

# Coverage Optimization Strategy of UWSNs Based on Improved Sparrow Search Algorithm

Lu Liu<sup>1</sup>, Yuming Mao<sup>2,\*</sup>

<sup>1</sup> School of Information Science and Electrical Engineering, Shandong Jiaotong University, Jinan 250000, China

<sup>2</sup> School of Information Science and Electrical Engineering, Shandong Jiaotong University, Jinan 250000, China

\* Corresponding author: Yuming Mao (Email: 34293593@qq.com)

**Abstract:** Randomly distributed nodes lead to low coverage rate and low connectivity in the underwater wireless sensor networks (UWSNs). An UWSNs node coverage optimization strategy based on the Improved Sparrow Search Algorithm (ISSA) is designed to improve coverage rate and connectivity. The ISSA algorithm effectively balances the global and local optimization capabilities of nodes by introducing nonlinear inertia weights and Levy flight strategy, thereby avoiding the algorithm from getting trapped in local optima. Simulation experiments demonstrate that ISSA achieves optimal deployment results in a short time. It not only improves network coverage but also maintains good network connectivity and reduces node redundancy, thereby effectively reducing network energy consumption. Compared to other common algorithms, ISSA can increase network coverage up to 8.4% and improve network utilization up to 11.8%, and meets the deployment requirements.

**Keywords:** Underwater Wireless Sensor Networks; Coverage optimization; Sparrow Search Algorithm; Connectivity.

## 1. Introduction

An Underwater Wireless Sensor Network (UWSN) is a network composed of a large number of sensor nodes randomly distributed underwater, possessing sensing and communication capabilities, and collaborating in a self-organizing manner<sup>[1,2]</sup>. UWSNs can sense the state of the network coverage area, and monitor objects. Currently, UWSNs have been widely applied in fields such as ocean monitoring, maritime disaster early warning, and national defense security<sup>[3]</sup>. In UWSNs, network coverage can reflect the service quality of a network<sup>[4]</sup>, and how to improve the networks' coverage is a hot research topic.

At present, some results have been achieved to improve coverage of UWSNs<sup>[5,6]</sup>. Arivudainambi proposed a Cuckoo Search algorithm to improve network coverage and reduce energy consumption, aiming to enhance network coverage efficiency and prolong network lifespan<sup>[7]</sup>. Khan describes an ADAC-BC algorithm that enables nodes to deploy and adjust independently to achieve optimal coverage in wireless sensor networks where nodes can move freely and are evenly distributed<sup>[8]</sup>. Deng Xiaohua proposed a spring-based coverage optimization algorithm designed to effectively avoid coverage holes or abnormal structures<sup>[9]</sup>. Hashim optimized the relevant parameters of a hybrid Wireless Sensor Network using the Artificial Bee Colony algorithm while limiting the number of sink nodes. This not only improved the network's coverage performance but also reduced network energy consumption<sup>[10]</sup>. Mao Yuming proposed an anchor node deployment algorithm based on the spring system model. This algorithm abstracts anchor nodes as interconnected nodes through springs, allowing some anchor nodes to undergo stretching and contraction movements under the action of a resultant force, thereby improving network performance<sup>[11]</sup>. Guo Chao designed an inertia-weight-based Simulated Particle Swarm Optimization algorithm, enhancing global search capability, reducing algorithm runtime, and minimizing repeated coverage<sup>[12]</sup>. Liu Peng proposed a Spark-based parallel genetic algorithm to optimize

the deployment of underwater sensor networks by computing the extremum of the Shubert multi-modal function<sup>[13]</sup>. In 3D space, Al Turjman optimized the deployment of relay nodes while considering network lifetime and reliability<sup>[14,15]</sup>.

This article proposes a coverage algorithm for Underwater Wireless Sensor Networks based on the Improved Sparrow Search Algorithm (ISSA) to eliminate the shortcomings of existing algorithms. By introducing nonlinear inertia weights and a Levy flight strategy, ISSA balances the global search and local exploitation processes, overcoming the issue of the original Sparrow Search Algorithm easily getting stuck in local optima and enhancing the algorithm's optimization capability. Through simulations, this algorithm demonstrates significant advantages in optimizing UWSNs deployment, offering both fast convergence and excellent deployment effectiveness and network connectivity, thus meeting the deployment requirements of UWSNs.

## 2. Node Coverage Model

### 2.1. Description of Underwater Space Coverage Problem

In practical applications, in order to improve the coverage rate of UWSNs, it is necessary to solve the coverage hole problem caused by randomly throwing sensor nodes. The solution is to use the relevant algorithms for overlapping coverage, eliminating holes, and enhancing network coverage, so as to achieve effective monitoring of underwater areas. In this paper, a method of discretization is proposed, which draws on the idea of grid division of two-dimensional plane coverage. The underwater monitoring area is divided into several grids, and the target points are abstracted into grid points, which are dealt with in the form of point coverage problem. This method effectively resolves the underwater area coverage problem, improving network coverage and monitoring effectiveness. In practical applications, in order to improve the coverage rate of UWSNs, it is necessary to solve the coverage hole problem caused by randomly throwing sensor nodes. The solution is to use the relevant algorithms

for overlapping coverage, eliminating holes, and enhancing network coverage ,so as to achieve effective monitoring of underwater areas. In this paper, a method of discretization is proposed, which draws on the idea of grid division of two-dimensional plane coverage. The underwater monitoring area is divided into several grids, and the target points are abstracted into grid points, which are dealt with in the form of point coverage problem. This method effectively resolves the underwater area coverage problem, improving network coverage and monitoring effectiveness.

## 2.2. Underwater Wireless Sensor Node Sensing Model

The sensing model is a theoretical model, which is used to describe the abstract features of the sensing range and sensing strength of nodes in UWSNs. Therefore, the primary task of UWSNs research is to establish appropriate node sensing models that facilitate the optimization of node deployment and coverage. In this paper, the underwater spherical binary perception model is adopted, as shown in Figure 1:

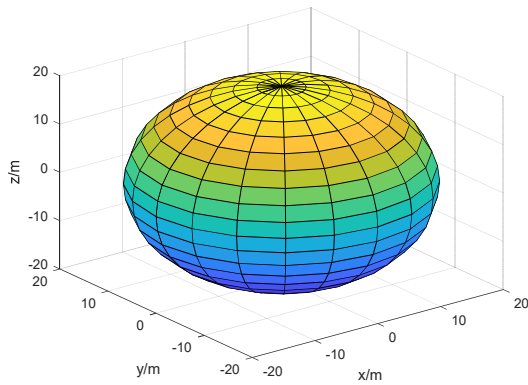


Figure 1. Underwater Wireless Sensor Node Sensing Model

In the underwater space, the coordinates of the moving target  $q$  are represented as  $(x_q, y_q, z_q)$ , and the coordinates of the  $n$ th sensor node are given as  $(x_n, y_n, z_n)$ . The Euclidean distance between two points, denoted as  $d(s, q)$ , signifies the straight-line distance between them. When  $d(s, q)$  is less than or equal to the sensing radius  $r$ , sensor node  $S_n$  can perceive the moving target  $q$ ; otherwise, it cannot perceive it. The expression for  $d(s, q)$  is as follows:

$$d(s, q) = \sqrt{(x_q - x_n)^2 + (y_q - y_n)^2 + (z_q - z_n)^2} \quad (1)$$

The probability that the moving target  $q$  is detected by the  $n$ th sensor node  $S_n$  is denoted as  $p(s, q)$ , where  $p(s, q)$  is defined as follows:

$$p(s, q) = \begin{cases} 1 & d(s, q) < r \\ 0 & d(s, q) \geq r \end{cases} \quad (2)$$

## 2.3. Coverage Calculation

Assuming the monitoring area is a cubic space with dimensions of  $n \times n \times n$  ( $m^3$ ), it is discretized into  $n$  equal divisions along the  $x$ ,  $y$ , and  $z$  axes, resulting in a total of  $n^3$  intersections. During the initialization of sensor nodes, the count of coverable intersections, denoted as  $p_{cover}$ , is set to 0. Subsequently, the distances  $D(i, n)$  between the  $i$ th sensor and these intersections are calculated individually. The

calculation formula is as follows:

$$D(i, n) = \sqrt{(x_q - x_n)^2 + (y_q - y_n)^2 + (z_q - z_n)^2} \quad (3)$$

When the intersection can be perceived by the sensor node,  $p_{cover}$  is incremented by 1; otherwise, it remains unchanged.

$$P_{cover} = \begin{cases} P_{cover} + 1 & 0 < D(i, n) \leq r \\ P_{cover} & D(i, n) > r \end{cases} \quad (4)$$

Therefore, the formula for sensor node coverage rate is as follows:

$$R_{cover} = \frac{P_{cover}}{n^3} \quad (5)$$

Using equation (5) as the objective function for solving the coverage optimization problem in UWSNs with the Improved Sparrow Search Algorithm, we aim to find the maximum coverage rate  $R_{cover}$ .

## 3. Basic Principles and Algorithm Introduction

### 3.1. Basic Sparrow Search Algorithm

The Sparrow Search Algorithm is inspired by foraging and anti-predatory behaviors in sparrow populations. It models the behavior of sparrow populations when searching for food by dividing the population into two types: discoverers and joiners. Discoverers are responsible for exploring the search space, providing search information to attract joiners, and establishing warning mechanisms to protect the population from predators. This algorithm has been widely used to solve optimization problems.

In the sparrow population, discoverers with higher fitness values find food faster. The formula for updating the position of discoverers is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot Iter_{max}}\right) & R_2 < ST \\ X_{i,j}^t + Q \cdot L & R_2 \geq ST \end{cases} \quad (6)$$

In the above equation,  $X_{i,j}^t$  represents the positional information of the  $i$ th sparrow in the  $j$ th dimension, where  $t$  denotes the current iteration count, and  $Iter_{max}$  represents the maximum number of iterations.  $\alpha$  represents a positive number less than 1, while  $R_2$  and  $ST$  represent the warning value and safety value, respectively, with their values falling within the ranges  $[0,1]$  and  $[0.5,1]$ .  $Q$  represents a random number following a normal distribution.  $L$  represents a multidimensional matrix, with each element equal to 1.

Followers optimize their positions by following the update strategy of discoverers, further searching for the optimal solution. The formula for updating follower positions is as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{X_{worst}^t - X_{i,j}^t}{i^2}\right) & i > \frac{n}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot L & i \leq \frac{n}{2} \end{cases} \quad (7)$$

In the equation above,  $X_p$  represents the current best position of the discoverer,  $X_{\text{worst}}$  represents the current global worst position, and  $A$  represents a  $1 \times d$  matrix with each element randomly assigned as 1 or -1.

Due to the threat of natural predators, security considerations are essential to successfully obtain food. Therefore, to address potential risks, 10% to 20% of individuals in the population are chosen to serve as sentinels. The update formula for sentinel positions is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^t + \beta \cdot |X_{i,j}^t - X_{\text{best}}^t| & f_i > f_g \\ X_{i,j}^t + K \left( \frac{|X_{i,j}^t - X_{\text{worst}}^t|}{(f_i - f_w)} \right) & f_i = f_g \end{cases} \quad (8)$$

In the equation above,  $X_{\text{best}}$  represents the global best position,  $\beta$  represents a random number following a standard normal distribution, and  $K$  belongs to the range  $[-1, 1]$ .  $f_i$  denotes the fitness value of the current individual sparrow.  $f_g$  and  $f_w$  respectively represent the fitness values of the global best and global worst individuals.

## 3.2. Improved Sparrow Search Algorithm

### 3.2.1. Nonlinear Inertia Weight

In metaheuristic algorithms, finding a balance between global exploration and local exploitation is crucial. A significant issue in the Sparrow Search Algorithm is the lack of proper control over step size. Once the optimal solution is found, other individuals quickly converge to the vicinity of the optimal solution, making it challenging for the algorithm to effectively control both the global exploration and local exploitation processes. This often leads to the algorithm getting trapped in local optima.

To address this issue, a nonlinear inertia weight is introduced to control the search range and convergence rate, allowing for a better balance between global and local exploration. Larger weight values can increase the search range, promoting global exploration, while smaller weight values are more conducive to local exploitation, accelerating the convergence rate and avoiding getting trapped in local optima. The formula for calculating the nonlinear inertia weight is as follows:

$$\omega = \exp \left( 1 - \frac{t_{\text{max}} + t}{t_{\text{max}} - t} \right) \quad (9)$$

After incorporating the nonlinear inertia weight, the position update formula for discoverers is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^t \cdot \omega & R_2 < ST \\ X_{i,j}^t + Q & R_2 \geq ST \end{cases} \quad (10)$$

### 3.2.2. Levy Flight Strategy

In the standard Sparrow Search Algorithm, the roles of individuals are simplistic, which can lead to individuals of the same role converging to the same optimal solution, causing the algorithm to get stuck in local optima. For high-dimensional complex problems, discoverers only conduct random searches within a broader range than followers, and this approach can limit the algorithm's ability to escape local optima. Therefore, there is a need to enhance the standard Sparrow Search Algorithm to strike a balance between global

exploration and local exploitation, thereby improving the algorithm's search efficiency and accuracy.

Hence, the Levy flight strategy<sup>[16]</sup> is introduced in the follower position update. Levy flight can assist the algorithm in searching for the optimal solution more effectively and enable it to traverse more distant regions in the solution space during the search.

The random step size for Levy flight is as follows:

$$S = \frac{\mu}{|v|^{1/\varphi}} \quad (11)$$

Where  $\mu$  and  $v$  are random numbers following a normal distribution, and  $\mu \sim N(0, \sigma_\mu^2)$ ,  $v \sim N(0, \sigma_v^2)$ . The definitions of  $\sigma_\mu$  and  $\sigma_v$  are as follows:

$$\begin{cases} \sigma_\mu = \left( \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] \beta 2^{(\beta-1)/2}} \right)^{1/\beta} \\ \sigma_v = 1 \end{cases} \quad (12)$$

In the equation,  $\Gamma$  represents the standard gamma function. The value of  $\beta$  typically falls within the interval  $[0, 2]$ , and in this paper,  $\beta$  is set to 1.5.

After incorporating Levy flight for joiners, their position update formula is as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp \left( \frac{X_{\text{worst}}^t - X_{i,j}^t}{t^2} \right) & i > \frac{n}{2} \\ X_p^{t+1} + |X_{i,j}^t - X_p^{t+1}| \cdot A^+ \cdot L \cdot S & i \geq \frac{n}{2} \end{cases} \quad (13)$$

## 3.3. Coverage Optimization Strategy

ISSA aims to maximize the detection coverage area in the water domain by optimizing the positions of sensor nodes in the network. During the optimization process, each sparrow represents a set of sensor deployment nodes. Therefore, the number of sparrows corresponds to the number of generated solutions. Through the evaluation and selection of these solutions, the optimal sparrow represents the best coverage scheme. The coverage optimization strategy for UWSNs based on ISSA follows the process outlined below:

a) Initialization of parameters. Parameters including coverage area (Area), communication radius (R), number of nodes (N), population size (pop), maximum iteration (Max-iteration), spatial dimension (dim), the number of producers (PD), the number of sparrows who perceive the danger (SD), safety threshold (ST), etc., are initialized.

b) Population initialization. The sparrow population is randomly initialized, and their fitness values are computed based on equation (5). Sparrows are sorted based on their fitness values to identify the best and worst individuals.

c) Position update. Discoverers update their positions using equation (10), joiners update their positions using equation (11), and sentinels update their positions using equation (12).

d) Check if the current iteration count has reached the predefined maximum value. If it has, the execution stops, and the results are returned. Otherwise, proceed to step 3) for the next iteration.

The corresponding workflow for the above steps is

depicted in Figure 2.

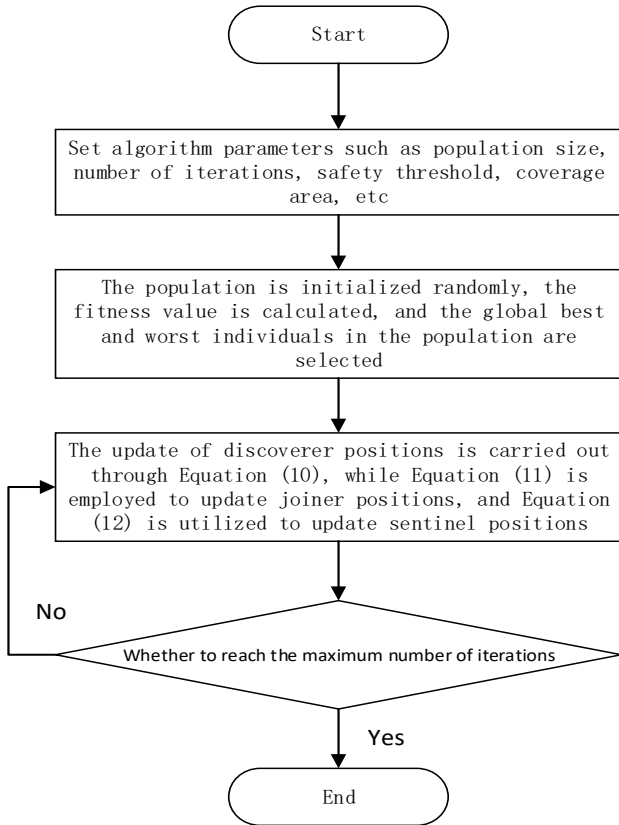


Figure 2. Coverage Optimization Strategy Flowchart

## 4. Literature References

### 4.1. Experimental Environment

It is assumed that the monitoring area is a cube with three sides length  $L = 100$ , the number of underwater sensor nodes  $N$  is 10,20,30,40,50 respectively, and the sensing radius of underwater sensor  $R = 15$ . The sparrow search algorithm parameter settings are shown in Table 1. In this paper, Matlab R2018a software is used for simulation experiments to evaluate the performance of ISSA in UWNs node coverage.

Table 1. Algorithm Parameter Settings

parameter name	parameter values
pop	30
Max-iteration	50
PD	0.2
SD	0.2
ST	0.7

### 4.2. Experimental Environment

Figure 3. shows the simulation results of the sensor node deployment without optimization when  $N=30$ . The simulation results show that there is an overlap in the sensing range of sensor nodes, which leads to unnecessary waste of resources. Furthermore, the distance between some sensor nodes is too far to communicate, which reduces the connectivity of the sensor.

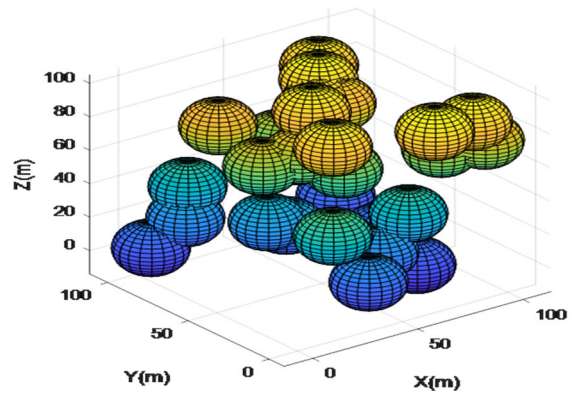


Figure 3. Unoptimized Sensor Node Deployment Results ( $N=30$ )

Figure 4. shows the deployment effect of sensor nodes optimized by ISSA when  $N = 30$ . After ISSA optimization, the overlap among sensor nodes observed in Fig. 3 is eliminated, ensuring the connectivity of the sensor network across the entire deployment area. The optimized deployment makes the sensor nodes evenly distributed. From the results of Figure 5, it can be seen that after optimization, the coverage of sensor networks has increased significantly, which has been significantly improved compared with that before optimization. When the number of nodes is 40 and 50, the improvement effect is the most obvious, which can be improved by nearly 13.8 %.

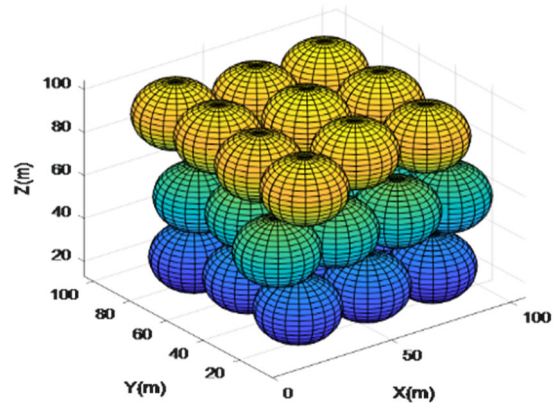


Figure 4. Optimized Sensor Node Deployment Results ( $N=30$ )

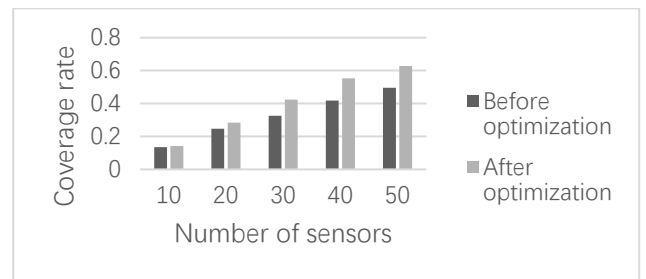


Figure 5. Coverage Rates for Different Numbers of Sensor Nodes under the ISSA Algorithm

Node utilization in sensor networks is an important index to evaluate the degree of redundancy, which is determined by comparing the actual coverage with the theoretical coverage. The higher node utilization indicates that the utilization of sensor networks is more efficient, thereby reducing

redundancy and overlap. In other words, when the node utilization increases, it means that the utilization of sensor nodes is more effective, which effectively avoids the waste of resources and the problem of repeated coverage. Since the

sensor model used in this paper is a sphere binary perception model, the theoretical coverage of the sensor network is  $N * \frac{4 \pi r^3}{3} / L^3$ .

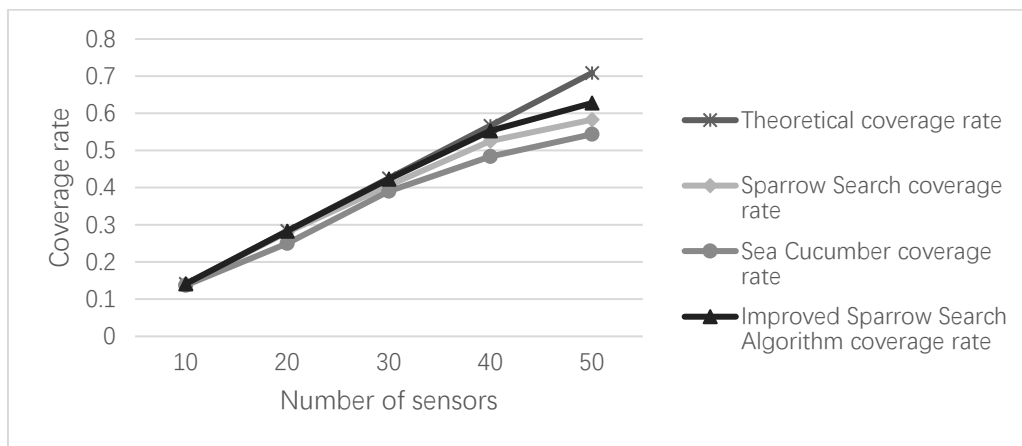


Figure 6. Sensor Node Coverage Rates under Different Algorithms.

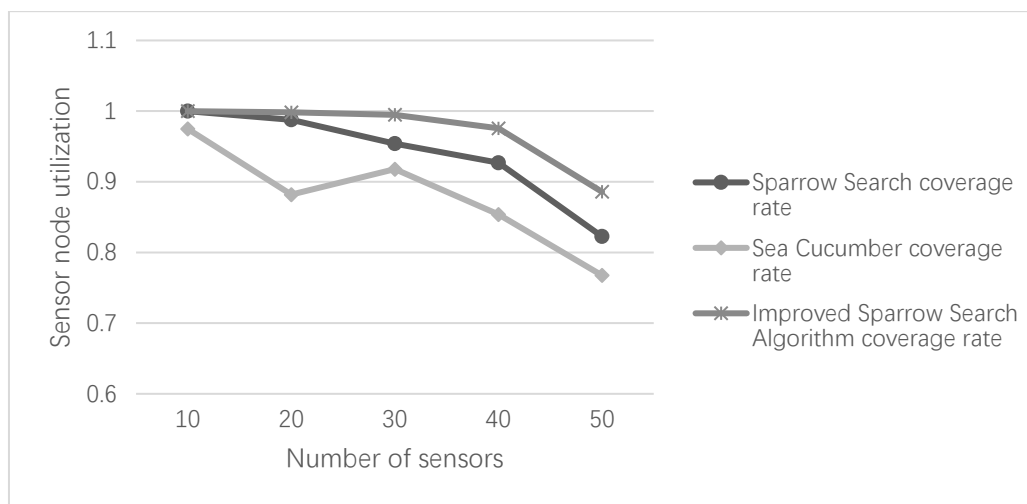


Figure 7. Sensor Node Utilization under Different Algorithms.

Figure 6. and Figure 7. show that the coverage and node utilization of ISSA are better than other algorithms. When the number of sensors is less than 40, the proposed algorithm can approach the theoretical coverage, and the node utilization rate reaches 97.6 % after algorithm optimization, which can effectively reduce the energy loss of UWSNs. However, as the number of sensors increases, the gap between the actual network coverage and the theoretical network coverage gradually expands, and the node utilization rate decreases. This is due to the increase in the number of sensor nodes, the effect of the optimization algorithm is weakened. When reaching a certain number, further increasing the number of sensor nodes will no longer significantly increase the network coverage, but will increase redundancy and overlap, resulting in reduced node utilization.

## 5. Conclusion

Aiming at the problems of uneven distribution of nodes, low coverage rate and low connectivity rate due to random deployment of underwater nodes, an improved sparrow search algorithm for underwater wireless sensor network node coverage optimization strategy is proposed. By introducing nonlinear inertia weight and Levy flight strategy, the global and local optimization of nodes are effectively

balanced, and the algorithm is effectively avoided from falling into local optimal solution. Simulation experiments show that compared with the standard sparrow algorithm and salp algorithm, ISSA has better network deployment effect, and maintains higher node utilization while improving coverage, which can effectively reduce network energy consumption. Compared with other algorithms, ISSA can increase the coverage rate by up to 8.4 %, and the network utilization rate is increased by 11.8 %, which meets the deployment requirements of UWSNs.

## References

- [1] K.Y.Chen, m.d. Ma, E. Cheng, et al. A survey on MAC protocols for underwater wireless sensor networks[J]. IEEE communications Surveys&Tutorials,2014,16(3): 1433-1447.
- [2] Z.W. Guo,H.J. Luo,F. Hong, et al. Current progress and research issues in underwater sensor network [J]. Computer Research and Development, 2010,47 (3):377-389.
- [3] Mahalle P N, Shelar P A, Shinde G R, et al. Applications of UWSN[M]. The Underwater World for Digital Data Transmission. Springer, Singapore,2021:55-64.
- [4] Choudhary M, Goyal N. Node Deployment Strategies in Underwater Wireless Sensor Network [C] International

- Conference on advance Computing and Innovative Technologies in Engineering,2021:773-779.
- [5] J.H. Luo, Fan Liying,S. Wu, et al. Research on localization algorithms based on acoustic communication for underwater sensor networks[J]. *Sensors*,2017,18(1):67.
- [6] Yashwanth N, Sujatha B R. Node Localization Performance for UWSN Topologies[J]. *Journal of Communications* Vol. 14, No. 9, September,2019,14(9):839-844.
- [7] D. Arivudainambi, S. Balaji, T. S. Poorani. Sensor deployment for target coverage in underwater wireless sensor network [C] *International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks*,2017:1-6.
- [8] Alam S M N, Haas Z J. Coverage and connectivity in three-dimensional networks with random node deployment[J]. *Ad Hoc Networks*,2015,34:157-169.
- [9] X.H. Deng, Z.Y. Yu, R.X. Tang, et al. An optimized node deployment solution based on a virtual spring force algorithm for wireless sensor network applications[J]. *Sensors*, 2019,19(8):1817.
- [10] Hashim H A, Ayinde B O, Abido M A. Optimal placement of relay nodes in wireless sensor network using artificial bee colony algorithm[J]. *Journal of Network and Computer Applications*, 2016, 64: 239-248.
- [11] Y.M. Mao. Optimal deployment of anchor nodes based on3D spring model system[J]. *Telecommunication technology*, 2020,37(04):1170-1173.
- [12] C. Guo,Y.X. Yang,R.L. Hu, et al. WSN coverage optimization based on particle swarm optimization [J]. *Research on Computer Application*, 2020,37(04):1170-1173.
- [13] P. Liu, S. Ye, C.Wang, et al. Spark-Based Parallel Genetic Algorithm for Simulating a Solution of Optimal Deployment of an Underwater Sensor Network[J]. *Sensors*,2019, 19(12):2717.
- [14] F. M. Al-Turjman, H.S.H., and M. A. Ibnkahla, Efficient deployment of wireless sensor networks targeting environmentmonitoring applications. *Computer Communications*, 2013. 36(2): p. 135-148.
- [15] F. M. Al-Turjman, H.S.H., and M. Ibnkahla, Quantifying connectivity in wireless sensor networks with grid-based deployments. *Journal of Network and Computer Applications*, 2013. 36(1): p. 368-377.
- [16] S.P. WANG, X.P. YANG, X.Q. WANG, et al. Avirtual force algorithm-lévy-embedded grey wolf optimization algorithm for wireless sensor network coverage optimization[J]. *Sensors*,2019,19(12):2735-2735.