Research Article

A Learning Sparrow Search Algorithm

Chengtian Ouyang [ID]**, Donglin Zhu** [ID]**, and Fengqi Wang**

School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi 341000, China

Correspondence should be addressed to Donglin Zhu; 6920190624@mail.jxust.edu.cn

This paper solves the drawbacks of traditional intelligent optimization algorithms relying on 0 and has good results on CEC 2017 and benchmark functions, which effectively improve the problem of algorithms falling into local optimality. The sparrow search algorithm (SSA) has significant optimization performance, but still has the problem of large randomness and is easy to fall into the local optimum. For this reason, this paper proposes a learning sparrow search algorithm, which introduces the lens reverse learning strategy in the discoverer stage. The random reverse learning strategy increases the diversity of the population and makes the search method more flexible. In the follower stage, an improved sine and cosine guidance mechanism is introduced to make the search method of the discoverer more detailed. Finally, a differential-based local search is proposed. The strategy is used to update the optimal solution obtained each time to prevent the omission of high-quality solutions in the search process. LSSA is compared with CSSA, ISSA, SSA, BSO, GWO, and PSO in 12 benchmark functions to verify the feasibility of the algorithm. Furthermore, to further verify the effectiveness and practicability of the algorithm, LSSA is compared with MSSCS, CSin, and FA-CL in CEC 2017 test function. The simulation results show that LSSA has good universality. Finally, the practicability of LSSA is verified by robot path planning, and LSSA has good stability and safety in path planning.

1. Introduction

In recent decades, the swarm intelligence optimization algorithm has been favored by many scholars due to its simple structure and high solving efficiency. Plenty of intelligent optimization algorithms have appeared continuously, such as monarch butterfly optimization (MBO) [1], smooth mount algorithm (SMA) [2], mother search algorithm (MSA) [3], hunter games search (HGS) [4], naked mole-rat algorithm (NMRA) [5], and Harris hawks optimization (HHO) [6]. Xue and Shen [7] proposed a sparrow search algorithm (SSA) that simulates the nature of sparrows foraging for food with the advantages of simple principle, fewer adjustment parameters, less programming difficulty, etc. Compared with grey wolf optimizer (GWO) [8], particle swarm optimization (PSO) [9], and genetic algorithm (GA) [10] in function optimization, it has better search results. Although excellent ability has been strongly confirmed, however, the SSA algorithm also has its own shortcomings. SSA is highly dependent on a certain role in the group and lacks learning ability. It is still easy to fall into the local optimum on high-dimensional complex problems.

At present, lots of scholars have carried out a series of studies so as to minify the shortcomings of the SSA itself. Scholars have studied and improved it separately, for the sake of further improving the optimization ability of the SSA. For example, Lu et al. [11] proposed a chaotic sparrow search algorithm (CSSA). The algorithm first used the tent mapping based on random variables to generate a better individual sparrow sequence and introduced tent perturbation as well as Gaussian mutation in the optimization process to carry out the solution found by the sparrow. The update prevents the algorithm from appearing premature; furthermore, the effectiveness of the algorithm is verified in the test function and image segmentation issues. At the same time, they stated an improved sparrow search algorithm (ISSA) to apply to the multithreshold image segmentation problem [12] and achieved meaningful results. In the process of algorithm optimization, the idea of a bird swarm algorithm is applied. The benchmark function and

multithreshold image segmentation based on the variance between classes and Kapur entropy verify that the improved algorithm has strong search and development capabilities. Mao and Zhang [13] illustrated an improved sparrow search algorithm that combines Cauchy mutation and opposition-based learning. At first, a sin chaotic mechanism with an unlimited number of folds was used to initialize the population, and the previous generation global optimal solution was employed into the position update formula. For the aim of generating a new solution of higher quality, the information exchange of the algorithm was sped up, and the adaptive weight strategy was introduced to coordinate the local and global search capabilities, and the fusion Cauchy mutation and reverse learning strategy were utilized to perform disturbance mutation at the optimal position, which improved the ability of the algorithm to jump out of the local optimum. Eight test functions verify that the algorithm has been greatly improved in global optimization. Liu et al. [14], in order to better apply the SSA in 3D path planning, also adopted chaos strategies to enhance the diversity of the population and used adaptive inertial weights to balance the convergence speed and exploration capabilities of the algorithm. Finally, they adopted the Cauchy–Gaussian mutation strategy to get rid of the ability of the later algorithm stagnation, through the effect of planning the route, so that the improved SSA with a strong search ability is able to plan the route which is safer. Wang et al. [15] proposed a chaotic map sparrow search algorithm, which uses a dynamic adaptive weight mechanism to control the search range of the sparrow and finally uses reverse learning and Gaussian mutation to prevent the algorithm from falling into local optimality and balance the development of the algorithm and searchability as well. The 12 test functions proved that the improved algorithm has strong optimization capabilities. Zhang and Ding [16] also proposed a chaotic sparrow search algorithm, which strengthens the global search ability of the algorithm by introducing logistic mapping, adaptive hyperparameter, and mutation operation. The effectiveness of the algorithm is verified by the test function, and then, it is applied to the random configuration network. The results show that the proposed model has good regression accuracy.

The abovementioned authors have carried out a lot of experiments to verify the advantages of the proposed algorithm, but still there are some shortcomings:

(1) Most papers put forward chaos theory. Chaos theory itself has uncertainty and cannot change the randomness of the algorithm. Therefore, a flexible search mechanism is needed to improve the situation.

(2) The existing literature does not fundamentally change the optimization mechanism of the algorithm itself and lacks learning ability, so there is still a probability of falling into the local optimum when encountering high-dimensional complex problems.

(3) At present, the improved algorithms are only tested on the function of the optimal value dimension 0,

which lacks rationality and cannot fully explain the effectiveness of the algorithm.

(4) Researchers only consider the update of the best location, but ignore the update of the worst location of SSA. The search method of SSA is closely related to the worst position.

Based on the above shortcomings, this paper proposes a learning sparrow search algorithm, with the help of lens reverse learning and random reverse learning to improve the learning ability of the algorithm and adapt to various complex models. The improved sine and cosine algorithm is used to guide the followers to update the position and improve the search precision. As a result, the local search based on the difference is used to update the optimal solution to improve the quality of the solution. LSSA is compared with CSSA, ISSA, SSA, beetle swarm optimization (BSO) [17], GWO, and PSO in 12 benchmark functions to verify the feasibility of the algorithm. In order to further verify its effectiveness and practicability, LSSA is compared with multistrategy serial cuckoo search algorithm (MSSCS) [18], CSsin [19], and firefly algorithm with courtship learning (FA-CL) [20] in CEC 2017 test function. All the three algorithms are verified in the CEC test function; finally, the results show that the LSSA algorithm has strong universality. The contributions of this paper are as follows:

(i) The fusion of two opposition-based learning strategies is proposed to improve the algorithm's global search capability

(ii) An improved sine and cosine algorithm is proposed to improve the flexibility of the algorithm

(iii) A local search based on the difference is proposed to improve the quality of the solution each time

(iv) It has a good effect on the benchmark function and the CEC 2017 function and, at the same time, minifies the defect of the algorithm close to the origin

(v) LSSA is applied to robot path planning, and good results are achieved

This reminder of this paper is organized as follows. Section 2 introduces the basic sparrow search algorithm and analyzes it. Section 3 describes the process and validation of LSSA. Section 4 shows the experiment and analysis of each algorithm on benchmark function and CEC 2017 test function. Section 5 provides discussion and future research directions.

2. Sparrow Search Algorithm

The SSA is divided into three phases: discoverer, follower, and investigator. As the name implies, the discoverer discovers food, searches for food, and provides direction for other individuals in the population. Therefore, the discoverer searches for a wide range of food, which accounts for 20% of the population. The location update formula for the discoverer is

$$X_{i,j}^{t+1} = \begin{cases} X_{i,j}^{t} \cdot \exp\left(\dfrac{-h}{\alpha \cdot M}\right), & \text{if } R_2 < \text{ST}, \\ \\ X_{i,j}^{t} + Q \cdot L, & \text{if } R_2 \geq \text{ST}. \end{cases} \quad (1)$$

In formula (1), h represents the current number of iterations, M is the maximum number of iterations, $X_{i,j}$ denotes the current position of the ith sparrow in the jth dimension, $\alpha \varepsilon [0, 1]$ is a random number, R_2 and ST represent warning and safety values, respectively, and when $R_2 < [0, 1]$, $\text{ST} < [0.5, 1]$, Q is a normal distribution of random numbers, and L means 1 with all the elements of $1 \times D$. When $R_2 < \text{ST}$, this indicates that the community environment is safe at this time, no predators are found around them, and the discoverer can perform a wide search mechanism. When $R_2 \geq \text{ST}$, this indicates that the individual within the group has discovered the predator and issued an alert, that all individuals in the group will make antipredatory actions, and that the discoverer will lead the follower to a safe location.

Followers perform food searches after the discoverer and neighborhood searches around the discoverer's location. Followers' location updates the formula as follows:

$$X_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\dfrac{X_{\text{worst}}^{t} - X_{i,j}^{t}}{i^2}\right), & \text{if } i > \dfrac{n}{2}, \\ \\ X_{P}^{t+1} + \left|X_{i,j}^{t} - X_{P}^{t+1}\right| \cdot A^{+} \cdot L, & \text{otherwise}. \end{cases} \quad (2)$$

In formula (2), X_p is the optimal position currently occupied by the discoverer, X_{worst} represents the worst position currently, and A is a matrix of $1 \times d$, where the value of each element is 1 or -1, and $A^{+} = A^{T}(AA^{T})^{-1}$. When $i > n/2$, at this point, the sparrow population will counterattack when it senses danger.

Investigators are randomly selected individuals within the population. When predators invade, they will send out signals to make sparrows escape to a safe position. The behavior formula of investigators is as follows:

$$X_{i,j}^{t+1} = \begin{cases} X_{\text{best}}^{t} + \beta \cdot \left|X_{i,j}^{t} - X_{\text{best}}^{t}\right|, & \text{if } f_i \neq f_g, \\ \\ X_{i,j}^{t} + K \cdot \left(\dfrac{\left|X_{i,j}^{t} - X_{\text{worst}}^{t}\right|}{(f_i - f_w) + \varepsilon}\right), & \text{if } f_i = f_g. \end{cases} \quad (3)$$

In formula (3), X_{best} is the current global optimal location, β is the control step parameter, which is a random number with the normal distribution of mean value 0 and variance 1, $K \in [-1, 1]$ is a random number, f_i is the fitness value of the sparrow, f_g and f_w are the best and worst fitness values in the current search range, respectively, and ε is the smallest real number to prevent the denominator from 0. When $f_i \neq f_g$, it means that the sparrow is at the boundary of the population and vulnerable to predators, so it needs to adjust its position. When $f_i = f_g$, this indicates that the sparrow individuals in the population are aware of the

danger and need to be close to other sparrows in order to avoid the danger. K represents the direction of the sparrow movement and can also control the movement step.

2.1. Performance Analysis. SSA can be divided into three stages: discoverer, follower, and scout. From the three formulas, it can be seen that sparrow individuals depend on the search of the discoverer stage. The idea of adaptive weight is introduced in formula (1), but the adaptive weight still has defects in the face of high-dimensional complex functions and cannot open up a global vision. Therefore, it is necessary to make use of lens reverse learning and random reverse learning to dig out more hidden positions, but also increase the diversity of the population and make full preparation for the optimization in the later stage. Formula (2) has the defect of near-zero points; hence, nonlinear sine-cosine guidance is used to balance the local and global search. From the overall formula, the update distance between the front and back position of SSA is far, so the blind area between them becomes more. The local search based on the difference can improve the search precision and reduce the scope of the blind areas.

3. Learning Sparrow Search Algorithm

3.1. Opposition-Based Learning Strategy Based on Lens Principle. The discoverer leads other individuals to search for food, and the search method directly affects the overall search performance, so the discoverer must have a wide range and flexible search mechanism. To solve these problems, researchers have proposed the corresponding learning mechanism [21–23]. The general opposition-based learning strategy only solves the problem in a certain space [24–26], which still has monotonicity and risk of local optimization. As for this phenomenon, this paper provides two fusion opposition-based learning strategies to jointly improve the searchability of the discoverer. The opposition-based learning strategy based on the lens principle [27] is used to effectively update the location of the discoverer. The schematic diagram is shown in Figure 1. It is the opposition-based learning solution of the lens principle that is flexible and diverse, which is conducive to mining new solutions in the unknown area and increasing the diversity of the population. The principle is as follows:

In a certain space, suppose an individual P of height h, and individual X_p is the projection of individual P onto the X-axis. A lens of focal length f is placed on the base point position O, O is the midpoint of $[a_j, b_j]$, and a_j and b_j represent the upper and lower limits of the jth dimension of the current solution. An image P' of height H' is obtained by the lens imaging process, and its projection on the coordinate axis is X_P' (reverse point). At this point, X_P' is the new individual generated by X_p through the opposition-based learning strategy based on the principle of lens imaging. The schematic diagram is shown in Figure 1.

As shown in Figure 1, the corresponding reverse point X_P' of individual X_p is obtained by taking O as the base point, which can be obtained by the lens imaging principle:
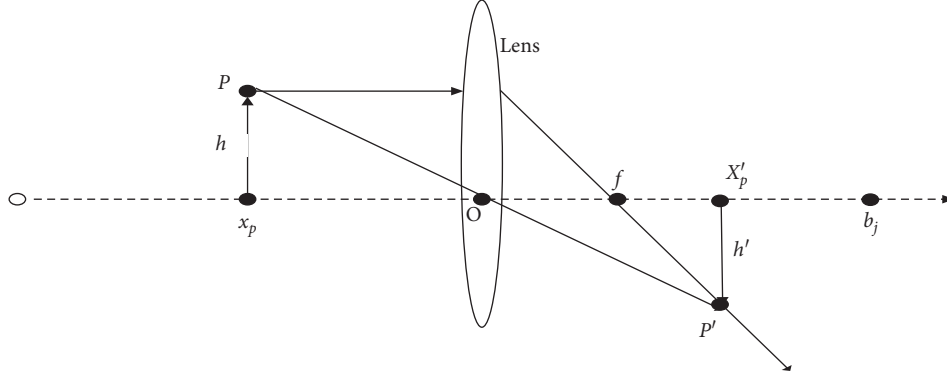
FIGURE 1: Lens schematic diagram.

$$\frac{a + b/2 - x_p}{x_p' - a + b/2} = \frac{h}{h'}. \tag{4}$$

Let $h/h' = k$, and k is the scaling factor. After transformation, the reverse point can be obtained as

$$x_p' = \frac{a + b}{2} + \frac{a + b}{2k} - \frac{x_p}{k}. \tag{5}$$

Thus, when $k = 1$,

$$x_p' = a + b - x_p. \tag{6}$$

Formula (6) is called a general opposition-based learning strategy. From the above formulas, it is known that the general learning strategy is only a specific case of lens imaging opposition-based learning strategy, and the new individuals obtained by the general opposition-based learning strategy are fixed each time. In high-dimensional complex functions, new individuals with the fixed range also have the possibility of falling into the local optimum, and they are monotonic. By adjusting the parameter K, the new individuals based on the lens imaging learning strategy are dynamic, which improves the diversity of the population.

In this paper, we generalize the formula to the d-dimensional space:

$$x_P'^j = \frac{a_j + b_j}{2} + \frac{a_j + b_j}{2k} - \frac{x_P^j}{k}. \tag{7}$$

In formula (7), x_p^j and $x_p'^j$ are the j-dimensional components of x_p and x_p', respectively, and a_j and b_j represent the j-dimensional components of the upper and lower bounds of decision variables, respectively.

3.2. Opposition-Based Learning of the Worst Position.
After the discoverer has searched, the worst position they get is not necessarily reliable. From formulas (2) and (3), it is known that the worst solution will affect the later stage of optimization, and the minimum value will give followers a better search range. This means that updating at the worst location is extremely important, which is also a point that scholars tend to ignore, only pursuing the optimal location and ignoring the integrity of the algorithm. This paper uses the random opposition-based mechanism to update the worst position; the specific formula is as follows:

$$x_{\text{worst}}'(t) = a_j + \text{rand} \cdot (b_j - x_{\text{worst}}). \tag{8}$$

3.3. Guidance Strategy Based on Improved Sine-Cosine Algorithm.
In the follower location update formula of SSA, the follower searches the location of the follower immediately after the discoverer, and there are few dynamic parameters, so it is easy to limit the search range of sparrow population and blindness, which limits the searchability of the algorithm. To solve these problems, the strategy of sine-cosine guidance [28–30] is used in the follower stage to dynamically update the sparrow's individual position and expand the search scope by using the sine-cosine characteristics. The formula for updating follower positions with sine-cosine strategy is

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \cdot \sin(r_2) \cdot |r_3 \cdot X_P^t - X_i^t|, & r_4 \leq 0.5, \\ X_i^t + r_1 \cdot \cos(r_2) \cdot |r_3 \cdot X_P^t - X_i^t|, & r_4 > 0.5, \end{cases} \tag{9}$$

$$r_1 = a - t \cdot \frac{a}{M}. \tag{10}$$

In the above formula, r_1 is a parameter, determined by the number of iterations, and it is the key to determine the individual search range. As the number of iterations increases, r_1 gets smaller and smaller, and the sparrow search range is also smaller and smaller. a is a constant, and the value of a in this paper is 2. r_2 is a random number in the range $[0, 2\pi]$, which determines the individual movement distance; r_3 and r_4 are random numbers in $[0, 2]$ and $[0, 1]$, respectively.

It can be seen from the formula that r_1 uses linear decline to balance the search scope, but this approach is easily trapped into local optimal [31, 32] when facing high-dimensional complexity functions. Therefore, this paper adopts nonlinear decline to set r_1 to balance local and global search [33]. The specific formula is as follows:

$$r_1 = c + \frac{b}{\exp\left(4 \times (t/M)^4\right) + 1}. \tag{11}$$

Among them, b is the fixed value of 0.1 and c is the regulating factor. After many experiments, when $c = 0.9$, the best effect is achieved. The introduction of an improved sine-cosine guidance strategy reduces the blindness of sparrow searches, accelerates information exchange between individuals in the population and those in the best and worst positions, and makes followers more purposeful in their searches. According to the characteristics of the above formulas, it is clear that the nonlinear decreasing parameters make the search more detailed and improve the convergence accuracy of the algorithm.

3.4. Local Search Based on Difference.

Sparrows do not always get reliable optimal solutions for each search. Precocious phenomena occur when local extremes are encountered, which paralyzes the algorithm. In order to overcome this limitation, a differential local search is proposed to get rid of the attraction of local extremes and improve the quality of the solution. P_1 and P_2 are the historic optimal solution and the historic suboptimal solution of the population, respectively, in the process of SSA optimization. In this paper, the difference between the two solutions is used to guide P_1 to search for a reliable optimal solution in the neighborhood. Differential guidance is accurate. This strategy enables sparrows to search between solutions to avoid missing high-quality solutions and blind searches. The implementation is as follows:

$$P_1' = P_1 + r \cdot C_t \cdot (P_1 - P_2), \tag{12}$$

where P_1' is the updated historical optimal solution, r is the uniform random number between $[-1, 1]$, and C_t is the local scaling factor. In the early stage of the algorithm, P_1 is far away from the optimal solution, so a larger search range is needed to speed up the search speed. In the later stage, the distance is relatively short, and a smaller search range is needed to achieve higher mining accuracy. Therefore, the idea of inertia weight is introduced here, and the linear decline strategy is adopted. The range gradually shrinks as the number of iterations increases:

$$C_{t+1} = C_t \cdot \left(\frac{1 - i}{M}\right). \tag{13}$$

The greedy strategy is adopted to preserve the solution P_1' obtained by local search as follows:

$$P_1 = \begin{cases} P_1', & \text{fit}(P_1') > \text{fit}(P_1), \\ P_1, & \text{otherwise}, \end{cases} \tag{14}$$

where $\text{fit}(x)$ is the fitness value of x.

3.5. Learning Sparrow Search Algorithm.

Compared with other algorithms, the SSA has better performance, but it has more random parameters, which leads to increased randomness and the probability of falling into the local extreme

value. Therefore, a learning sparrow search algorithm is proposed in this paper. Two learning mechanisms are introduced in the finder stage, and the opposition-based learning based on the lens principle is adopted to enlarge the finder search range, improve the diversity of the population, and make the search method more flexible. In the follower stage, an improved sine and cosine strategy is introduced to adjust the sine and cosine by adopting a nonlinear decreasing method to reduce the blind search in the follower stage and make its search method more detailed. In the end, a local search based on difference is expounded to update the optimal solution and promote the quality of the solution during the iteration. The algorithm flow chart is shown in Figure 2. The specific pseudocode is shown in Algorithm 1.

3.6. Algorithm Validity Test.

For making it obvious that LSSA improves the optimization mechanism of SSA and verify the scientificity and effectiveness of the LSSA algorithm, this paper takes the Schwefel function as an example and gives the individual distribution map of the two algorithms in the optimization process. Let the maximum number of iterations be 20 and the population number be 50. The function model diagram is shown in Figure 3. The individual distribution diagram of the two algorithms is shown in Figures 4 and 5.

From Figures 4 and 5, the LSSA algorithm converges fast, and most individuals are close to the optimal value, from the initial individual to the final individual distribution, while the SSA algorithm is still in a decentralized state, and the convergence speed is slow. Therefore, the LSSA algorithm's search mechanism is broad and detailed, and it can quickly find the best in the optimization process.

3.7. Time Complexity Analysis.

Time complexity is an important index to judge an algorithm and determine the rationality of the algorithm. Let the population size of the algorithm in this paper be P, the maximum number of iterations be M, the dimension be D, and the ratio coefficients of discoverers and followers are R_1 and R_2, respectively. The time complexity of this paper is analyzed as follows:

From a macropoint of view, the time of the intelligent optimization algorithm is $O(P \times M \times D)$, so is the sparrow search algorithm. The improved sparrow search algorithm not only does not change the structure of the algorithm but also increases the number of cycles; in this way, its time complexity is $O(P \times M \times D)$, the same as the basic sparrow search algorithm.

From a microscopic point of view, the improved sparrow search algorithm increases a certain amount of computational complexity, and the opposition-based learning of the lens and the opposition-based learning of the worst position, respectively, increase the complexity of $O(R_1 \times P \times M \times D)$ and $O(M)$. The computational complexity of introducing the sine and cosine guiding mechanism is $O(R_2 \times P \times M \times D)$, and the time complexity of introducing the local search based on the difference is $O(M)$. It can be seen that the introduction of each strategy does not improve

```
Input
M: Maximum number of iterations
PD: Discoverer
SD: Individuals who are aware of the danger
R₂: Alert value
N: Population sparrows
Output: X_best, f_g
Initialize population
t = 1;
While (t < M)
Find the position of the best and worst sparrow individuals according to fitness values.
R₂ = rand(1)
For i = 1 : PD
Update the location of the discoverers according to formulas (1) and (7);
End for
Update the worst location found by the discoverer according to formula (8);
For i = (PD + 1) : N
Update the location of the followers according to formulas (2) and (9);
End for
For l = 1 : SD
Get the individual position of a sparrow that is aware of danger according to formula (3);
End for
Get the Location of the New Optimal Individual;
Update the individual location according to formulas (12)–(14);
t = t + 1
End while
Return: X_best, f_g
```

ALGORITHM 1: The framework of the LSSA.

the order of magnitude of the sparrow search algorithm, and the time complexity is still $O(P \times M \times D)$.

4. Benchmark Function Test

In order to better verify the optimization ability of the LSSA algorithm, this paper first selects 10 standard test functions for verification and compares them with the six algorithms of PSO, GWO, SSA, ISSA, CSSA, and BSO. BSO is a new and hot research fusion algorithm in recent years. The specific parameters are shown in the literature. The test function information table is shown in Table 1. F_1–F_6 are complex unimodal functions, and F_7–F_9 are high-dimensional complex functions; the rest is a fixed-dimensional function, where F_1–F_9 are tested in 30 and 100 dimensions. The population size and maximum iteration number of each algorithm are 100 and 500, respectively. The two learning factors in the particle swarm algorithm are $c1 = c2 = 1.429$ and weight $w = 0.729$, and each algorithm runs 30 times independently and calculates the best value (best), average (Ave), and standard deviation (std) of the running results. Three indicators comprehensively evaluate the optimization ability of each algorithm in function. For performance evaluation, simulations are performed on Windows 10 having Matlab 2019a, Intel(R) Core (TM) i5-10200H CPU @ 2.40 GHz with 16 GB RAM.

From Tables 2 and 3, the LSSA algorithm exhibits good optimization effects in both 30 and 100 dimensions. In the unimodal function, both LSSA and basic SSA algorithm can

find the optimal value, indicating that the LSSA algorithm does not reduce the optimization ability of the algorithm itself, which shows the rationality of the LSSA algorithm. In the multimodal function, LSSA can show a strong optimization ability and has better convergence accuracy than other algorithms, and it does not significantly weaken the optimization ability of the LSSA algorithm in the 100 dimensions. In the fixed dimension function F_{10}–F_{12}, the LSSA algorithm has good stability and can find the same value almost every time, which is close to the theoretical optimal value.

In order to better describe the optimization process and convergence speed of each algorithm, the 30-dimensional convergence graph of each algorithm on each function is given, as shown in Figure 6.

As shown in Figure 6, LSSA has great advantages in the optimization speed and convergence accuracy of each function. It converges quickly on the unimodal function and has better antilocal attraction ability on the multimodal function. It can be seen that the LSSA algorithm gets rid of the constraints of the original algorithm's search mechanism and develops a better search space.

5. CEC 2017 Function Test

5.1. Algorithm Complexity. According to the requirements of CEC 2017 test standard, the complexity of the proposed algorithm needs to be calculated. Therefore, the following code is used to calculate the running time T_0 of LSSA:
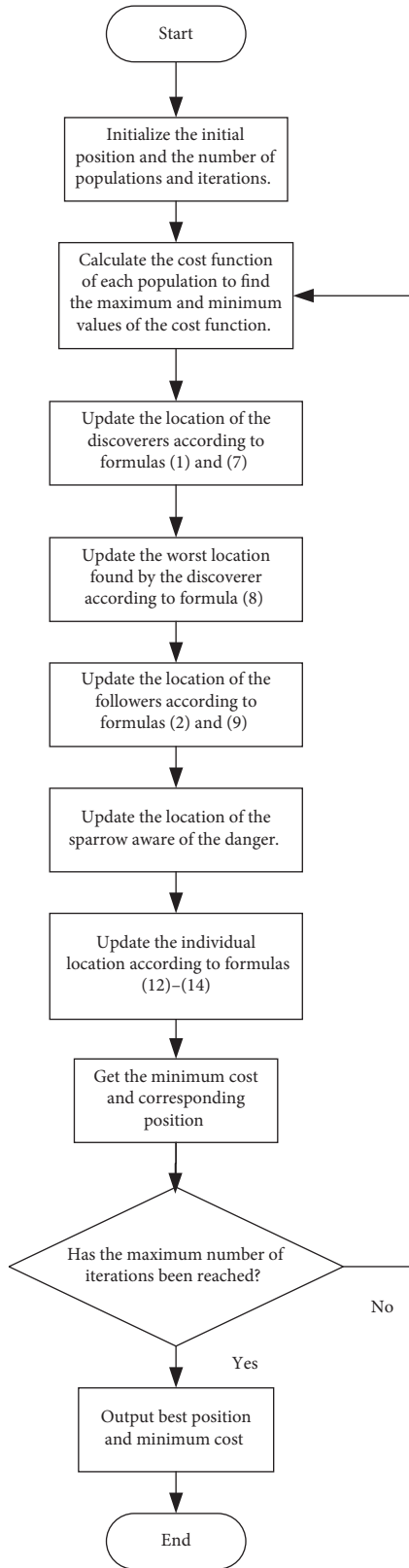
FIGURE 2: Algorithm flowchart.
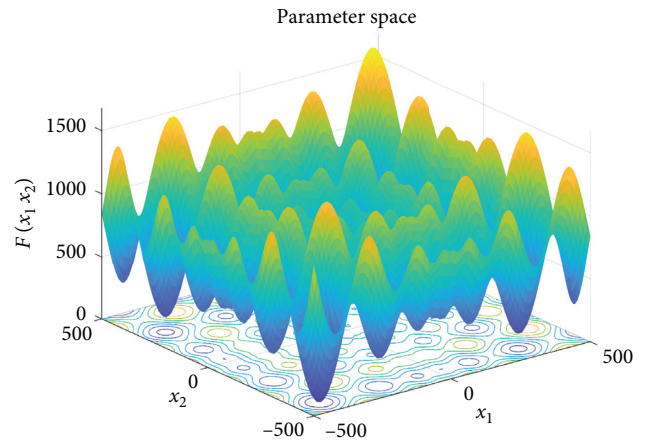
FIGURE 3: Schematic diagram of the Schwefel function.

for $i = 1 : 1,000,000$

$x = 0.55 + \text{double}(i);$ $x = x + x;$ $x = x/2;$ $x = x * x;$
$x = \text{sqrt}(x);$ $x = \log(x);$ $x = \exp(x);$ $x = x/(x + 2);$

end

T_1 is the calculation time of the F18 function under 200,000 evaluation times, and T_2 is the average calculation time of the F_{18} function for 5 times under the same condition. The specific algorithm complexity is shown in Table 4.

5.2. Function Test. Experiments on benchmark functions alone cannot show the universality and effectiveness of the algorithm. To better illustrate the effectiveness of the LSSA algorithm and avoid the situation that the LSSA algorithm depends on the optimal value of 0, the algorithm is tested on CEC 2017 test function [34], and the number of evaluations is $10,000 * \text{dim}$, the dimension is 50 or 30, and the population number is 50. The test results of LSSA and SSA, CSSA, MSSCS, CSsin, and FA-CL are compared, and the specific parameters of each algorithm are shown in Table 5. Each algorithm runs 30 times independently and calculates five indexes of the results of each algorithm running on the function, namely, the best, the worst, the median, the mean, and the standard deviation. Each index can clearly reflect the optimization ability of each algorithm. The best value in each indicator is treated in bold font. At the same time, the Wilcoxon rank test is used to show whether there is a significant difference in each algorithm, and the experiment is carried out at the significance level of 5%. " + " means that the LSSA algorithm is better than other algorithms in the optimization effect, "−" means the opposite, and " = " means the same performance. Finally, the comparison of each algorithm is counted. The test results are shown in Tables 6 and 7.

(a)

(b)
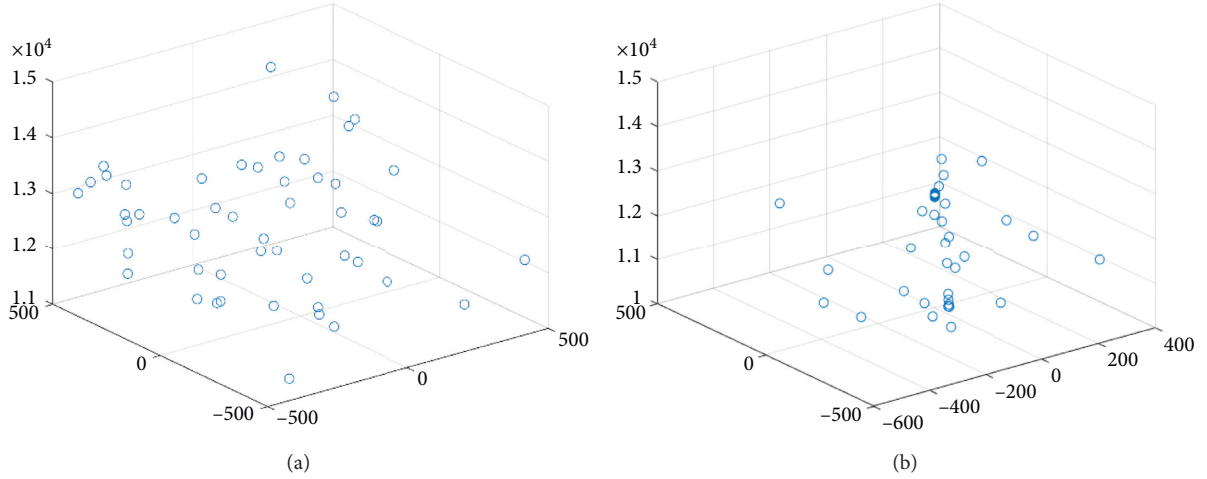
FIGURE 4: Individual distribution of SSA. (a) SSA individual initialization map. (b) Individual distribution of SSA in 100 generations.

(a)

(b)
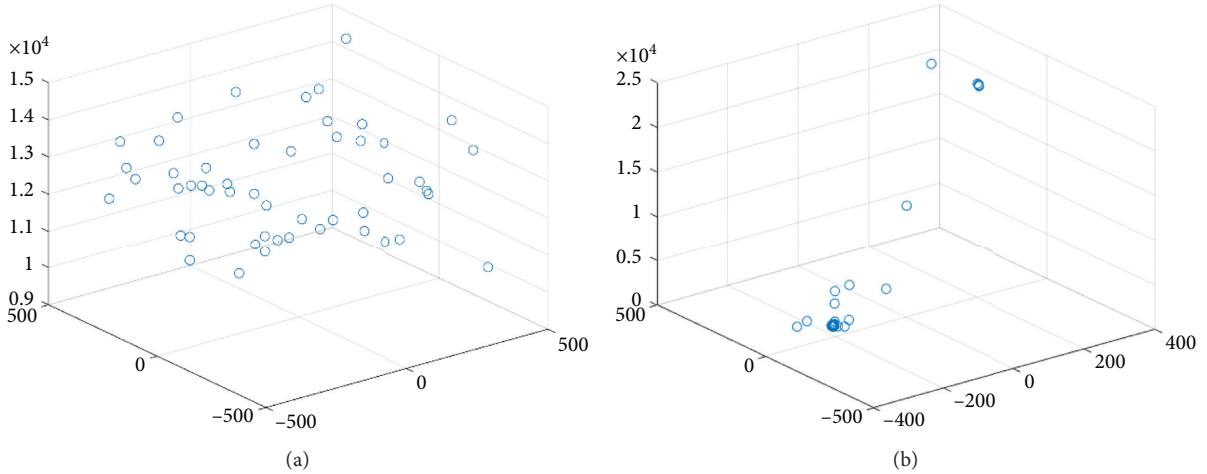
FIGURE 5: Individual distribution of LSSA. (a) LSSA individual initialization map. (b) Individual distribution of LSSA in 100 generations.

From Tables 6 and 7, it is apparent that LSSA has a better optimization effect in both 30 and 50 dimensions, and each function is close to the theoretical optimal value, but the effect on the F_4 function is poor. Although MSSCS and CSsin have better optimization effects, they show the very poor effects on F_1, F_3, and F_{12-13} in 50 dimensions. It can be seen that these two algorithms have limitations in this kind of problem. Other algorithms perform poorly and rarely approach the theoretical optimal value. From the statistical test, the LSSA algorithm is significantly different from other algorithms, showing better advantages, and some functions are similar to MSSCS and CSsin. Generally speaking, the LSSA algorithm has high universality and is more suitable for some complex optimization problems than other algorithms.

6. Robot Path Planning

This paper takes the classic robot path planning case to explore it. In path planning, each sparrow is a feasible path.

Suppose that there are n feasible paths, and dimension D is determined by the number of lines from the starting point to the target point. The grid method is used for environment modeling, and the grid method is to use 1×1. According to the grid value, the obstacles in the equivalent position are calculated. Grid number 0 is defined as the feasible area, and 1 is the obstacle area. Then, the robot can plan the path on the grid assigned to 0, and dimension D is the column number of the grid map. The cost function of the path length of the ith sparrow is shown in the following equation:

$$f(x) = \sum_{j=1}^{D-1} \sqrt{\left(x_{j+1} + x_j\right)^2 + \left(y_{j+1} - y_j\right)^2}. \tag{15}$$

In equation (12), j is the jth dimension of a sparrow.

In order to better verify the practicability of the improved algorithm, LSSA is applied to robot path planning and SSA is used for comparative experiments. The number of populations is 50, and the number of iterations is 20. Other environmental parameters are consistent with the

TABLE 1: Test function.

Function	Dimensions	Interval	Min
$F_1(x) = \sum_{i=1}^n x_i^2$	30/100	[−100, 100]	0
$F_2(x) = \sum_{i=1}^n \left(\sum_{j=1}^i x_j\right)^2$	30/100	[−100, 100]	0
$F_3(x) = \max_i\{|x_i|, 1 \le i \le n\}$	30/100	[−100, 100]	0
$F_4(x) = \sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30/100	[−30, 30]	0
$F_5(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	30/100	[−100, 100]	0
$F_6(x) = \sum_{i=1}^n i x_i^4 + \text{random}[0, 1)$	30/100	[−1.28, 1.28]	0
$F_7(x) = \sum_{i=1}^n -x_i \sin(\sqrt{|x_i|})$	30/100	[−500, 500]	−418.9829n
$F_8 = \frac{\pi}{n}\left\{10 \sin(\pi y_1) + \sum_{i=1}^{n-1}(y_i - 1)^2[1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2\right\} + y_i = 1 + x_i + 1/4$	30/100	[−50, 50]	0
$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$			
$F_9 = 418.9829n - \sum_{i=1}^n x_i \sin(\sqrt{|x_i|})$	30/100	[−500, 500]	0
$F_{10}(x) = (1.5 - x_1 + x_1 x_2)^2 + (2.25 - x_1 + x_1 x_2^2)^2 + (2.625 - x_1 + x_1 x_2^3)^2$	2	[−4.5, 4.5]	0
$F_{11}(x) = 100(x_1^2 - x_2)^2 + (x_1 - 1)^2 + (x_3 - 1)^2 + 90(x_3^2 - x_4)^2 + 10.1((x_2 - 1)^2 + (x_4 - 1)^2) + 19.8(x_2 - 1)(x_4 - 1)$	4	[−10, 10]	0
$F_{12}(x) = \left(0.002 + \sum_{i=1}^{25} 1/i + (x_1 - a_{1i})^6 + (x_2 - a_{2i})^6\right)^{-1}$ where $a = \begin{pmatrix} -32 & -16 & 0 & 16 & 32 & -32 & \dots & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & \dots & 32 & 32 & 32 \end{pmatrix}$	2	[−65.536, 65.536]	0.998

TABLE 2: Comparison table of optimization effect of each algorithm (30 dimensions and fixed dimensions).

Function	Algorithm	Best	Ave	Std
$F_1(x)$	LSSA	0	0	0
	BSO	1.8337	9.2351	8.0567
	CSSA	0	0	0
	ISSA	0	0	0
	SSA	0	0	0
	GWO	$2.9726E-42$	$9.9797E-41$	$2.3113E-40$
	PSO	$1.562E-12$	$7.3034E-11$	$1.5384E-10$
$F_2(X)$	LSSA	0	0	0
	BSO	$9.4317E-10$	5.8438	19.6417
	CSSA	0	0	0
	ISSA	0	0	0
	SSA	0	$7.0603E-92$	$3.8713E-91$
	GWO	$1.2722E-15$	$1.6521E-11$	$5.6379E-11$
	PSO	2.4597	6.8764	4.6663
$F_3(X)$	LSSA	0	0	0
	BSO	0.0417	1.5546	1.2080
	CSSA	0	0	0
	ISSA	0	$3.7672E-158$	$2.0634E-157$
	SSA	0	0	0
	GWO	$1.0249E-11$	$1.3528E-10$	$1.3629E-10$
	PSO	0.06008	0.1756	0.08657
$F_4(X)$	LSSA	$1.3707E-10$	$8.2877E-06$	$1.5080E-05$
	BSO	233.8862	971.7400	841.0884
	CSSA	$3.47E-09$	$7.3760E-05$	$1.0942E-04$
	ISSA	$1.0562E-07$	$1.2681E-04$	$2.7208E-04$
	SSA	$7.6415E-09$	$3.1722E-05$	$5.9606E-05$
	GWO	45.1861	46.6720	0.7848
	PSO	10.0094	102.2121	51.6541
$F_5(X)$	LSSA	0	$8.3405E-32$	$1.7901E-31$
	BSO	0	$3.6491E-21$	$1.9700E-20$
	CSSA	$2.4651E-32$	$6.1469E-29$	$1.7338E-28$
	ISSA	$4.6878E-29$	$1.3501E-25$	$3.1929E-25$
	SSA	$3.0814E-32$	$2.7691E-29$	$7.1572E-29$
	GWO	$3.3762E-07$	$1.0734E-06$	$3.7183E-07$
	PSO	$8.9363E-32$	$1.9760E-30$	$5.1909E-30$
$F_6(X)$	LSSA	$7.5495E-06$	$3.6715E-04$	$3.4039E-04$
	BSO	$1.8821E-03$	$8.0499E-03$	$4.1052E-03$
	CSSA	$1.5867E-05$	$1.7275E-04$	$1.2860E-04$
	ISSA	$1.0528E-05$	$2.2030E-04$	$1.7169E-04$
	SSA	$8.2819E-06$	$1.6418E-04$	$1.7567E-04$
	GWO	$1.8171E-04$	$5.6482E-04$	$3.1679E-04$
	PSO	$7.1929E-03$	$1.8162E-02$	$7.2361E-03$
$F_7(X)$	LSSA	-12569.4866	-12151.0951	724.9186
	BSO	-12569.4866	-11205.6626	1020.4548
	ISSA	-10141.2347	-9080.1757	568.2124
	CSSA	-10022.9333	-9045.4556	574.4343
	SSA	-10258.1903	-8724.1633	718.2918
	GWO	-8262.1714	-6362.1144	709.0298
	PSO	-8324.1386	-6844.9716	752.6817
$F_8(X)$	LSSA	$3.5924E-19$	$1.5863E-15$	$3.8647E-15$
	BSO	0.1323	0.9448	0.6457
	CSSA	$8.7326E-15$	$2.3930E-12$	$4.4060E-12$
	ISSA	$1.9914E-14$	$2.6662E-12$	$5.2284E-12$
	SSA	$3.4524E-15$	$1.4887E-11$	$3.5684E-11$
	GWO	$1.4151E-06$	0.01468	$9.2774E-03$
	PSO	$6.4837E-14$	0.08643	0.2527

TABLE 2: Continued.

Function	Algorithm	Best	Ave	Std
$F_9(X)$	LSSA	$4.9797E-04$	719.9689	812.9308
	BSO	$3.8182E-04$	1272.5502	983.9231
	CSSA	-9082.3593	-8132.3934	763.6265
	ISSA	2191.2870	3460.1076	584.0178
	SSA	2587.7399	4017.9491	668.5352
	GWO	4518.4944	6050.6684	801.1268
	PSO	4601.8502	6257.0871	952.5740
$F_{10}(X)$	LSSA	0	0	0
	BSO	0	0.3556	0.3866
	CSSA	0	$1.8488E-33$	$7.0361E-33$
	ISSA	0	0	0
	SSA	0	$8.2173E-33$	$4.5008E-32$
	GWO	$3.2417E-10$	$4.6721E-08$	$3.9041E-08$
	PSO	0	0	0
$F_{11}(X)$	LSSA	$7.1484E-17$	$1.0144E-09$	$3.9939E-09$
	BSO	0	0.4704	1.7733
	CSSA	$8.9576E-13$	$1.1160E-08$	$2.0300E-08$
	ISSA	$2.8781E-12$	$1.1195E-07$	$2.9258E-07$
	SSA	$2.6178E-13$	$2.3804E-07$	$1.0194E-06$
	GWO	$3.5322E-06$	0.1966	0.4975
	PSO	$6.6552E-07$	$8.9930E-04$	$8.7012E-03$
$F_{12}(X)$	LSSA	0.998	0.998	0
	BSO	0.998	1.8886	1.4519
	CSSA	0.998	1.0641	0.3622
	ISSA	0.998	2.3662	3.3602
	SSA	0.998	2.4984	3.3421
	GWO	0.998	1.7255	0.9724
	PSO	0.998	1.1305	0.3436

TABLE 3: Comparison table of optimization effect of each algorithm (100 dimensions).

Function	Algorithm	Best	Ave	Std
$F_1(x)$	LSSA	0	0	0
	BSO	195.6467	757.4745	391.4064
	CSSA	0	0	0
	ISSA	0	0	0
	SSA	0	0	0
	GWO	$5.0820E-18$	$2.6308E-17$	$2.1089E-17$
	PSO	1.0668	12.4586	30.0289
$F_2(X)$	LSSA	0	0	0
	BSO	0.0680	154.2623	315.1105
	CSSA	0	$1.9637E-206$	0
	ISSA	0	0	0
	SSA	0	$1.0517E-212$	0
	GWO	0.4399	11.8139	13.5850
	PSO	6831.9888	4185.5172	7632.0184
$F_3(X)$	LSSA	0	0	0
	BSO	0.04267	1.7935	1.3896
	CSSA	0	$1.1926E-184$	0
	ISSA	0	$1.3563E-144$	$7.4285E-144$
	SSA	0	$1.9421E-165$	0
	GWO	$2.0983E-03$	0.05209	0.0865
	PSO	7.3782	9.5753	1.2667

TABLE 3: Continued.

Function	Algorithm	Best	Ave	Std
$F_4(X)$	LSSA	$2.9056E-09$	$2.6894E-05$	$2.4742E-05$
	BSO	1077.0368	$1.3748E+04$	$1.3239E+04$
	CSSA	$4.2237E-08$	$2.5499E-04$	$3.4620E-04$
	ISSA	$2.1761E-07$	$2.6971E-04$	$6.66094E-04$
	SSA	$9.1735E-09$	$7.9045E-05$	$1.3805E-04$
	GWO	95.6787	96.9734	0.9208
	PSO	651.0884	3221.6846	6748.9024
$F_5(X)$	LSSA	$2.5013E-11$	$9.1120E-08$	$1.8203E-07$
	BSO	244.9986	777.9455	539.4453
	CSSA	$4.0102E-10$	$2.6523E-07$	$3.6092E-07$
	ISSA	$4.5456E-11$	$1.9716E-07$	$3.2439E-07$
	SSA	$2.2699E-10$	$9.0256E-08$	$1.3690E-07$
	GWO	5.3138	6.7272	0.6593
	PSO	1.0675	4.4194	5.3378
$F_6(X)$	LSSA	$2.4150E-06$	$1.2529E-04$	$8.5801E-05$
	BSO	0.02903	0.1136	0.06599
	CSSA	$3.137E-05$	$3.1911E-04$	$3.8926E-04$
	ISSA	$1.1119E-05$	$2.3439E-04$	$1.9717E-04$
	SSA	$2.4052E-06$	$2.3133E-04$	$1.8927E-04$
	GWO	$9.4823E-04$	$2.6521E-03$	$1.1372E-03$
	PSO	0.4998	1.1219	0.3632
$F_7(X)$	LSSA	-41082.3952	-36443.6428	4817.9562
	BSO	-40261.0069	-33461.9674	3925.7756
	ISSA	-26395.4326	-24633.4955	981.1676
	CSSA	-26934.0704	-24515.6636	1327.8950
	SSA	-27320.1874	-24637.4984	1078.1888
	GWO	-22404.2115	-16606.7711	2437.1531
	PSO	-23435.5935	-20056.7171	2277.0807
$F_8(X)$	LSSA	$1.1891E-15$	$1.9561E-12$	$3.1614E-12$
	BSO	0.1178	0.9327	0.6212
	CSSA	$2.6323E-15$	$1.1686E-11$	$2.9995E-11$
	ISSA	$2.4668E-10$	$2.1636E-07$	$1.1233E-06$
	SSA	$2.1460E-15$	$7.6085E-12$	$2.1375E-11$
	GWO	$2.1287E-06$	0.01593	$9.1126E-03$
	PSO	$2.3296E-13$	0.08298	0.1774
$F_9(X)$	LSSA	0.4063	4843.1939	4063.8163
	BSO	1803.1299	7726.7640	4238.1062
	CSSA	$1.4004E+04$	$1.7119E+04$	1070.6257
	ISSA	2191.2870	3460.1076	584.0178
	SSA	$1.4549E+04$	$1.7425+04$	1198.2007
	GWO	$2.0063E+04$	$2.4918E+4$	2683.4939
	PSO	$1.6965E+04$	$2.2306E+04$	4297.4782

above CEC 2017 test. Each algorithm works in a 15×15 model, and the optimal route is shown in Figure 7. In order to eliminate the chance, each algorithm is tested 10 times, and the optimal route, average route, and worst route of each algorithm are counted. Three indicators are used to measure the stability and feasibility of each algorithm in this experiment. The optimization statistics of each algorithm are shown in Table 8.

It can be seen from Table 8 and Figure 7 that the minimum cost of LSSA planning is 19.7990, while the minimum cost of SSA is 25.4558. It can be seen that the route planning ability of LSSA is strong, and through the average value and the worst value, it can be seen that the route planned by LSSA has good stability. Therefore, LSSA has a

good effect on robot path planning and can plan a more stable and safe route.

7. Conclusions

A learning sparrow search algorithm is proposed in this paper, overcoming the shortcomings of the SSA. The lens reverse learning and random reverse learning are introduced in the discoverer's position and the worst position, respectively, which make the discoverer's search method more flexible. Then, the improved sine-cosine guidance makes the follower search more detailed. Finally, the local search based on the difference is used to update the optimal solution, which improves the quality of each solution.

(a)

(b)

(c)

(d)

(e)

(f)
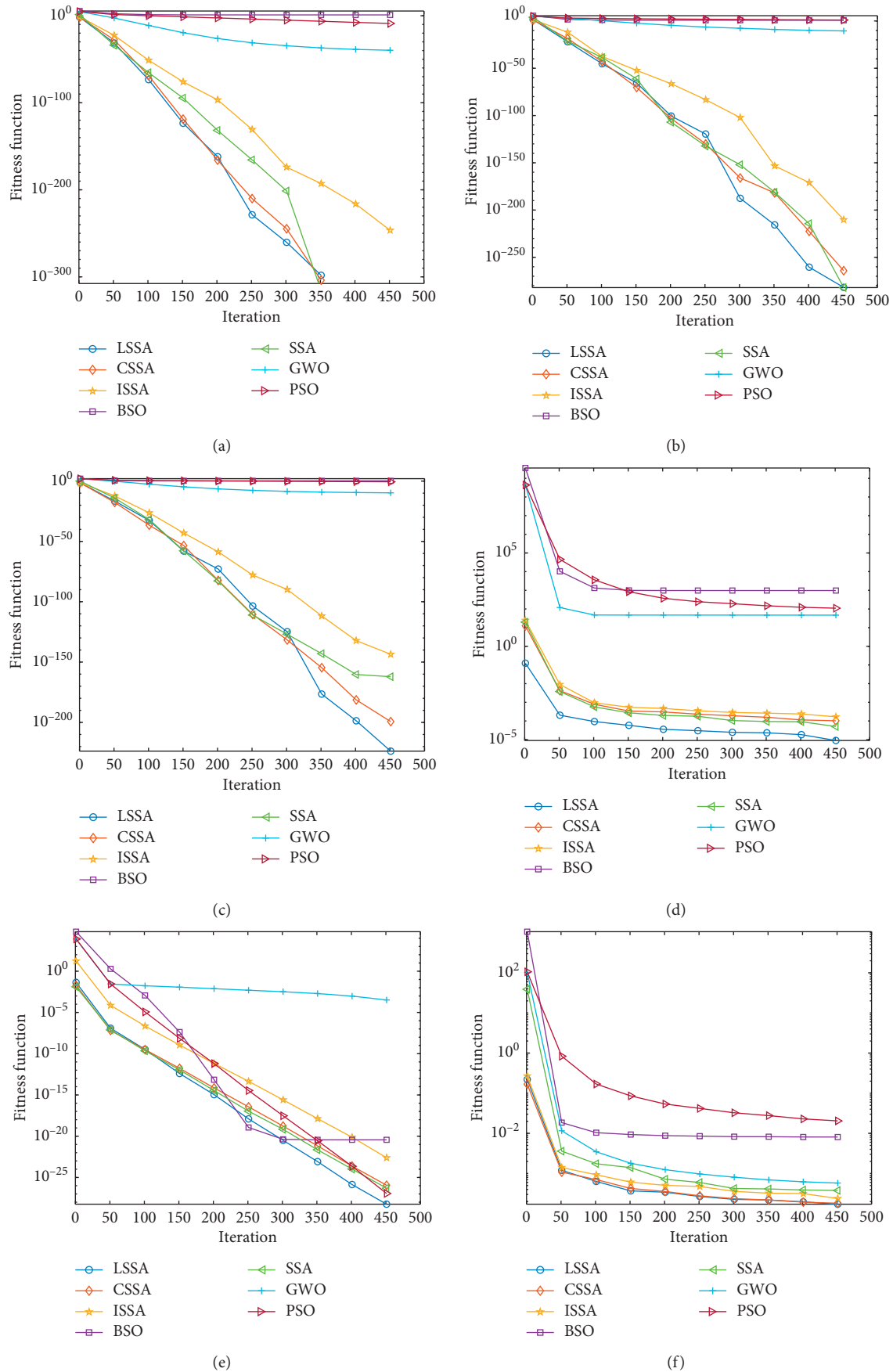
FIGURE 6: Continued.
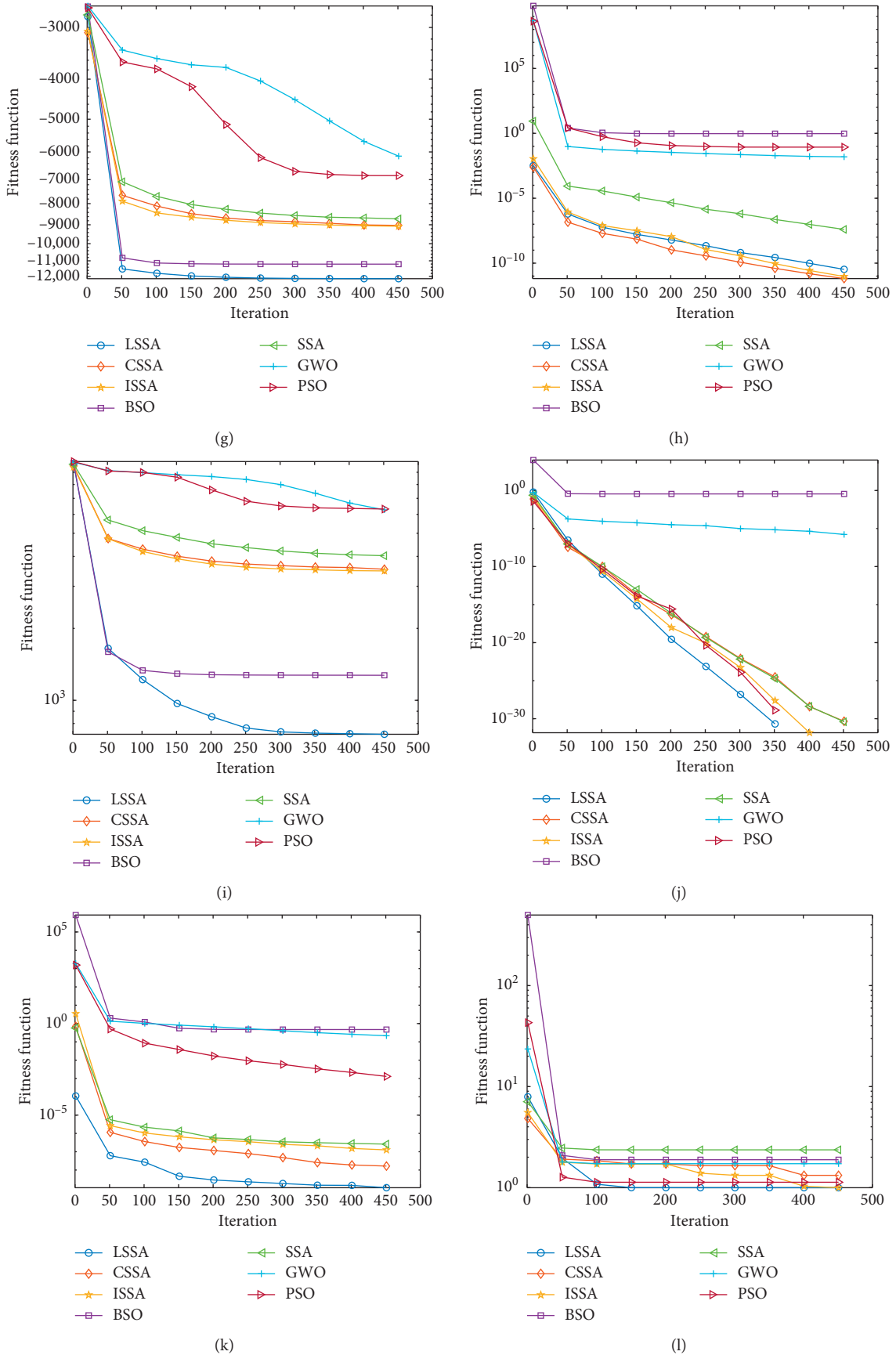
FIGURE 6: Convergence diagram of each algorithm. (a) F_1. (b) F_2. (c) F_3. (d) F_4. (e) F_5. (f) F_6. (g) F_7. (h) F_8. (i) F_9. (j) F_{10}. (k) F_{11}. (l) F_{12}.

Table 4: Complexity of the LSSA algorithm.

Dim	T_0	T_1	T_2	$|T_2 - T_1|/T_0$
30	0.0838	4.9412	3.9684	11.6085
50		9.1498	5.3006	54.1031

Table 5: Parameters of each algorithm.

Algorithm	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
Parameter	Discoverers $= 0.2 \times N$ Investigators $= 0.6 \times N$	Discoverers $= 0.2 \times N$ Investigators $= 0.6 \times N$	$\alpha = 0.01$ $\beta = 1.5$ $P_a = 0.25$ $C = 0.2$ $PA_{max} = 0.35$ $PA_{min} = 0.25$	$P_{max} = 0.75$ $P_{max} = 0.25$ Freq $= 0.5$	$\alpha = 0.01$ $\beta_{min} = 0.2$ $\beta = 1$ $\gamma = 1$	Discoverers $= 0.2 \times N$ Investigators $= 0.6 \times N$

Table 6: Test results of each algorithm in CEC 2017(dim = 30).

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_1(x)$	Best	**1.01E + 02**	1.03E + 02	1.00E + 10	1.00E + 10	8.21E + 05	1.10E + 02
	Worst	2.03E + 04	1.98E + 04	1.00E + 10	1.00E + 10	1.85E + 06	**1.86E + 04**
	Median	2.12E + 03	4.55E + 03	1.00E + 10	1.00E + 10	1.20E + 06	**1.84E + 03**
	Mean	4.80E + 03	6.27E + 03	1.00E + 10	1.00E + 10	1.28E + 06	**4.47E + 03**
	Std	5.74E + 03	5.94E + 03	0.00E + 00	0.00E + 00	2.96E + 05	**5.61E + 03**
	P	0.8073(=)	0.0484(+)	1.21E − 12(+)	1.21E − 12(+)	3.02E − 11(+)	
$F_3(x)$	Best	3.52E + 02	**3.00E + 02**	3.08E + 02	3.14E + 02	8.98E + 02	**3.00E + 02**
	Worst	2.48E + 03	**3.00E + 02**	1.07E + 03	8.75E + 02	5.10E + 03	**3.00E + 02**
	Median	7.72E + 02	**3.00E + 02**	4.21E + 02	4.15E + 02	1.91E + 03	**3.00E + 02**
	Mean	9.05E + 02	**3.00E + 02**	4.82E + 02	4.55E + 02	2.25E + 03	**3.00E + 02**
	Std	2.93E + 01	1.77E − 03	1.64E + 01	1.27E + 02	2.17E + 01	**8.17 E − 04**
	P	3.02E − 11(+)	0.0292(+)	3.02E − 11(+)	3.02E − 11(+)	3.02E − 11(+)	
$F_4(X)$	Best	**4.00E + 02**	4.28E + 02	**4.00E + 02**	**4.00E + 02**	4.68E + 02	**4.00E + 02**
	Worst	5.21E + 02	5.17E + 02	4.89E + 02	4.67E + 02	5.32E + 02	**4.64E + 02**
	Median	4.74E + 02	4.90E + 02	4.84E + 02	4.04E + 02	5.15E + 02	**4.01E + 02**
	Mean	4.72E + 02	4.94E + 02	4.65E + 02	4.18E + 02	5.06E + 02	**4.09E + 02**
	Std	2.93E + 01	2.12E + 01	3.30E + 01	2.41E + 01	2.17E + 01	**1.64E + 01**
	P	1.41E − 09(+)	4.08E − 11(+)	1.10E − 08(+)	0.1241(=)	3.02E − 11(+)	
$F_5(X)$	Best	5.69E + 02	6.14E + 02	5.36E + 02	**5.26E + 02**	6.36E + 02	5.54E + 02
	Worst	8.27E + 02	8.01E + 02	5.73E + 02	**5.69E + 02**	7.33E + 02	5.96E + 02
	Median	7.56E + 02	7.03E + 02	5.58E + 02	**5.53E + 02**	6.91E + 02	5.79E + 02
	Mean	7.53E + 02	7.01E + 02	5.57E + 02	**5.52E + 02**	6.84E + 02	5.77E + 02
	Std	5.71E + 01	4.74E + 01	**9.13E + 00**	9.40E + 00	2.59E + 01	1.08E + 01
	P	2.87E − 10(+)	3.02E − 11(+)	4.31E − 08(−)	1.17E − 09(−)	3.02E − 11(+)	
$F_6(X)$	Best	6.13E + 02	6.09E + 02	**6.00E + 02**	**6.00E + 02**	6.28E + 02	**6.00E + 02**
	Worst	6.59E + 02	6.37E + 02	**6.00E + 02**	**6.00E + 02**	6.57E + 02	**6.00E + 02**
	Median	6.40E + 02	6.21E + 02	**6.00E + 02**	**6.00E + 02**	6.46E + 02	**6.00E + 02**
	Mean	6.40E + 02	6.21E + 02	**6.00E + 02**	**6.00E + 02**	6.45E + 02	**6.00E + 02**
	Std	1.12E + 02	6.40E + 00	1.08E + 01	6.79E − 03	9.34E + 01	**1.06E − 03**
	P	3.02E − 11(+)	3.02E − 11(+)	4.34E − 05(+)	0.002156(+)	3.02E − 11(+)	
$F_7(X)$	Best	1.02E + 03	8.66E + 02	7.82E + 02	7.69E + 02	8.86E + 02	**7.60E + 02**
	Worst	1.37E + 03	1.26E + 03	8.27E + 02	8.07E + 02	1.26E + 03	**8.04E + 02**
	Median	1.27E + 03	9.66E + 02	8.07E + 02	7.92E + 02	1.07E + 03	**7.87E + 02**
	Mean	1.23E + 03	1.00E + 03	8.08E + 02	7.90E + 02	1.07E + 03	**7.86E + 02**
	Std	1.12E + 02	1.08E + 02	1.03E + 01	**8.80E + 00**	9.34E + 01	1.08E + 01
	P	3.02E − 11(+)	3.02E − 11(+)	8.48E − 09(+)	0.20620(=)	3.02E − 11(+)	

TABLE 6: Continued.

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_8(X)$	Best	$9.22E+02$	$8.64E+02$	$8.42E+02$	$8.40E+02$	$8.74E+02$	$\mathbf{8.39E+02}$
	Worst	$1.03E+03$	$1.01E+03$	$9.03E+02$	$8.75E+02$	$9.80E+02$	$\mathbf{8.77E+02}$
	Median	$9.79E+02$	$9.45E+02$	$8.81E+02$	$8.61E+02$	$9.26E+02$	$\mathbf{8.60E+02}$
	Mean	$9.75E+02$	$9.42E+02$	$8.78E+02$	$8.59E+02$	$9.23E+02$	$\mathbf{8.58E+02}$
	Std	$2.41E+01$	$2.93E+01$	$1.51E+01$	$1.00E+01$	$2.42E+01$	$9.56E+00$
	P	$3.02E-11(+)$	$7.39E-11(+)$	$2.32E-06(+)$	$0.6414(=)$	$3.34E-11(+)$	
$F_9(X)$	Best	$4.50E+03$	$1.95E+03$	$\mathbf{9.00E+02}$	$\mathbf{9.00E+02}$	$2.27E+03$	$\mathbf{9.00E+02}$
	Worst	$5.47E+03$	$5.86E+03$	$9.49E+02$	$\mathbf{9.26E+02}$	$6.12E+03$	$9.93E+02$
	Median	$5.39E+03$	$5.15E+03$	$9.05E+02$	$9.05E+02$	$4.53E+03$	$\mathbf{9.04E+02}$
	Mean	$5.30E+03$	$4.74E+03$	$\mathbf{9.07E+02}$	$\mathbf{9.07E+02}$	$4.31E+03$	$9.11E+02$
	Std	$5.53E+02$	$9.76E+02$	$3.25E+02$	$\mathbf{7.14E+00}$	$6.15E+02$	$1.79E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$0.66272(=)$	$0.9823(=)$	$3.02E-11(+)$	
$F_{10}(X)$	Best	$4.01E+03$	$4.06E+03$	$3.24E+03$	$\mathbf{2.31E+03}$	$3.61E+03$	$2.76E+03$
	Worst	$6.07E+03$	$5.93E+03$	$4.81E+03$	$3.95E+03$	$6.19E+03$	$\mathbf{3.92E+03}$
	Median	$5.07E+03$	$4.71E+03$	$4.10E+03$	$3.38E+03$	$5.19E+03$	$\mathbf{3.36E+03}$
	Mean	$5.16E+03$	$4.83E+03$	$4.03E+03$	$\mathbf{3.35E+03}$	$5.03E+03$	$3.36E+03$
	Std	$5.53E+02$	$5.42E+02$	$\mathbf{3.25E+02}$	$3.97E+02$	$6.15E+02$	$\mathbf{3.25E+02}$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$1.01E-08(+)$	$0.91170(=)$	$8.99E-11(+)$	
$F_{11}(X)$	Best	$1.18E+03$	$1.14E+03$	$\mathbf{1.10E+03}$	$1.11E+03$	$1.16E+03$	$1.11E+03$
	Worst	$1.42E+03$	$1.48E+03$	$\mathbf{1.14E+03}$	$\mathbf{1.14E+03}$	$1.32E+03$	$1.15E+03$
	Median	$1.29E+03$	$1.30E+03$	$\mathbf{1.12E+03}$	$\mathbf{1.12E+03}$	$1.22E+03$	$\mathbf{1.12E+03}$
	Mean	$1.30E+03$	$1.30E+03$	$\mathbf{1.12E+03}$	$\mathbf{1.12E+03}$	$1.23E+03$	$\mathbf{1.12E+03}$
	Std	$6.50E+01$	$7.29E+01$	$8.27E+00$	$7.78E+00$	$4.49E+01$	$\mathbf{7.72E+00}$
	P	$3.02E-11(+)$	$3.34E-11(+)$	$0.2707(=)$	$0.2588(=)$	$3.02E-11(+)$	
$F_{12}(X)$	Best	$4.55E+04$	$1.32E+04$	$1.75E+04$	$\mathbf{8.77E+03}$	$1.29E+06$	$2.07E+04$
	Worst	$3.89E+05$	$1.35E+05$	$1.00E+10$	$1.00E+10$	$9.36E+06$	$\mathbf{1.05E+05}$
	Median	$1.42E+05$	$4.27E+04$	$1.00E+10$	$1.00E+10$	$3.59E+06$	$\mathbf{5.48E+04}$
	Mean	$1.64E+05$	$4.83E+04$	$5.67E+09$	$\mathbf{5.34E+09}$	$4.18E+06$	$5.37E+04$
	Std	$\mathbf{1.04E+04}$	$2.58E+04$	$6.46E+09$	$5.07E+09$	$3.20E+04$	$2.18E+04$
	P	$6.01E-08(+)$	$1.83E-05(+)$	$0.000386(+)$	$0.01727(+)$	$3.02E-11(+)$	
$F_{13}(X)$	Best	$1.45E+03$	$2.79E+03$	$1.67E+03$	$\mathbf{1.35E+03}$	$5.41E+04$	$\mathbf{1.35E+03}$
	Worst	$5.02E+04$	$6.67E+04$	$7.40E+03$	$1.00E+10$	$2.11E+05$	$\mathbf{1.57E+03}$
	Median	$6.67E+03$	$8.49E+03$	$2.52E+03$	$1.42E+03$	$1.19E+05$	$\mathbf{1.41E+03}$
	Mean	$1.05E+04$	$1.44E+04$	$2.89E+03$	$6.67E+08$	$1.13E+05$	$\mathbf{1.43E+03}$
	Std	$1.04E+04$	$1.84E+04$	$1.18E+03$	$2.54E+09$	$3.20E+04$	$\mathbf{6.46E+01}$
	P	$1.46E-10(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$0.1857(=)$	$3.02E-11(+)$	
$F_{14}(X)$	Best	$2.70E+03$	$1.59E+03$	$1.45E+03$	$\mathbf{1.43E+03}$	$3.41E+03$	$\mathbf{1.43E+03}$
	Worst	$4.36E+04$	$1.74E+04$	$1.49E+03$	$\mathbf{1.46E+03}$	$5.26E+04$	$\mathbf{1.46E+03}$
	Median	$9.44E+03$	$4.69E+03$	$1.47E+03$	$\mathbf{1.44E+03}$	$2.86E+04$	$\mathbf{1.44E+03}$
	Mean	$1.12E+04$	$5.25E+03$	$1.47E+03$	$\mathbf{1.44E+03}$	$2.54E+04$	$\mathbf{1.44E+03}$
	Std	$8.21E+03$	$3.19E+03$	$8.17E+00$	$\mathbf{8.03E+00}$	$1.32E+04$	$8.04E+00$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$6.70E-11(+)$	$0.1580(=)$	$3.02E-11(+)$	
$F_{15}(X)$	Best	$2.01E+03$	$1.79E+03$	$1.57E+03$	$\mathbf{1.51E+03}$	$1.94E+04$	$\mathbf{1.51E+03}$
	Worst	$4.62E+04$	$4.34E+04$	$1.62E+03$	$\mathbf{1.53E+03}$	$6.21E+04$	$1.54E+03$
	Median	$1.01E+04$	$4.19E+03$	$1.59E+03$	$\mathbf{1.52E+03}$	$4.36E+04$	$\mathbf{1.52E+03}$
	Mean	$1.62E+04$	$9.43E+03$	$1.59E+03$	$\mathbf{1.52E+03}$	$4.08E+04$	$\mathbf{1.52E+03}$
	Std	$3.42E+02$	$1.06E+04$	$1.26E+01$	$\mathbf{5.28E+00}$	$3.21E+02$	$2.08E+02$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$0.1536(=)$	$3.02E-11(+)$	
$F_{16}(X)$	Best	$2.25E+03$	$2.20E+03$	$\mathbf{1.71E+03}$	$1.76E+03$	$2.49E+03$	$1.82E+03$
	Worst	$3.62E+03$	$3.39E+03$	$2.45E+03$	$\mathbf{2.42E+03}$	$3.68E+03$	$\mathbf{2.35E+03}$
	Median	$2.96E+03$	$2.77E+03$	$\mathbf{2.02E+03}$	$2.05E+03$	$3.07E+03$	$2.03E+03$
	Mean	$2.93E+03$	$2.74E+03$	$\mathbf{2.05E+03}$	$2.07E+03$	$3.09E+03$	$2.07E+03$
	Std	$3.42E+02$	$3.21E+02$	$2.08E+02$	$1.69E+02$	$3.21E+02$	$\mathbf{1.53E+02}$
	P	$6.70E-11(+)$	$2.61E-10(+)$	$0.5011(=)$	$0.6734(=)$	$3.02E-11(+)$	

TABLE 6: Continued.

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_{17}(X)$	Best	$1.98E+03$	$1.79E+03$	$1.76E+03$	$\mathbf{1.73E+03}$	$1.79E+03$	$\mathbf{1.73E+03}$
	Worst	$2.88E+03$	$2.71E+03$	$\mathbf{1.89E+03}$	$1.99E+03$	$2.54E+03$	$1.96E+03$
	Median	$2.47E+03$	$2.23E+03$	$1.79E+03$	$1.80E+03$	$2.08E+03$	$\mathbf{1.77E+03}$
	Mean	$2.47E+03$	$2.26E+03$	$1.80E+03$	$1.82E+03$	$2.13E+03$	$\mathbf{1.79E+03}$
	Std	$2.33E+02$	$2.21E+02$	$2.87E+01$	$6.63E+01$	$2.40E+02$	$\mathbf{5.62E+01}$
	P	$3.02E-11(+)$	$1.78E-10(+)$	$0.005828(+)$	$0.07978(=)$	$9.76E-10(+)$	
$F_{18}(X)$	Best	$2.71E+04$	$8.67E+03$	$1.15E+04$	$\mathbf{1.87E+03}$	$5.21E+04$	$1.89E+03$
	Worst	$5.02E+05$	$2.96E+05$	$7.70E+04$	$2.81E+03$	$6.73E+05$	$\mathbf{2.51E+03}$
	Median	$1.15E+05$	$1.07E+05$	$3.44E+04$	$1.98E+03$	$2.36E+05$	$\mathbf{1.98E+03}$
	Mean	$1.56E+05$	$1.12E+05$	$3.59E+04$	$2.07E+03$	$2.47E+05$	$\mathbf{2.01E+03}$
	Std	$1.60E+04$	$7.56E+04$	$1.24E+04$	$2.33E+02$	$5.79E+05$	$\mathbf{2.63E+00}$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$0.7844(=)$	$3.02E-11(+)$	
$F_{19}(X)$	Best	$2.06E+03$	$2.27E+03$	$1.94E+03$	$\mathbf{1.91E+03}$	$2.16E+05$	$\mathbf{1.91E+03}$
	Worst	$5.35E+04$	$5.54E+04$	$1.97E+03$	$\mathbf{1.92E+03}$	$2.39E+06$	$\mathbf{1.92E+03}$
	Median	$6.25E+03$	$7.81E+03$	$1.95E+03$	$\mathbf{1.91E+03}$	$1.39E+06$	$\mathbf{1.91E+03}$
	Mean	$1.37E+04$	$1.49E+04$	$1.95E+03$	$\mathbf{1.91E+03}$	$1.32E+06$	$\mathbf{1.91E+03}$
	Std	$1.60E+04$	$1.56E+04$	$7.42E+00$	$\mathbf{2.22E+00}$	$5.79E+05$	$2.63E+00$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$0.7505(=)$	$3.02E-11(+)$	
$F_{20}(X)$	Best	$2.26E+03$	$2.25E+03$	$\mathbf{2.02E+03}$	$\mathbf{2.02E+03}$	$2.26E+03$	$2.04E+03$
	Worst	$3.05E+03$	$2.82E+03$	$2.34E+03$	$\mathbf{2.29E+03}$	$2.69E+03$	$2.31E+03$
	Median	$2.58E+03$	$2.40E+03$	$2.16E+03$	$2.15E+03$	$2.31E+03$	$\mathbf{2.08E+03}$
	Mean	$2.62E+03$	$2.44E+03$	$2.14E+03$	$2.13E+03$	$2.37E+03$	$\mathbf{2.12E+03}$
	Std	$2.11E+02$	$1.49E+02$	$8.20E+01$	$8.12E+01$	$1.20E+02$	$\mathbf{6.43E+01}$
	P	$4.50E-11(+)$	$6.07E-11(+)$	$5.01E-01(=)$	$0.9351(=)$	$1.33E-10(+)$	
$F_{21}(X)$	Best	$2.42E+03$	$2.39E+03$	$\mathbf{2.20E+03}$	$\mathbf{2.20E+03}$	$2.37E+03$	$\mathbf{2.20E+03}$
	Worst	$2.59E+03$	$2.57E+03$	$2.40E+03$	$\mathbf{2.38E+03}$	$2.60E+03$	$2.39E+03$
	Median	$2.50E+03$	$2.47E+03$	$2.38E+03$	$2.36E+03$	$2.45E+03$	$\mathbf{2.35E+03}$
	Mean	$2.50E+03$	$2.47E+03$	$2.36E+03$	$2.35E+03$	$2.45E+03$	$\mathbf{2.34E+03}$
	Std	$1.95E+03$	$4.42E+01$	$6.33E+01$	$\mathbf{4.11E+01}$	$9.25E-01$	$1.19E+03$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$2.13E-05(+)$	$0.5691(=)$	$3.69E-11(+)$	
$F_{22}(X)$	Best	$\mathbf{2.30E+03}$	$\mathbf{2.30E+03}$	$\mathbf{2.30E+03}$	$\mathbf{2.30E+03}$	$2.31E+03$	$\mathbf{2.30E+03}$
	Worst	$7.73E+03$	$8.05E+03$	$5.48E+03$	$5.34E+03$	$\mathbf{2.31E+03}$	$5.83E+03$
	Median	$6.52E+03$	$5.69E+03$	$2.30E+03$	$2.30E+03$	$\mathbf{2.31E+03}$	$\mathbf{2.30E+03}$
	Mean	$5.96E+03$	$4.81E+03$	$3.00E+03$	$3.10E+03$	$\mathbf{2.31E+03}$	$2.53E+03$
	Std	$1.95E+03$	$2.19E+03$	$1.19E+03$	$1.19E+03$	$\mathbf{9.25\ E-01}$	$8.77E+02$
	P	$1.49E-06(+)$	$3.91E-02(+)$	$6.55E-04(+)$	$8.89E-02(+)$	$8.48E-09(-)$	
$F_{23}(X)$	Best	$2.77E+03$	$2.71E+03$	$2.66E+03$	$2.67E+03$	$2.82E+03$	$\mathbf{2.40E+03}$
	Worst	$3.00E+03$	$3.00E+03$	$\mathbf{2.73E+03}$	$\mathbf{2.73E+03}$	$3.01E+03$	$2.76E+03$
	Median	$2.91E+03$	$2.83E+03$	$\mathbf{2.71E+03}$	$\mathbf{2.71E+03}$	$2.93E+03$	$2.72E+03$
	Mean	$2.90E+03$	$2.84E+03$	$2.71E+03$	$2.71E+03$	$2.92E+03$	$\mathbf{2.70E+03}$
	Std	$5.30E+01$	$6.12E+01$	$\mathbf{1.40E+01}$	$1.49E+01$	$5.79E+01$	$6.38E+01$
	P	$3.02E-11(+)$	$1.96E-10(+)$	$9.93E-02(=)$	$1.22E-01(=)$	$3.02E-11(+)$	
$F_{24}(X)$	Best	$2.95E+03$	$2.92E+03$	$2.86E+03$	$2.86E+03$	$2.94E+03$	$\mathbf{2.60E+03}$
	Worst	$3.33E+03$	$3.15E+03$	$2.94E+03$	$2.91E+03$	$3.23E+03$	$\mathbf{2.91E+03}$
	Median	$3.08E+03$	$2.98E+03$	$2.90E+03$	$2.89E+03$	$3.07E+03$	$\mathbf{2.88E+03}$
	Mean	$3.10E+03$	$3.00E+03$	$2.90E+03$	$2.89E+03$	$3.07E+03$	$\mathbf{2.87E+03}$
	Std	$1.52E+01$	$5.59E+01$	$\mathbf{1.75E+00}$	$1.30E+01$	$2.19E+01$	$7.35E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$6.38E-03(+)$	$9.63E-02(=)$	$3.02E-11(+)$	
$F_{25}(X)$	Best	$\mathbf{2.88E+03}$	$\mathbf{2.88E+03}$	$\mathbf{2.88E+03}$	$\mathbf{2.88E+03}$	$2.89E+03$	$\mathbf{2.88E+03}$
	Worst	$2.94E+03$	$2.94E+03$	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$2.96E+03$	$\mathbf{2.89E+03}$
	Median	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$2.94E+03$	$\mathbf{2.89E+03}$
	Mean	$2.90E+03$	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$\mathbf{2.89E+03}$	$2.93E+03$	$\mathbf{2.89E+03}$
	Std	$1.52E+01$	$1.32E+01$	$1.75E+00$	$1.61E+00$	$2.19E+01$	$\mathbf{6.12E-01}$
	P	$6.74E-06(+)$	$6.38E-03(+)$	$8.77E-02(=)$	$8.77E-01(=)$	$5.07E-10(+)$	

TABLE 6: Continued.

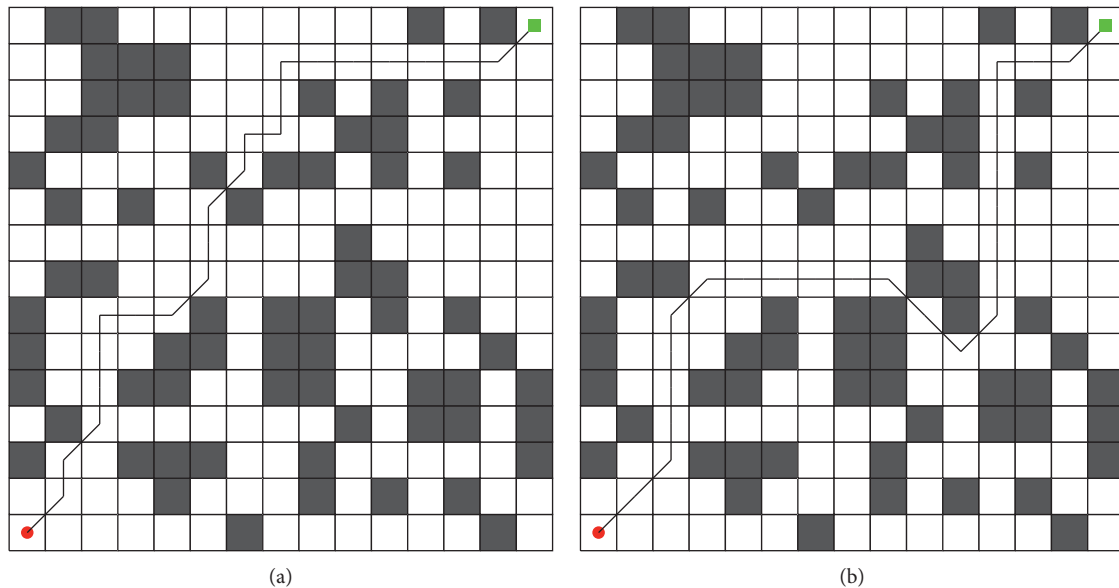| F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_{26}(X)$	Best	$3.67E+03$	$2.90E+03$	**$2.80E+03$**	**$2.80E+03$**	$2.83E+03$	**$2.80E+03$**
	Worst	$7.70E+03$	$7.46E+03$	$4.58E+03$	$4.54E+03$	$7.86E+03$	**$4.28E+03$**
	Median	$6.38E+03$	$5.99E+03$	**$2.90E+03$**	$3.46E+03$	$4.81E+03$	**$2.90E+03$**
	Mean	$6.33E+03$	$5.64E+03$	$3.33E+03$	$3.60E+03$	$4.77E+03$	**$2.94E+03$**
	Std	$8.67E+02$	$1.37E+03$	$6.98E+02$	$7.26E+02$	$1.87E+03$	**$2.55E+02$**
	P	$3.34E-11(+)$	$3.32E-06(+)$	$5.28E-02(=)$	$6.19E-01(=)$	$6.57E-02(=)$	
$F_{27}(X)$	Best	$3.22E+03$	$3.20E+03$	$3.18E+03$	**$3.17E+03$**	$3.36E+03$	**$3.17E+03$**
	Worst	$3.39E+03$	$3.32E+03$	$3.21E+03$	$3.22E+03$	$3.67E+03$	**$3.21E+03$**
	Median	$3.27E+03$	$3.24E+03$	$3.20E+03$	$3.20E+03$	$3.47E+03$	**$3.20E+03$**
	Mean	$3.27E+03$	$3.25E+03$	$3.20E+03$	$3.20E+03$	$3.48E+03$	**$3.20E+03$**
	Std	$4.95E+01$	$2.64E+01$	$3.67E+01$	$1.08E+01$	$9.39E+00$	**$9.01E+00$**
	P	$3.02E-11(+)$	$1.21E-10(+)$	$9.05E-02(=)$	$5.94E-02(=)$	$3.02E-11(+)$	
$F_{28}(X)$	Best	**$3.10E+03$**	**$3.10E+03$**	**$3.10E+03$**	**$3.10E+03$**	$3.18E+03$	**$3.10E+03$**
	Worst	$3.25E+03$	$3.26E+03$	$3.21E+03$	**$3.20E+03$**	$3.21E+03$	$3.26E+03$
	Median	$3.11E+03$	**$3.10E+03$**	**$3.10E+03$**	**$3.10E+03$**	$3.20E+03$	**$3.10E+03$**
	Mean	$3.14E+03$	$3.14E+03$	$3.11E+03$	**$3.12E+03$**	$3.20E+03$	$3.14E+03$
	Std	$4.95E+01$	$6.30E+01$	$3.67E+01$	$3.92E+01$	**$9.39E+00$**	$5.94E+01$
	P	$1.33E-02(+)$	$8.07E-01(=)$	$5.27E-06(-)$	$6.29E-05(-)$	$1.00E-03(+)$	
$F_{29}(X)$	Best	$3.80E+03$	$3.60E+03$	$3.32E+03$	$3.29E+03$	$3.88E+03$	**$3.28E+03$**
	Worst	$4.69E+03$	$4.43E+03$	$3.63E+03$	**$3.56E+03$**	$4.84E+03$	$3.63E+03$
	Median	$4.09E+03$	$3.98E+03$	$3.47E+03$	$3.44E+03$	$4.34E+03$	**$3.41E+03$**
	Mean	$4.16E+03$	$4.00E+03$	$3.46E+03$	$3.44E+03$	$4.33E+03$	**$3.43E+03$**
	Std	$2.40E+02$	$2.05E+02$	$8.23E+01$	**$7.44E+01$**	$2.72E+02$	$7.74E+01$
	P	$3.02E-11(+)$	$3.69E-11(+)$	$1.05E-01(=)$	$5.49E-01(=)$	$3.02E-11(+)$	
$F_{30}(X)$	Best	$5.80E+03$	$5.30E+03$	$7.68E+03$	**$5.27E+03$**	$1.15E+06$	$5.33E+03$
	Worst	$2.42E+04$	$2.13E+04$	$1.17E+04$	**$6.27E+03$**	$4.46E+06$	$6.93E+03$
	Median	$9.94E+03$	$7.31E+03$	$9.56E+03$	$5.74E+03$	$2.21E+06$	**$5.67E+03$**
	Mean	$1.10E+04$	$9.26E+03$	$9.70E+03$	**$5.74E+03$**	$2.52E+06$	$5.80E+03$
	Std	$4.95E+01$	$4.15E+03$	$1.08E+03$	$2.81E+02$	$8.37E+05$	**$3.67E+01$**
	P	$2.37E-10(+)$	$2.88E-06(+)$	$3.02E-11(+)$	$0.9117(=)$	$3.02E-11(+)$	
$+/=/-$		$28/1/0$	$28/1/0$	$18/9/2$	$5/22/2$	$27/1/1$	

(a)　　　　　　　　　　　　　　　　　　　　　　　　　(b)

FIGURE 7: Minimum path planning. (a) LSSA. (b) SSA.

Table 7: Test results of each algorithm in CEC 2017(dim = 50).

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_1(x)$	Best	$1.06E + 02$	$1.02E + 02$	$1.00E + 10$	$1.00E + 10$	$3.48E + 06$	$\mathbf{1.01E + 02}$
	Worst	$2.08E + 04$	$2.29E + 04$	$1.00E + 10$	$1.00E + 10$	$6.99E + 06$	$\mathbf{9.93E + 03}$
	Median	$2.31E + 03$	$1.15E + 03$	$1.00E + 10$	$1.00E + 10$	$5.22E + 06$	$\mathbf{1.19E + 03}$
	Mean	$3.05E + 03$	$2.98E + 03$	$1.00E + 10$	$1.00E + 10$	$5.16E + 06$	$\mathbf{2.39E + 03}$
	Std	$4.04E + 03$	$4.48E + 03$	$\mathbf{0.00E + 00}$	$\mathbf{0.00E + 00}$	$7.96E + 05$	$2.74E + 03$
	P	$0.3870(=)$	$0.7171(=)$	$1.21E - 12(+)$	$1.21E - 12(+)$	$3.02E - 11(+)$	
$F_3(x)$	Best	$1.18E + 03$	$3.01E + 02$	$1.47E + 04$	$1.25E + 04$	$2.25E + 04$	$\mathbf{3.00E + 02}$
	Worst	$1.02E + 04$	$3.09E + 02$	$4.03E + 04$	$3.89E + 04$	$7.91E + 04$	$\mathbf{3.01E + 02}$
	Median	$5.66E + 03$	$3.03E + 02$	$2.51E + 04$	$1.99E + 04$	$4.52E + 04$	$\mathbf{3.00E + 02}$
	Mean	$5.60E + 03$	$3.03E + 02$	$2.63E + 04$	$2.22E + 04$	$4.48E + 04$	$\mathbf{3.00E + 02}$
	Std	$2.82E + 03$	$1.55E + 00$	$6.32E + 03$	$7.45E + 03$	$4.63E + 01$	$\mathbf{1.09E - 01}$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$3.02E - 11(+)$	$3.02E - 11(+)$	$3.02E - 11(+)$	
$F_4(X)$	Best	$4.01E + 02$	$4.08E + 02$	$\mathbf{4.00E + 02}$	$4.09E + 02$	$4.75E + 02$	$\mathbf{4.00E + 02}$
	Worst	$6.17E + 02$	$7.30E + 02$	$6.61E + 02$	$\mathbf{5.39E + 02}$	$6.86E + 02$	$5.40E + 02$
	Median	$4.72E + 02$	$5.33E + 02$	$5.46E + 02$	$\mathbf{4.29E + 02}$	$6.03E + 02$	$4.73E + 02$
	Mean	$4.88E + 02$	$5.35E + 02$	$5.27E + 02$	$\mathbf{4.47E + 02}$	$5.87E + 02$	$4.60E + 02$
	Std	$6.85E + 01$	$6.81E + 01$	$7.76E + 01$	$3.46E + 01$	$4.63E + 01$	$\mathbf{3.38E + 01}$
	P	$2.52E - 01(=)$	$3.16E - 05(+)$	$1.77E - 03(+)$	$3.15E - 02(-)$	$9.92E - 11(+)$	
$F_5(X)$	Best	$8.19E + 02$	$8.19E + 02$	$6.21E + 02$	$5.84E + 02$	$7.43E + 02$	$\mathbf{5.75E + 02}$
	Worst	$9.22E + 02$	$9.11E + 02$	$6.89E + 02$	$\mathbf{6.74E + 02}$	$9.00E + 02$	$6.90E + 02$
	Median	$8.77E + 02$	$8.64E + 02$	$6.55E + 02$	$6.36E + 02$	$7.94E + 02$	$\mathbf{6.34E + 02}$
	Mean	$8.69E + 02$	$8.66E + 02$	$6.54E + 02$	$\mathbf{6.34E + 02}$	$7.97E + 02$	$6.35E + 02$
	Std	$2.72E + 01$	$2.24E + 01$	$1.94E + 01$	$2.25E + 01$	$3.72E + 01$	$\mathbf{1.99E + 01}$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$1.06E - 03(+)$	$9.23E - 01(=)$	$3.02E - 11(+)$	
$F_6(X)$	Best	$6.38E + 02$	$6.33E + 02$	$\mathbf{6.00E + 02}$	$\mathbf{6.00E + 02}$	$6.43E + 02$	$\mathbf{6.00E + 02}$
	Worst	$6.65E + 02$	$6.63E + 02$	$\mathbf{6.00E + 02}$	$\mathbf{6.00E + 02}$	$6.67E + 02$	$\mathbf{6.00E + 02}$
	Median	$6.53E + 02$	$6.50E + 02$	$\mathbf{6.00E + 02}$	$\mathbf{6.00E + 02}$	$6.60E + 02$	$\mathbf{6.00E + 02}$
	Mean	$6.53E + 02$	$6.49E + 02$	$\mathbf{6.00E + 02}$	$\mathbf{6.00E + 02}$	$6.59E + 02$	$\mathbf{6.00E + 02}$
	Std	$7.26E + 00$	$9.15E + 00$	$2.81E - 02$	$\mathbf{9.62E - 03}$	$1.36E + 02$	$2.70E - 02$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$3.03E - 02(+)$	$2.39E - 04(-)$	$3.02E - 11(+)$	
$F_7(X)$	Best	$1.30E + 03$	$1.17E + 03$	$8.65E + 02$	$\mathbf{8.35E + 02}$	$1.27E + 03$	$8.59E + 02$
	Worst	$1.82E + 03$	$1.82E + 03$	$9.60E + 02$	$9.36E + 02$	$1.99E + 03$	$\mathbf{9.26E + 02}$
	Median	$1.75E + 03$	$1.56E + 03$	$9.14E + 02$	$9.02E + 02$	$1.70E + 03$	$\mathbf{8.95E + 02}$
	Mean	$1.68E + 03$	$1.52E + 03$	$9.16E + 02$	$8.98E + 02$	$1.67E + 03$	$\mathbf{8.96E + 02}$
	Std	$1.37E + 02$	$1.94E + 02$	$2.30E + 01$	$2.08E + 01$	$1.36E + 02$	$\mathbf{1.44E + 01}$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$1.58E - 04(+)$	$3.63E - 01(=)$	$3.02E - 11(+)$	
$F_8(X)$	Best	$1.10E + 03$	$1.08E + 03$	$8.83E + 02$	$\mathbf{8.79E + 02}$	$1.03E + 03$	$9.20E + 02$
	Worst	$1.25E + 03$	$1.26E + 03$	$9.76E + 02$	$\mathbf{9.65E + 02}$	$1.20E + 03$	$1.00E + 03$
	Median	$1.19E + 03$	$1.19E + 03$	$9.30E + 02$	$\mathbf{9.29E + 02}$	$1.10E + 03$	$9.63E + 02$
	Mean	$1.18E + 03$	$1.19E + 03$	$9.32E + 02$	$\mathbf{9.30E + 02}$	$1.10E + 03$	$9.61E + 02$
	Std	$4.05E + 01$	$4.04E + 01$	$1.80E + 01$	$2.23E + 01$	$3.66E + 01$	$\mathbf{1.79E + 01}$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$4.11E - 07(-)$	$1.61E - 06(-)$	$3.02E - 11(+)$	
$F_9(X)$	Best	$1.08E + 04$	$1.03E + 04$	$9.38E + 02$	$9.22E + 02$	$1.07E + 04$	$\mathbf{9.01E + 02}$
	Worst	$1.46E + 04$	$1.38E + 04$	$1.77E + 03$	$\mathbf{1.18E + 03}$	$2.04E + 04$	$1.66E + 03$
	Median	$1.31E + 04$	$1.30E + 04$	$1.12E + 03$	$\mathbf{1.01E + 03}$	$1.52E + 04$	$1.06E + 03$
	Mean	$1.29E + 04$	$1.27E + 04$	$1.16E + 03$	$\mathbf{1.03E + 03}$	$1.54E + 04$	$1.10E + 03$
	Std	$8.52E + 02$	$8.73E + 02$	$1.86E + 02$	$\mathbf{6.47E + 01}$	$8.38E + 02$	$1.83E + 02$
	P	$3.02E - 11(+)$	$3.02E - 11(+)$	$1.45E - 01(=)$	$3.18E - 01(=)$	$3.02E - 11(+)$	
$F_{10}(X)$	Best	$6.33E + 03$	$6.32E + 03$	$5.62E + 03$	$4.96E + 03$	$6.23E + 03$	$\mathbf{4.67E + 03}$
	Worst	$9.53E + 03$	$1.00E + 04$	$7.60E + 03$	$\mathbf{6.57E + 03}$	$9.50E + 03$	$6.86E + 03$
	Median	$8.14E + 03$	$7.73E + 03$	$6.47E + 03$	$5.81E + 03$	$8.16E + 03$	$\mathbf{5.70E + 03}$
	Mean	$8.12E + 03$	$7.82E + 03$	$6.52E + 03$	$5.77E + 03$	$8.02E + 03$	$\mathbf{5.75E + 03}$
	Std	$6.74E + 02$	$8.10E + 02$	$4.72E + 02$	$\mathbf{4.02E + 02}$	$8.38E + 02$	$6.04E + 02$
	P	$1.07E - 09(+)$	$3.08E - 08(+)$	$5.46E - 06(+)$	$2.57E - 07(+)$	$3.35E - 08(+)$	

TABLE 7: Continued.

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_{11}(X)$	Best	$1.20E+03$	$1.21E+03$	$\mathbf{1.14E+03}$	$1.18E+03$	$1.25E+03$	$\mathbf{1.14E+03}$
	Worst	$1.58E+03$	$1.41E+03$	$1.18E+03$	$1.40E+03$	$1.42E+03$	$\mathbf{1.17E+03}$
	Median	$1.36E+03$	$1.29E+03$	$1.16E+03$	$1.29E+03$	$1.33E+03$	$\mathbf{1.15E+03}$
	Mean	$1.37E+03$	$1.30E+03$	$1.16E+03$	$1.30E+03$	$1.34E+03$	$\mathbf{1.15E+03}$
	Std	$8.19E+01$	$4.72E+01$	$9.46E+00$	$5.71E+01$	$3.85E+01$	$\mathbf{8.10E+00}$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.34E-03(+)$	$8.19E-01(=)$	$3.02E-11(+)$	
$F_{12}(X)$	Best	$4.63E+05$	$\mathbf{1.22E+05}$	$1.00E+10$	$1.00E+10$	$1.04E+07$	$1.68E+05$
	Worst	$3.71E+06$	$3.83E+06$	$1.00E+10$	$1.00E+10$	$5.74E+07$	$\mathbf{2.25E+06}$
	Median	$1.30E+06$	$9.68E+05$	$1.00E+10$	$1.00E+10$	$2.55E+07$	$\mathbf{9.61E+05}$
	Mean	$1.64E+06$	$1.35E+06$	$1.00E+10$	$1.00E+10$	$2.57E+07$	$\mathbf{1.09E+06}$
	Std	$9.69E+05$	$9.69E+05$	$\mathbf{0.00E+00}$	$\mathbf{0.00E+00}$	$1.06E+05$	$5.84E+05$
	P	$4.68E-02(+)$	$4.64E-01(=)$	$1.21E-12(+)$	$1.21E-12(+)$	$3.02E-11(+)$	
$F_{13}(X)$	Best	$4.81E+03$	$\mathbf{3.09E+03}$	$1.00E+10$	$1.00E+10$	$2.31E+05$	$4.40E+03$
	Worst	$5.11E+04$	$\mathbf{4.77E+04}$	$1.00E+10$	$1.00E+10$	$6.39E+05$	$4.81E+04$
	Median	$1.90E+04$	$1.15E+04$	$1.00E+10$	$1.00E+10$	$4.16E+05$	$\mathbf{1.03E+04}$
	Mean	$2.35E+04$	$1.66E+04$	$1.00E+10$	$1.00E+10$	$4.02E+05$	$\mathbf{1.55E+04}$
	Std	$1.54E+04$	$1.36E+04$	$0.00E+00$	$0.00E+00$	$1.06E+05$	$\mathbf{1.16E+04}$
	P	$4.84E-02(+)$	$9.82E-01(=)$	$1.21E-12(+)$	$1.21E-12(+)$	$3.02E-11(+)$	
$F_{14}(X)$	Best	$2.89E+03$	$4.47E+03$	$1.64E+03$	$1.50E+03$	$2.95E+04$	$\mathbf{1.46E+03}$
	Worst	$2.11E+05$	$1.68E+05$	$1.93E+03$	$1.59E+03$	$4.22E+05$	$\mathbf{1.55E+03}$
	Median	$2.70E+04$	$4.00E+04$	$1.78E+03$	$1.54E+03$	$1.14E+05$	$\mathbf{1.50E+03}$
	Mean	$4.95E+04$	$4.51E+04$	$1.77E+03$	$1.54E+03$	$1.35E+05$	$\mathbf{1.50E+03}$
	Std	$4.91E+04$	$4.08E+04$	$6.03E+01$	$\mathbf{1.73E+01}$	$8.34E+04$	$2.07E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$4.69E-08(+)$	$3.02E-11(+)$	
$F_{15}(X)$	Best	$2.05E+03$	$2.48E+03$	$1.77E+03$	$1.61E+03$	$7.52E+04$	$\mathbf{1.56E+03}$
	Worst	$3.68E+04$	$3.60E+04$	$3.15E+03$	$\mathbf{2.00E+03}$	$2.26E+05$	$3.01E+03$
	Median	$1.48E+04$	$1.89E+04$	$1.92E+03$	$\mathbf{1.68E+03}$	$1.48E+05$	$1.69E+03$
	Mean	$1.65E+04$	$1.66E+04$	$1.99E+03$	$\mathbf{1.72E+03}$	$1.50E+05$	$1.77E+03$
	Std	$1.05E+04$	$9.16E+03$	$2.56E+02$	$\mathbf{1.03E+02}$	$4.96E+02$	$2.66E+02$
	P	$3.69E-11(+)$	$4.08E-11(+)$	$9.53E-07(+)$	$8.19E-01(=)$	$3.02E-11(+)$	
$F_{16}(X)$	Best	$2.94E+03$	$2.87E+03$	$\mathbf{2.08E+03}$	$2.14E+03$	$3.04E+03$	$2.30E+03$
	Worst	$4.63E+03$	$4.17E+03$	$3.21E+03$	$3.35E+03$	$4.71E+03$	$\mathbf{3.18E+03}$
	Median	$3.68E+03$	$3.53E+03$	$2.70E+03$	$\mathbf{2.62E+03}$	$4.12E+03$	$2.79E+03$
	Mean	$3.77E+03$	$3.59E+03$	$2.67E+03$	$\mathbf{2.62E+03}$	$3.96E+03$	$2.78E+03$
	Std	$4.72E+02$	$3.67E+02$	$2.93E+02$	$3.02E+02$	$4.96E+02$	$\mathbf{2.24E+02}$
	P	$1.78E-10(+)$	$2.37E-10(+)$	$1.91E-01(=)$	$1.56E-02(-)$	$5.49E-11(+)$	
$F_{17}(X)$	Best	$2.35E+03$	$2.72E+03$	$2.11E+03$	$\mathbf{2.01E+03}$	$3.00E+03$	$\mathbf{2.01E+03}$
	Worst	$4.02E+03$	$4.18E+03$	$2.71E+03$	$\mathbf{2.70E+03}$	$4.31E+03$	$2.78E+03$
	Median	$3.42E+03$	$3.43E+03$	$2.47E+03$	$\mathbf{2.31E+03}$	$3.57E+03$	$2.43E+03$
	Mean	$3.37E+03$	$3.39E+03$	$2.43E+03$	$\mathbf{2.32E+03}$	$3.56E+03$	$2.40E+03$
	Std	$4.05E+02$	$3.57E+02$	$1.74E+02$	$\mathbf{1.62E+02}$	$3.02E+02$	$1.92E+02$
	P	$2.87E-10(+)$	$4.08E-11(+)$	$6.10E-01(=)$	$7.01E-02(=)$	$3.02E-11(+)$	
$F_{18}(X)$	Best	$9.41E+04$	$7.26E+04$	$\mathbf{2.51E+04}$	$6.30E+04$	$3.20E+05$	$4.45E+04$
	Worst	$9.83E+05$	$4.50E+05$	$3.71E+05$	$\mathbf{2.02E+05}$	$2.59E+06$	$2.64E+05$
	Median	$2.48E+05$	$1.63E+05$	$1.17E+05$	$1.16E+05$	$1.23E+06$	$\mathbf{8.99E+04}$
	Mean	$2.78E+05$	$1.75E+05$	$1.39E+05$	$1.27E+05$	$1.23E+06$	$\mathbf{1.01E+05}$
	Std	$1.61E+05$	$8.15E+04$	$8.07E+04$	$\mathbf{3.92E+04}$	$1.18E+06$	$4.43E+04$
	P	$8.89E-10(+)$	$1.75E-05(+)$	$3.92E-02(+)$	$3.85E-03(+)$	$3.02E-11(+)$	
$F_{19}(X)$	Best	$2.98E+03$	$2.38E+03$	$2.06E+03$	$1.96E+03$	$1.03E+05$	$\mathbf{1.93E+03}$
	Worst	$4.46E+04$	$4.43E+04$	$2.73E+03$	$2.03E+03$	$3.66E+06$	$\mathbf{2.00E+03}$
	Median	$2.05E+04$	$1.69E+04$	$2.29E+03$	$1.99E+03$	$1.39E+06$	$\mathbf{1.95E+03}$
	Mean	$2.28E+04$	$2.13E+04$	$2.32E+03$	$1.99E+03$	$1.61E+06$	$\mathbf{1.95E+03}$
	Std	$1.63E+04$	$1.32E+04$	$1.37E+02$	$\mathbf{1.57E+01}$	$1.18E+06$	$1.60E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.02E-11(+)$	$3.20E-09(+)$	$3.02E-11(+)$	

TABLE 7: Continued.

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_{20}(X)$	Best	$2.72E+03$	$2.61E+03$	$2.25E+03$	$2.30E+03$	$2.50E+03$	$\mathbf{2.15E+03}$
	Worst	$3.87E+03$	$3.80E+03$	$2.84E+03$	$2.86E+03$	$3.82E+03$	$\mathbf{2.72E+03}$
	Median	$3.23E+03$	$3.19E+03$	$2.62E+03$	$2.56E+03$	$3.20E+03$	$\mathbf{2.44E+03}$
	Mean	$3.28E+03$	$3.20E+03$	$2.60E+03$	$2.56E+03$	$3.21E+03$	$\mathbf{2.46E+03}$
	Std	$3.19E+02$	$3.65E+02$	$\mathbf{1.38E+02}$	$1.39E+02$	$3.14E+02$	$1.50E+02$
	P	$3.34E-11(+)$	$1.33E-10(+)$	$6.20E-04(+)$	$9.07E-03(+)$	$1.21E-10(+)$	
$F_{21}(X)$	Best	$2.50E+03$	$2.55E+03$	$2.41E+03$	$\mathbf{2.40E+03}$	$2.44E+03$	$2.41E+03$
	Worst	$2.85E+03$	$2.86E+03$	$2.50E+03$	$2.47E+03$	$2.78E+03$	$\mathbf{2.46E+03}$
	Median	$2.68E+03$	$2.68E+03$	$2.47E+03$	$\mathbf{2.43E+03}$	$2.63E+03$	$2.43E+03$
	Mean	$2.69E+03$	$2.68E+03$	$2.46E+03$	$\mathbf{2.43E+03}$	$2.61E+03$	$2.43E+03$
	Std	$9.83E+01$	$7.97E+01$	$2.11E+01$	$1.87E+01$	$2.24E+03$	$\mathbf{1.44E+01}$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$1.47E-07(+)$	$8.30E-01(=)$	$6.07E-11(+)$	
$F_{22}(X)$	Best	$8.06E+03$	$8.10E+03$	$\mathbf{2.30E+03}$	$\mathbf{2.30\ E+03}$	$2.32E+03$	$\mathbf{2.30E+03}$
	Worst	$1.21E+04$	$1.11E+04$	$8.54E+03$	$\mathbf{8.35E+03}$	$1.20E+04$	$9.50E+03$
	Median	$9.61E+03$	$9.92E+03$	$\mathbf{7.47E+03}$	$7.55E+03$	$1.03E+04$	$8.38E+03$
	Mean	$9.79E+03$	$9.66E+03$	$6.95E+03$	$\mathbf{6.90E+03}$	$9.82E+03$	$7.88E+03$
	Std	$1.08E+03$	$8.33E+02$	$1.65E+03$	$1.91E+03$	$2.24E+03$	$\mathbf{1.60E+03}$
	P	$3.96E-08(+)$	$1.10E-08(+)$	$2.28E-05(-)$	$4.08E-05(-)$	$3.96E-08(+)$	
$F_{23}(X)$	Best	$3.02E+03$	$3.09E+03$	$2.82E+03$	$\mathbf{2.73E+03}$	$3.06E+03$	$2.74E+03$
	Worst	$3.48E+03$	$3.44E+03$	$2.94E+03$	$\mathbf{2.91E+03}$	$3.59E+03$	$2.94E+03$
	Median	$3.18E+03$	$3.27E+03$	$2.87E+03$	$\mathbf{2.87E+03}$	$3.31E+03$	$2.89E+03$
	Mean	$3.22E+03$	$3.26E+03$	$\mathbf{2.87E+03}$	$2.87E+03$	$3.31E+03$	$2.89E+03$
	Std	$1.18E+02$	$7.68E+01$	$\mathbf{2.76E+01}$	$3.47E+01$	$1.28E+02$	$3.69E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$2.16E-03(-)$	$2.38E-03(-)$	$3.02E-11(+)$	
$F_{24}(X)$	Best	$3.22E+03$	$3.17E+03$	$\mathbf{2.99E+03}$	$3.00E+03$	$3.32E+03$	$3.00E+03$
	Worst	$3.71E+03$	$3.58E+03$	$3.15E+03$	$\mathbf{3.14E+03}$	$3.78E+03$	$\mathbf{3.14E+03}$
	Median	$3.36E+03$	$3.40E+03$	$3.07E+03$	$3.06E+03$	$3.54E+03$	$\mathbf{3.05E+03}$
	Mean	$3.38E+03$	$3.39E+03$	$3.08E+03$	$3.06E+03$	$3.53E+03$	$\mathbf{3.05E+03}$
	Std	$1.19E+02$	$9.85E+01$	$\mathbf{3.43E+01}$	$3.68E+01$	$3.30E+01$	$3.59E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$5.57E-03(+)$	$5.30E-01(=)$	$3.02E-11(+)$	
$F_{25}(X)$	Best	$\mathbf{2.96E+03}$	$2.98E+03$	$2.96E+03$	$2.96E+03$	$3.01E+03$	$\mathbf{2.96E+03}$
	Worst	$3.11E+03$	$3.12E+03$	$3.08E+03$	$3.06E+03$	$3.14E+03$	$\mathbf{3.03E+03}$
	Median	$3.07E+03$	$3.08E+03$	$3.03E+03$	$2.98E+03$	$3.12E+03$	$\mathbf{3.02E+03}$
	Mean	$3.05E+03$	$3.07E+03$	$3.03E+03$	$2.99E+03$	$3.11E+03$	$\mathbf{3.00E+03}$
	Std	$3.83E+01$	$2.94E+01$	$3.74E+01$	$2.99E+01$	$3.30E+01$	$\mathbf{2.53E+01}$
	P	$7.60E-07(+)$	$2.87E-10(+)$	$1.32E-04(+)$	$8.19E-01(=)$	$1.29E-09(+)$	
$F_{26}(X)$	Best	$\mathbf{2.90E+03}$	$\mathbf{2.90E+03}$	$\mathbf{2.90E+03}$	$\mathbf{2.90E+03}$	$1.03E+04$	$\mathbf{2.90E+03}$
	Worst	$1.13E+04$	$1.13E+04$	$5.58E+03$	$\mathbf{5.52E+03}$	$1.40E+04$	$5.80E+03$
	Median	$\mathbf{2.90E+03}$	$6.53E+03$	$5.13E+03$	$4.87E+03$	$1.19E+04$	$\mathbf{2.90E+03}$
	Mean	$5.81E+03$	$6.49E+03$	$5.02E+03$	$4.18E+03$	$1.19E+04$	$\mathbf{3.95E+03}$
	Std	$3.69E+03$	$3.61E+03$	$\mathbf{6.09E+02}$	$1.15E+03$	$9.27E+02$	$1.24E+03$
	P	$1.86E-01(=)$	$8.07E-01(=)$	$2.61E-02(+)$	$2.12E-01(=)$	$3.02E-11(+)$	
$F_{27}(X)$	Best	$3.35E+03$	$3.43E+03$	$3.24E+03$	$3.20E+03$	$3.80E+03$	$\mathbf{3.17E+03}$
	Worst	$4.07E+03$	$3.84E+03$	$3.38E+03$	$3.34E+03$	$5.08E+03$	$\mathbf{3.33E+03}$
	Median	$3.57E+03$	$3.57E+03$	$3.30E+03$	$3.27E+03$	$4.44E+03$	$\mathbf{3.26E+03}$
	Mean	$3.57E+03$	$3.62E+03$	$3.31E+03$	$\mathbf{3.26E+03}$	$4.47E+03$	$\mathbf{3.26E+03}$
	Std	$1.45E+02$	$1.16E+02$	$4.11E+01$	$\mathbf{3.49E+01}$	$4.30E+01$	$3.81E+01$
	P	$3.02E-11(+)$	$3.02E-11(+)$	$3.18E-04(+)$	$8.88E-01(=)$	$3.02E-11(+)$	
$F_{28}(X)$	Best	$\mathbf{3.26E+03}$	$\mathbf{3.26E+03}$	$\mathbf{3.26E+03}$	$\mathbf{3.26E+03}$	$3.29E+03$	$\mathbf{3.26E+03}$
	Worst	$3.32E+03$	$3.43E+03$	$3.31E+03$	$\mathbf{3.30E+03}$	$3.51E+03$	$3.31E+03$
	Median	$3.30E+03$	$3.31E+03$	$\mathbf{3.26E+03}$	$\mathbf{3.26E+03}$	$3.37E+03$	$\mathbf{3.26E+03}$
	Mean	$3.30E+03$	$3.31E+03$	$3.27E+03$	$\mathbf{3.26E+03}$	$3.37E+03$	$\mathbf{3.26E+03}$
	Std	$1.69E+01$	$3.82E+01$	$1.89E+01$	$\mathbf{8.18E+00}$	$4.30E+01$	$1.49E+01$
	P	$1.73E-06(+)$	$8.35E-08(+)$	$3.18E-03(+)$	$9.06E-08(-)$	$7.39E-11(+)$	

TABLE 7: Continued.

F	Index	SSA	CSSA	MSSCS	CSsin	FA-CL	LSSA
$F_{29}(X)$	Best	$4.07E+03$	$3.90E+03$	$3.35E+03$	$3.36E+03$	$4.99E+03$	$\mathbf{3.34E+03}$
	Worst	$5.69E+03$	$5.59E+03$	$4.01E+03$	$\mathbf{3.93E+03}$	$6.57E+03$	$3.99E+03$
	Median	$4.75E+03$	$4.75E+03$	$3.77E+03$	$\mathbf{3.68E+03}$	$5.78E+03$	$3.70E+03$
	Mean	$4.84E+03$	$4.80E+03$	$3.74E+03$	$\mathbf{3.65E+03}$	$5.80E+03$	$3.69E+03$
	Std	$3.75E+02$	$4.58E+02$	$1.60E+02$	$1.66E+02$	$4.30E+02$	$\mathbf{1.50E+02}$
	P	$3.02E-11(+)$	$3.69E-11(+)$	$1.76E-01(=)$	$3.79E-01(=)$	$3.02E-11(+)$	
$F_{30}(X)$	Best	$6.92E+05$	$6.25E+05$	$6.18E+05$	$6.05E+05$	$2.64E+07$	$\mathbf{5.88E+05}$
	Worst	$2.24E+06$	$2.30E+06$	$\mathbf{7.72E+05}$	$8.91E+05$	$7.04E+07$	$7.79E+05$
	Median	$1.27E+06$	$1.11E+06$	$6.94E+05$	$6.32E+05$	$3.43E+07$	$\mathbf{6.13E+05}$
	Mean	$1.31E+06$	$1.17E+06$	$6.92E+05$	$6.44E+05$	$3.67E+07$	$\mathbf{6.31E+05}$
	Std	$4.24E+05$	$3.82E+05$	$4.34E+04$	$5.26E+04$	$4.30E+01$	$\mathbf{5.23E+04}$
	P	$5.49E-11(+)$	$1.61E-10(+)$	$5.46E-06(+)$	$2.42E-02(+)$	$3.02E-11(+)$	
$+/=/-$		26/3/0	25/4/0	22/4/3	10/12/7	29/0/0	

TABLE 8: Statistics table of the optimization route by the algorithm.

Map size	Performance index	LSSA	SSA
$15 * 15$	Shortest	19.7990	25.4558
	Average	21.6846	32.9983
	Worst	25.4558	39.5980

Through 12 standard test functions, LSSA is proved to be better than SSA, BSO, CSSA, ISSA, GWO, and PSO. At the same time, in order to avoid LSSA only depending on the zero point, this paper compares LSSA with MSSCS, CSsin, and FA-CL which have been verified by CEC function in recent years. The results show that the LSSA algorithm has good universality, while other algorithms have limitations in some functions. Finally, the practicability of LSSA is verified by robot path planning. The LSSA test effect is satisfying, but there are some shortcomings; for instance, during the function optimization process, the time consumption is large and cannot show the best effect on some functions of CEC 2017, showing an unstable effect. Of course, the increase of time is inevitable because we add workload. In the future, we need to do three aspects of work, the first is how to balance the time and optimization ability of the algorithm, the second is how to improve the stability of the algorithm on the basis of the previous, and the third is how to get a better application in practical complex engineering. On the contrary, we will also try to analyze and optimize in MBO, SMA, MSA, HGS, and HHO and better apply to practical problems.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] Y. Feng, S. Deb, G. G. Wang et al., "Monarch butterfly optimization: a comprehensive review," *Expert Systems with Applications*, vol. 168, Article ID 114418, 2020.

[2] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: a new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, 2020.

[3] G.-G. Wang, "Moth search algorithm: a bio-inspired metaheuristic algorithm for global optimization problems," *Memetic Computing*, vol. 10, no. 2, pp. 151–164, 2018.

[4] Y. Yang, H. Chen, A. A. Heidari, and A. H. Gandomi, "Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts," *Expert Systems with Applications*, vol. 177, Article ID 114864, 2021.

[5] R. Salgotra and U. Singh, "The naked mole-rat algorithm," *Neural Computing and Applications*, vol. 31, no. 12, pp. 8837–8857, 2019.

[6] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: algorithm and applications," *Future Generation Computer Systems*, vol. 97, pp. 849–872, 2019.

[7] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: sparrow search algorithm," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 22–34, 2020.

[8] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[9] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proceedings of the 1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, vol. 5, IEEE, Hyatt, OR, USA, 1997.

[10] C. R. Houck, J. Joines, and M. G. Kay., "A genetic algorithm for function optimization: a Matlab implementation," *Ncsu-ie Tr*, vol. 95, no. 9, pp. 1–10, 1995.

[11] X. Lu, X. Mu, J. Zhang, and Z. Wang, "Chaos sparrow search optimization algorithm," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 1–10, 2020.

[12] X. Lu, X. Mu, and J. Zhang, "Multi-threshold image segmentation based on improved sparrow search algorithm," *Systems Engineering and Electronics*, vol. 43, no. 2, pp. 318–327, 2021.

[13] Q. Mao and Q. Zhang, "Improved sparrow algorithm combining Cauchy mutation and opposition-based learning," *Journal of Computer Science and Exploration*, vol. 15, pp. 1–12, 2020.

[14] G. Liu, C. Shu, Z. Liang, B. Peng, and L. Cheng, "A modified sparrow search algorithm with application in 3D route planning for UAV," *Sensors*, vol. 21, no. 4, p. 1224, 2021.

[15] P. Wang, Y. Zhang, and H. Yang, "Research on economic optimization of microgrid cluster based on chaos sparrow search algorithm," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 5556780, 18 pages, 2021.

[16] C. Zhang and S. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowledge-Based Systems*, vol. 220, no. 10, Article ID 106924, 2021.

[17] T. Wang and L. Yang, "Beetle swarm optimization algorithm: theory and application," 2018, https://arxiv.org/abs/1808.00206.

[18] H. Peng, Z. Zeng, C. Deng, and Z. Wu, "Multi-strategy serial cuckoo search algorithm for global optimization," *Knowledge-Based Systems*, vol. 214, Article ID 106729, 2021.

[19] R. Salgotra, U. Singh, S. Saha et al., "Improving cuckoo search: incorporating changes for CEC 2017 and CEC 2020 benchmark problems," in *Proceedings of the 2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–7, IEEE, Glasgow, UK, 2020.

[20] H. Peng, W. Zhu, C. Deng, and Z. Wu, "Enhancing firefly algorithm with courtship learning," *Information Sciences*, vol. 543, pp. 18–42, 2021.

[21] W. Li, G. G. Wang, and A. H. Gandomi, "A survey of learning-based intelligent optimization algorithms," *Archives of Computational Methods in Engineering*, pp. 1–19, 2021.

[22] W. Li, G.-G. Wang, and A. H. Alavi, "Learning-based elephant herding optimization algorithm for solving numerical optimization problems," *Knowledge-Based Systems*, vol. 195, Article ID 105675, 2020.

[23] W. Li and G. G. Wang, "Elephant herding optimization using dynamic topology and biogeography-based optimization based on learning for numerical optimization," *Engineering with Computers*, pp. 1–29, 2021.

[24] H. R. Tizhoosh, "Opposition-based learning: a new scheme for machine intelligence," *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)*, vol. 1, pp. 695–701, 2005.

[25] J. He, Z. Peng, D. Cui, and Q. Li, "A teaching and learning optimization algorithm based on multi-inverse learning," *Engineering Science and Technology*, vol. 51, no. 6, pp. 159–167, 2019.

[26] S. Rahnamayan, H. R. Tizhoosh, and M. M. A. Salama, "Opposition-based differential evolution," *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 1, pp. 64–79, 2008.

[27] L. Wen, T. Wu, M. Tang et al., "Gray wolf optimization algorithm based on lens imaging learning strategy," *Acta Automatica Sinica*, vol. 46, no. 10, pp. 2148–2164, 2020.

[28] S. Mirjalili, "SCA: a sine cosine algorithm for solving optimization problems," *Knowledge-based Systems*, vol. 96, pp. 120–133, 2016.

[29] R. Sindhu, R. Ngadiran, Y. M. Yacob, N. A. H. Zahri, and M. Hariharan, "Sine-cosine algorithm for feature selection with elitism strategy and new updating mechanism," *Neural Computing and Applications*, vol. 28, no. 10, pp. 2947–2958, 2017.

[30] C. Ouyang and D. Zhu, "Multi-strategy improved sparrow search algorithm based on K-means," *Electro-Optics and Control*, vol. 1–9, 2021.

[31] W. Fengtao, Y. Zhang, J. Li, and Y. Shi, "Improved sine and cosine algorithm based on dynamic classification strategy," *System Engineering and Electronic Technology*, vol. 1–14, 2021.

[32] Li Yu, Y. Zhao, and J. Liu, "A levy flight sine cosine algorithm for global optimization problems," *International Journal of Distributed Systems and Technologies (IJDST)*, vol. 12, no. 1, pp. 49–66, 2021.

[33] J. Luo, Z. Liu, P. Zhang, X. Liu, and L. Zheng, "The application of improved bird swarm algorithm based on nonlinear factors in dynamic energy consumption management," *Journal of Electronics & Information Technology*, vol. 42, no. 3, pp. 729–736, 2020.

[34] N. H. Awad, M. Z. Ali, J. J. Liang et al., "Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization," Technical Report, Zhengzhou University, Zhengzhou, China, 2016.