axioms

Article

HPSBA: A Modified Hybrid Framework with Convergence Analysis for Solving Wireless Sensor Network Coverage Optimization Problem

Mengjian Zhang [1,2], Deguang Wang [1,3], Ming Yang [1,3], Wei Tan [4] and Jing Yang [1,3,*]

[1] Electrical Engineering College, Guizhou University, Guiyang 550025, China
[2] School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China
[3] Guizhou Provincial Key Laboratory of Internet + Intelligent Manufacturing, Guiyang 550025, China
[4] College of Forestry, Guizhou University, Guiyang 550025, China
* Correspondence: jyang7@gzu.edu.cn

Abstract: Complex optimization (CO) problems have been solved using swarm intelligence (SI) methods. One of the CO problems is the Wireless Sensor Network (WSN) coverage optimization problem, which plays an important role in Internet of Things (IoT). A novel hybrid algorithm is proposed, named hybrid particle swarm butterfly algorithm (HPSBA), by combining their strengths of particle swarm optimization (PSO) and butterfly optimization algorithm (BOA), for solving this problem. Significantly, the value of individual scent intensity should be non-negative without consideration of the basic BOA, which is calculated with absolute value of the proposed HPSBA. Moreover, the performance of the HPSBA is comprehensively compared with the fundamental BOA, numerous potential BOA variants, and tried-and-true algorithms, for solving the twenty-six commonly used benchmark functions. The results show that HPSBA has a competitive overall performance. Finally, when compared to PSO, BOA, and MBOA, HPSBA is used to solve the node coverage optimization problem in WSN. The experimental results demonstrate that the HPSBA optimized coverage has a higher coverage rate, which effectively reduces node redundancy and extends WSN survival time.

Keywords: particle swarm optimization; butterfly optimization algorithm; hybrid algorithm; convergence analysis; Wireless Sensor Network; node coverage

check for updates

Citation: Zhang, M.; Wang, D.; Yang, M.; Tan, W.; Yang, J. HPSBA: A Modified Hybrid Framework with Convergence Analysis for Solving Wireless Sensor Network Coverage Optimization Problem. *Axioms* **2022**, *11*, 675. https://doi.org/10.3390/axioms11120675

Academic Editor: Muhammad Azeem Akbar

Received: 23 October 2022
Accepted: 24 November 2022
Published: 27 November 2022

1. Introduction

With the emergence of heuristic intelligent optimization algorithms, new methods have been provided for solving complex engineering problems. The principle is mostly to imitate the biological habits of foraging and courtship of the biological community. According to the theoretical principle of meta-heuristic optimization algorithm, it can be simply divided into four categories (See Figure 1). Typical swarm intelligence algorithms (SI-based) are: Particle swarm optimization (PSO) [1], Cuckoo search (CS) [2], Grey wolf optimizer (GWO) [3], Whale optimization algorithm (WOA) [4], Marine predators algorithm (MPA) [5], Ant colony optimization (ACO) [6], Firefly algorithm (FA) [7], Moth-flame optimization (MFO) [8], Grasshopper optimization algorithm (GOA) [9], Butterfly optimization algorithm (BOA) [10]. Evolution algorithms (Ev-based) are: Genetic algorithm (GA) [11], Differential evolution (DE) [12], Biogeography-based optimizer (BBO) [13], Genetic programming (GP) [14]. Algorithms based on physical characteristics (Phy-based) are: Simulated annealing (SA) [15], Gravitational search algorithm (GSA) [16], Harmony search (HS) [17], Sine cosine algorithm (SCA) [18], Equilibrium optimizer (EO) [19], Gradient-based optimizer (GBO) [20]. Algorithms based on human social behavior (Hu-based) are: Teaching learning based optimization (TLBO) [21], Tabu search (TS) [22], Socio evolution and learning optimization (SELO) [23], Political optimizer (PO) [24]. For a more detailed review, we can refer to the literature [25,26].

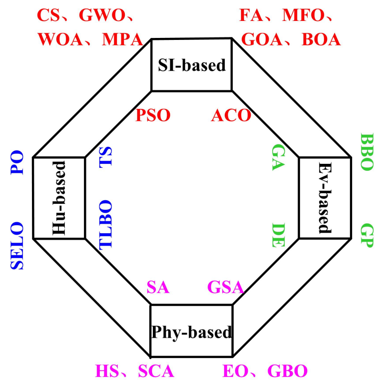


Figure 1. Classification of heuristic optimization algorithms.

According to the biological characteristics of natural animals (such as oviparous animals or mammals, and insects) and plants, swarm intelligence optimization algorithms can be divided into three categories. Imitating animal habits such as: Particle swarm optimization (PSO) [1], Cuckoo search (CS) [2], Grey wolf optimizer (GWO) [3], Whale optimization algorithm (WOA) [4], Marine predators algorithm (MPA) [5], etc. Imitating the habits of insects such as: Ant colony optimization (ACO) [6], Firefly algorithm (FA) [7], Moth-flame optimization (MFO) [8], Grasshopper optimization algorithm (GOA) [9], Butterfly optimization algorithm (BOA) [10], etc. Imitating plant characteristics such as: Flower pollination algorithm (FPA) [27], Tree-seed algorithm (TSA) [28], etc.

PSO is a typical and widely used intelligent optimization algorithm. It has the advantage of fast convergence. Some of the recent works on hybrid PSO with other SI algorithms are as follows: hybrid PSO and DE [29], hybrid PSO and GSA [30], hybrid GA and PSO [31,32], hybrid PSO and SSA [33], etc. Butterfly optimization algorithm [10] is a novel swarm intelligent algorithm proposed by Arora and Singh, which has been used to solve the wireless sensor network node localization problem [34], and optimization training of wavelet neural network [35]. However, BOA is susceptible to local optima and suffers from premature convergence. There are several recent works on BOA as follows: improved BOA [36], modified BOA [37], hybrid BOA and PSO [38], etc.

The energy consumption of the wireless sensor network (WSN) is limited by the large number of sensor nodes [39], which is crucial to the Internet of Things (IoT). It can provide users with accurate and comprehensive real-time data by processing the detection data of sensing objects of the mutual cooperation between nodes. WSN has been widely used in military, transportation, environmental monitoring and other fields [40]. The coverage problem is one of the key tasks in the research field of the WSN, which reflects the quality of the service. Coverage ratio is an important indicator for evaluating the performance of the WSN nodes.

In a working area, sensor nodes are usually arranged at random for the initial stage. High-density nodes will result from this method, resulting in a low coverage rate that directly impacts monitoring quality [41,42]. As a result, optimizing sensor node coverage is critical to increasing WSN coverage ratio in the work area. SI optimization algorithms have recently made significant contributions to the WSN's problem of optimizing node coverage. Wang et al. [43] proposed a resampled PSO to solve the coverage control problem in IoT. Yang et. al [44] used the improved FA to solve sensor coverage problem, which considered the target coverage and network connectivity of sensor nodes. Miao et al. [45] proposed a GWO-EH algorithm to address the WSN node coverage optimization problem. Wang et al. [46] proposed the topology optimization of coverage-oriented method for a WSN based on wolf pack algorithm (WPA). Dao et al. [47] proposed a WSN coverage optimization model based on an improved Archimedes optimization algorithm for the working area. Although the above-mentioned SI algorithms have achieved success, they are still difficult to rid of local optima when optimization problems become more challenging, which makes it necessary to find new methods.

Because of the advantages of BOA with simple structure and few adjustment parameters, and PSO with fast convergence, a novel Chaotic hybrid butterfly optimization algorithm with particle swarm optimization (HPSOBOA) [38] was proposed for solving the high-dimensional optimization problems. However, both of BOA and PSO can easily fall into a local optimum and have low convergence accuracy, and HPSOBOA also has poor results for engineering optimization problems, which makes further research necessary for them. Significantly, the value of individual scent intensity should be a non-negative in nature without consideration in the basic BOA and others. Thus, a novel hybrid particle swarm butterfly algorithm (HPSBA) is proposed on the basis of HPSOBOA. The main contributions and highlights are as follows:

- A novel hybrid particle swarm butterfly algorithm is proposed. This combination strikes a balance between exploitation and exploration. We design that the control strategy of parameter c is based on Logistic map, and the parameter ω is based on adaptive adjustment strategy of the HPSBA for improving the optimization speed, convergence accuracy and global search capability. Moreover, the individual scent intensity value is calculated with absolute value of the proposed HPSBA.
- To ensure that the proposed algorithm works, we compare the optimization results of twenty-six benchmark functions with ten intelligent optimization algorithms. According to the mean value (Mean), standard deviation (Std), Wilcoxon rank-sum (WRS) test findings, and convergence curves, the simulation results show that HPSBA has a competitive overall performance.
- The node optimization coverage problem of the WSN is solved using the proposed HPSBA. The application and advantages of the HPSBA are also discussed.

The remaining sections of this study are as follows: The mathematical model for the WSN's node coverage optimization (NCO) problem is established in Section 2, which goes over the underlying concepts of the PSO and BOA. The proposed HPSBA is explained in detail in Section 3. The outcomes of the algorithms' comparison experiments are presented in Section 4. In Section 5, HPSBA is applied to solve the WSN's NCO problem. Section 6 concludes with a discussion of the next steps.

2. Basic Knowledge

2.1. Particle Swarm Optimization

There are two important characteristics of PSO algorithm [1] are the position and velocity of the particles. The position and velocity of the particles are updated as Equations (1) and (2).

$$v_i^{t+1} = \omega \cdot v_i^t + c_1 \cdot rand_1 \times (p_{best} - x_i^t) + c_2 \cdot rand_2 \times (g_{best} - x_i^t) \tag{1}$$

$$x_i^{t+1} = x_i^t + v^{t+1} \tag{2}$$

where v_i^t and v_i^{t+1} are the velocity of the i-th particle when the iteration number is t and $t+1$, respectively. p_{best} and g_{best} represent the initial individual best position and global best position of the particle. $rand_1$ and $rand_2$ are the random number in (0, 1), and usually $c_1 = c_2 = 2$. ω is the inertia weight coefficient.

2.2. Butterfly Optimization Algorithm

In BOA [10], each butterfly in the group has a unique sense and individual perception ability. The intensity of fragrance perception is generated between individuals. Figure 2 presents the food foraging of butterflies in the 2-D search space.

The following is an expression of the intensity of scent that other butterflies perceive:

$$F(x) = cI^a \tag{3}$$

where $F(x)$ denotes the scent intensity function, c denotes the sensory modality, and I indicates stimulus intensity, that is, the function fitness value. a denotes the intensity factor, and the value range of parameter a is [0, 1]. The sensory modality c is calculated as follows:

$$c_{t+1} = c_t + [0.025/(c_t \cdot T_{\max})] \tag{4}$$

where the initial value of c is set to 0.01 in basic BOA [10]. However, the parameter c can be set to any value within $[0, \infty)$ in theory. $T_{\max}$ is the maximum number of iterations.

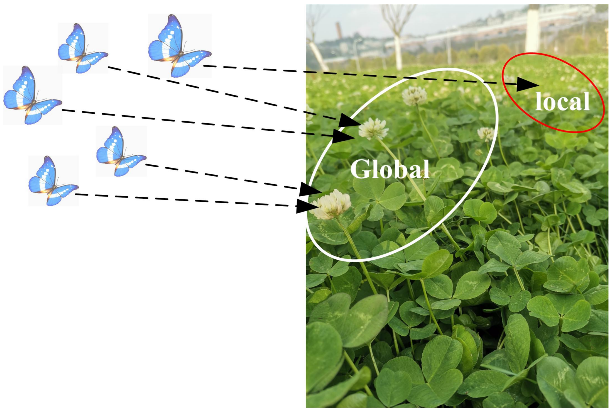


Figure 2. Food foraging of butterflies.

The switching probability SP determines the global search and local search of the BOA. The position update formula is expressed as follows:

$$x_i^{t+1} = \begin{cases} x_i^t + (r^2 \times g^* - x_i^t) \times F_i, SP \geq rand \\ x_i^t + (r^2 \times x_j^t - x_k^t) \times F_i, SP < rand \end{cases} \tag{5}$$

where x_i^t denotes the spatial position of the i-th butterfly in the t-th iteration. $g*$ is the best position of all butterfly individuals in the current iteration. x_j^t and x_k^t are the position of the j-th and k-th butterfly individual when the iteration number is t, respectively. r is a random number in (0, 1), and F_i is the scent intensity value of the i-th butterfly.

2.3. Node Coverage Optimization Problem Model

For the two-dimensional point coverage problem in WSN [39], it is assumed that there are n detection points to be covered in the two-dimensional coverage area, and the coverage nodes use homogeneous sensors, that is, the sensors have the same sensing radius. Supposing the sensing radius is r_s, thus r_c is regarded as the communication radius, and the unit of them is meter. The sensing radius is the maximum distance at which the received signal strength of a node is greater than the inherent noise, that is, the sensing range of the node. The communication radius is the maximum distance for transmitting data or signals between nodes, generally.

Assuming that the monitoring area contains n target points, the position coordinates of the i-th target point to be monitored are (x_i, y_i), and the sensor nodes s position coordinates are (x_s, y_s). Then, the Euclidean distance that the sensor can cover the target to be monitored can be expressed as:

$$d(i,s) = \sqrt{(x_s - x_i)^2 + (y_s - y_i)^2} \tag{6}$$

The binary perception model [48,49] is used in this study, and the sensor node s covers the probability p that the target node i will be monitored, it can be defined as:

$$p(i,s) = \begin{cases} 0, d(i,s) \geq r_s \\ 1, d(i,s) < r_s \end{cases} \tag{7}$$

We divide the two-dimensional plane area to be deployed along the x and y axes with the step length q, and then the length of each segment is $l = q$, and the intersection of the coverage area is q^2. The probability that the monitoring point set T in the coverage area is interpreted by the node set S. The node coverage rate is defined as:

$$\mathrm{Cov} = \frac{p_{\mathrm{cov}}}{q^2} = \frac{\sum_{i=1}^{S} p(i,s)}{q^2} \tag{8}$$

Assuming that the coverage area is a square, the side length is L, and r_s denotes the node sensing radius. Theoretically, the number of deployed nodes can be calculated in the coverage area. The schematic diagram of the nodes full coverage is as Figure 3.

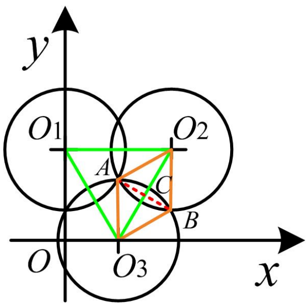


Figure 3. Node coverage diagram.

In Figure 3, O_1, O_2 and O_3 indicate the positions of the three nodes, the triangle $O_1O_2O_3$ is an equilateral triangle where $O_3A = r_s$, that is, sensing radius of the sensor, $\angle AO_3B = \pi/3$, $AB = BO_3 = O_3A = r_s$. According to the nature of the circle, $AB \perp O_2O_3$, $\angle AO_3C = 1/2\angle AO_3B = \pi/6$. According to the law of cosines, the length of the line segment O_3C can be expressed as follows:

$$L_{O_3C} = L_{O_3A} \times \cos(\angle AO_3C) = r_s \times \cos(\pi/6) = \frac{\sqrt{3}}{2} r_s \tag{9}$$

Thus, the number of nodes in the coverage area can be calculated by Equation (10) in theory.

$$M = \left(\frac{L}{\sqrt{3}/2 \cdot r_s + r_s} + 1 \right)^2 \tag{10}$$

According to the above analysis, the NCO problem can be simplified as a constrained optimization problem, the expression is as follows:

$$\max f(x) = Cov, \text{s.t} \begin{cases} g_1 = \sum_{i=1}^{S} p(i,s) \geq 0, \\ g_2 = \sum_{i=1}^{S} p(i,s) - q^2 \geq 0, \\ g_3 = d(i,s) - r_s \geq 0, \\ g_4 = S - M \geq 0. \end{cases} \tag{11}$$

where r_s denotes perception radius of the node, $p(i,s)$ indicates the probability of the target node i monitored and covered by the sensor node s, and $d(i,s)$ indicates the Euclidean distance between sensor node s and monitored target node i. M is the theoretical number of nodes in the coverage area, and S is the set of coverage nodes in the monitoring area.

3. Method

In this section, a novel HPSBA is proposed to improve the optimization speed, convergence accuracy and global search capability between PSO and BOA. It is possible to strike a balance between exploitation and exploration and benefits strengths of both algorithms by combining them. Significantly, the individual scent intensity value is calculated with absolute value of the proposed HPSBA. Furthermore, HPSBA is used to solve the node optimization coverage problem of the WSN for Internet of Things (IoT).

3.1. Hybrid Particle Swarm Butterfly Algorithm (HPSBA)

3.1.1. Algorithmic Population Initialization

Supposing the expression of randomly generated initial solution is defined as follow in the D-dimensional search space.

$$X_i = L_b + (U_b - L_b) \cdot rand \tag{12}$$

where X_i presents the spatial position of the i-th butterfly individual $(i = 1, 2, 3, \cdots, N)$ in the butterfly swarm, N denotes the number of initial individual solutions. L_b and U_b are the upper and lower bounds of the search space, and *rand* is a random number matrix between (0, 1).

3.1.2. Algorithmic Exploration

The exploration stage of the proposed HPSBA is expressed as:

$$V_{i\text{-}1}^t = \omega V_{i\text{-}1}^t + C_1 r_1 \times (p_b - X_{i\text{-}1}^t) + C_2 r_2 \times (g_b - X_{i\text{-}1}^t) \tag{13}$$

$$X_i^t = X_{i\text{-}1}^t + V_{i\text{-}1}^t \tag{14}$$

where ω is the inertia weight coefficient. C_1 and C_2 indicate adjustment parameters, respectively. X_i^t and X_{i-1}^t represent the position of the i-th and $(i-1)$-th agent at t. V_i^t and V_{i-1}^t are the velocity of the i-th and $(i-1)$-th agent when t-th iteration, respectively. p_b and g_b represent the initial global best position of the agent. r_1 and r_2 are the random number in (0, 1).

3.1.3. Algorithmic Exploitation

The exploitation stage of the proposed HPSBA is expressed as:

$$X_i^{t+1} = \begin{cases} \omega \cdot X_i^t + r^2 \cdot (g_b - X_i^t) \times |F_i|, SP \geq rand \\ \omega \cdot X_i^t + r^2 \cdot (X_k^t - X_j^t) \times |F_i|, SP < rand \end{cases} \tag{15}$$

where ω indicates adaptive adjustment parameter. X_i^{t+1} and X_i^t represent the position of the i-th particle at $t+1$ and t, respectively. X_j^t and X_k^t are the positions of the j-th and k-th individuals randomly selected from the solution. F_i is the scent intensity value of the i-th individual. Most notably, the value of individual scent intensity is a non-negative, thus we take the absolute value of F_i in the proposed HPSBA.

3.1.4. The Chaotic Adjusting Strategies

Chaos theory has a lot of application research in SI algorithms, such as chaotic population initialization [38], chaotic adjusting strategies of the control parameters [50], etc. The expression of the Logistic map is defined in Ref. [51]. The chaotic sequence of the Logistic map is (0, 1). When $\mu = 4$, the mapping will produce strong chaotic phenomena. In this paper, the control parameter c of the proposed HPSBA is expressed as:

$$c(t) = 4 \cdot c \cdot (1 - c) \tag{16}$$

The inertial weight coefficient ω has a direct impact on the particle flight speed of the PSO algorithm, and can modify the algorithm's global and local search capabilities. We adopt an adaptive adjustment strategy with chaos, and its expression is as follows:

$$\omega(t) = \omega_u - (\omega_u - \omega_l) \cdot t / T_{max} \tag{17}$$

where ω_u = 0.9, ω_l = 0.2, T_{max} is the maximum iteration number of the algorithm. For the control parameters c and ω, $T_{max} = 500$ and different $c(0)$ values are taken, and the corresponding chaotic sequences are shown in Figure 4. For selecting the best initial value of the control parameters, Schwefel 1.2 and Solomon functions are used to perform optimization tests. The optimization results are shown in Table 1.

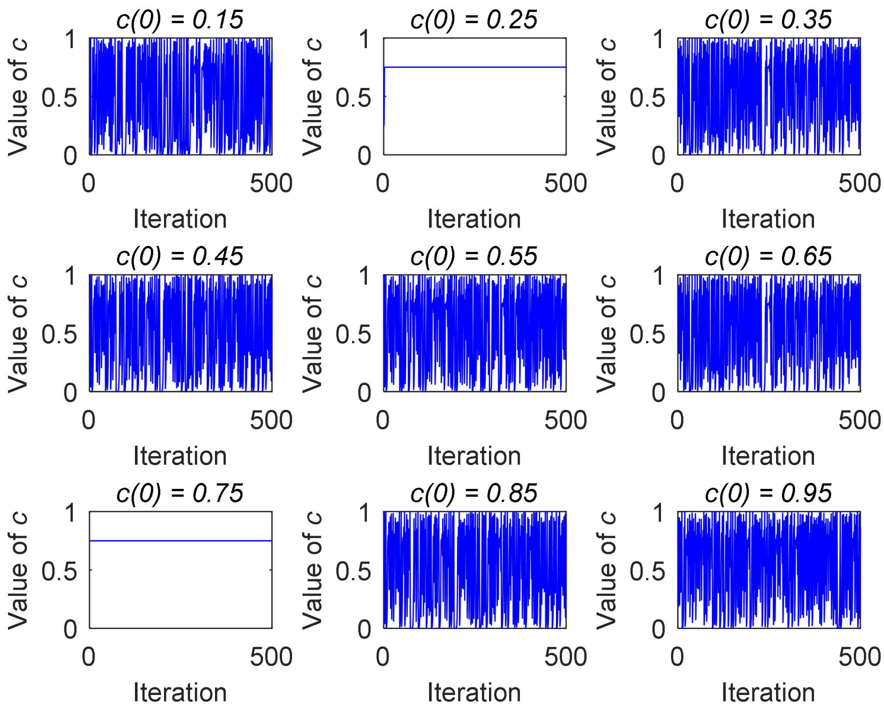


Figure 4. Chaotic sequence with different $c(0)$ values.

Table 1. Optimal value with different c(0) values.

c(0)	Schwefel 1.2			Solomon		
	Mean	Std	Time/s	Mean	Std	Time/s
0.15	5.28E-299	0	1.15	3.85E-301	0	0.17
0.25	4.75E-300	0	1.11	1.49E-223	0	0.17
0.35	7.70E-300	0	1.12	5.73E-301	0	0.17
0.45	3.49E-299	0	1.11	1.87E-301	0	0.17
0.55	4.54E-299	0	1.13	2.40E-301	0	0.17
0.65	3.16E-299	0	1.11	8.90E-301	0	0.17
0.75	7.87E-298	0	1.11	9.75E-222	0	0.17
0.85	4.82E-299	0	1.12	4.40E-301	0	0.17
0.95	3.16E-299	0	1.12	1.93E-301	0	0.17

As seen from Figure 4, when $c(0) = 0.25$ and $c(0) = 0.75$, according to the property of Logistic map, the control strategy of parameter c falls into a fixed point. That is, the initial value of parameter c cannot be set 0.25 or 0.75 in this study.

According to Figure 4 and Table 1, when $c(0) = 0.35$, HPSBA can obtain the best value for the Schwefel 1.2 function (Uni-modal). For the Solomon function (Multi-modal), although the optimal value is not the best with $c(0) = 0.35$, its search value is in the same order of magnitude as the best search value. They take the same time to search the optimal value. In summary, the initial value of parameter c is set to 0.35 in the following experiments.

Figure 5 shows the optimization process of the proposed algorithm. From Figure 5, the new fitness value (F_{new}) and new agent positions are obtained in the exploration stage, which is the local optimum. Then, according to the parameters value in exploration, the proposed HPSBA will get the global optimum in the theory. Finally, the best fitness and agent position are output.

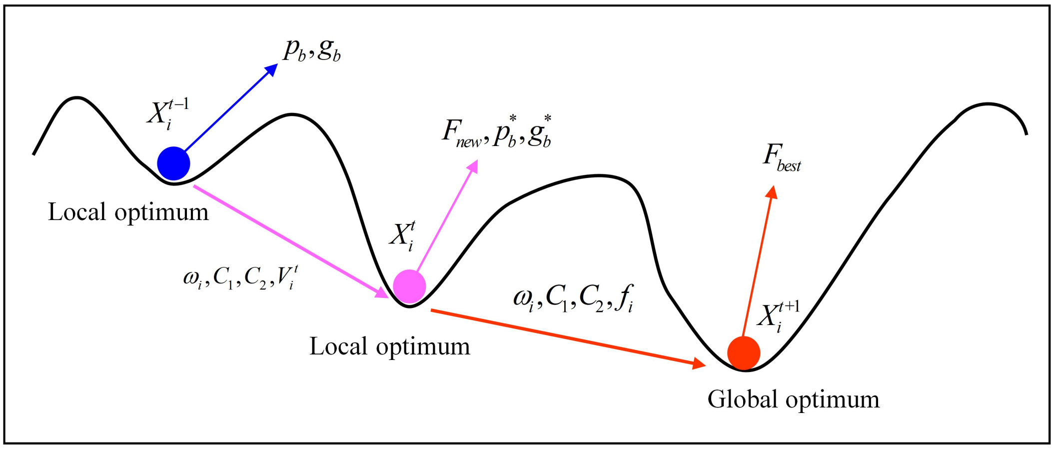


Figure 5. The optimization process of HPSBA.

3.2. Complexity Analysis of the HPSBA

To better understand computational complexity the proposed HPSBA, the time and space complexity of HPSBA are given in this section.

3.2.1. Time Complexity

Assuming that the population size is n, the search space dimension is d, and the maximum iteration is T_{max}. The complexity of the HPSBA includes: the population initialization complexity is $O(nd)$, the fitness value calculation complexity is $O(nd)$, the global and local search location update complexity is $O(n^2 \log n)$, the fitness value sorting complexity is $O(n^2)$, and the control parameter update complexity of the algorithm is $O(nd)$. The HPSBA's total time complexity can be shown as follows by looking at all of the aforementioned components:

$$O(HPSBA) = O(nd) + O(T_{max})O\left(nd + n^2 \log_n + n^2 + nd\right) \tag{18}$$

The time complexity of BOA is:

$$O(BOA) = O(nd) + O(T_{max})\, O\left(n^2 \log_n + n + nd\right) \tag{19}$$

3.2.2. Space Complexity

The space complexity of an algorithm is regarded as the storage space consumed by the algorithm. The population size is n and the dimension is d. The hybrid algorithm is used to calculate the space complexity. The total space complexity of the proposed HPSBA is $O(nd)$. The butterfly optimization algorithm uses n search agents to calculate the space complexity, and the total space complexity of the BOA is $O(nd)$. Therefore, the total space complexity of the basic BOA is the same as the total space complexity of the HPSBA. Therefore, the proposed algorithm has a reliable and effective space efficiency.

3.3. The Pseudo-Code and Flowchart of the HPSBA

The pseudo-code of the HPSBA is presented in Algorithm 1.

Algorithm 1: Pseudo-code of HPSBA

Input: Set the parameters C_1, C_2, ω_{min} and ω_{max}, define sensory modality c, power exponent a and switch probability SP, and initialize population of the agents X_i and V_i $(i = 1, 2, \cdots, n)$ randomly.

Output: Best position and optimal value.

Calculate the fitness value of each agent and find the best agents p_b and g_b.
while $t = 1 : T_{max}$ **do**
    Update the parameters ω using Equation (17)
    for *each search agent* **do**
        Calculate the F_{new} and perceived intensity value F_i;
        if $p_b > F_{new}$ **then**
            Update the position p_b of best individual agent;
        end
        if $g_b > F_{new}$ **then**
            Update the position g_b of best agents;
        end
        Obtain the new velocity of the new agent using Equation (13);
        Update the position of current search agent using Equation (14);
        Set a random number r in (0,1).
        if $r < SP$ **then**
            Move towards best position by Equation (15);
        else
            Move towards best position by Equation (15);
        end
        Limit the boundaries of each agent;
        Calculate the new fitness value of each agent;
        Update the best position and fitness value of each agent;
    end
    Update the parameters c using Equation (16).
    $t = t + 1$
end

In addition, the flowchart of the proposed HPSBA is shown as Figure 6.

3.4. Convergence Analysis of the HPSBA

Theorem 1. *The population position vector sequence $\{X(t), t \geq 0\}$ of the proposed hybrid HPSBA method is a finite homogeneous Markov process.*

Proof. The search space of any optimization algorithm is limited, so the population position vector sequence $\{X(t), t \geq 0\}$ of the hybrid particle swarm butterfly algorithm is also limited. In addition, the position vector of the population in the optimization process is determined by the odor behavior $F_i(t)$ and the flight speed $V_i(t)$. It can be seen that $X(t+1)$ is only related to $X(t)$, namely $\{X(t), t \geq 0\}$ is a Markov chain. Individuals gradually approach the optimal position base on the search space's fitness value, that is, when $f(x_{t+1}) > f(x_t)$, the movement of the population is adjusted, which only has to do with time t. In summary, the population position vector sequence $\{X(t), t \geq 0\}$ of the HPSBA is a finite homogeneous Markov process. □

The essence of the HPSBA belongs to the category of random search algorithm, so the convergence criterion of random optimization algorithm [52] is used to prove the convergence of the hybrid algorithm HPSBA.

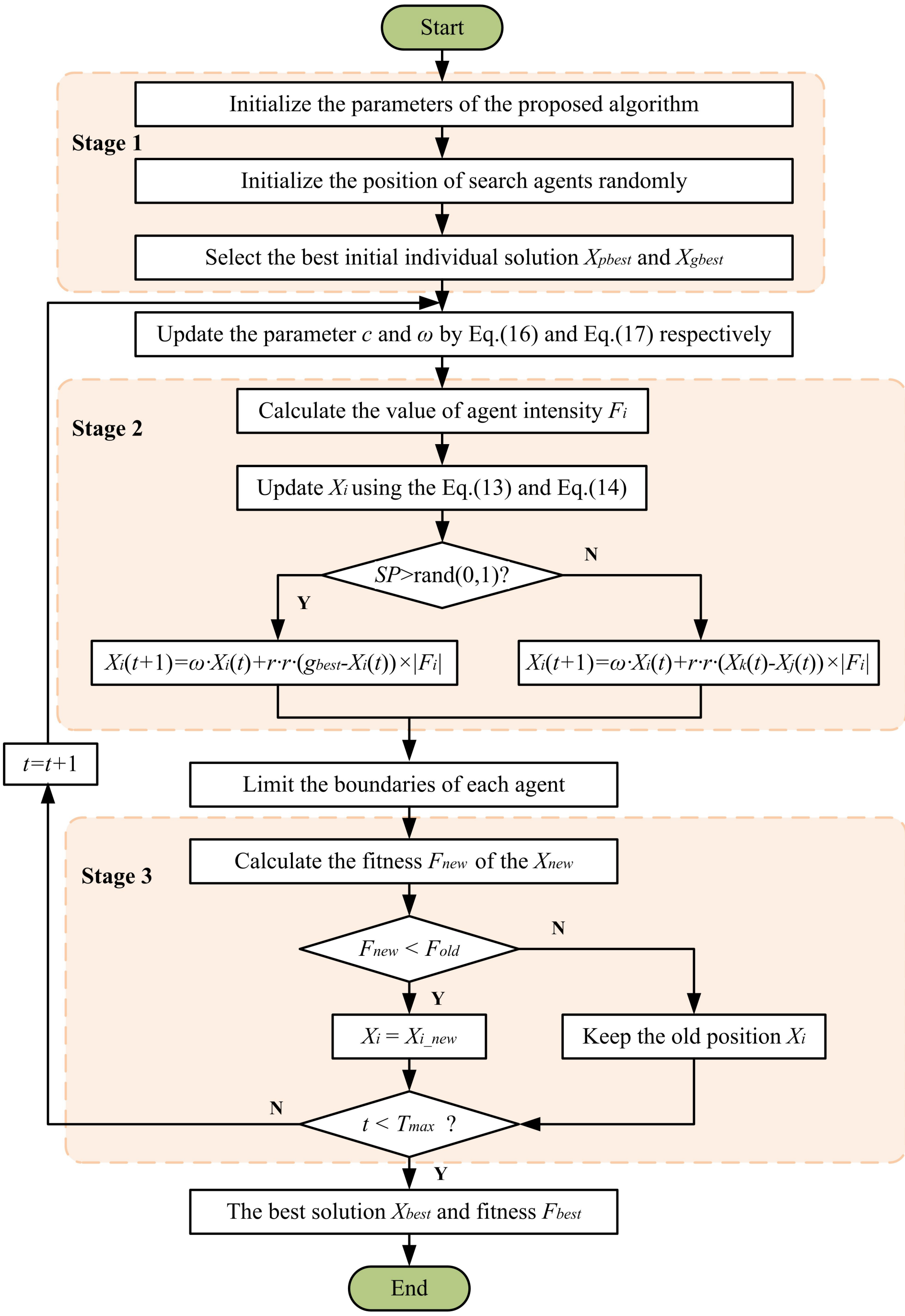


Figure 6. The flowchart of the proposed HPSBA.

3.4.1. Convergence Criterion

For the problem $< Y, f >$, there is a random optimization algorithm Z [53]. The result of the k-th iteration is x_k, and the result of the next iteration is $x_{k+1} = D(x_k, \zeta)$. Y represents the space of potential solutions, f denotes the fitness function, and ζ is the solution searched in Z iteration of the algorithm. The search's lower bound of the Lebesgue measure space [52] is defined as follows:

$$\sigma = \inf\{t|U(x \in Y|f(x) < t) > 0\} \tag{20}$$

where $U(X)$ denotes the Lebesgue measure on the set X, and the optimum can be defined as:

$$R_{\xi,M} = \begin{cases} \{x \in Y | f(x) < \sigma + \xi\}, \sigma \; limited \\ \{x \in Y | f(x) < -\epsilon\}, \; \sigma = -\infty \end{cases} \tag{21}$$

where ξ denotes greater than zero, and ϵ is a sufficiently large positive number. If the algorithm can find a point in $R_{\xi,M}$, that is, the algorithm may have reached an acceptable global optimal point or an approximate global optimal point.

Condition 1: $f(Z(x,\zeta)) \leq f(x)$, and if $\zeta \in Y, f(Z(x,\zeta)) \leq f(\zeta)$.

Condition 2: If $\forall B \in Y$, s.t. $U(B) > 0$, then

$$\prod_{k=0}^{\infty}(1 - U_k(B)) = 0 \tag{22}$$

where $U_k(B)$ denotes the measure of probability of algorithm Z searching solution on set B at the k-th iteration.

Theorem 2. *(Conditions necessary and sufficient for global convergence) Supposing f is measurable, the measurable space Y denotes a measurable subset of R_n, and algorithm Z fulfills both Conditions 1 and 2, and $\{x_k\}_{k=0}^{\infty}$ is the algorithm Z used to generate the solution sequence. So there is a probability measure:*

$$\lim_{k\to\infty} P(x_k \in R_{\xi,M}) = 1 \tag{23}$$

Thus, the algorithm Z converges globally. $P(x_k, R_{\xi,M})$ is the probability measure of the solution x_k in $R_{\xi,M}$ of the iterative search step of the algorithm.

3.4.2. Convergence Analysis

Lemma 1. *Condition 1 is met by HPSBA, which states that the hybrid algorithm's direction of population optimization is monotonic.*

Lemma 2. *The HPSBA population state space's general state has a transition probability of one to the optimal state, that is, $\lim_{t\to\infty} P^{(t)}(\zeta_i \to \zeta_j) = 1$.*

Proof. Assuming the population state $\zeta(j)$ is the optimal solution, if the algorithm converges, after infinite state transitions, the probability of its state space from the general state to the optimal state should be 1. Due to

$$\lim_{t\to\infty} P^{(t)}(\zeta_i \to \zeta_j) = \sum_{k=1}^{N} P^{(t)}(\zeta_{ik} \to \zeta_{jk}) \tag{24}$$

Each iteration of the HPSBA population state is based on the transfer of individual odor to the optimal state, that is, the position of the worst individual state of the population is updated. Therefore

$$\lim_{t\to\infty} P^{(t)}(\zeta_i \to \zeta_j) = 1 \tag{25}$$

□

Lemma 3. *HPSBA satisfies Condition 2.*

Theorem 3. *HPSBA converges to the global optimum, namely $\lim_{t\to\infty} P\{X(t) \in G | X(0) = \Phi_0\} = 1$.*

Proof. Since HPSBA satisfies Condition 1 and Condition 2, in each iteration of the algorithm, the individual will choose to update the retention mechanism of the optimal individual. That is, when the iteration has a tendency to be infinite, $\lim_{n\to\infty} P(x_k, R_{\xi,M}) = 1$, and $\{x_k\}_{k=0}^{\infty}$ is the solution sequence generated by HPSBA iteration. It can be concluded from Theorem 2 that HPSBA is a globally convergent algorithm. $\square$

4. Analyses and Results of Numerical Optimization

Twenty-six benchmark functions are used to test the proposed hybrid algorithm performance, which are listed in Table 2. There are two categories of the benchmark functions. F1–F15 are unimodal functions, which notes the category as U. F16–F26 are multimodal functions, which notes the category as M.

4.1. Parameter Setting of Comparison Algorithms

To verify the performance of HPSBA for solving numerical optimization problems, ten algorithms are employed as competitors for the experiment. The proposed algorithm is compared with five standard algorithms (they are PSO [1], GWO [3], BOA [10], EO [19], MPA [5]), and four BOA variants (they are LBOA [34], CBOA [36], HPSOBOA [38], and IBOA [35]), and SOGWO [54].

The comparison algorithm parameter settings for numerical optimization experiments are shown in Table 3. All of the experimental series were carried out using MATLAB 2018a and an Intel(R) Core (TM) i5-10210U CPU @2.11G with 8G RAM in this study. Furthermore, to better set the number of nodes in the node optimization coverage experiment, the dimensions of the function are respectively set to 30 and 100 in the numerical optimization problems, that is, the setting range of the number of nodes in 2-D monitoring area.

4.2. Comparison Results of HPSBA with Others (Dim = 30)

In order to ensure the reasonableness and fairness of the comparison results, the dimension of the test function in the numerical optimization experiment is set to 30, and the maximum number of iterations is set to 500. For the same test function, each algorithm is independently run 30 times. The mean (Mean), standard deviation (Std) are calculated according to the statistical value. Table 4 shows the comparison results of eleven algorithms with Dim = 30. Results of Wilcoxon rank-sum test calculated at a significance level of $\alpha = 0.05$ are also listed in Tables 4 and 5. The second last row indicates the number of success (+), failure (–), and approximate ($\approx$) of the compared algorithms with respect to HPSBA. The last row shows the rank of the compared algorithms.

4.2.1. Analysis of the Numerical Results

As can be seen from Table 4, for the benchmark functions F1, F2, F3, F4, F6, F8, F9, F11, F14, F20, and F25, the proposed algorithm outperforms all comparative algorithms. For the F16, F17, F19, F24, and F26, HPSBA achieved the theoretical optimal value. For the F18, HPSBA, HPSOBOA and CBOA have the superior results over the other algorithms. For the F5, F21, F22, and F23, MPA have the best results. For the F7, PSO achieved the theoretical optimal value. For the F10, EO has the best result. For the F12, although the Means of GWO, EO, SOGWO, and HPSBA are the same, the Std of EO is the smallest. For the F13, GWO, EO, and SOGWO achieved the theoretical optimal value.

Although the performance of the PSO algorithm is poor, the optimization time is the shortest, which shows that the optimization speed of the algorithm is relatively strong. The conclusions that can be drawn from the results presented in Table 4 are that the rankings of the comparison algorithms are HPSBA > MPA > EO = HPSOBOA = CBOA > LBOA = SOGWO > IBOA > GWO > PSO > BOA.

Table 2. Twenty-six benchmark functions.

Name	Formula	Search Range	Dim	f_{min}	Category
Sphere	$F_1 = \sum_{i=1}^{Dim} x_i^2$	[−100,100]	30/100	0	U
Schwefel 2.22	$F_2 = \sum_{i=1}^{Dim} \lvert x_i \rvert + \prod_{i=1}^{Dim} \lvert x_i \rvert$	[−10,10]	30/100	0	U
Schwefel 1.2	$F_3 = \sum_{i=1}^{Dim} \left(\sum_{j=1}^{i} x_j \right)^2$	[−10,10]	30/100	0	U
Schwefel 2.21	$F_4 = \max\{\lvert x_i \rvert, 1 \le i \le Dim\}$	[−10,10]	30/100	0	U
Step	$F_5 = \sum_{i=1}^{Dim} (x_i + 0.5)^2$	[−10,10]	30/100	0	U
Quartic	$F_6 = \sum_{i=1}^{Dim} i x_i^4 + rand(0,1)$	[−1.28,1.28]	30/100	0	U
Exponential	$F_7 = \exp\left(0.5 \sum_{i=1}^{Dim} x_i\right)$	[−10,10]	30/100	0	U
Sum Power	$F_8 = \sum_{i=1}^{Dim} \lvert x_i \rvert^{(i+1)}$	[−1,1]	30/100	0	U
Sum Square	$F_9 = \sum_{i=1}^{Dim} i x_i^2$	[−10,10]	30/100	0	U
Rosenbrock	$F_{10} = \sum_{i=1}^{Dim} \left(100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right)$	[−10,10]	30/100	0	U
Zakharov	$F_{11} = \sum_{i=1}^{Dim} x_i^2 + \left(\sum_{i=1}^{Dim} 0.5 i x_i\right)^2 + \left(\sum_{i=1}^{Dim} 0.5 i x_i\right)^4$	[−5.12,5.12]	30/100	0	U
Trid	$F_{12} = (x_1 - 1)^2 + \sum_{i=2}^{Dim} i(2x_i^2 - x_{i-1})^2$	[−5,5]	30/100	0	U
Elliptic	$F_{13} = \sum_{i=1}^{Dim} (10^6)^{\frac{i-1}{D-1}} x_i^2$	[−100,100]	30/100	0	U
Cigar	$F_{14} = x_1^2 + 10^6 \sum_{i=2}^{Dim} x_i^2$	[−100,100]	30/100	0	U
Tablet	$F_{15} = 10^6 x_1^2 + \sum_{i=2}^{Dim} x_i^6$	[−10,10]	30/100	0	U
Rastrigin	$F_{16} = \sum_{i=1}^{Dim} \left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$	[−5.12,5.12]	30/100	0	M
NCRastrigin	$F_{17} = \sum_{i=1}^{Dim} \left(y_i^2 - 10\cos(2\pi y_i) + 10\right), y_i = \begin{cases} x_i, \lvert x_i \rvert < 0.5, \\ round(2x_i)/2, \lvert x_i \rvert < 0.5 \end{cases}$	[−5.12,5.12]	30/100	0	M
Ackley	$F_{18} = -20\exp\left(-0.2\sqrt{\frac{1}{Dim}\sum_{i=1}^{Dim} x_i^2}\right) - \exp\left(\frac{1}{Dim}\sum_{i=1}^{Dim}\cos(2\pi x_i)\right) + 20 + e$	[−20,20]	30/100	0	M
Griewank	$F_{19} = \frac{1}{4000}\sum_{i=1}^{Dim} x_i^2 - \prod_{i=1}^{Dim}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	[−600,600]	30/100	0	M
Alpine	$F_{20} = \sum_{i=1}^{Dim} \lvert x_i \cdot \sin(x_i) + 0.1 x_i \rvert$	[−10,10]	30/100	0	M
Penalized 1	$F_{21} = \frac{\pi}{Dim}\left\{\sum_{i=1}^{Dim-1} (y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_{Dim-1})^2 + 10\sin^2(\pi y_1)\right\} + \sum_{i=1}^{Dim} u(x_i, 10, 100, 4)$, $y_i = 1 + \frac{x_i+1}{4}, u_{y_i,a,k,m} = \begin{cases} k(x_i - a)^m, x_i > a, \\ 0, -a \le x_i \le a, \\ k(-x_i - a)^m, x_i < a \end{cases}$	[−10,10]	30/100	0	M
Penalized 2	$F_{22} = \frac{1}{10}\left\{\sin^2(\pi x_1) + \sum_{i=1}^{Dim-1} (x_i - 1)^2\left[1 + \sin^2(3\pi x_{i+1})\right] + (x_{Dim-1})^2\left(1 + \sin^2(2\pi x_{i+1})\right)\right\} + \sum_{i=1}^{Dim} u(x_i, 5, 100, 4)$	[−5,5]	30/100	0	M
Levy	$F_{23} = \sin^2(3\pi x_1) + \sum_{i=1}^{Dim-1} (x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})] + \lvert x_{Dim} - 1 \rvert \cdot [1 + \sin^2(2\pi x_{Dim})]$	[−2,2]	30/100	0	M
Weierstrass	$F_{24} = \sum_{i=1}^{Dim}\left(\sum_{k=0}^{k_{\max}}\left[a^k\cos\left(2\pi b^k(x_i + 0.5)\right)\right]\right) - Dim\sum_{k=0}^{k_{\max}}\left[a^k\cos\left(2\pi b^k \cdot 0.5\right)\right], a = 0.5, b = 3, k_{\max} = 20$	[−1,1]	30/100	0	M
Solomon	$F_{25} = 1 - \cos(2\pi\sqrt{\sum_{i=1}^{Dim} x_i^2}) + 0.1\sqrt{\sum_{i=1}^{Dim} x_i^2}$	[−20,20]	30/100	0	M
Bohachevsky	$F_{26} = \sum_{i=1}^{Dim}\left[x_i^2 + 2x_{i+1}^2 - 0.3\cdot\cos(3\pi x_i) - 0.4\cdot\cos(4\pi x_{i+1}) + 0.7\right]$	[−5,5]	30/100	0	M

Table 3. Parameter settings of the comparison algorithms.

Algorithms	Parameter Settings
PSO	$N = 30, c_1 = c_2 = 2, v_{max} = 1, v_{min} = -1, \omega = 0.7$
GWO	$N = 30, a_{first} = 2, a_{final} = 0$
BOA	$N = 30, a = 0.1, c(0) = 0.01, SP = 0.6$
EO	$N = 25, a_1 = 2, a_2 = 1, GP = 0.5, \lambda \in (0, 1)$
MPA	$N = 30, p = 0.5, FADs = 0.2$
LBOA	$N = 30, a = 0.1, c(0) = 0.01, p = 0.6, \gamma = 1.5$
CBOA	$N = 30, a(0) = 0.1, c(0) = 0.01, p = 0.6, r(0) = 0.33, \mu = 4$
HPSOBOA	$N = 30, a_{first} = 0.1, a_{final} = 0.3, c_0 = 0.01, p = 0.6, x_0 = 0.315, \rho = 0.295, c_1 = c_2 = 0.5$
IBOA	$N = 30, a = 0.1, c(0) = 0.01, and p is dynamic.$
SOGWO	$N = 50, a_{first} = 2, a_{final} = 0$
HPSBA	$N = 30, a = 0.1, c(0) = 0.35, SP = 0.6, \mu = 4, \omega_u = 0.9, \omega_l = 0.2, C_1 = C_2 = 2$

4.2.2. Convergence Behavior Analysis

Figure 7 shows the 2D search space for twenty-six benchmark functions in three dimensional visualization. Figure 8 shows the convergence curves of the comparison algorithms for the functions F1 to F4 (unimodal functions), F16 to F19 and F24 (multimodal functions) when Dim = 30. As can be seen from Figure 8, the comparison curves confirm the superiority of HPSBA over the PSO, GWO, BOA, EO, MPA and other comparison algorithms for functions F1 to F4. In addition, for F16, F17, and F19, HPSBA, HPSOBOA, IBOA, MPA and EO can obtain the best value of the functions in theory. For F24, there are two comparison algorithms can obtain the optimal value, called HPSBA and EO algorithm from the Figure 8. From the curves of the proposed HPSBA, the performance of the algorithm needs to be further improved, especially in terms of convergence speed.

4.3. Comparison Results of HPSBA with Others (Dim = 100)

Table 5 shows the comparison results of eleven algorithms with Dim = 100. Results of the Wilcoxon rank-sum test calculated at a significance level of $\alpha = 0.05$ are listed in Table 5. Where the number of success (+), failure (–), and approximate (≈) of the compared algorithms with respect to HPSBA are listed in the second last row. The last row shows the rank of the compared algorithms.

4.3.1. Analysis of the Numerical Results

As can be seen from Table 5, for the benchmark functions F1, F3, F4, F6, F8, F9, F11, F14, F15, F20, and F25, the proposed algorithm outperforms all comparative algorithms. For the F16, F17, F19, F24, and F26, HPSBA achieved the theoretical optimal value. For the F2, CBOA has the best result. For the F18, HPSBA, HPSOBOA and CBOA have the superior results over the other algorithms. For the F5, F10, F21, and F23, HPSOBOA has the best results. For the F7 and F12, EO algorithm has the best results. For the F13, GWO, EO, and SOGWO achieved the theoretical optimal value. For the F22, PSO algorithm has the best result.

It can be seen from Table 5 that with the increase of the problem dimensions, the optimization speed of PSO algorithm decreases to be some extent. This shows that the difficulty of solving a problem increases with its complexity. The conclusion that can be drawn from the results presented in Table 5 are that the rankings of the comparison algorithms is HPSBA > HPSOBOA > CBOA > EO = MPA > LBOA > IBOA > PSO = SOGWO = GWO > BOA.

Table 4. Comparison results of eleven algorithms: Dim = 30.

Functions		PSO	GWO	BOA	EO	MPA	LBOA	CBOA	HPSOBOA	IBOA	SOGWO	HPSBA
F1	Mean	1.16E-01	1.87E-27	7.70E-11	3.35E-40	5.72E-23	3.54E-12	2.20E-30	3.59E-152	7.47E-15	3.46E-33	**3.29E-252**
	Std	3.89E-02	3.11E-27	6.78E-12	1.70E-39	5.83E-23	3.57E-12	4.29E-30	7.89E-153	1.72E-15	7.28E-33	**0.00E+00**
F2	Mean	6.35E-01	9.50E-17	2.35E-08	6.78E-24	2.53E-13	1.24E-09	3.92E-19	5.04E-60	4.97E+12	8.25E-20	**2.52E-134**
	Std	2.05E-01	7.46E-17	6.49E-09	6.18E-24	2.68E-13	2.10E-09	6.67E-19	2.06E-59	1.22E+13	9.41E-20	**6.87E-134**
F3	Mean	4.16E+00	7.10E-08	5.31E-11	2.38E-11	2.71E-06	2.77E-12	1.11E-30	4.05E-153	9.32E-15	1.97E-09	**7.60E-300**
	Std	9.64E-01	1.69E-07	5.83E-12	1.04E-10	7.11E-06	2.71E-12	3.47E-30	1.13E-153	1.11E-15	8.79E-09	**0.00E+00**
F4	Mean	3.36E-01	7.05E-08	2.65E-08	2.10E-11	3.39E-10	2.48E-09	1.48E-19	1.05E-77	8.05E-12	1.82E-09	**6.02E-152**
	Std	4.82E-02	4.84E-08	2.94E-09	4.31E-11	2.12E-10	3.13E-09	2.49E-19	7.23E-79	1.06E-12	1.47E-09	**1.69E-152**
F5	Mean	7.12E-02	5.40E-01	5.23E+00	7.61E-06	**3.42E-08**	3.40E+00	4.55E+00	4.12E-02	3.36E+00	3.46E-01	5.38E+00
	Std	3.20E-02	3.28E-01	6.84E-01	6.22E-06	**1.73E-08**	6.50E-01	5.83E-01	2.40E-02	7.86E-01	2.66E-01	6.36E-01
F6	Mean	2.60E-01	1.79E-03	1.99E-03	1.36E-03	1.28E-03	1.92E-03	1.17E-04	2.31E-04	3.10E-04	1.20E-03	**9.21E-05**
	Std	8.64E-02	9.30E-04	5.51E-04	9.12E-04	7.43E-04	9.45E-04	1.17E-04	3.77E-04	2.58E-04	5.60E-04	**9.73E-05**
F7	Mean	**0.00E+00**	3.19E-58	4.94E-11	7.18E-66	7.18E-66	6.36E-21	3.84E-19	1.53E-62	7.09E-14	8.16E-61	8.51E-16
	Std	**0.00E+00**	1.20E-57	1.34E-10	1.02E-78	1.40E-69	2.34E-20	1.16E-18	6.12E-63	3.06E-13	4.36E-60	4.27E-15
F8	Mean	7.23E-07	1.75E-95	8.88E-14	1.97E-134	1.98E-60	8.27E-16	1.46E-36	1.02E-156	4.45E-19	2.00E-116	**1.70E-307**
	Std	1.17E-06	9.25E-95	5.52E-14	9.42E-134	4.89E-60	9.08E-16	6.48E-36	8.09E-158	2.44E-19	9.70E-116	**0.00E+00**
F9	Mean	7.51E-01	2.35E-28	6.94E-11	1.36E-41	4.83E-24	3.11E-12	8.60E-31	2.02E-152	9.61E-15	2.03E-34	**3.19E-263**
	Std	2.87E-01	4.02E-28	8.22E-12	3.56E-41	6.23E-24	4.11E-12	1.77E-30	2.89E-153	1.34E-15	2.28E-34	**0.00E+00**
F10	Mean	5.99E+01	2.72E+01	2.89E+01	**2.53E+01**	2.51E+01	2.88E+01	2.89E+01	2.71E+01	2.89E+01	2.68E+01	2.89E+01
	Std	3.87E+01	8.55E-01	2.70E-02	**1.54E-01**	3.89E-01	3.25E-02	3.73E-02	6.30E+00	3.53E-02	8.00E-01	3.37E-02
F11	Mean	1.47E+00	3.17E-28	6.67E-11	3.42E-41	1.23E-23	3.52E-12	2.81E-30	6.89E-153	8.32E-15	1.18E-33	**1.28E-252**
	Std	8.03E-01	5.15E-28	7.26E-12	1.26E-40	2.24E-23	3.22E-12	7.66E-30	1.17E-153	1.52E-15	2.59E-33	**0.00E+00**
F12	Mean	4.22E+00	**6.67E-01**	9.74E-01	**6.67E-01**	6.67E-01	9.18E-01	9.76E-01	1.00E+00	9.35E-01	**6.67E-01**	**6.67E-01**
	Std	1.82E+00	3.76E-05	8.43E-03	**3.08E-10**	3.87E-08	2.48E-02	9.00E-03	1.25E-05	1.81E-02	4.37E-06	1.86E-04
F13	Mean	6.08E-31	**0.00E+00**	2.80E-21	**0.00E+00**	3.55E-174	5.49E-26	1.77E-34	2.30E-148	8.87E-31	**0.00E+00**	2.44E-302
	Std	2.42E-30	**0.00E+00**	8.91E-21	**0.00E+00**	0.00E+00	1.14E-25	5.23E-34	1.14E-147	4.33E-30	**0.00E+00**	0.00E+00
F14	Mean	1.16E-24	2.82E-205	1.92E-17	7.30E-207	1.34E-61	3.31E-18	3.49E-31	1.95E-147	5.73E-23	4.59E-228	**6.12E-296**
	Std	2.26E-24	0.00E+00	2.08E-17	0.00E+00	7.35E-61	4.01E-18	7.94E-31	4.53E-147	1.22E-22	0.00E+00	**0.00E+00**
F15	Mean	3.02E-30	6.90E-261	4.54E-19	8.38E-255	8.23E-94	1.65E-19	1.09E-34	1.92E-153	3.69E-22	**1.06E-313**	3.61E-304
	Std	1.65E-29	0.00E+00	8.61E-19	0.00E+00	3.33E-93	3.98E-19	5.42E-34	6.79E-153	7.04E-22	**0.00E+00**	0.00E+00
F16	Mean	2.37E+02	4.02E+00	6.54E+01	1.89E-15	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	2.27E+00	**0.00E+00**
	Std	5.65E+01	3.88E+00	9.09E+01	1.04E-14	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	3.20E+00	**0.00E+00**
F17	Mean	2.76E+02	8.31E+00	1.24E+02	2.33E-01	3.96E-07	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	8.87E+00	**0.00E+00**
	Std	7.80E+01	4.39E+00	7.02E+01	6.26E-01	2.17E-06	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	5.03E+00	**0.00E+00**
F18	Mean	2.47E-01	9.05E-14	2.75E-08	8.47E-15	8.53E-13	2.47E-09	**8.88E-16**	**8.88E-16**	7.10E-12	4.14E-14	**8.88E-16**
	Std	7.83E-02	1.67E-14	2.47E-09	1.80E-15	5.41E-13	1.38E-09	**0.00E+00**	**0.00E+00**	7.10E-13	2.89E-15	**0.00E+00**
F19	Mean	3.41E+01	3.23E-03	9.73E-12	**0.00E+00**	**0.00E+00**	1.79E-13	**0.00E+00**	**0.00E+00**	4.18E-16	2.70E-03	**0.00E+00**
	Std	5.57E+00	8.81E-03	1.06E-11	**0.00E+00**	**0.00E+00**	3.96E-13	**0.00E+00**	**0.00E+00**	1.90E-15	5.90E-03	**0.00E+00**
F20	Mean	1.53E-01	4.74E-04	3.47E-09	6.29E-09	6.61E-14	6.42E-14	1.00E-19	9.42E-60	6.67E-12	3.52E-04	**7.05E-136**
	Std	9.79E-02	7.66E-04	7.60E-09	3.44E-08	4.58E-14	1.86E-13	1.28E-19	2.99E-59	7.69E-13	5.84E-04	**3.83E-135**
F21	Mean	7.09E+00	5.03E-02	5.39E-01	3.46E-03	**7.59E-05**	2.90E-01	4.78E-01	2.47E-03	1.48E+00	3.38E-02	5.49E-01
	Std	3.04E+00	2.12E-02	1.58E-01	1.89E-02	**4.15E-04**	9.21E-02	1.36E-01	2.64E-03	2.31E-01	1.49E-02	1.37E-01
F22	Mean	8.06E-03	7.07E-01	3.40E+00	2.18E-02	**3.45E-03**	2.37E+00	3.00E+00	4.07E+00	2.63E+00	5.15E-01	3.42E+00
	Std	**4.77E-03**	2.10E-01	4.83E-01	4.72E-02	**1.65E-02**	6.28E-01	5.74E-01	2.15E+00	5.90E-01	1.88E-01	5.79E-01
F23	Mean	3.88E-01	1.67E+00	1.18E+01	1.52E-01	**1.38E-01**	9.31E+00	1.01E+01	8.89E-01	1.06E+01	1.25E+00	1.09E+01
	Std	2.43E-01	1.02E+00	2.10E+00	3.22E-01	**1.89E-01**	2.67E+00	2.71E+00	1.18E+00	1.94E+00	8.19E-01	3.23E+00
F24	Mean	5.70E+00	4.93E+00	1.23E+00	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	4.61E+00	**0.00E+00**
	Std	2.76E+00	2.04E+00	2.35E+00	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.67E+00	**0.00E+00**
F25	Mean	7.11E-01	2.79E-01	7.66E-01	9.95E-02	9.95E-02	2.99E-02	1.68E-32	6.43E-02	9.95E-02	2.89E-01	**4.07E-301**
	Std	3.41E-01	1.49E-01	2.18E-01	2.08E-12	7.05E-17	4.64E-02	3.64E-32	4.99E-02	1.24E-06	1.46E-01	**0.00E+00**
F26	Mean	1.24E+00	**0.00E+00**	8.02E-11	**0.00E+00**	**0.00E+00**	4.45E-12	**0.00E+00**	**0.00E+00**	9.79E-15	**0.00E+00**	**0.00E+00**
	Std	5.92E-01	**0.00E+00**	8.59E-12	**0.00E+00**	**0.00E+00**	5.45E-12	**0.00E+00**	**0.00E+00**	1.33E-15	**0.00E+00**	**0.00E+00**
+/-/≈		1/25/0	0/24/2	0/26/0	1/20/5	4/18/4	0/23/3	0/20/6	0/20/6	0/23/3	1/22/3	∼
Rank		7	6	8	3	2	4	3	3	5	4	1

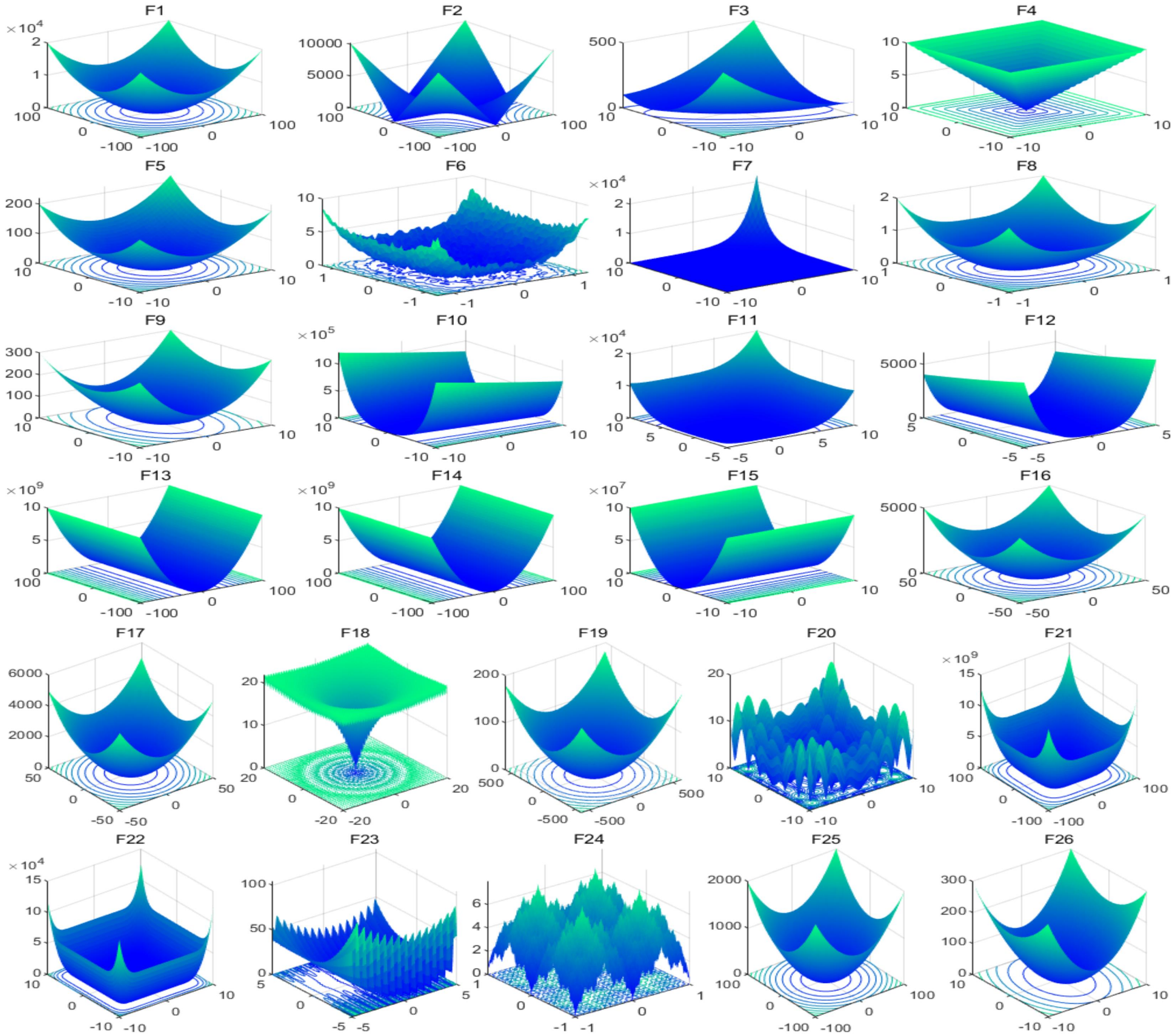


Figure 7. Search space of twenty-six benchmark functions.

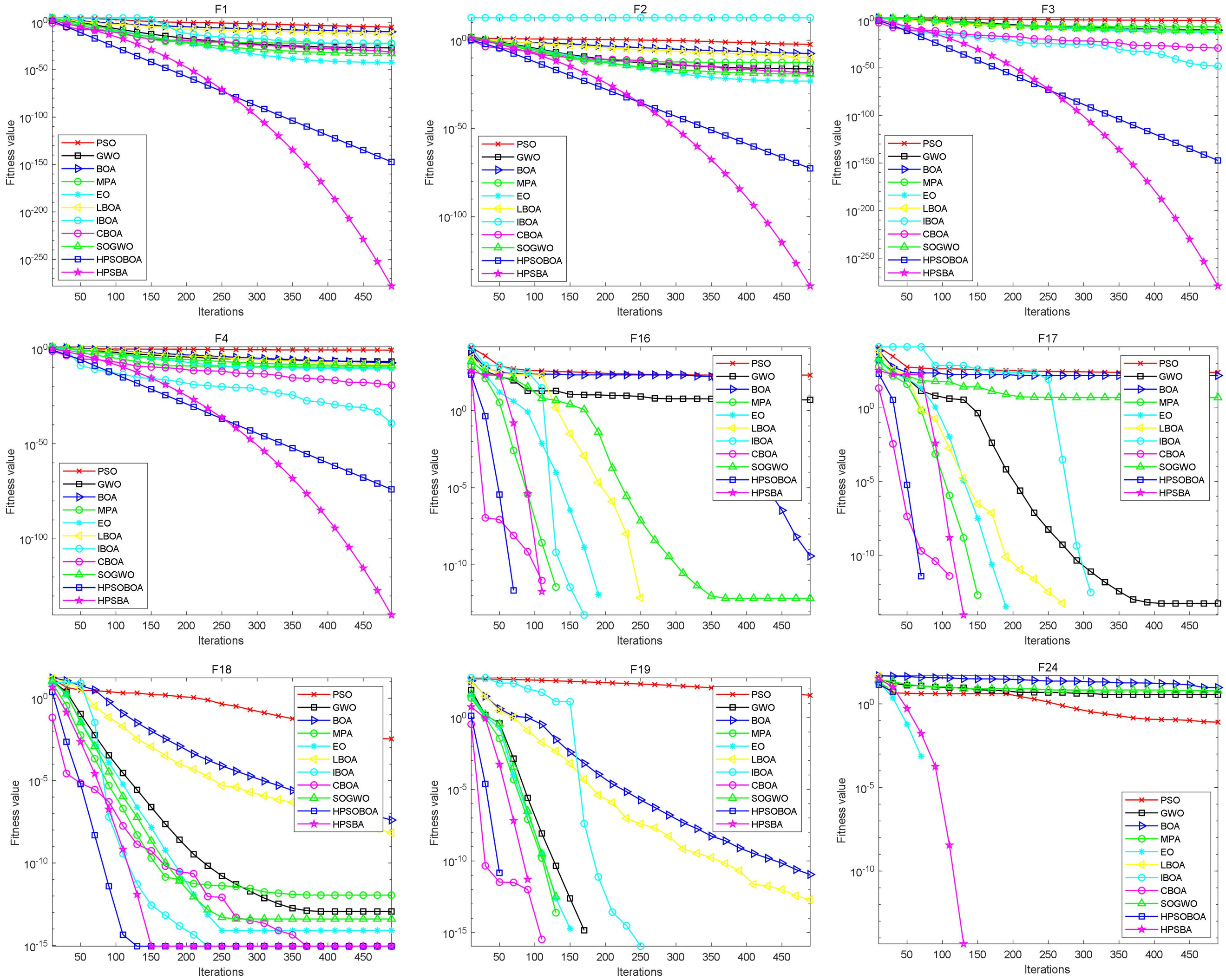


Figure 8. Convergence curves of comparison algorithms when Dim = 30.

4.3.2. Boxplot Results Analysis

To better explain the stability of the comparison algorithms for solving the high dimensional optimization problems. Figure 9 presents the boxplot results of the eleven algorithms on the four test functions (F3, F8, F20, and F25). 30 independent runs of each algorithm are conducted for the same test function.

Table 5. Comparison results of eleven algorithms: Dim = 100.

Functions		PSO	GWO	BOA	EO	MPA	LBOA	CBOA	HPSOBOA	IBOA	SOGWO	HPSBA
F1	Mean	8.45E+00	1.48E-12	8.75E-11	4.14E-29	1.77E-19	6.12E-12	6.86E-30	3.31E-152	9.05E-15	6.44E-15	**7.12E-299**
	Std	7.28E-01	1.83E-12	9.69E-12	5.38E-29	2.10E-19	7.39E-12	1.51E-29	1.35E-152	1.87E-15	8.86E-15	**0.00E+00**
F2	Mean	1.82E+01	4.04E-08	4.71E+50	1.85E-17	1.43E-11	1.52E+50	**3.74E-18**	1.48E+36	8.10E+49	1.60E-09	4.35E+50
	Std	2.23E+00	1.35E-08	1.92E+51	1.36E-17	1.15E-11	6.27E+50	**6.73E-18**	2.80E+35	3.95E+50	6.14E-10	1.42E+51
F3	Mean	2.80E+02	4.54E+00	6.08E-11	3.15E-01	1.04E-01	3.40E-12	4.42E-31	1.48E-152	9.74E-15	1.39E+00	**1.06E-299**
	Std	7.38E+01	4.04E+00	5.91E-12	1.11E+00	1.50E-01	3.36E-12	8.91E-31	1.35E-152	1.11E-15	1.98E+00	**0.00E+00**
F4	Mean	1.77E+00	5.79E-02	2.98E-08	2.64E-01	2.40E-08	2.86E-09	1.20E-19	1.08E-77	8.47E-12	1.60E-02	**7.43E-152**
	Std	1.71E-01	6.77E-02	2.71E-09	1.45E+00	1.16E-08	2.88E-09	1.58E-19	5.19E-79	1.17E-12	1.52E-02	**2.49E-152**
F5	Mean	5.47E+00	9.20E+00	2.21E+01	2.94E+00	2.56E+00	2.03E+01	2.12E+01	**1.48E-01**	2.07E+01	7.67E+00	2.24E+01
	Std	9.43E-01	9.87E-01	1.06E+00	5.46E-01	7.71E-01	1.60E+00	1.18E+00	**1.27E-01**	1.06E+00	9.06E-01	8.81E-01
F6	Mean	6.02E+01	7.14E-03	2.11E-03	2.44E-03	1.87E-03	2.01E-03	1.03E-04	1.13E-04	2.77E-04	4.90E-03	**6.85E-05**
	Std	1.38E+01	2.79E-03	8.96E-04	1.41E-03	9.55E-04	1.18E-03	7.42E-05	1.28E-04	2.49E-04	1.70E-03	**6.09E-05**
F7	Mean	0.00E+00	9.84E-135	1.71E-23	**7.16E-218**	1.70E-202	2.74E-28	4.05E-32	1.93E-207	1.92E-20	2.96E-155	4.15E-27
	Std	0.00E+00	5.39E-134	8.68E-23	**0.00E+00**	0.00E+00	1.39E-27	1.12E-31	0.00E+00	8.47E-20	1.62E-154	2.27E-26
F8	Mean	9.97E-02	1.28E-66	7.32E-14	1.16E-129	2.23E-60	9.96E-16	4.86E-37	9.98E-157	4.00E-19	2.38E-64	**1.82E-306**
	Std	2.20E-01	4.84E-66	6.04E-14	6.03E-129	4.85E-60	1.62E-15	1.49E-36	7.46E-158	2.59E-19	1.30E-63	**0.00E+00**
F9	Mean	2.82E+02	6.63E-13	8.60E-11	1.94E-29	8.17E-20	3.97E-12	1.13E-30	4.23E-152	9.06E-15	1.38E-15	**4.72E-299**
	Std	5.17E+01	5.34E-13	8.92E-12	2.56E-29	5.83E-20	3.66E-12	2.60E-30	1.02E-152	1.95E-15	1.01E-15	**0.00E+00**
F10	Mean	1.24E+03	9.79E+01	9.89E+01	9.66E+01	9.69E+01	9.88E+01	9.89E+01	**9.07E+01**	9.89E+01	9.77E+01	9.89E+01
	Std	2.28E+02	5.73E-01	**2.93E-02**	1.08E+00	8.66E-01	3.76E-02	4.76E-02	**2.30E+01**	3.48E-02	7.22E-01	3.49E-02
F11	Mean	3.65E+02	7.60E-13	8.13E-11	3.57E-29	3.77E-20	4.82E-12	3.26E-30	9.41E-153	1.00E-14	1.93E-15	**4.59E-299**
	Std	7.04E+01	6.45E-13	6.27E-12	1.09E-28	3.04E-20	3.99E-12	1.45E-29	5.44E-153	1.66E-15	1.49E-15	**0.00E+00**
F12	Mean	6.01E+02	6.67E-01	9.98E-01	**6.67E-01**	6.67E-01	9.95E-01	9.98E-01	1.00E+00	9.96E-01	6.67E-01	9.99E-01
	Std	1.55E+02	3.47E-05	8.04E-04	**3.93E-08**	1.41E-06	9.93E-04	5.10E-04	8.11E-05	8.74E-04	5.49E-06	4.19E-04
F13	Mean	1.06E-33	**0.00E+00**	5.13E-22	**0.00E+00**	3.61E-169	3.15E-25	3.19E-34	1.33E-150	4.62E-31	**0.00E+00**	1.53E-302
	Std	5.50E-33	**0.00E+00**	1.69E-21	**0.00E+00**	0.00E+00	8.18E-25	1.63E-33	3.20E-150	1.56E-30	**0.00E+00**	0.00E+00
F14	Mean	6.38E-24	1.24E-205	3.84E-17	2.66E-201	1.86E-63	4.88E-18	3.01E-31	6.40E-150	9.63E-23	2.87E-186	**3.05E-298**
	Std	1.76E-23	0.00E+00	6.24E-17	0.00E+00	1.02E-62	7.45E-18	6.15E-31	1.34E-149	2.26E-22	0.00E+00	**0.00E+00**
F15	Mean	1.88E-32	8.39E-261	1.23E-19	3.97E-253	9.63E-92	3.93E-19	1.55E-34	1.71E-153	3.26E-22	6.24E-310	**1.69E-303**
	Std	5.59E-32	0.00E+00	2.55E-19	0.00E+00	4.94E-91	9.39E-19	8.12E-34	5.23E-153	4.34E-22	0.00E+00	**0.00E+00**
F16	Mean	4.87E+02	9.55E+00	1.77E-06	3.79E-15	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	9.04E+00	**0.00E+00**
	Std	6.30E+01	8.37E+00	9.70E-06	2.08E-14	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	5.80E+00	**0.00E+00**
F17	Mean	4.34E+02	2.34E+01	7.57E+01	1.00E-01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	5.92E-17	1.32E+01	**0.00E+00**
	Std	6.22E+01	2.09E+01	2.31E+02	3.05E-01	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	3.24E-16	9.07E+00	**0.00E+00**
F18	Mean	2.52E+00	8.15E-08	3.08E-08	3.27E-14	2.87E-11	2.76E-09	**8.88E-16**	**8.88E-16**	7.78E-12	4.06E-09	**8.88E-16**
	Std	1.91E-01	3.11E-08	2.55E-09	6.08E-15	1.33E-11	2.31E-09	**0.00E+00**	**0.00E+00**	9.15E-13	1.29E-09	**0.00E+00**
F19	Mean	1.38E+02	4.01E-03	6.69E-11	**0.00E+00**	**0.00E+00**	1.98E-12	**0.00E+00**	**0.00E+00**	9.25E-15	1.83E-03	**0.00E+00**
	Std	1.42E+01	1.15E-02	2.72E-11	**0.00E+00**	**0.00E+00**	1.80E-12	**0.00E+00**	**0.00E+00**	4.81E-15	5.61E-03	**0.00E+00**
F20	Mean	1.33E+01	3.92E-03	2.01E-09	3.66E-18	3.01E-12	3.26E-11	1.05E-19	1.15E-57	7.60E-12	2.49E-03	**1.27E-151**
	Std	3.35E+00	2.70E-03	1.85E-09	2.04E-18	2.42E-12	5.92E-11	1.32E-19	6.30E-57	9.27E-13	1.50E-03	**3.56E-152**
F21	Mean	1.13E-01	2.05E-01	9.83E-01	2.83E-02	3.74E-02	7.49E-01	9.33E-01	**1.49E-03**	7.70E-01	1.51E-01	1.09E+00
	Std	8.19E-02	4.33E-02	8.89E-02	8.01E-03	9.89E-03	1.14E-01	1.20E-01	**6.36E-04**	1.10E-01	4.35E-02	7.82E-02
F22	Mean	**1.02E+00**	5.66E+00	9.99E+00	5.16E+00	6.05E+00	9.99E+00	9.98E+00	9.70E+00	9.94E+00	4.96E+00	9.99E+00
	Std	**1.98E-01**	4.20E-01	5.25E-03	1.35E+00	2.96E+00	4.55E-03	4.85E-03	6.11E-01	1.34E-01	4.19E-01	2.48E-03
F23	Mean	2.35E+01	1.81E+01	6.84E+01	3.96E+00	4.54E+00	6.05E+01	6.85E+01	**1.94E+00**	6.54E+01	1.37E+01	6.69E+01
	Std	7.20E+00	4.80E+00	4.20E+00	1.59E+00	1.69E+00	6.39E+00	4.94E+00	**8.69E-01**	5.49E+00	3.09E+00	5.65E+00
F24	Mean	5.16E+01	1.67E+01	2.66E+00	3.20E-05	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.33E+01	**0.00E+00**
	Std	9.49E+00	1.08E+01	3.18E+00	1.75E-04	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**	4.05E+00	**0.00E+00**
F25	Mean	3.97E+00	6.80E-01	4.00E-01	2.29E-01	2.49E-01	3.33E-02	1.09E-32	8.86E-02	9.95E-02	6.47E-01	**7.39E-301**
	Std	1.06E+00	2.51E-01	3.16E-03	1.50E-01	1.52E-01	4.77E-02	3.81E-32	3.58E-02	3.00E-06	2.53E-01	**0.00E+00**
F26	Mean	6.60E+01	1.01E-13	8.60E-11	**0.00E+00**	**0.00E+00**	5.22E-12	**0.00E+00**	**0.00E+00**	7.87E-15	2.96E-16	**0.00E+00**
	Std	6.71E+00	1.50E-13	8.49E-12	**0.00E+00**	**0.00E+00**	6.98E-12	**0.00E+00**	**0.00E+00**	1.50E-15	4.86E-16	**0.00E+00**
+/-/≈		1/25/0	1/25/0	0/26/0	2/21/3	0/21/5	0/23/3	1/19/6	4/16/6	0/24/2	1/25/0	~
Rank		7	7	8	4	4	5	3	2	6	7	1

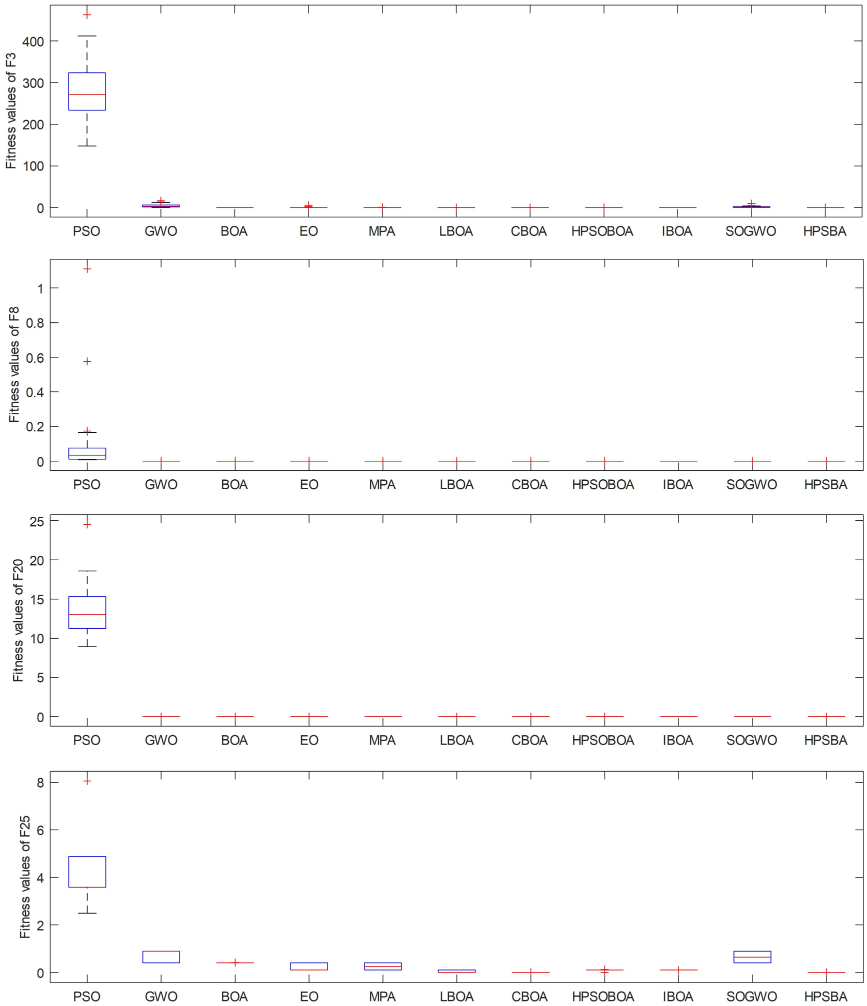


Figure 9. Box plot results for the function F3, F8, F20 and F25.

5. Nodes Coverage Optimization in WSN

Through the above experiments, it can be known that HPSBA performs better in numerical optimization problems, but for practical problems, the effectiveness of the proposed algorithm remains to be verified. In this section, we apply the proposed HPSBA without parameter ω in Equation (14) to the NOC problem of the WSN. The model description and objective function of the problem are detailed in Section 2.3.

5.1. Parameter Setting and Pseudo Code of Node Coverage Using HPSBA

To confirm the performance of the HPSBA for solving the NOC problem, there are two group experiments are designed as follows: (1) To study the performance of the HPSBA in node optimization coverage problem, three comparison algorithms, PSO, BOA, and MBOA, are employed as competitors. (2) The HPSBA algorithm is applied to the problem of optimal coverage of nodes with obstacles.

Equation (10) can be used to determine the number of sensor nodes needed to cover the theoretical area. The node coverage area, the number of sensor nodes, and the parameter settings of the simulation experiment are shown in Table 6.

Table 6. Node coverage parameter settings.

Parameters	Setting Values	
Side length of coverage area/m	100 × 100	100 × 100
Number of nodes	45	40, 45, 50
Perception radius/m	10	10
Communication radius r_c/m	20	20
Maximum iterations (T_{max})	100, 150, 200	150
Boundary threshold/m	$r_s/3$	$r_s/3$

The simulations of node optimization coverage of the WSN are conducted using PSO, BOA, MBOA, and HPSBA. The population and number of iterations of comparison algorithms are respectively set 30 and 150. Additionally, the following is a discussion of the parameters for the comparison methods:

- PSO based node optimization coverage: Each target node runs the PSO to become a deployed node. Parameters that are considered for coverage are: inertial weight = 0.7, cognitive and social scaling parameters $c_1 = c_2 = 2$.
- BOA based node optimization coverage: Each target node runs the BOA to become a deployed node. Parameters that are considered for coverage are: probability switch weight $SP = 0.8$, cognitive and social scaling parameters $c = 1$, and $a = 0.1$.
- MBOA based node optimization coverage: Each target node runs the MBOA to become a deployed node. Parameters that are considered for coverage are: probability switch weight $SP = 0.5$, cognitive and social scaling parameters $c(0) = r_1 = 0.35$ with chaotic adjust strategy, and $a = 0.1$.
- HPSBA based node optimization coverage: Each target node runs the proposed HPSBA to become a deployed node. Parameters that are considered for coverage are set as follows: initial value of inertial weight = 0.9, probability switch weight $SP = 0.6$, cognitive and social scaling parameters $c_1 = c_2 = 2, a = 0.1$ and $c(0) = 0.35$.

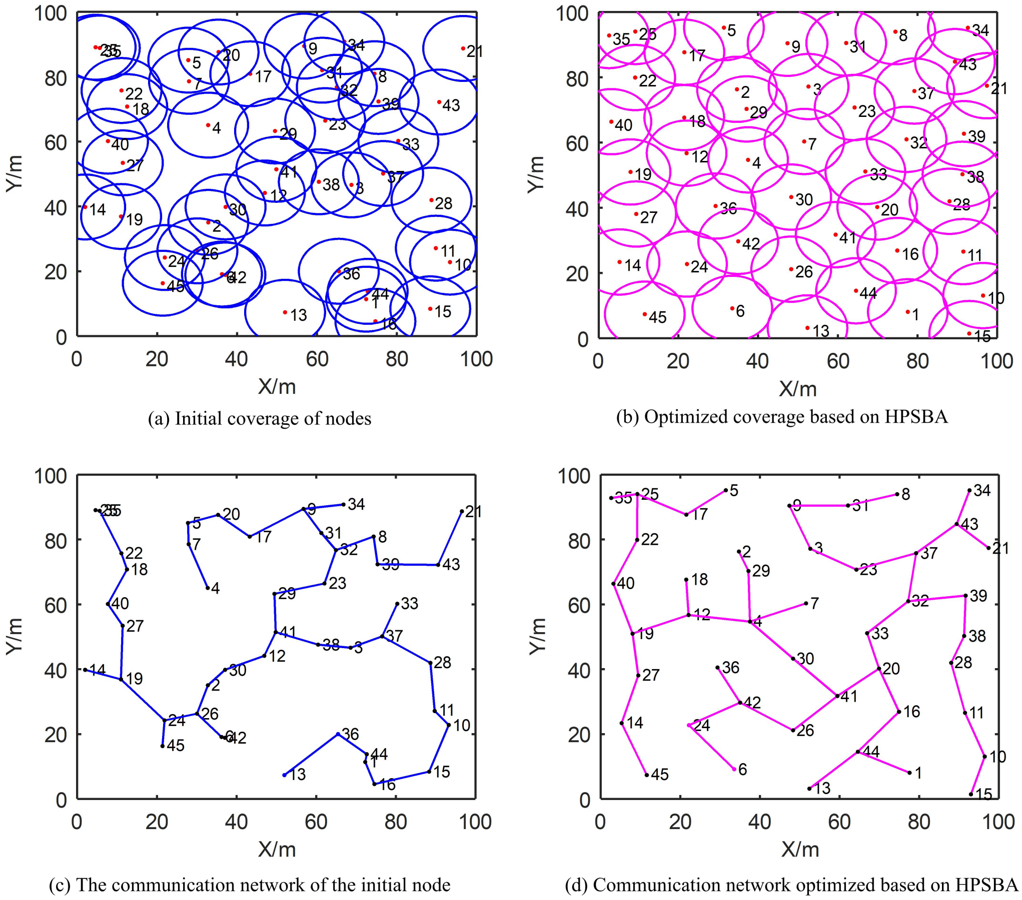
5.2. Results Analyses of Coverage Optimization Problem

5.2.1. The Effect of the Number of Nodes on Coverage

To further test the optimization performance of the HPSBA for coverage optimization problem, the coverage rates of various algorithms in the monitoring area under different numbers of sensor nodes are compared. Deploy sensor nodes in a 100 m × 100 m square monitoring area, with that the sensing radius is set 10 m, the communication radius is set 20 m, and the maximum iterations is set 150. Comparison algorithms are used when sensor nodes numbers are 40, 45, and 50, respectively. When nodes are 45 with 150 iterations, the coverage simulation results are shown in Figure 10. Other parameters remain unchanged, and the variation trend of the coverage rate with the number of nodes under different coverage strategies is shown in Figure 10a.

As seen from Figure 10 that the initial random coverage is shown in Figure 10a with 45 sensor nodes. As the number of iterations of HPSBA reaches 150 times, the coverage position of the nodes is shown in Figure 10b. According to the HPSBA to optimize the sensor node coordinate positions before and after coverage, the minimum spanning tree Prim algorithm [55] is used to draw the node communication network in the coverage area, as shown in Figure 10c,d.

As seen from Figure 10c,d, the uniformity of the communication distance between the initially deployed nodes is poor, the sink node is located in the center of the coverage area, and the data transmission distance between the nodes is longer. Energy consumption is large, and the optimized communication distance between nodes is more uniform, there are multiple convergence nodes, and the location is located at the boundary, thereby enhancing the reliability of the network, thereby reducing the energy consumption of node data transmission, and extending the network life.



(a) Initial coverage of nodes

(b) Optimized coverage based on HPSBA

(c) The communication network of the initial node

(d) Communication network optimized based on HPSBA

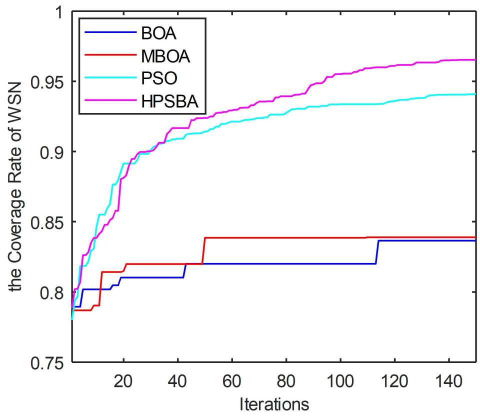
Figure 10. Node coverage and communication distribution diagram.

The coverage curves of node coverage based on BOA, MBOA, PSO and HPSBA node are shown in Figure 11. The coverage effect of the BOA is poor. HPSBA is in the leading position after 10 iterations. The final coverage rates of the four algorithms after 150 iterations are: 83.65%, 83.90%, 94.12%, and 96.54%.

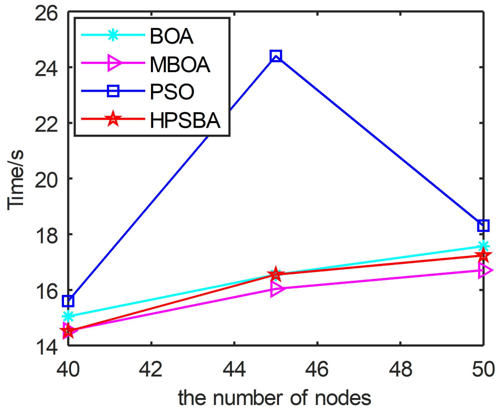
In the case of different numbers of nodes, Figure 12 provides an illustration of the calculated fitness average values. The proposed HPSBA coverage strategy is better than BOA, MBOA and PSO in the WSN node coverage with different number of nodes. Moreover, it is evident that the most effective simulation results are obtained with the proposed HPSBA, and the coverage rate reaches 93.15% in nodes = 40, 96.54% in nodes $N = 45$, and 98.42% in nodes $N = 50$. Compared to the standard BOA, the coverage rate is increased by the percentage points of 11.49, 12.89 and 12.41 in $N = 40$, $N = 45$ and $N = 50$, respectively. It is noteworthy that as the coverage optimization problem (80-dimension when $N = 40$, 90-dimension when $N = 45$, 100-dimension when $N = 50$) grows in size, HPSBA has more outstanding advantages over the basic BOA.

5.2.2. The Effect of the Number of Iterations on Coverage

To verify the influence of the number of algorithm iterations on the coverage rate, we set different iteration numbers, 100, 150, and 200, respectively. The optimized coverage results with 45 sensor nodes of the four comparison algorithms are shown in Table 7.



(a) Coverage curves



(b) Time of different nodes

Figure 11. Coverage comparison curves and optimized coverage time.

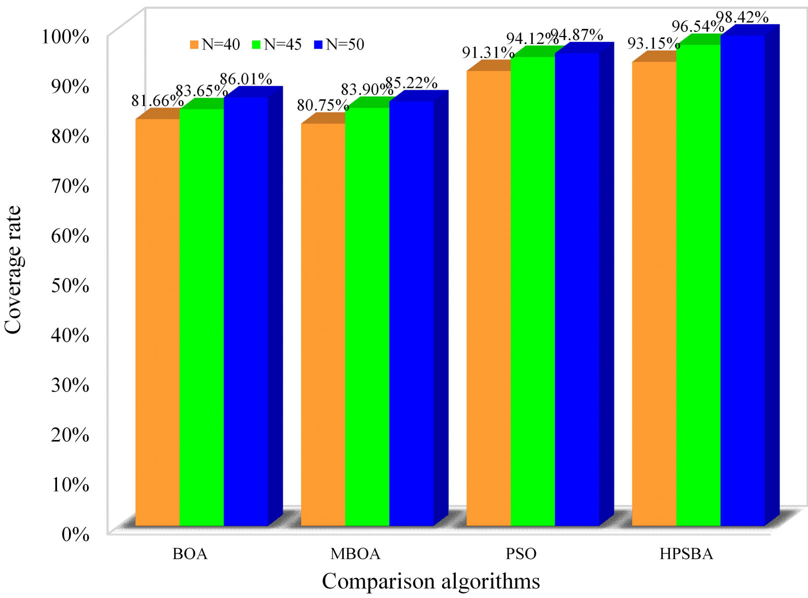


Figure 12. Coverage rates of different number of nodes.

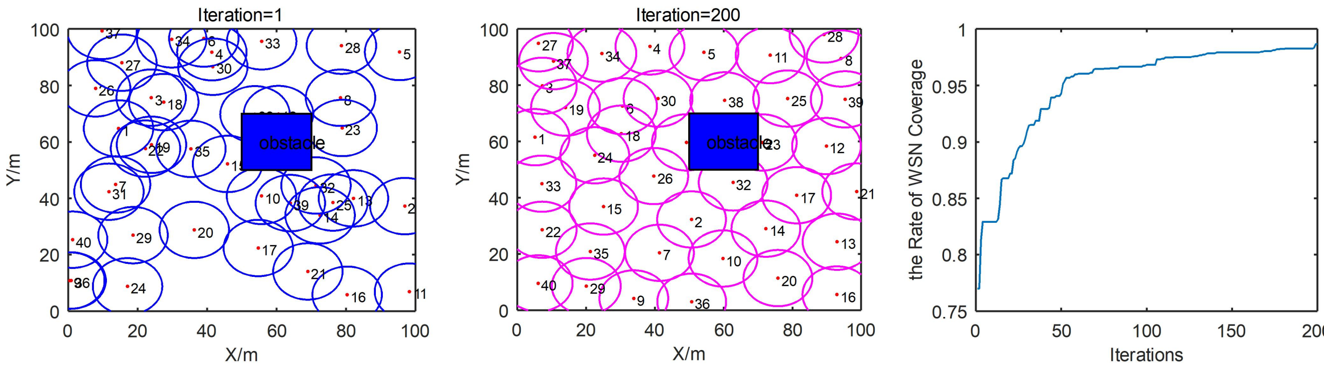
As seen from Table 7, it is evident that the most effective simulation results are obtained with the proposed HPSBA, and the coverage rate reaches 93.28% in $T_{max} = 100$, 96.54% in $T_{max} = 150$, and 96.32% in $T_{max} = 200$. Compared to the standard BOA, the coverage rate is increased by the percentage points of 8.49, 12.89, and 9.90 in T_{max} = 100, 150 and 200, respectively. It is noteworthy that as the coverage optimization problem (80-dimension when $N = 40$, 90-dimension when $N = 45$, 100-dimension when $N = 50$) grows in size, the proposed HPSBA demonstrates a superior advantage to the basic BOA. Furthermore, the proposed HPSBA compared with MBOA, the coverage rate is increased by the percentage points of 8.17, 12.64. HPSBA compared with PSO, the coverage rate is increased by the percentage points of 1.08, 2.42, and 2.00. Overall, when it comes to the WSN node coverage optimization problem, HPSBA outperforms the other three competitors.

Table 7. Node coverage rate of different iterations.

Item	T_{Max} = 100		T_{Max} = 150		T_{Max} = 200	
	Cov/%	**Time/s**	**Cov/%**	**Time/s**	**Cov/%**	**Time/s**
BOA	84.79	12.21	83.65	16.55	86.42	40.98
MBOA	85.11	10.93	83.90	16.04	85.82	21.10
PSO	92.20	11.94	94.12	24.4	94.32	51.92
HPSBA	93.28	11.09	96.54	16.55	96.32	21.31

5.2.3. Node Obstacle Avoidance Coverage Based on HPSBA

The proposed HPSBA has a positive effect on the sensor network's coverage of two-dimensional nodes when compared to the previous experimental results. To verify the effectiveness of the HPSBA node obstacle avoidance coverage problem, a node coverage experiment is designed under obstacles: the node coverage area is a two-dimensional plane of 100 m × 100 m, the number of sensor nodes is 40, the sensing radius r_s is set 10 m, communication radius $r_c = 20$ m, the number of iterations is 200, and the obstacle is a rectangular area of 20 m × 20 m. The optimization results of the coverage of obstacle avoidance nodes based on HPSBA and the coverage curve of 200 iterations are depicted in Figure 13.



(a) Iteration=1 (b) Iteration=200 (c) Coverage convergence curve

Figure 13. Coverage of obstacle avoidance nodes based on HPSBA.

As seen from Figure 13, HPSBA has a better application effect in the coverage of obstacle avoidance nodes. After 200 iterations, the node coverage in the set coverage area increased from 77.01% to 98.67%, an increase point of 21.66, and the time consumption is 21.34 s.

6. Conclusions

Aiming at the uneven distribution of nodes and low coverage in the random coverage of sensor networks, a hybrid particle swarm butterfly algorithm (HPSBA) is proposed for deploying WSN nodes. HPSBA improves the convergence speed through Logistic map and adaptive adjustment strategies, and the algorithm's accuracy at convergence is also improved. Through the optimization experiment of twenty-six benchmark functions with ten comparison swarm intelligence algorithms, the optimization results show that HPSBA's optimization ability and convergence accuracy are improved, and the stability is enhanced.

For the WSN node coverage problem, HPSBA can effectively coordinate the global exploration and local development capabilities of the algorithm. Compared with other algorithms, HPSBA effectively improves the coverage of WSN nodes while using fewer nodes, thus reducing it. The configuration cost of the network is reduced. However, the energy of nodes is usually taken into account in the real node coverage. More advanced algorithms [56] will be considered in the future work.

We will concentrate on the following tasks in future work: (i) In order to guarantee optimization precision, we will further develop HPSBA in light of the high complexity of its main framework. (ii) The proposed HPSBA will be further applied to solve multi-objective optimization problem, such as energy, distance, and uniformity for WSN in a three-dimensional environment, etc, between nodes.

Author Contributions: Methodology, M.Z.; software, M.Z.; writing—original draft preparation, M.Z. and J.Y.; supervision, D.W. and M.Y.; funding acquisition, D.W., W.T. and J.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The work was supported by NNSF of China (No.61640014), Industrial Project of Guizhou province (No. Qiankehe Zhicheng [2022]017, [2019]2152), Engineering Research Center of Guizhou Education Department (No. Qianjiaoji[2022]043), Innovation group of Guizhou Education Department under Grant Qianjiaohe (No.KY[2021]012), Science and Technology Fund of Guizhou Province under Grant Qiankehe (No.[2020]1Y266), Qiankehejichu [No.ZK[2022]Yiban103], Science and Technology Foundation of Guizhou University (Guidateganghezi [2021]04).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN'95-International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; Volume 4, pp. 1942–1948.
2. Gandomi, A.H.; Yang, X.S.; Alavi, A.H. Cuckoo search algorithm: A metaheuristic approach to solve structural optimization problems. *Eng. Comput.* **2013**, *29*, 17–35. [CrossRef]
3. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
4. Mirjalili, S.; Lewis, A. The whale optimization algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67.
5. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377. [CrossRef]
6. Dorigo, M.; Di Caro, G. Ant colony optimization: A new meta-heuristic. In Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406), Washington, DC, USA, 6–9 July 1999; Volume 2, pp. 1470–1477.
7. Yang, X.S. *Nature-Inspired Metaheuristic Algorithms*; Luniver Press: Frome, UK, 2010.
8. Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl.-Based Syst.* **2015**, *89*, 228–249. [CrossRef]
9. Saremi, S.; Mirjalili, S.; Lewis, A. Grasshopper optimisation algorithm: Theory and application. *Adv. Eng. Softw.* **2017**, *105*, 30–47. [CrossRef]
10. Arora, S.; Singh, S. Butterfly optimization algorithm: A novel approach for global optimization. *Soft Comput.* **2019**, *23*, 715–734. [CrossRef]
11. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73. [CrossRef]
12. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
13. Simon, D. Biogeography-based optimization. *IEEE Trans. Evol. Comput.* **2008**, *12*, 702–713. [CrossRef]
14. Koza, J.R. Genetic programming II: Automatic discovery of reusable subprograms. *Cambridge MA USA* **1994**, *13*, 32.
15. Kirkpatrick, S.; Gelatt, C.D., Jr.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [CrossRef] [PubMed]
16. Rashedi, E.; Nezamabadi-pour, H.; Saryazdi, S. GSA: A Gravitational Search Algorithm. *Inf. Sci.* **2009**, *179*, 2232–2248. [CrossRef]
17. Geem, Z.W.; Kim, J.H.; Loganathan, G.V. A new heuristic optimization algorithm: Harmony search. *Simulation* **2001**, *76*, 60–68. [CrossRef]
18. Mirjalili, S. SCA: A sine cosine algorithm for solving optimization problems. *Knowl.-Based Syst.* **2016**, *96*, 120–133. [CrossRef]
19. Faramarzi, A.; Heidarinejad, M.; Stephens, B.; Mirjalili, S. Equilibrium optimizer: A novel optimization algorithm. *Knowl.-Based Syst.* **2020**, *191*, 105190. [CrossRef]
20. Ahmadianfar, I.; Bozorg-Haddad, O.; Chu, X. Gradient-based optimizer: A new metaheuristic optimization algorithm. *Inf. Sci.* **2020**, *540*, 131–159. [CrossRef]
21. Rao, R.V.; Savsani, V.J.; Vakharia, D. Teaching–learning-based optimization: An optimization method for continuous non-linear large scale problems. *Inf. Sci.* **2012**, *183*, 1–15. [CrossRef]
22. Glover, F. Tabu search: A tutorial. *Interfaces* **1990**, *20*, 74–94. [CrossRef]
23. Kumar, M.; Kulkarni, A.J.; Satapathy, S.C. Socio evolution & learning optimization algorithm: A socio-inspired optimization methodology. *Future Gener. Comput. Syst.* **2018**, *81*, 252–272.
24. Askari, Q.; Younas, I.; Saeed, M. Political Optimizer: A novel socio-inspired meta-heuristic for global optimization. *Knowl.-Based Syst.* **2020**, *195*, 105709. [CrossRef]
25. Tu, J.; Chen, H.; Wang, M.; Gandomi, A.H. The colony predation algorithm. *J. Bionic Eng.* **2021**, *18*, 674–710. [CrossRef]
26. Li, C.; Chen, G.; Liang, G.; Luo, F.; Zhao, J.; Dong, Z.Y. Integrated optimization algorithm: A metaheuristic approach for complicated optimization. *Inf. Sci.* **2022**, *586*, 424–449. [CrossRef]

27. Yang, X.S. Flower pollination algorithm for global optimization. In *International Conference on Unconventional Computing and Natural Computation*; Springer: Berlin/Heidelberg, Germany 2012; pp. 240–249.
28. Kiran, M.S. TSA: Tree-seed algorithm for continuous optimization. *Expert Syst. Appl.* **2015**, *42*, 6686–6698. [CrossRef]
29. Liu, H.; Cai, Z.; Wang, Y. Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. *Appl. Soft Comput.* **2010**, *10*, 629–640. [CrossRef]
30. Razmjooy, N.; Ramezani, M. Training wavelet neural networks using hybrid particle swarm optimization and gravitational search algorithm for system identification. *Int. J. Mechatron. Electr. Comput. Technol.* **2016**, *6*, 2987–2997.
31. Garg, H. A hybrid GSA-GA algorithm for constrained optimization problems. *Inf. Sci.* **2019**, *478*, 499–523. [CrossRef]
32. Gong, Y.J.; Li, J.J.; Zhou, Y.; Li, Y.; Chung, H.S.H.; Shi, Y.H.; Zhang, J. Genetic learning particle swarm optimization. *IEEE Trans. Cybern.* **2015**, *46*, 2277–2290. [CrossRef]
33. El Sehiemy, R.A.; Selim, F.; Bentouati, B.; Abido, M. A novel multi-objective hybrid particle swarm and salp optimization algorithm for technical-economical-environmental operation in power systems. *Energy* **2020**, *193*, 116817. [CrossRef]
34. Arora, S.; Singh, S. Node localization in wireless sensor networks using butterfly optimization algorithm. *Arab. J. Sci. Eng.* **2017**, *42*, 3325–3335. [CrossRef]
35. Tan, L.S.; Zainuddin, Z.; Ong, P. Wavelet neural networks based solutions for elliptic partial differential equations with improved butterfly optimization algorithm training. *Appl. Soft Comput.* **2020**, *95*, 106518. [CrossRef]
36. Zhi, Y.; Weiqing, W.; Haiyun, W.; Khodaei, H. Improved butterfly optimization algorithm for CCHP driven by PEMFC. *Appl. Therm. Eng.* **2020**, *173*, 114766. [CrossRef]
37. Shams, I.; Mekhilef, S.; Tey, K.S. Maximum power point tracking using modified butterfly optimization algorithm for partial shading, uniform shading, and fast varying load conditions. *IEEE Trans. Power Electron.* **2020**, *36*, 5569–5581. [CrossRef]
38. Mengjian, Z.; Daoyin, L.; Qin, T.; Jing, Y. A Chaotic Hybrid Butterfly Optimization Algorithm with Particle Swarm Optimization for High-Dimensional Optimization Problems. *Symmetry* **2020**, *12*, 1800. [CrossRef]
39. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayirci, E. Wireless sensor networks: A survey. *Comput. Netw.* **2002**, *38*, 393–422. [CrossRef]
40. Rashid, B.; Rehmani, M.H. Applications of wireless sensor networks for urban areas: A survey. *J. Netw. Comput. Appl.* **2016**, *60*, 192–219. [CrossRef]
41. Liao, W.H.; Kao, Y.; Wu, R.T. Ant colony optimization based sensor deployment protocol for wireless sensor networks. *Expert Syst. Appl.* **2011**, *38*, 6599–6605. [CrossRef]
42. Adulyasas, A.; Sun, Z.; Wang, N. Connected Coverage Optimization for Sensor Scheduling in Wireless Sensor Networks. *IEEE Sensors J.* **2015**, *15*, 3877–3892. [CrossRef]
43. Wang, X.; Zhang, H.; Fan, S.; Gu, H. Coverage Control of Sensor Networks in IoT Based on RPSO. *IEEE Internet Things J.* **2018**, *5*, 3521–3532. [CrossRef]
44. Yang, M.; Wang, A.; Sun, G.; Zhang, Y. Deploying charging nodes in wireless rechargeable sensor networks based on improved firefly algorithm. *Comput. Electr. Eng.* **2018**, *72*, 719–731. [CrossRef]
45. Miao, Z.; Yuan, X.; Zhou, F.; Qiu, X.; Song, Y.; Chen, K. Grey wolf optimizer with an enhanced hierarchy and its application to the wireless sensor network coverage optimization problem. *Appl. Soft Comput.* **2020**, *96*, 106602. [CrossRef]
46. Wang, S.; You, H.; Yue, Y.; Cao, L. A novel topology optimization of coverage-oriented strategy for wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2021**, *17*, 1550147721992298. [CrossRef]
47. Dao, T.K.; Chu, S.C.; Nguyen, T.T.; Nguyen, T.D.; Nguyen, V.T. An Optimal WSN Node Coverage Based on Enhanced Archimedes Optimization Algorithm. *Entropy* **2022**, *24*, 1018. [CrossRef] [PubMed]
48. Abdollahzadeh, S.; Navimipour, N.J. Deployment strategies in the wireless sensor network: A comprehensive review. *Comput. Commun.* **2016**, *91–92*, 1–16. [CrossRef]
49. Zhang, M.; Yang, J.; Qin, T. An Adaptive Three-Dimensional Improved Virtual Force Coverage Algorithm for Nodes in WSN. *Axioms* **2022**, *11*, 199. [CrossRef]
50. Zhang, M.; Wang, D.; Yang, J. Hybrid-Flash Butterfly Optimization Algorithm with Logistic Mapping for Solving the Engineering Constrained Optimization Problems. *Entropy* **2022**, *24*, 525. [CrossRef]
51. May, R.M. Simple mathematical models with very complicated dynamics. *Nature* **1976**, *261*, 459–467. [CrossRef]
52. Solis, F.J.; Wets, R.J.B. Minimization by random search techniques. *Math. Oper. Res.* **1981**, *6*, 19–30. [CrossRef]
53. Zhang, M.-J.; Long, D.-Y.; Wang, X.; Yang, J. Research on Convergence of Grey Wolf Optimization Algorithm Based on Markov Chain. *Acta Electron. Sin.* **2020**, *48*, 1587–1595. [CrossRef]
54. Dhargupta, S.; Ghosh, M.; Mirjalili, S.; Sarkar, R. Selective Opposition based Grey Wolf Optimization. *Expert Syst. Appl.* **2020**, *151*, 113389. [CrossRef]
55. Knowles, J.; Corne, D. A new evolutionary approach to the degree-constrained minimum spanning tree problem. *IEEE Trans. Evol. Comput.* **2000**, *4*, 125–134. [CrossRef]
56. LaTorre, A.; Molina, D.; Osaba, E.; Poyatos, J.; Del Ser, J.; Herrera, F. A prescription of methodological guidelines for comparing bio-inspired optimization algorithms. *Swarm Evol. Comput.* **2021**, *67*, 100973. [CrossRef]