best male location; and F_f is the female fighting power, as shown in Eq (28).

$$F_f = exp\left(-\frac{f_{best,m}}{f_i}\right) \tag{28}$$

2) Mating pattern: Position update between males and females through mating, with eggs laid after mating, and ACGSOA further determining whether the eggs hatch. The position is updated as shown in Eqs (29) and (30),

$$X_{i,m}(g+1) = c_3 \times rand \times M_m \times \left(\left(Q \times X_{i,f}(g)\right) - X_{i,m}(g)\right) + X_{i,m}(g) \tag{29}$$

$$X_{i,f}(g+1) = c_3 \times rand \times M_f \times \left(\left(Q \times X_{i,m}(g)\right) - X_{i,f}(g)\right) + X_{i,f}(g) \tag{30}$$

where $X_{i,m}$ is the location of the ***i-th*** male; $X_{i,f}$ is the location of the ***i-th*** female; and M_m and M_f are the mating ability of males and females, respectively, as shown in Eqs (31) and (32),

$$M_m = exp\left(-\frac{f_{i,f}}{f_{i,m}}\right) \tag{31}$$

$$M_f = exp\left(-\frac{f_{i,m}}{f_{i,f}}\right) \tag{32}$$

where $f_{i,m}$ is the fitness value of the ***i-th*** male position; $f_{i,f}$ is the fitness value of the ***i-th*** female position; and if the egg hatches, a new individual is produced. The worst male and female positions are found, and the new individual is used to replace the worst individual.

4.6. ACGSOA steps

ACGSOA implementation process is divided into the following steps.

Step 1: Set the relevant parameters in the algorithm. The number of sensors ***N***, the number of populations ***pop_size***, the maximum number of iterations ***G***, and the boundary parameters x_l and x_u.

Step 2: The sensor node locations and randomly generated population locations are initialized using the chaotic operator, and the populations are Gaussian mutated.

Step 3: Divide the population into two groups.

Step 4: Calculate the population fitness and find the best individuals.

Step 5: Calculate temperature and food quantity.

Step 6: Determine if ***Q*** is greater than the food threshold of 0.25. If ***Q*** is less than the food threshold, find the current global optimal fitness according to Eqs (19) and (21). If ***Q*** exceeds the food threshold, determine whether ***Temp*** is greater than the temperature threshold of 0.6.

Step 7: If ***Temp*** is greater than the temperature threshold, the individual position is updated according to Eq (24). If ***Temp*** is less than the temperature threshold, further determine whether ***rand*** > 0.6 or ***rand*** < 0.6.

Step 8: If ***rand*** > 0.6, the population enters fight mode, and the position is updated according to Eqs (25) and (27). If ***rand*** < 0.6, the population enters mating mode and is updated according to Eqs (29) and (30).

Step 9: Replace the worst individual and update the global best fitness.

Step10: If the total number of iterations ***G*** does not arrive, skip to step 4. If the maximum number of iterations ***G*** has arrived, output the coverage strategy.

The flowchart of ACGSOA is shown in Figure 5.

4.7. ACGSOA complexity analysis

In this paper, the size of the population is set to ***N***, the maximum number of iterations is ***G***, and the solving spatial dimension is ***D***. The time complexity of ACGSOA is analyzed according to the criterion related to the algorithm's time complexity. The final result is that the time complexity of population initialization using the chaotic operator in ACGSOA is $O(N)$, the time complexity of updating the position using the Gaussian variant operator change is $O(D)$, the time complexity of finding the local optimum is $O(N \times D)$, and the total time complexity of ACGSOA is $O(G \times N \times D)$. As a result, the overall time complexity of ACGSOA is comparable to that of SO and does not increase the operational cost of the algorithm.

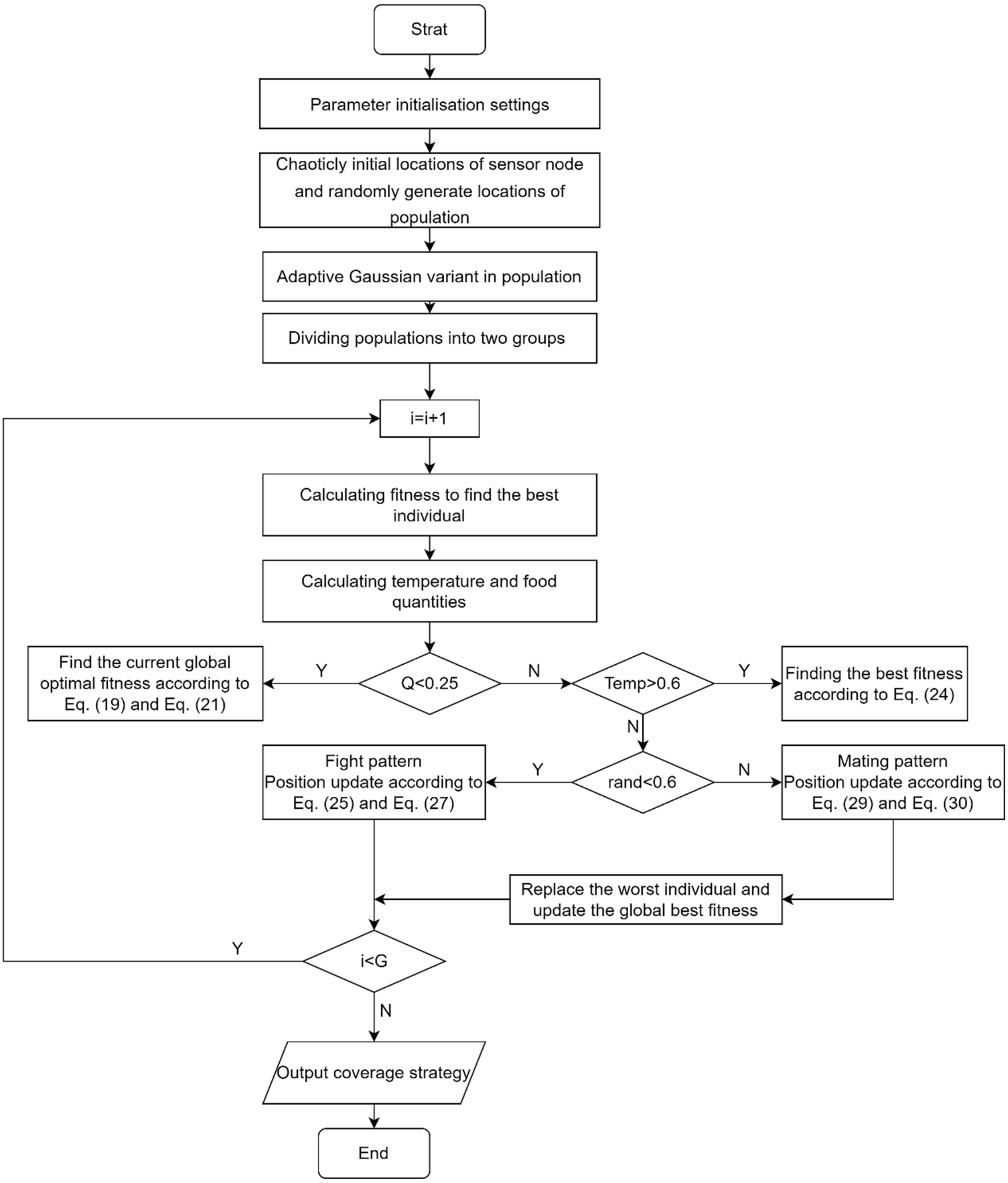


Figure5. Flowchart of ACGSOA in SEMWSNs.

The pseudo-code of ACGSOA proposed in this paper is shown in Algorithm 1.

Algorithm 1. Adaptive Chaotic Gaussian Variant Snake Optimization Algorithm
1: Initialize Problem Setting ($\boldsymbol{M}$, ***pop_size***, $\boldsymbol{G}$, $\boldsymbol{i}$, x_l, and x_u)
2: The randomly generated population positions are initialized using chaotic operators using Eqs (12) and (13).
3: Adaptive Gaussian variant in population.
4: Divide population ***pop_size*** to 2 equal groups N_m and N_f using Eqs (14) and (15).
5: while $(i < G)$ do
6:     Find the best individual

Continued on next page

Algorithm 1. Adaptive Chaotic Gaussian Variant Snake Optimization Algorithm

Calculate temperature (***Temp***) and food quantity (***Q***) using Eqs (18) and (23)
if ($Q < 0.25$) then
  Find the current global optimal fitness using Eqs (19) and (21)
else
   if ($Temp > 0.6$) then
      update individual position using Eq (24)
   else
      if ($rand < 0.6$) then
         Fight pattern Eqs (25) and (27)
      else
         Mating pattern Eqs (29) and (30)
         replace the worst individual and update the global best fitness
      end if
   end if
end if
end while
Return best solution.

5. Results and discussion

To verify the validity of ACGSOA proposed in this paper in improving the coverage of SEMWSNs, a series of simulation experiments were conducted. ACGSOA is compared with SO, WOA, ABC, and FOA. The results of the experiments covered in this paper are averages derived from the effects of 100 experiments. The simulation experiments included comparing node deployment locations, algorithm running time, network power consumption, and coverage. Different types of simulations were performed with different numbers of sensors to reflect the usefulness of SEMWSNs. In addition, all simulations were acted on a computer equipped with an i5-12400F CPU @ 3.20GHz, and the fitness function used in the algorithms follows Eq (7).

To compare the performance of different algorithms for SEMWSNs coverage optimization, a unified standard parameter is used to ensure the objectivity and impartiality of the simulation experiments. In the simulation experiments, the population size of all algorithms was set to 30, and the maximum number of iterations was set to 1000. Detailed parameter settings for the four algorithms are shown in Table 1.

This paper assumes that the sensor nodes of SEMWSNs are deployed in a square monitoring area of $W \times L$. Table 2 lists the experimental parameters of the SEMWSNs node deployment area and the simulation results for different constraints. All the results in Table 2 are the average results obtained after 100 simulations. When the simulation area is 50×50 m^2, the sensing radius R_s of the sensor nodes is set to 5 m, and the communication radius R_c is set to 10 ± 2 m. When the simulation area is 30×30 m^2, R_s is set to 2.5 m, and R_c is set to 5 ± 1 m. The number of sensor nodes is indicated by ***N*** and consists of 54, 44, 33, and 27 sensor nodes.

Table 1. Detailed parameter settings for the algorithm.

Name of algorithm	Basic parameters of the algorithm
ACGSOA	$pop = 30,\ G = 1000$, $c_1 = 0.5$, $c_2 = 0.05$, $c_3 = 2$
SO	$pop = 30,\ G = 1000$, $c_1 = 0.5$, $c_2 = 0.5$, $c_3 = 2$
WOA	$pop = 30,\ G = 1000,\ a = 2\ to\ 0,\ b = 1,\ l = [-1,1]$
ABC	$pop = 30,\ G = 1000,\ limit = 20,\ a = 1$
FOA	$pop = 30,\ G = 1000,\ s = 0.2$

Table 2. Results of ACGSOA compared with SO, WOA, ABC and FOA under different constraints.

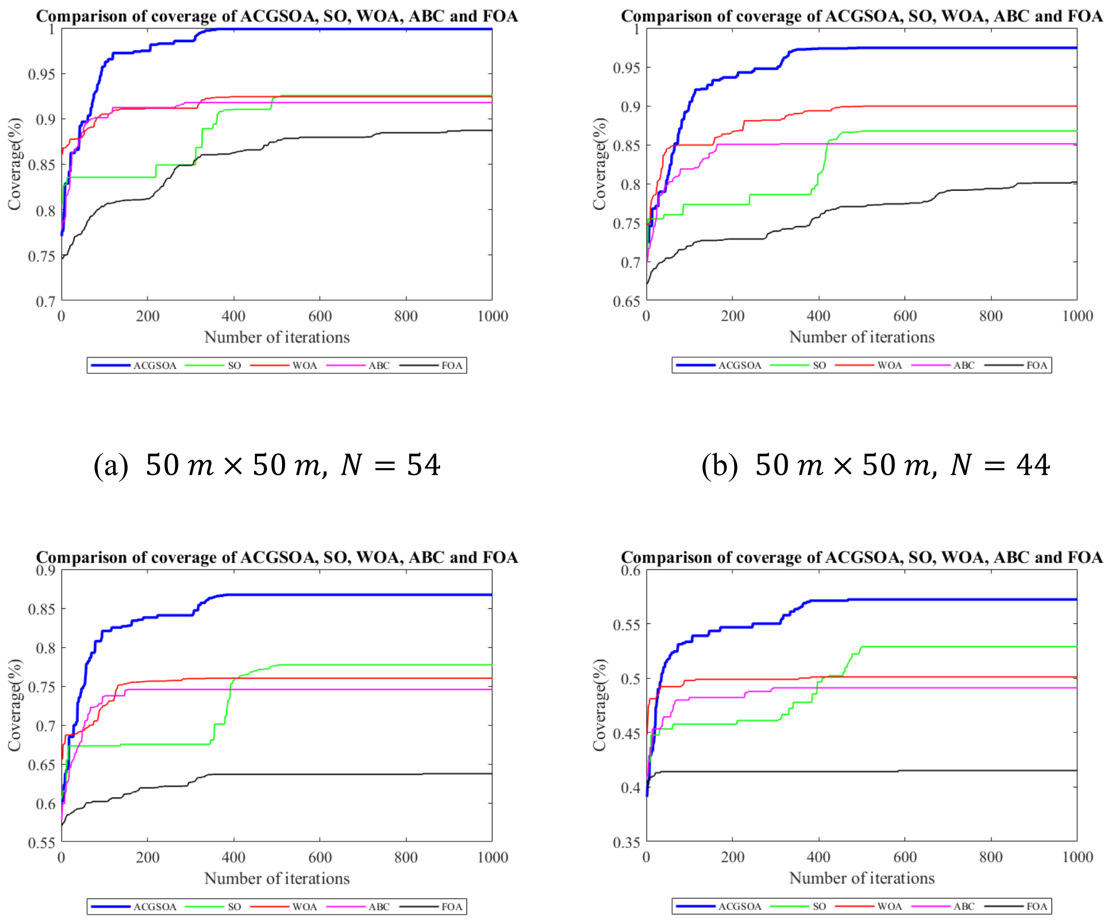
Algorithm	Factor Variables	50 × 50 $N = 54$	50 × 50 $N = 44$	50 × 50 $N = 33$	30 × 30 $N = 27$
ACGSOA	Coverage rate (%)	99.76	97.24	87.24	57.00
	Consumed execution time (s)	17.491	13.292	9.820	3.66
SO	Coverage rate (%)	92.56	86.76	77.72	52.89
	Consumed execution time (s)	24.06	22.12	20.61	7.30
WOA	Coverage rate (%)	92.44	89.96	76.00	50.11
	Consumed execution time (s)	19.27	14.06	11.75	4.70
ABC	Coverage rate (%)	91.80	85.12	74.56	49.11
	Consumed execution time (s)	18.45	14.28	11.73	5.33
FOA	Coverage rate (%)	88.73	80.20	63.74	41.52
	Consumed execution time (s)	36.15	30.33	21.99	8.54

Figure 6(a)–(d) visualizes ACGSOA, SO, WOA, ABC, and FOA simulation results for optimizing coverage of SEMWSNs under different constraints. The data in the figures are the average results obtained after 100 simulation experiments were conducted. Figure 6 shows that this paper's proposed ACGSOA node coverage optimization has the most robust performance, indicating the best quality when applied to SEMWSNs. According to Figure 6(a), the optimized ACGSOA was used for SEMWSNs under $50m \times 50m,\ N = 54$, and the final ACGSOA node coverage is all higher than that of SO, WOA, ABC, and FOA. The coverage rates for SO, WOA, ABC, and FOA were 92.56%, 92.44%, 91.80%, and 88.73%, respectively. By viewing Figure 4(a), FOA has a poor initial probability of 75% and the algorithm's slow convergence rate, and the result is not satisfactory at 88.73.

Analyzing FOA, it is due to the inability of FOA to make full use of the population information that results in FOA falling into local extremes and a decrease in convergence accuracy. For SO, WOA, and ABC, the coverage results are similar, with SO and WOA being slightly better, SO, WOA and ABC are improved compared to FOA. However, there are still some problems. This paper can see that SO converges slowly by analyzing SO. According to Table 2, This paper can learn that SO takes longer to run compared to both WOA and BOA algorithms, and the reason for this is that SO divides the population into two groups. The two groups cannot make full use of the location information. WOA and ABC suffer from premature convergence, low solution accuracy, and the same tendency to fall into localization. In ACGSOA, a new chaotic operator is designed to enhance the convergence speed of the algorithm, and an adaptive Gaussian variant operator is also designed to improve the search capability of ACGSOA, which is conducive to jumping out of the local extreme. Under the condition of $50m \times 50m,\ N = 54$, the experiment results show that the coverage of ACGSOA is as high as 99.76%, and the coverage rate of ACGSOA is improved by 7.20%, 7.32%, 7.96%, and 11.03%, respectively, compared with the other four algorithms. The above result shows that the improvement method proposed in this paper can effectively enhance ACGSOA's optimization-seeking ability and get the best coverage rate with fewer iterations.

Figure 6(a) shows that the convergence speed of ACGSOA is significantly improved compared to the other four algorithms. WOA reaches a maximum of 92.44% after 400 iterations. ABC reaches a maximum of 91.80% after 300 iterations. SO reaches a maximum of 92.56% after 500 iterations. FOA reaches a maximum of 88.73% after 900 iterations. However, ACGSOA reached 95.80% after 100 iterations. The above results demonstrate that the novel chaotic operator and the new adaptive Gaussian variational operator designed in this paper effectively improve the algorithm's convergence speed and global search ability. Further analysis of Figure 6(b)–(d) shows the same conclusion.

Figure 6(b) shows a ten wireless sensor nodes reduction compared to Figure 6(a). ACGSOA optimized coverage in Figure 6(b) is also as high as 97.24%, which is 4.68%, 4.8%, 5.44%, and 8.51% higher than SO, WOA, ABC, and FOA coverage results in Figure 6(a), respectively. Comparing the results of the two photos demonstrates that ACGSOA can achieve better coverage with fewer nodes deployed and can effectively reduce deployment costs. The same conclusion is reached by analyzing Figure 6(c),(d).



(a) $50\,m \times 50\,m,\ N = 54$ (b) $50\,m \times 50\,m,\ N = 44$

(c) $50\,m \times 50\,m,\ N = 33$ (d) $30\,m \times 30\,m,\ N = 27$

Figure 6. Iteration curves under different constraints.

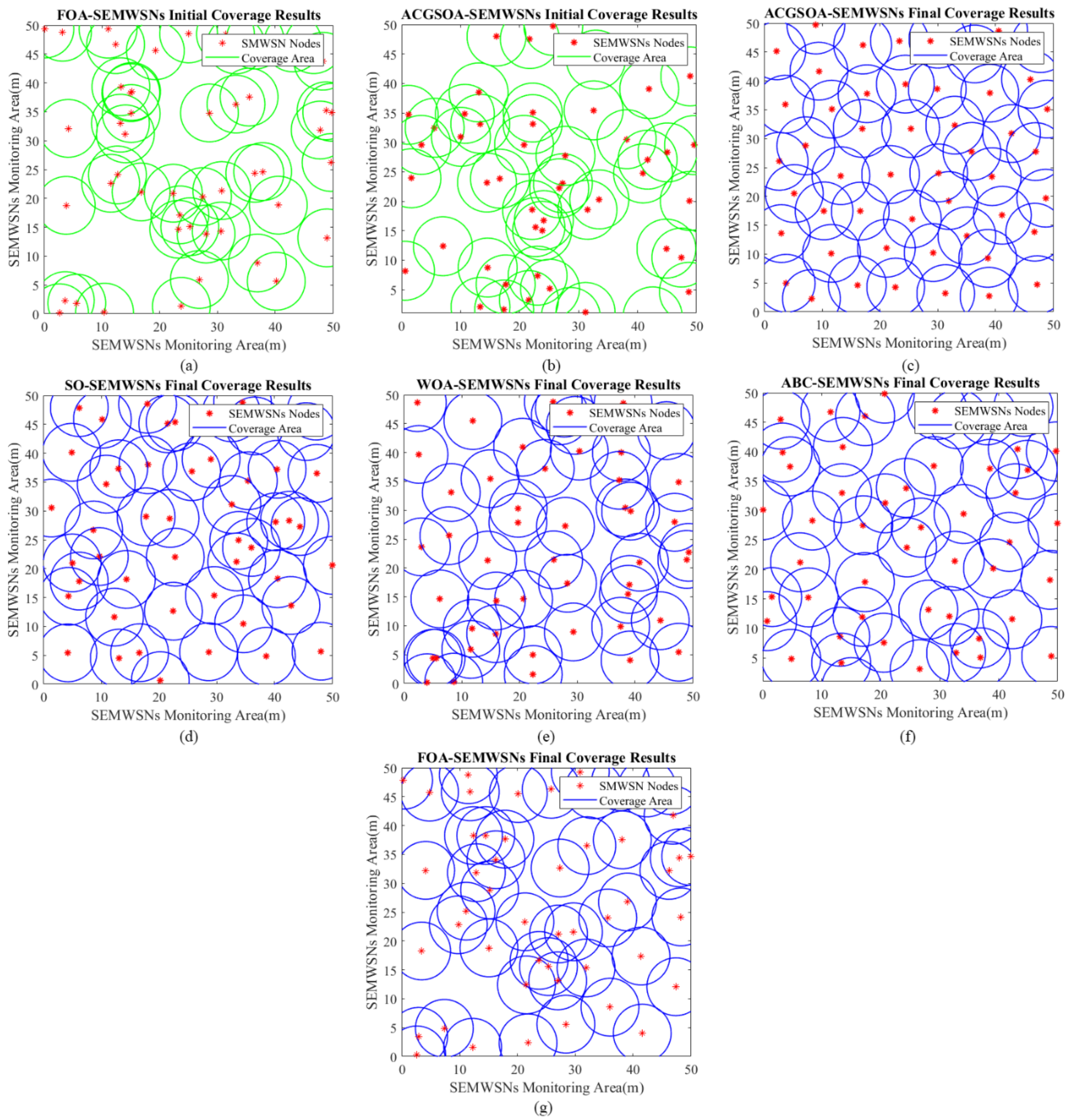


Figure 7. Coverage results after algorithm optimization with $50\ \mathrm{m} \times 50\ \mathrm{m}$, $N = 54$.

Figure 7(a), (b) shows the initial coverage effect of FOA and ACGSOA in the context of constraint (1). Based on the images, it can be learned that the distribution of randomly deployed nodes in FOA and ACGSOA is highly uneven. The initial distribution includes a large amount of coverage redundancy and coverage gaps, which seriously wastes unlimited sensor node resources and increases the deployment cost. Figure 7(c)–(g) shows the coverage results of applying the five algorithms to SEMWSNs in the context of constraint (1). According to Figures 7(e) and 5(g), it can be seen that the optimized deployment results using WOA and FOA show a better distribution result. However, there are still more coverage redundant and gaps, and the node coverage after optimization by two algorithms still needs to be optimized. Figure 7(d), (f), and (c) show that the application of SO, ABC, and

ACGSOA to SEMWSNs achieves better results, but the coverage rate is better with ACGSOA. Using ACGSOA algorithm for SEMWSNs node coverage optimization yields a more uniform node distribution, and the result has fewer coverage gaps and redundancy relative to other algorithms. The maximum coverage of ACGSOA is almost one hundred percent.

Figure 8(a)–(e) shows the coverage results of applying the four algorithms to SEMWSNs in the context of constraint (2). According to Figure 8, it can be seen that ACGSOA is still the algorithm with the highest coverage rate when reducing the number of nodes by a certain amount and is equally close to complete coverage. Comparing Figure 8(a) with Figure 7(b)–(d), it is easy to conclude that ACGSOA has a more uniform distribution of nodes, significantly fewer coverage gaps, redundancy, and the best coverage. This result shows that ACGSOA has less impact on coverage and lower economic cost in reducing the number of SEMWSNs nodes, maximizing the benefits. Under the same target region size, node sensing distance, and communication distance conditions, this paper explores the trend of coverage change as the number of nodes changes. The simulation results show that the higher the number of nodes at a given range, the better the coverage effect and the smaller the coverage redundancy and gaps within the monitoring area.

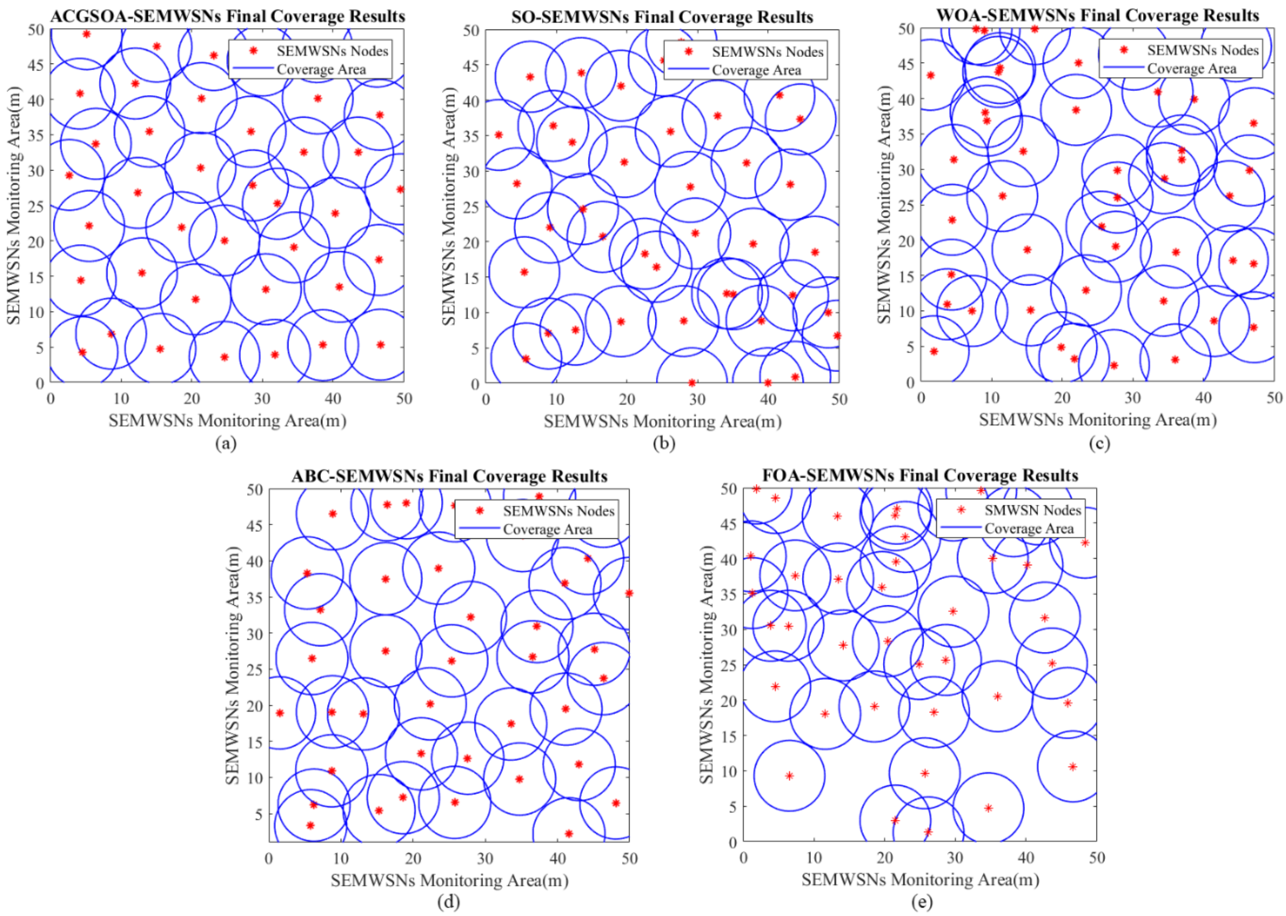


Figure 8. Coverage results after algorithm optimization with $50\ \mathrm{m} \times 50\ \mathrm{m}$, $N = 44$.

In this paper, a failure rate of 8% was set at the time of deployment to prevent individual node failures from causing a drop in coverage rate. For the simulation experiments, the actual number of nodes used in Figures 7 and 8 are 50 and 40, respectively.

Figure 9 represents the variation of ACGSOA, SO, WOA, ABC, and FOA coverage with the number of sensor nodes in the SEMWSNs. It was found that the coverage of SEMWSNs increased with the number of nodes. When the number of nodes is the same, ACGSOA has a more significant advantage over SO, WOA, ABC, and FOA. With the same parameters, ACGSOA can be better adapted to different numbers of sensor nodes, allowing for a more comprehensive application of coverage optimization. Also, the increased network coverage indicates improved network monitoring quality and data accuracy, making ACGSOA advisable for coverage optimization of SEMWSNs.

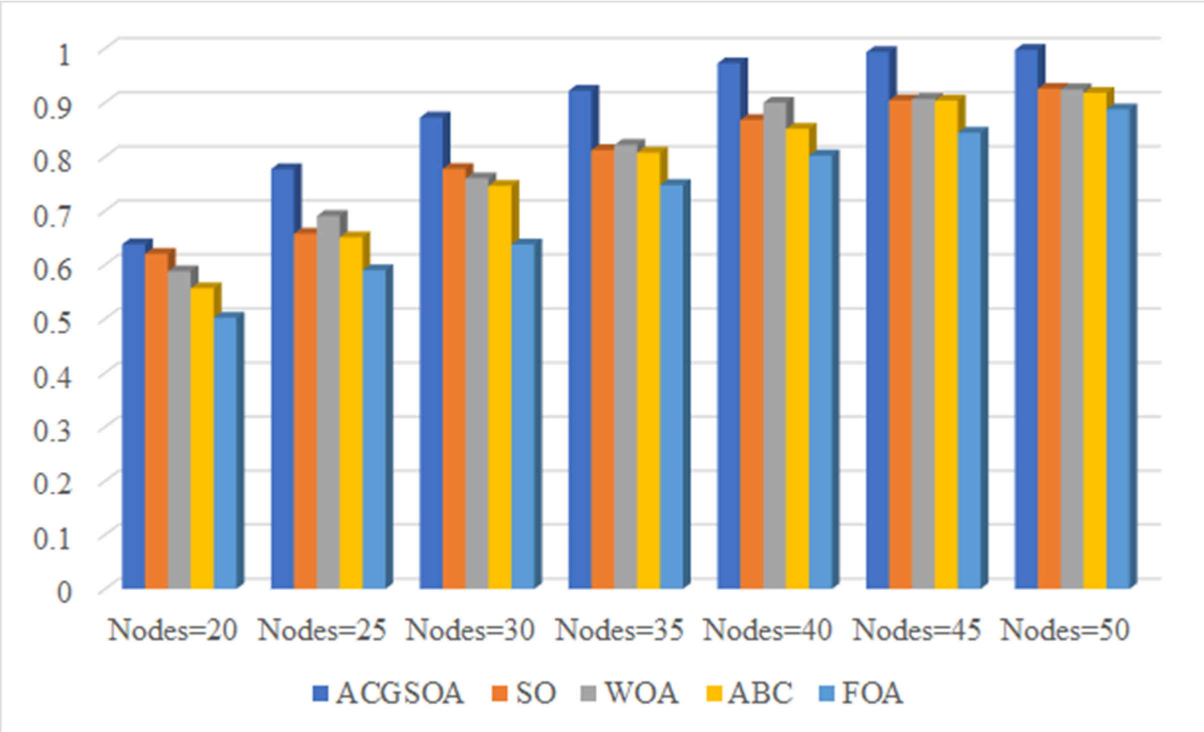


Figure 9. Trends in coverage of SEMWSNs.

According to the above analysis and results, it can be learned that ACGSOA dramatically improves the coverage of SEMWSNs compared to SO, WOA, ABC, and FOA. Compared with the other three algorithms, ACGSOA achieves 95% coverage using fewer nodes. At the same time, SO, WOA, ABC, and FOA deploy too many nodes and are prone to coverage redundancy, affecting data transmission accuracy. According to Figure 9, this paper calculates the power consumption of SEMWSNs with different deployment algorithms when SEMWSNs are deployed. Some critical parameters of the power consumption calculation model in SEMWSNs are as follows: $E_{elec} = 50\, nJ/bit$, $k = 1\, Mbit$, $n = 3$, $\varepsilon_{amp} = 100\, pJ/bit/m^2$. The power consumption results of different algorithms, as shown in Table 3. All the results in Table 3 are the average results obtained after 100 simulations. This paper focuses on calculating the power consumption generated by exchanging data with all other nodes within the communication range of a certain node. The results show that SEMWSNs show the highest power consumption after optimal deployment using FOA, while SEMWSNs exhibit the lowest power consumption after optimal deployment using ACGSOA. ACGSOA proposed in this paper accomplishes the objective of reducing the power consumption of SEMWSNs.

Table 3. Power consumption of different coverage algorithms.

	$N = 35$	$N = 40$	$N = 45$	$N = 50$
ACGSOA	49.97 (J)	63.93 (J)	79.11 (J)	91.23 (J)
SO	63.01 (J)	78.35 (J)	94.53 (J)	126.63 (J)
WOA	62.08 (J)	74.06 (J)	106.79 (J)	122.71 (J)
ABC	54.27 (J)	76.17 (J)	103.56 (J)	116.97 (J)
FOA	73.11 (J)	80.17 (J)	114.16 (J)	138.86 (J)

6. Conclusions

To improve the coverage of SEMWSNs, a novel adaptive chaotic Gaussian variant snake optimization algorithm (ACGSOA) is designed in this paper. The innovation of ACGSOA lies in the design of various optimization operators to enhance the performance of ACGSOA. A new chaotic operator is designed, which speeds up the algorithm's convergence. A new Gaussian variant operator is also proposed, effectively preventing SEMWSNs from falling into local optima during deployment. Subsequently, ACGSOA is compared with SO, WOA, ABC, and FOA to demonstrate its effectiveness for coverage optimization of SEMWSNs. According to the simulation results, the coverage capability of ACGSOA far exceeds that of other algorithms and effectively saves node resources. The application of ACGSOA in the coverage optimization of SEMWSNs significantly improves the coverage rate while reducing the number of nodes used and provides a research basis for further development of intelligent agriculture.

Although ACGSOA proposed in this paper has demonstrated its superior performance through simulation, it still needs some improvement due to the limitations of research capability and environmental conditions. In this paper, the sensor nodes in SEMWSNs are statically and randomly distributed in the monitoring area. However, some application scenarios require the sensor nodes to be distributed as mobile monitoring data. In the future, the distribution of sensor nodes as mobile monitoring data to improve the practicality and adaptability of the algorithm will be investigated in some application scenarios. Future research can place the network in a 3D scene and be heterogeneous. The impact of environmental factors such as temperature, noise, and obstacles on SEMWSNs nodes will be considered in the future.

Acknowledgments

This paper was funded by the National Natural Science Foundation of China, grant number 61962053, the Shihezi University High-level Talent Research Start-up Fund Project, grant number RCZK2018C39, the project of Youth and middleaged Scientific and Technological Innovation Leading Talents Program of the Corps, grant number 2018CB006, the Corps innovative talents plan, grant number 2020CB001, the China Postdoctoral Science Foundation, grant number 220531, Funding Project for High Level Talents Research in Shihezi University, grant number RCZK2018C38, Project of Shihezi University, grant number ZZZC201915B.

Conflict of interest

The authors declare no competing interests.

Supplementary

The datasets generated during this current study are not publicly available due to privacy but are available from the corresponding author upon reasonable request. Email: tm_inf@shzu.edu.cn.

References

1. A. Rejeb, K. Rejeb, A. Abdollahi, F. Al-Turjman, H. Treiblmaier, The Interplay between the Internet of Things and agriculture: A bibliometric analysis and research agenda, *Internet Things*, **19** (2022), 100580. https://doi.org/10.1016/j.iot.2022.100580
2. T. Ojha, S. Misra, N. S. Raghuwanshi, Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges, *Comput. Electron. Agric.*, **118** (2015), 66–84. https://doi.org/10.1016/j.compag.2015.08.011
3. S. M. Mohamed, H. S. Hamza, I. A. Saroit, Coverage in mobile wireless sensor networks (M-WSN): A survey, *Comput. Commun.*, **110** (2017), 133–150. https://doi.org/10.1016/j.comcom.2017.06.010
4. J. Zhou, Y. Zhang, Z. Li, R. Zhu, A. zeman, Stochastic scheduling of a power grid in the presence of EVs, RESs, and risk index with a developed lightning search algorithm, *J. Cleaner Prod.*, **364** (2022), 132473. https://doi.org/10.1016/j.jclepro.2022.132473
5. M. K. B. Melhem, L. Abualigah, R. A. Zitar, A. G. Hussien, D. Oliva, Comparative study on arabic text classification: challenges and opportunities, in *Classification Applications with Deep Learning and Machine Learning Technologies*, Springer, Cham, (2023), 217–224. https://doi.org/10.1007/978-3-031-17576-3_10
6. N. Milhem, L. Abualigah, M. H. Nadimi-Shahraki, H. Jia, A. E. Ezugwu, A. G. Hussien, Enhanced MapReduce performance for the distributed parallel computing: application of the big data, in *Classification Applications with Deep Learning and Machine Learning Technologies*, Springer, Cham, (2023), 191–203. https://doi.org/10.1007/978-3-031-17576-3_8
7. C. Li, Y. Liu, Y. Zhang, M. Xu, J. Xiao, J. Zhou, A novel nature-inspired routing scheme for improving routing quality of service in power grid monitoring systems, *IEEE Syst. J.*, **2022** (2022), 1–12. https://doi.org/10.1109/jsyst.2022.3192856
8. L. Cao, Y. Yue, Y. Cai, Y. Zhang, A novel coverage optimization strategy for heterogeneous wireless sensor networks based on connectivity and reliability, *IEEE Access*, **9** (2021), 18424–18442. https://doi.org/10.1109/access.2021.3053594
9. J. Lloret, S. Sendra, L. Garcia, J. M. Jimenez, A wireless sensor network deployment for soil moisture monitoring in precision agriculture, *Sensors*, **21** (2021), 7243. https://doi.org/10.3390/s21217243
10. J. Amutha, S. Sharma, J. Nagar, WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: review, approaches and open issues, *Wireless Pers. Commun.*, **111** (2020), 1089–1115. https://doi.org/10.1007/s11277-019-06903-z
11. S. Q. Ong, G. Nair, R. D. A. Dabbagh, N. F. Aminuddin, P. Sumari, L. Abualigah, et al., Comparison of pre-trained and convolutional neural networks for classification of jackfruit artocarpus integer and artocarpus heterophyllus, in *Classification Applications with Deep Learning and Machine Learning Technologies*, Springer, Cham, (2023), 129–141. https://doi.org/10.1007/978-3-031-17576-3_6

12. S. Verma, N. Sood, A. K. Sharma, Genetic algorithm-based optimized cluster head selection for single and multiple data sinks in heterogeneous wireless sensor network, *Appl. Soft Comput.*, **85** (2019), 105788. https://doi.org/10.1016/j.asoc.2019.105788
13. R. Priyadarshi, P. Rawat, V. Nath, Energy dependent cluster formation in heterogeneous wireless sensor network, *Microsyst. Technol.*, **25** (2019), 2313–2321. https://doi.org/10.1007/s00542-018-4116-7
14. R. Moazenzadeh, B. Mohammadi, Assessment of bio-inspired metaheuristic optimisation algorithms for estimating soil temperature, *Geoderma*, **353** (2019), 152–171. https://doi.org/10.1016/j.geoderma.2019.06.028
15. Y. Liu, C. Li, Y. Zhang, M. Xu, J. Xiao, J. Zhou, HPCP-QCWOA: High performance clustering protocol based on quantum clone whale optimization algorithm in integrated energy system, *Future Gener. Comput. Syst.*, **135** (2022), 315–332. https://doi.org/10.1016/j.future.2022.05.001
16. M. R. Tanweer, S. Suresh, N. Sundararajan, Self regulating particle swarm optimization algorithm, *Inf. Sci.*, **294** (2015), 182–202. https://doi.org/10.1016/j.ins.2014.09.053
17. C. Blum, Ant colony optimization: Introduction and recent trends, *Phys. Life Rev.*, **2** (2005), 353–373. https://doi.org/10.1016/j.plrev.2005.10.001
18. A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, *Future Gener. Comput. Syst.*, **97** (2019), 849–872. https://doi.org/10.1016/j.future.2019.02.028
19. Y. Yang, H. Chen, A. A. Heidari, A. H. Gandomi, Hunger games search: Visions, conception, implementation, deep analysis, perspectives, and towards performance shifts, *Expert Syst. Appl.*, **177** (2021), 114864. https://doi.org/10.1016/j.eswa.2021.114864
20. G. G. Wang, S. Deb, Z. Cui, Monarch butterfly optimization, *Neural Comput. Appl.*, **31** (2015), 1995–2014. https://doi.org/10.1007/s00521-015-1923-y
21. R. Zheng, A. G. Hussien, H. M. Jia, L. Abualigah, S. Wang, D. Wu, An improved wild horse optimizer for solving optimization problems, *Mathematics*, **10** (2022), 1311. https://doi.org/10.3390/math10081311
22. R. Priyadarshi, B. Gupta, A. Anurag, Wireless sensor networks deployment: a result oriented analysis, *Wireless Pers. Commun.*, **113** (2020), 843–866. https://doi.org/10.1007/s11277-020-07255-9
23. J. Liang, M. Tian, Y. Liu, J. Zhou, Coverage optimization of soil moisture wireless sensor networks based on adaptive Cauchy variant butterfly optimization algorithm, *Sci. Rep.*, **12** (2022), 1–17. https://doi.org/10.1038/s41598-022-15689-3
24. H. Mostafaei, M. U. Chowdhury, M. S. Obaidat, Border surveillance With WSN systems in a distributed manner, *IEEE Syst. J.*, **12** (2018), 3703–3712. https://doi.org/10.1109/jsyst.2018.2794583
25. S. Saha, R. K. Arya, Adaptive virtual anchor node based underwater localization using improved shortest path algorithm and particle swarm optimization (PSO) technique, *Concurrency Comput. Pract. Experi.*, **34** (2021), e6552. https://doi.org/10.1002/cpe.6552
26. N. T. Hanh, H. T. T. Binh, N. X. Hoai, M. S. Palaniswami, An efficient genetic algorithm for maximizing area coverage in wireless sensor networks, *Inf. Sci.*, **488** (2019), 58–75. https://doi.org/10.1016/j.ins.2019.02.059

27. J. W. Lee, J. J. Lee, Ant-colony-based scheduling algorithm for energy-efficient coverage of WSN, *IEEE Sens. J.*, **12** (2012), 3036–3046. https://doi.org/10.1109/jsen.2012.2208742
28. A. G. Hussien, L. Abualigah, R. Abu Zitar, F. A. Hashim, M. Amin, A. Saber, et al., Recent advances in harris hawks optimization: A comparative study and applications, *Electronics*, **11** (2022), 1919. https://doi.org/10.3390/electronics11121919
29. A. G. Hussien, M. Amin, A self-adaptive Harris Hawks optimization algorithm with opposition-based learning and chaotic local search strategy for global optimization and feature selection, *Int. J. Mach. Learn. Cybern.*, **13** (2022), 309–336. https://doi.org/10.1007/s13042-021-01326-4
30. R. Sharma, S. Prakash, HHO-LPWSN: harris hawks optimization algorithm for sensor nodes localization problem in wireless sensor networks, *EAI Endorsed Trans. Scalable Inf. Syst.*, **8** (2018), e5. https://doi.org/10.4108/eai.25-2-2021.168807
31. A. G. Hussien, A. A. Heidari, X. Ye, G. Liang, H. Chen, Z. Pan, Boosting whale optimization with evolution strategy and Gaussian random walks: an image segmentation method, *Eng. Comput.*, **2022** (2022), 1–45. https://doi.org/10.1007/s00366-021-01542-0
32. G. I. Sayed, G. Khoriba, M. H. Haggag, A novel chaotic salp swarm algorithm for global optimization and feature selection, *Appl. Intell.*, **48** (2018), 3462–3481. https://doi.org/10.1007/s10489-018-1158-6
33. R. R. Mostafa, A. G. Hussien, M. A. Khan, S. Kadry, F. A. Hashim, Enhanced coot optimization algorithm for Dimensionality Reduction, in *2022 Fifth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*, IEEE, Riyadh, Saudi Arabia, (2022), 43–48. https://doi.org/10.1109/WiDS-PSU54548.2022.00020
34. S. Wang, A. G. Hussien, H. Jia, L. Abualigah, R. Zheng, Enhanced remora optimization algorithm for solving constrained engineering optimization problems, *Mathematics*, **10** (2022), 1696. https://doi.org/10.3390/math10101696
35. A. G. Hussien, F. A. Hashim, R. Qaddoura, L. Abualigah, A. Pop, An enhanced evaporation rate water-cycle algorithm for global optimization, *Processes*, **10** (2022), 2254. https://doi.org/10.3390/pr10112254
36. H. Yu, H. Jia, J. Zhou, A. G. Hussien, Enhanced Aquila optimizer algorithm for global optimization and constrained engineering problems, *Math. Biosci. Eng.*, **19** (2022), 14173–14211. https://doi.org/10.3934/mbe.2022660
37. I. Strumberger, M. Tuba, N. Bacanin, E. Tuba, Cloudlet scheduling by hybridized monarch butterfly optimization algorithm, *J. Sens. Actuator Networks*, **8** (2019), 44. https://doi.org/10.3390/jsan8030044
38. I. Strumberger, E. Tuba, N. Bacanin, M. Beko, M. Tuba, Wireless sensor network localization problem by hybridized moth search algorithm, in *2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC)*, IEEE, Limassol, Cyprus, (2018), 316–321. https://doi.org/10.1109/IWCMC.2018.8450491
39. F. K. Onay, S. B. Aydemir, Chaotic hunger games search optimization algorithm for global optimization and engineering problems, *Math. Comput. Simul.*, **192** (2022), 514–536. https://doi.org/10.1016/j.matcom.2021.09.014
40. J. Daniel, S. F. V. Francis, S. Velliangiri, Cluster head selection in wireless sensor network using tunicate swarm butterfly optimization algorithm, *Wireless Networks*, **27** (2021), 5245–5262. https://doi.org/10.1007/s11276-021-02812-x

41. R. Deepa, R. Venkataraman, Enhancing Whale Optimization Algorithm with Levy Flight for coverage optimization in wireless sensor networks, *Comput. Electr. Eng.*, **94** (2021), 107359. https://doi.org/10.1016/j.compeleceng.2021.107359
42. F. A. Hashim, A. G. Hussien, Snake Optimizer: A novel meta-heuristic optimization algorithm, *Knowledge-Based Syst.*, **242** (2022), 108320. https://doi.org/10.1016/j.knosys.2022.108320