

## RESEARCH ARTICLE

# An Improved Chaos Sparrow Search Optimization Algorithm Using Adaptive Weight Modification and Hybrid Strategies

XUAN-YU ZHANG<sup>1,2</sup>, KAI-QING ZHOU<sup>1,2</sup>, PENG-CHENG LI<sup>1,2</sup>, YIN-HONG XIANG<sup>1,2</sup>, AZLAN MOHD ZAIN<sup>3</sup>, (Member, IEEE), AND AREZOO SARKHEYLI-HÄGELE<sup>4</sup>

<sup>1</sup>School of Communication and Electronic Engineering, Jishou University, Jishou, Hunan 416000, China

<sup>2</sup>Laboratory of Ethnic Cultural Heritage Digitization of Wuling Mountain Areas, Jishou University, Jishou, Hunan 416000, China

<sup>3</sup>UTM Big Data Center, Universiti Teknologi Malaysia, Skudai, Johor 81310, Malaysia

<sup>4</sup>Internet of Things and People Research Center, Department of Computer Science and Media Technology, Malmö University, 20506 Malmö, Sweden

Corresponding author: Kai-Qing Zhou (kqzhou@jsu.edu.cn)


This work was supported in part by the National Natural Science Foundation of China under Grant 62066016; in part by the Natural Science Foundation of Hunan Province, China, under Grant 2020JJ5458; in part by the Fundamental Research Grant Scheme of Malaysia under Grant R.J130000.7809.5F524; and in part by the Jishou University Graduate Research and Innovation Project under Grant JDY21067.

**ABSTRACT** Sparrow Search Algorithm (SSA) is a kind of novel swarm intelligence algorithm, which has been applied in-to various domains because of its unique characteristics, such as strong global search capability, few adjustable parameters, and a clear structure. However, the SSA still has some inherent weaknesses that hinder its further development, such as poor population diversity, weak local searchability, and falling into local optima easily. This manuscript proposes an improved chaos sparrow search optimization algorithm (ICSSOA) to overcome the mentioned shortcomings of the standard SSA. Firstly, the Cubic chaos mapping is introduced to increase the population diversity in the initialization stage. Then, an adaptive weight is employed to automatically adjust the search step for balancing the global search performance and the local search capability in different phases. Finally, a hybrid strategy of Levy flight and reverse learning is presented to perturb the position of individuals in the population according to the random strategy, and a greedy strategy is utilized to select individuals with higher fitness values to decrease the possibility of falling into the local optimum. The experiments are divided into two modules. The former investigates the performance of the proposed approach through 20 benchmark functions optimization using the ICSSOA, standard SSA, and other four SSA variants. In the latter experiment, the selected 20 functions are also optimized by the ICSSOA and other classic swarm intelligence algorithms, namely ACO, PSO, GWO, and WOA. Experimental results and corresponding statistical analysis revealed that only one function optimization test using the ICSSOA was slightly lower than the CSSOA and the WOA among the 20-function optimization. In most cases, the values for both accuracy and convergence speed are higher than other algorithms. The results also indicate that the ICSSOA has an outstanding ability to jump out of the local optimum.

**INDEX TERMS** Adaptive weighting modification, cubic chaos mapping, levy flight, reverse learning, sparrow search algorithm.

## I. INTRODUCTION

Swarm Intelligence (SI) is a method to solve complex realistic problems by simulating biological communication and cooperative behaviors and conveying information among

The associate editor coordinating the review of this manuscript and approving it for publication was Ludovico Minati .

individuals [1]. Recently, the SI has received wide attention from different aspects and obtained fruitful achievements because of its simple structure, easy implementation, good robustness, and wide applicability [2], [3], [4]. As one of the latest SI algorithms, a sparrow search algorithm (SSA) was proposed in 2020 by imitating the foraging and anti-predatory behaviors of sparrows [5]. Compared with other SI

algorithms, the SSA has been successfully applied in various fields because of its excellent characteristics, such as good global search capability, few adjustable parameters, and clear structure [6], [7], [8], [9], [10], [11].

Although the SSA has the mentioned unique highlights, some inherent bottlenecks still hinder its further investigation for resettling the industrial issues, such as poor population diversity, weak local searchability, and the tendency to fall into local optimal. Scholars have discussed a sea of studies to improve the performance of the standard SSA from various viewpoints. For example, Zhang and Ding employed a logical mapping to improve the population diversity in the initialization phase and introduced adaptive hyperparameters and mutation operators to enhance the optimal-seeking ability. Experiential results verified that the improved algorithm was superior to similar SSA variants by optimizing 13 test functions [12]. Ouyang *et al.* utilized a mirror reverse learning and a positive cosine mechanism in the global search and local convergence phases to improve convergence accuracy. A differential evolution approach was also presented to avoid missing high-quality solutions to enhance the ability to escape local optima. Finally, the algorithm's feasibility was verified by comparing three SSA variants and three classical SI algorithms [13]. Wang *et al.* used Bernoulli chaos mapping and adaptive weighting factors to improve the global search range of the SSA and used a hybrid Cauchy mutation and reverse learning to jump out of the local optimum. The experimental phase used different test functions to compare the two classical SI algorithms and verified that the improved algorithm has high convergence speed and solution accuracy [14]. Zhang *et al.* proposed an improved semi-supervised ensemble classifier using the SSA (AdaBoost-ISSA-S4VM) to enhance lung disease diagnosis classification accuracy. The experimental results illustrated that the classification model performed well on both labeled and unlabeled lung CT images [15]. Wu *et al.* presented a new greedy SSA. Firstly, a greedy strategy was used to increase the diversity of the initialized population. Secondly, a genetic operator was utilized to update the position of each iteration, and finally the adaptive weights were employed to improve the adaptability of the algorithm. This improved algorithm outperformed other algorithms through experimental comparison in terms of accuracy, optimization speed, and stability [16]. Other similar SSA variants and related applications could be found in [17], [18], [19], [20], [21], and [22]. Details of some of the above mentions of SSA improvements are shown in Table 1.

These discussed SSA variants are focused on enriching population diversity and avoiding falling into local optima. The main improvement ideas could be summarized into the following points: improving the initial population diversity using different hybrid strategies to find the optimum, jumping out of the local optimum dilemma, and that is how the mentioned SSA variants succeeded in showing better results than the standard SSA. However, the existing SSA variants still have some shortcomings:

- (1) The convergence speed is still relatively slow.
- (2) They do not consider how to balance the exploration capability in the early iterations and the exploitation capability in the late iterations.
- (3) They only consider how to reduce the risk of avoiding falling into a local optimum but do not propose a feasible solution to how the algorithm can escape from the local optimum after falling into a local optimum.

Based on the above findings, an improved chaos sparrow search optimization algorithm (ICSSOA) is presented to balance the exploration capability in the early iterations and the exploitation capability in the late iterations. The main improvements of the ICSSOA algorithm could be separated into the following points:

Firstly, the initial sparrow population is enriched with diversity by using the Cubic function for chaos initialization during population initialization.

Secondly, the introduction of adaptive weights balances the searchability in the early stage and the development ability in the later stage.

Finally, a hybrid strategy of Levy flight and reverse learning is used to enhance the ability of the algorithm to escape from the local optimum. The experiments are implemented further to validate the performance of ICSSOA from two dimensions. On the one hand, a longitudinal comparison is executed by optimizing the 20 benchmark functions among the SSA, the ICSSOA, and other four SSA variants (Chaos Sparrow Search Optimization Algorithm (CSSOA) [23], Improved Sparrow Search Algorithm (ISSA(a)) [24], Chaotic strategy and Adaptive inertia weight strategy Sparrow Search Algorithm (CASSA) [25] and Improved Sparrow Search Algorithm (ISSA(b)) [26]). On the other hand, a horizontal comparison is also carried out by optimizing the 20 benchmark functions among the ICSSOA and other SI algorithms (Grey Wolf Optimizer (GWO) [27], Particle Swarm Optimization (PSO) [28], Whale Optimization Algorithm (WOA) [29], and Ant Colony Optimization (ACO) [30], [31], [32]). In short, a systematical analysis of the ten algorithms is carried out by testing 11 unimodal and 9 multimodal test functions, thus verifying the superiority of ICSSOA in terms of merit-seeking capability, solution accuracy and convergence speed.

The rest sections of this paper are organized as follows. Section two recalls the related notions. Section three discusses the proposed ICSSOA in detail. Section four reveals the performance of the proposed algorithm throughout function optimization tasks and corresponding statistical analysis. Section five concludes the entire paper.

## II. STANDARD SPARRPW SEARCH ALGORITHM

The SSA refers to the process of predation and anti-predation behavior of sparrows for location updates, based on the following principles.

The sparrows in the population are divided into two categories, producers and followers. The two identities of the sparrow can be interchanged, and each sparrow has a

TABLE 1. Literature analysis.

Reference number	Features	Advantages	Application
Reference [12]	Logistic mapping Self-adaptive hyper -parameters Mutation operator	Enhance global search capability	Model parameter optimization
Reference [13]	Regularization parameter Reverse learning strategy An improved sine and cosine guidance mechanism	Enrich population diversity Improve search accuracy	Path Planning
Reference [14]	A differential-based local search Bernoulli chaotic mapping Dynamic adaptive weighting	Enrich population diversity Improv ability to jump out of local optima	Model optimization
Reference [15]	Cauchy mutation Reverse learning Sine cosine algorithm New labor cooperation structure	Improve global search capability Improved convergence ability	Data classification
Reference [16]	Greedy algorithm Genetic operators Adaptive weight	Increase the diversity of the population Balance global search and local development capabilities Increase the adaptability of the algorithm	Traveling salesman problem
Reference [17]	Sine map Adaptive adjustment of hyper-parameters Mutation strategy	Improve the global optimization ability of the algorithm	Image recognition

danger awareness mechanism. To be specific, each sparrow is aware of approaching danger or natural enemies and will immediately engage in anti-predatory behavior to ensure its safety. The producers themselves are high in energy and good at finding food, searching widely, leading other sparrows in their search and foraging for food. Followers follow the producer to obtain more food, and followers are always on the lookout for the producer to increase their food intake by grabbing food or foraging around the producer.

A. BASIC CONCEPTS

The standard SSA formula associated with part A is referred according to the reference [5].

Assuming N sparrows in D-dimensional space, the population matrix is shown in Eq. (1).

$$X = [x_1, x_2, \dots, x_N]^T, x_i = [x_{i,1}, x_{i,2}, \dots, x_{i,D}] \quad (1)$$

where  $x_{i,D}$  represents the position of the  $i_{th}$  sparrow in dimension D.

Producers are typically 10% to 20% of the population size and the location is updated using the Eq. (2).

$$x_{i,j}^{t+1} = \begin{cases} x_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot iter_{max}}\right) & R_2 < ST \\ x_{i,j}^t + Q \cdot L & R_2 \geq ST \end{cases} \quad (2)$$

where,  $t$  represents the current number of iterations.  $j = 1, 2, \dots, d$ .  $iter_{max}$  represents the maximum number of iterations.  $\alpha$  is a uniform random number in the range 0 to 1.  $R_2$  ( $R_2 \in [0, 1]$ ) and  $ST$  ( $ST \in [0.5, 1.0]$ ) represent the alert and safety values for sparrows respectively.  $Q$  is a random number that follows a standard normal distribution.  $L$  is a matrix of  $1 \times d$  and each element of the matrix is 1.  $R_2 < ST$  indicates that the producer is surrounded by no

natural predators and is in a relatively safe location and the producers enters a wide-area search mode. Otherwise,  $R_2 \geq ST$  means that the producer is aware of the presence of a natural predator, then the producer should go to another area to forage.

The follower position is updated due to the Eq. (3).

$$x_{i,j}^{t+1} = \begin{cases} Q \cdot \exp\left(\frac{x_{worst}^t - x_{i,j}^t}{i^2}\right) & i > \frac{n}{2} \\ x_p^{t+1} + |x_{i,j}^t - x_p^{t+1}| \bullet A^+ \bullet L & otherwise \end{cases} \quad (3)$$

where,  $x_{worst}^t$  represents the current position of the least adapted sparrow.  $x_p$  represents the position of the sparrow with the best current producer adaptation.  $A$  represents a matrix of  $1 \times d$ , and each element of the matrix is assigned a random value of 1 or -1.  $A^+ = A^T (AA^T)^{-1}$ . Moreover,  $i > \frac{n}{2}$  means that the  $i_{th}$  sparrow with the worse adaptation is likely to be hungry and so needs to travel to another location to forage.

B. DANGER AWARENESS MECHANISM

Some sparrows perceive the threat of predators when foraging and will abandon their current position and fly to another position. Sparrows that perceive danger generally make up 10%-20% of the population. The position of the sparrows that perceive danger is updated as shown in Eq. (4).

$$x_{i,j}^{t+1} = \begin{cases} x_{best}^t + \beta \cdot |x_{i,j}^t - x_{best}^t| & f_i > f_g \\ x_{i,j}^t + K \cdot \left(\frac{|x_{i,j}^t - x_{worst}^t|}{(f_i - f_w) + \varepsilon}\right) & f_i = f_g \end{cases} \quad (4)$$

$x_{best}$  is the current global optimum position.  $\beta$  is a standard normally distributed random step control parameter.  $K \in [-1, 1]$  is a uniform random number.  $f_i$  is the current fitness value of the sparrow.  $f_g$  and  $f_w$  are the current global best-fit and worst-fit values respectively.  $\varepsilon$  is the smallest constant to avoid division by zero error.

When  $f_i > f_g$ , indicating that the individual sparrow is at the edge of the population and vulnerable to attack by natural predators. When  $f_i = f_g$ , the individual sparrow is in the middle of the population and is aware of the danger and needs to move closer to other sparrows to escape from predators.

**C. BASIC SSA STEP**

According to the above sparrow foraging process, the overall execution process of the SSA algorithm includes the following six steps.

**Step 1:** Randomly initialize the sparrow population. Set parameters such as maximum number of iterations, population size, number of discoverers, number of sparrows sensing danger, safety value and set the objective function.

**Step 2:** Sort the sparrow population for fitness values to find the best individual and the current worst individual.

**Step 3:** The producer performs a location update according to Eq. (2).

**Step 4:** The follower performs a position update according to Eq. (3).

**Step 5:** Some sparrows are randomly selected to perceive danger and thus move, and their positions are updated as shown in Eq. (4).

**Step 6:** Evaluate all individuals. If the updated individual is better than the original one, replace the updated individual with that individual. If the maximum number of iterations is reached, the best sparrow position and the optimal fitness value are output. Otherwise, the current number of iterations is added by one and move to step 2.

**III. IMPROVED CHAOS SPARRPW SEARCH OPTIMIZATION ALGORITHM**

For the standard SSA, the producer does not maintain a good balance between the search of the early iteration and the development of the later iteration in the global search, failing to search extensively for the optimal solution in the early iteration and the slightly lower accuracy of the solution in the later iteration. The followers blindly follow the producer’s position, reducing population diversity and quickly falling into the dilemma of local optimum. To address the above problems, enriching the diversity of the population is the primary mechanism to maintain the dynamic balance of producer search and development, and improving the ability to escape local optima is the focus of ICSSOA research. This chapter will explain the ICSSOA in detail from the following aspects.

**A. CUBIC CHAOS MAPPING**

The Chaos is a nonlinear phenomenon that exists in nature and has been applied to optimize algorithms. It enriches the

diversity of populations and facilitates the algorithm to jump out of the local optimum because of its stochastic and ergodic nature. Cubic mapping is a typical chaotic mapping, and its standard form is shown in the Eq. (5) [33].

$$x_{n+1} = bx_n^3 - cx_n \tag{5}$$

where  $b$  and  $c$  are chaotic impact factors.

When  $c \in (2.3, 3)$ , the sequence generated by Cubic mapping is the chaos sequence. Feng et al. analyzed the maximum Lyapunov exponent for 16 common chaotic mappings such as Cubic mapping and corrected the Cubic mapping expression [34]. The experimental results demonstrated that the chaos of Cubic mapping is similar to that of worm mouth mapping and tent mapping, and it is better than the one-dimensional mappings such as Sine mapping and Circle mapping. The ICSSOA uses the corrected Cubic mapping initialized population specific expression as shown in Eq. (6) [35].

$$x_{n+1} = \rho x_n (1 - x_n^2) \tag{6}$$

where,  $x_n \in (0, 1)$  and  $\rho$  is the control parameter.

**B. ADAPTIVE WEIGHTING FACTOR**

The producer performs global exploration as drastically as possible in the early iterations to quickly find the global optimal solution, so a larger inertia weight is needed in the early iterations to lengthen the global search range of the discoverer. At the same time, a smaller inertia weight is needed in the late iterations to improve the local exploitation capability of the discoverer for accelerating the convergence speed and avoiding falling into the local optimal solution. Therefore, fusing adaptive weights proposes a new improvement to the producer position update Eq. (2), and the producer position improvement equation is shown in the Eq. (7).

$$x_{i,j}^{t+1} = \begin{cases} \omega \cdot x_{i,j}^t \cdot \exp\left(\frac{-i}{\alpha \cdot iter_{max}}\right) & R_2 < ST \\ \omega \cdot x_{i,j}^t + Q \cdot L & R_2 \geq ST \end{cases} \tag{7}$$

The specific calculation of  $\omega$  is shown in Eq. (8) [36].

$$\omega = \begin{cases} \omega_0 & t \leq t_0 \\ \left(\frac{1}{t}\right)^{0.9} & t > t_0 \end{cases} \tag{8}$$

where,  $\omega_0$  is the given positive real number.  $t$  is the current number of iterations.  $t_0$  is the given number of iterations.

In the sparrow search process, the producer improves its global search range with a larger step size in the early iteration and improves its local exploitation capability with a progressively smaller step size in the late iteration.

**C. A HYBRID STRATEGY OF LEVY FLIGHT AND REVERSE LEARNING**

Levy flight is a class of non-Gaussian stochastic processes. The probability distribution of step length is a heavy-tailed distribution of random walks. For SI optimization algorithms prone to fall into the dilemma of local optimum, Levy flight

possesses the ability to let the algorithm jump out of the local optimum by a giant stride with a higher probability of occurrence in the random walk. The sparrow position update formula based on Levy flight is shown in Eq. (9) [37].

$$x_{newi}^{t+1} = x_i^t + \gamma \oplus Levy(\lambda) \quad (9)$$

where,  $\gamma$  is the step control parameter.  $Levy(\lambda)$  is a random path search and satisfies Eq. (10).

$$Levy \sim u = t^{-\lambda} \quad 1 < \lambda \leq 3 \quad (10)$$

The generation step is shown in Eq. (11).

$$S = \frac{\mu}{|\nu|^{\frac{1}{\beta}}} \quad 1 \leq \beta \leq 2 \quad (11)$$

where,  $\mu$  and  $\nu$  are a random number that follows a normal distribution.  $\mu \sim N(0, \delta_\mu^2)$ .  $\nu \sim N(0, \delta_\nu^2)$ .  $\delta_\nu = 1$ . The calculation formula of parameter  $\delta_\mu$  is shown in Eq. (12).

$$\delta_\mu = \left\{ \frac{\Gamma(1+\beta) \sin(\pi\beta/2)}{\Gamma[(1+\beta)/2] 2^{1-\beta} \sqrt{\beta}} \right\}^{\frac{1}{\beta}} \quad (12)$$

where,  $\beta$  usually takes the value of the constant 1.5.

Reverse learning is a method to find the corresponding reverse solution by the current solution and retain the better solution after evaluation, so reverse learning has the feature of finding the better solution and is often used by SI algorithms to jump out of the local optimum. The sparrow position update based on reverse learning is shown in Eq. (13).

$$\begin{aligned} x_{best}^{t+1} &= ub + r \oplus (lb - x_{best}(t)) \\ x_{newi}^{t+1} &= x_i^t + b \oplus (x_{best}(t) - x_{best}^t) \end{aligned} \quad (13)$$

where,  $x_{best}^t(t)$  represents the optimal inverse solution of the  $i_{th}$  sparrow at the  $t_{th}$  iteration.  $ub$  and  $lb$  are the upper and lower spatial boundaries, and  $r$  is a standard uniformly distributed random number.  $b$  represents the information exchange control parameter, and its calculation formula is shown in Eq. (14).

$$b = \left( 1 - \frac{t}{iter_{max}} \right)^t \quad (14)$$

To further improve the SSA search capability, a dynamic selection strategy is adopted to update the position of the sparrow based on the above two methods, and Levy flight and reverse learning are alternately used to update the position with a certain probability. In the Levy flight strategy, the step factor is used to expand the search range and jump out of the local optimal dilemma. Meanwhile, in the reverse learning strategy, the reverse solution is used to increase the diversity of solutions and improve the algorithm's search optimization performance.

The dynamic selection strategy approach is as follows.

When  $rand \in (0, 0.5)$ , choose Eq. (9) Levy flight strategy for sparrow position update. Otherwise, choose Eq. (13) reverse learning for sparrow position update. After that, the greedy rule is used to decide whether the old position needs to be replaced by the new one and enter the next generation

by comparing the magnitude of the adaptation values of the old and new positions. The greedy rule is shown in Eq. (15).

$$\begin{aligned} x_i'(t+1) &= \begin{cases} x_i(t+1) & f(x_i(t+1)) \leq f(x_{newi}(t+1)) \\ x_{newi}(t+1) & f(x_i(t+1)) > f(x_{newi}(t+1)) \end{cases} \end{aligned} \quad (15)$$

where,  $x_{newi}(t+1)$  represents the new sparrow generated by the hybrid strategy.  $f(x)$  represents the current adaptation value of the sparrow.

#### D. ICSSOA PROCESS

The proposed ICSSOA algorithm based on the above three improvement ideas consists of the following eight steps.

**Step 1:** Initialize the sparrow population according to the Eq. (6). Set each parameter such as objective function, population number  $N$ , set problem dimension  $D$ , maximum number of iterations, percentage of discoverer sparrows, and percentage of warning sparrows.

**Step 2:** Sort the sparrow population for fitness values to find the current best individual and the current worst individual.

**Step 3:** Discoverer position update. The adaptive inertia weight factor is obtained using the Eq. (8), and then the producer location is updated by the Eq. (7).

**Step 4:** Follower location update. The follower position is updated by the Eq. (3).

**Step 5:** Alert sparrow position update. A number of sparrows are randomly selected and the warning sparrow position is updated by Eq. (4).

**Step 6:** Dynamic strategy selection. The sparrow position is updated according to the random probability selection strategy. When  $rand \in (0, 0.5)$ , the Levy flight strategy is selected, the step size is obtained by the Eq. (11) and Eq. (12), and then each sparrow position is updated using the Eq. (9). Otherwise, the reverse learning strategy is selected, the information exchange parameters are obtained by the Eq. (14), and then each sparrow position is updated using the Eq. (13).

**Step 7:** Then positions before and after the update are compared using the greedy rule and the fitness value is calculated. The sparrow position with the better fitness value is retained.

**Step 8:** Termination condition. The termination condition is determined by determining whether the current number of iterations reaches the maximum number of iterations, and the optimal solution is output if the maximum number of iterations is reached. Otherwise, move to step 2 and add one to the current number of iterations.

From the above steps, the algorithm flow chart of the ICSSOA is shown in Figure 1.

#### E. ICSSOA TIME COMPLEXITY ANALYSIS

In SSA, the time magnitudes for population initialization and parameter setting are  $n$  and  $C$ .

In the producer location update phase, the top 20% of sparrows need to be selected as producers for location update by

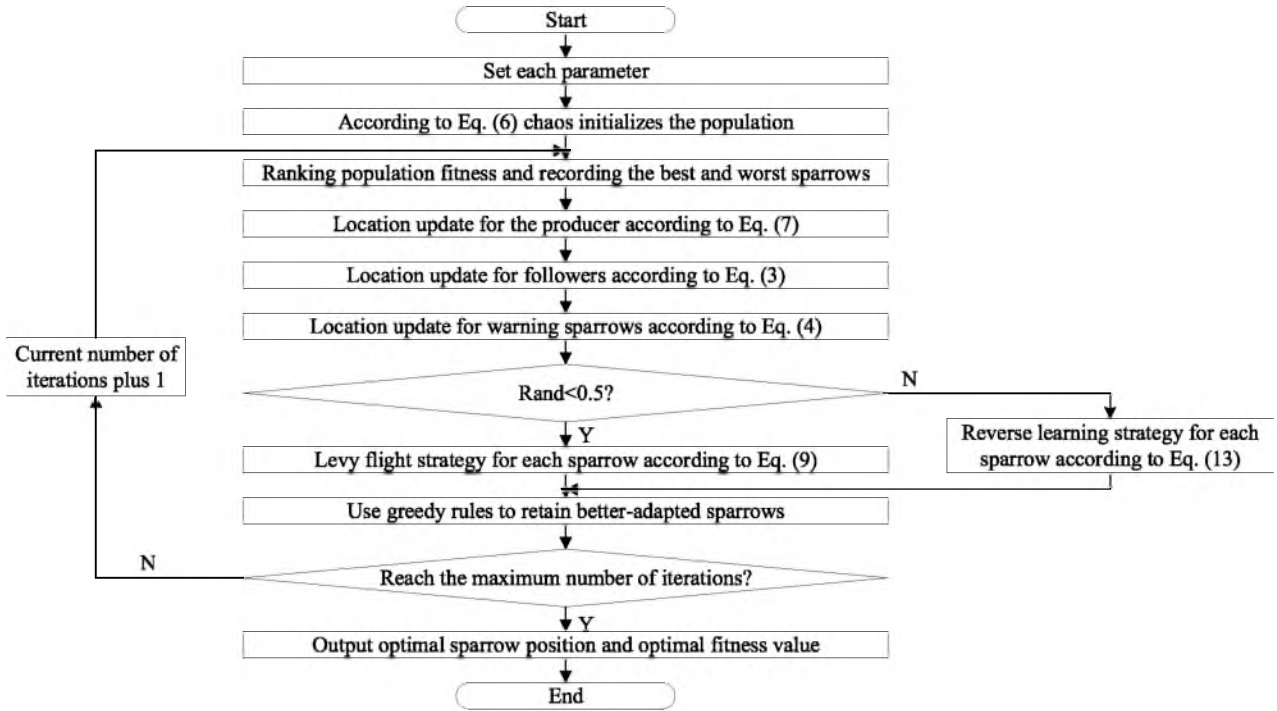


FIGURE 1. ICSSOA algorithm flow chart.

ranking the sparrow fitness and judging whether each dimension is beyond the set spatial range, if the total dimension is  $k$  dimensions, the sparrow fitness ranking and producer location update time magnitudes are  $n \times \log_2^n \times k$  and  $n \times k$ .

The follower position update phase requires the position update of the remaining sparrows as followers and the determination of whether each dimension of the individual is out of bounds, with a time scale of  $n \times k$ .

In the alert sparrow position update phase, some sparrows are randomly selected for position update and judgment is made on whether each dimension of an individual is out of bounds, with time magnitude  $n \times k$ .

In summary, the producer location update time magnitude is  $n \times \log_2^n \times k + n \times k$ . Both the follower location update time magnitude and the alert sparrow location update time magnitude are  $n \times k$ .

The time complexity of SSA is

$$O(n \times k + n \times \log_2^n \times k + n \times k + n \times k + n \times k) \approx O(n \times \log_2^n).$$

ICSSOA adds Cubic chaos mapping, adaptive weights for producer location updates, and hybrid policies to the standard SSA.

Chaos initialization uses chaos mapping for each sparrow for population initialization. The time magnitude is  $n \times k$ .

The adaptive weight judgment time magnitude is 1. The improved producer location update formula time magnitude is  $n \times \log_2^n \times k + n \times k$ .

The Levy flight step in the hybrid strategy calculates the time magnitude as a constant 1, then the time magnitude of Levy flight applied to each sparrow in the population is  $n \times k$ . The time magnitude for determining whether an individual is out of bounds for judgment is  $n \times k$ . The time magnitude of reverse learning to solve the reverse solution based on the current solution is 1. The time magnitude of reverse learning applied to each sparrow in the population is  $n \times k$ . The time magnitude for determining whether an individual is out of bounds for judgment is  $n \times k$ .

The time complexity of ICSSOA is

$$O(n \times k + n \times \log_2^n \times k + n \times k + n \times k + n \times k + n \times k + n \times k) \approx O(n \times \log_2^n).$$

#### IV. EXPERIMENT AND ANALYSIS

##### A. TEST FUNCTIONS

The experiments were conducted using a computer running in an environment such as ADM Ryzen 7 5800H @ 3.20GHz, 16G, Windows 10 operating system, and the implementation of the functional code using the programming language MATLABR2021b. The twenty benchmark functions used in use (F1-F11 for unimodal functions and F12-F20 for multimodal functions) are shown in Table 2. The unimodal function has only one local minima in the bounded interval, while the multimodal function has multiple local minima in the bounded interval. The unimodal function tests whether the algorithm can find the function's minimum value quickly, while the multimodal function considers whether the algorithm has good enough performance to jump out of the local optimum.

TABLE 2. List of test functions.

Number	Function expressions	Dim	Initial range	$f(x)_{min}$
F1	$f(x) = \sum_{i=1}^d x_i^2$	30,50,70	[-100,100]	0
F2	$f(x) = \sum_{i=1}^d  x_i  + \prod_{i=1}^d  x_i $	30,50,70	[-10,10]	0
F3	$f(x) = \sum_{i=1}^d \left( \sum_{j=1}^i x_j \right)^2$	30,50,70	[-100,100]	0
F4	$f(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	30,50,70	[-100,100]	0
F5	$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i)^2 + (x_i - 1)^2]$	30,50,70	[-30,30]	0
F6	$f(x) = \sum_{i=1}^d ([x_i + 0.5])^2$	30,50,70	[-100,100]	0
F7	$f(x) = \sum_{i=1}^d  x_i ^{i+1}$	30,50,70	[-1,1]	0
F8	$f(x) = \sum_{i=1}^d ix_i^2$	30,50,70	[-10,10]	0
F9	$f(x) = \sum_{i=1}^d x_i^2 + \left( \sum_{i=1}^d 0.5ix_i \right)^2 + \left( \sum_{i=1}^d 0.5ix_i \right)^4$	30,50,70	[-5,10]	0
F10	$f(x) = \sum_{i=1}^d ix_i^4 + random[0,1]$	30,50,70	[-1.28,1.28]	0
F11	$f(x) = \sum_{i=1}^{d/4} \left[ \begin{aligned} &(x_{4i-3} + 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 \\ &+ (x_{4i-2} - 2x_{4i-1})^4 + 10(x_{4i-3} - x_{4i})^4 \end{aligned} \right]$	30,50,70	[-4,5]	0
F12	$f(x) = \sin^2(\pi\omega_i)$ $+ \sum_{i=1}^{d-1} (\omega_i - 1)^2 [1 + 10\sin^2(\pi\omega_i + 1)] + (\omega_d - 1)^2 [1 + \sin^2(2\pi\omega_d)]$ $\omega_i = 1 + \frac{x_i - 1}{4}, i = 1, \dots, d$	30,50,70	[-10,10]	0
F13	$f(x) = 418.9829d - \sum_{i=1}^d x_i \sin(\sqrt{ x_i })$	30,50,70	[-500,500]	0
F14	$f(x) = \frac{\pi}{d} \left\{ 10\sin(\pi y_i) + \sum_{i=1}^d (y_i - 1)^2 [1 + 10\sin^2(\pi y_{i+1})] + (y_d - 1)^2 \right\}$ $y_i = 1 + \frac{x_i + 1}{4}$	30,50,70	[-50,50]	0
F15	$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$ $f(x) = 0.1 \left\{ \sin^2(3\pi x_i) + \sum_{i=1}^d (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] \right.$ $\left. + (x_i - 1)^2 [1 + \sin^2(2\pi x_i)] \right\} + \sum_{i=1}^d u(x_i, 5, 100, 4)$	30,50,70	[-50,50]	0
F16	$f(x) = 0.5 \sum_{i=1}^d (x_i^4 - 16x_i^2 + 5x_i)$	30,50,70	[-5,5]	-39.165d

TABLE 2. (Continued.) List of test functions.

F17	$f(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2}\right) - \exp\left(\frac{1}{d} \sum_{i=1}^d \cos(2\pi x_i)\right) + 20 + e$	30,50,70	[-32,32]	0
F18	$f(x) = \sum_{i=1}^d \frac{x_i^2}{4000} - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	30,50,70	[-600,600]	0
F19	$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$	30,50,70	[-5,5]	0
F20	$f(x) = \sum_{i=1}^d -x_i \sin(\sqrt{ x_i })$	30,50,70	[-500,500]	-418.98d

(PS: All test functions were derived from <http://www.sfu.ca/~ssurjano/>)

TABLE 3. List of test functions.

Name	Parameter
SSA	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2$
CSSOA	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2$
ISSA(a)	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2, \theta = 0.05$
CASSA	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2$
ISSA(b)	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2, v_{max} = 0.9, v_{min} = -0.3, \delta = 0.5$
ICSSOA	$N = 30, PD = 0.2, ST = 0.8, SD = 0.2, \alpha_0 = 1.5, t_0 = 125, \rho = 2.595, \beta = 1.5$
GWO	$N = 30, \alpha_{first} = 2, \alpha_{final} = 0$
PSO	$N = 30, c_1 = c_2 = 2, w_{max} = 0.9, w_{min} = 0.2, v_{min} = -v_{max}$
WOA	$N = 30, \alpha_{first} = 2, \alpha_{final} = 0, p = 0.5, b = 1$
ACO	$N = 30, Rho = 0.9, P0 = 0.2, step = 0.05$

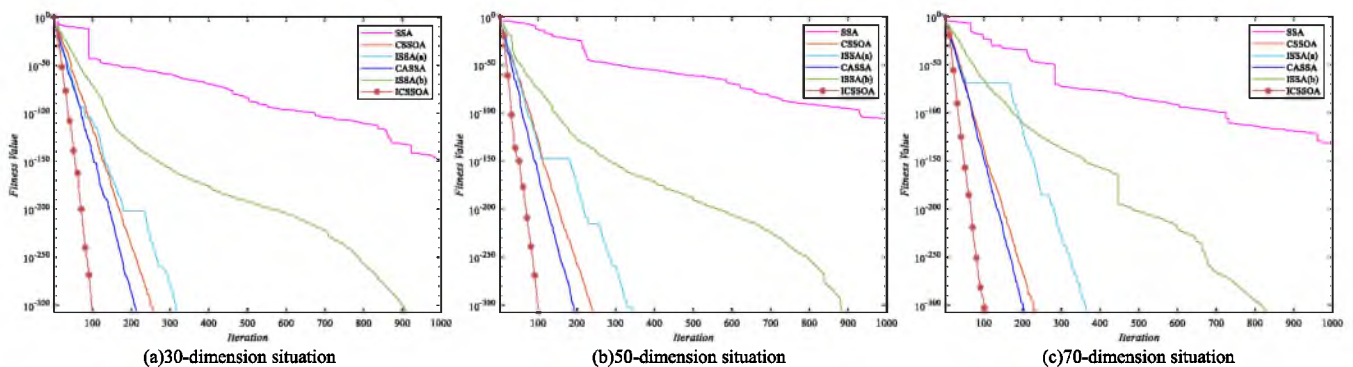


FIGURE 2. Performance comparison on F1.

**B. TEST FUNCTIONS EXPERIMENTAL RESULTS AND ANALYSIS**

To verify the performance of ICSSOA, the SSA and the four SSA variants were compared longitudinally based on the 20 benchmark functions in Table 2. The five SSAs include the SSA, CSSOA, ISSA(a), CASSA, and ISSA(b). Cross-sectional comparison of four different SI optimization algorithms. The four SI algorithms include GWO, PSO, WOA, and ACO. Each algorithm was run 30 times independently,

with 1000 iterations per run, and the parameters were set as shown in Table 3.

Figures 2 to 16 record the convergence plots of each algorithm while optimizing the benchmark functions F1 to F15. Functions F16 to F20 can be algorithmically judged based on the accuracy of the solutions in Table 4 because the algorithm converges too quickly. Moreover, six folded including purple solid lines, orange solid lines, cyan solid lines, blue solid lines, green solid lines, and brick red asterisk solid lines,



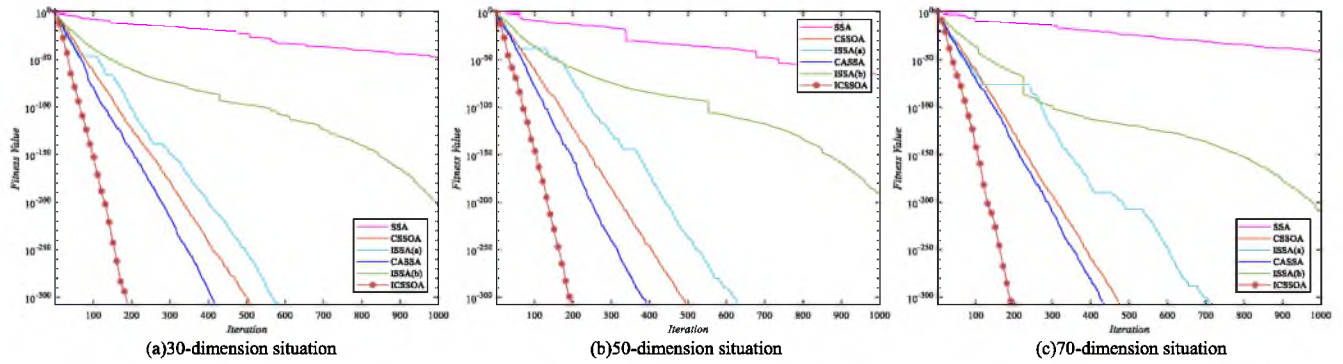


FIGURE 3. Performance comparison on F2.

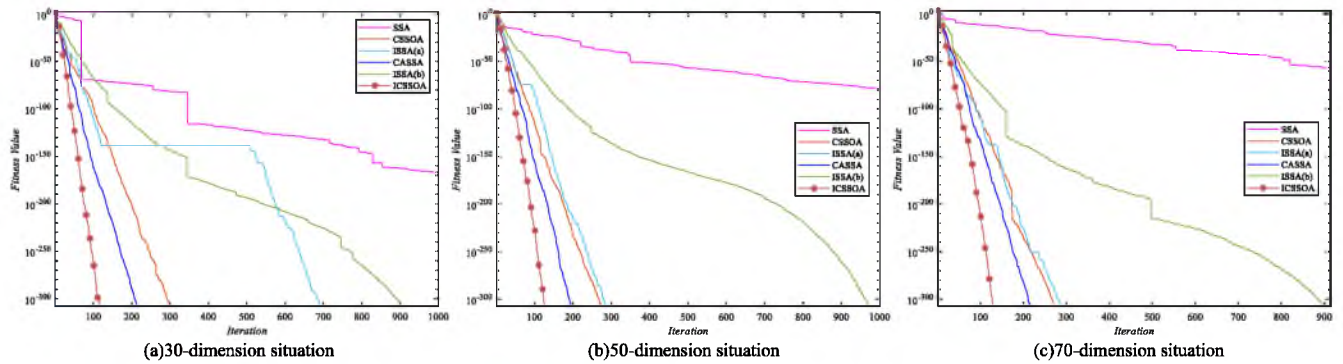


FIGURE 4. Performance comparison on F3.

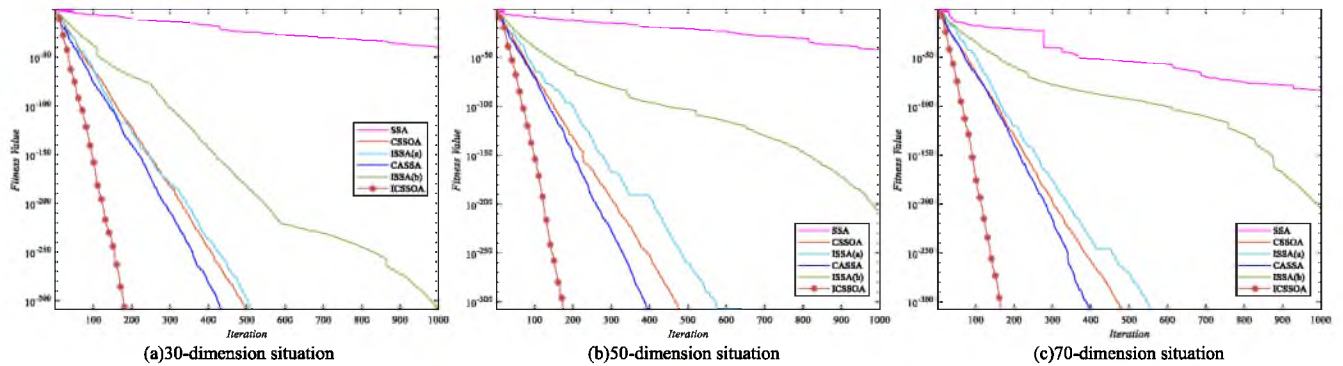


FIGURE 5. Performance comparison on F4.

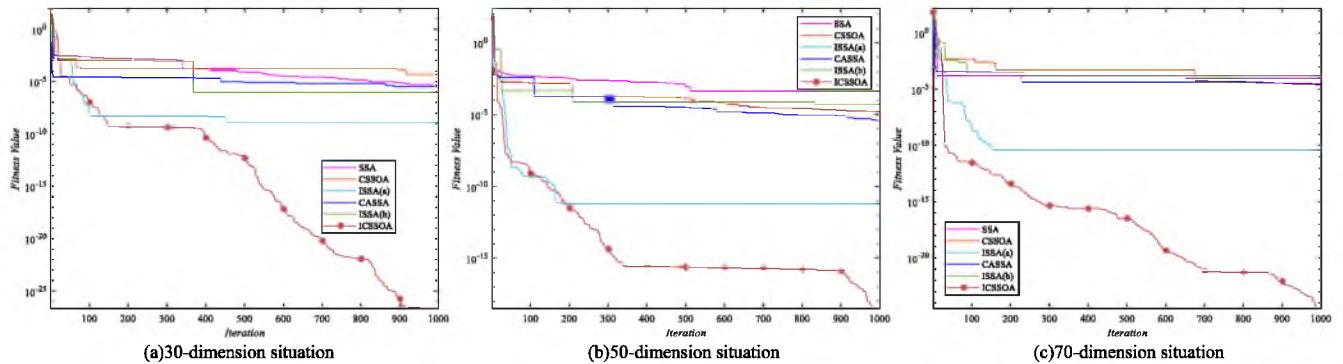


FIGURE 6. Performance comparison on F5.

represent the SSA, CSSOA, ISSA(a), CASSA, ISSA(b), and ICSSOA, respectively in the longitudinal comparison experimental results, respectively.

Table 4 uses mean and variance statistics to further analyze the performance of the six algorithms in the longitudinal experiment. The results of the algorithms are obtained

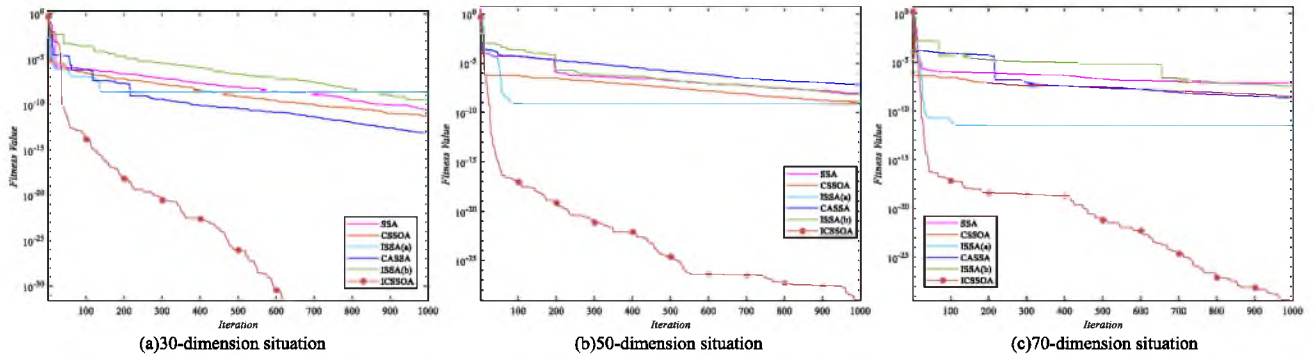


FIGURE 7. Performance comparison on F6.

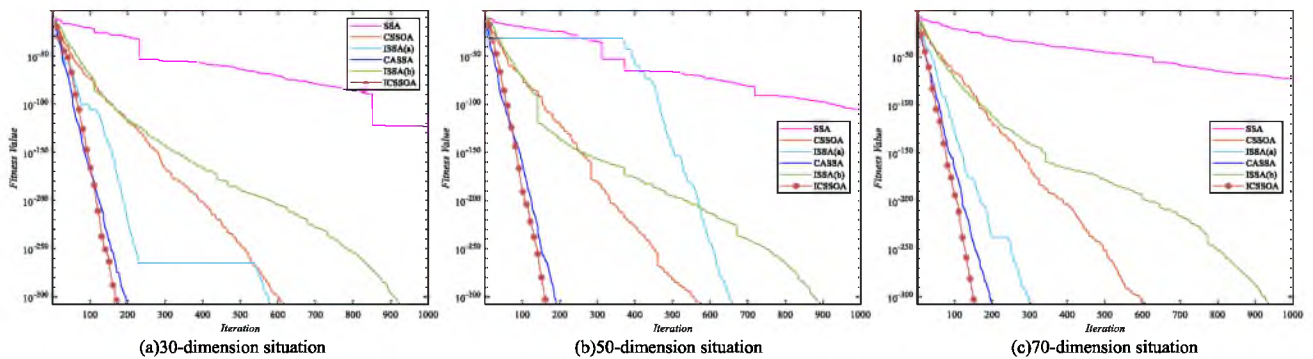


FIGURE 8. Performance comparison on F7.

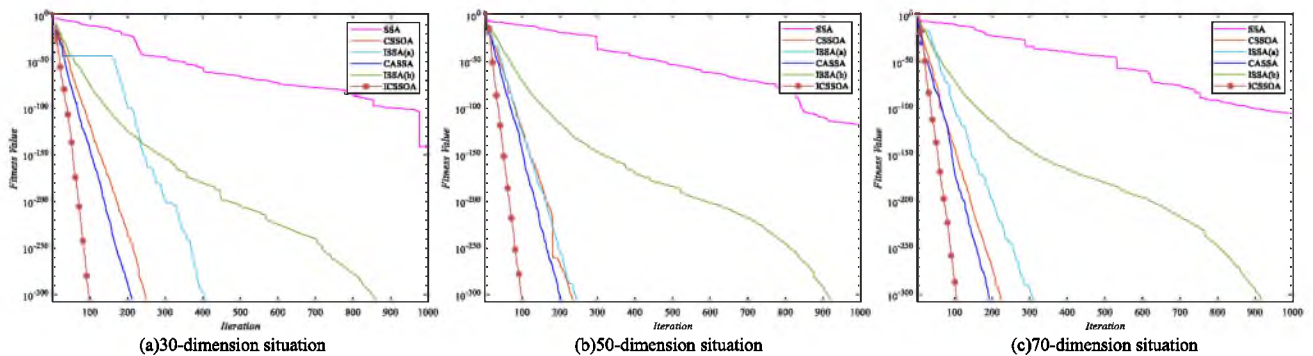


FIGURE 9. Performance comparison on F8.

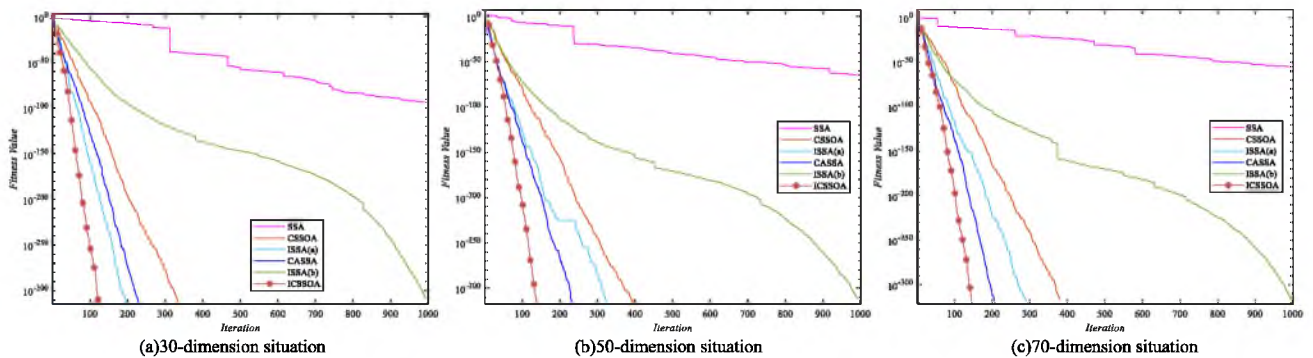


FIGURE 10. Performance comparison on F9.

by running each algorithm 30 times independently in 30, 50, and 70 dimensions. The related results are recorded in Table 4, and the bolded font shows the optimal values in the algorithms.

Figures 17 to 31 record the convergence plots of each algorithm while optimizing the benchmark functions F1 to F15. Functions F16 to F20 can be algorithmically judged based on the accuracy of the solutions in

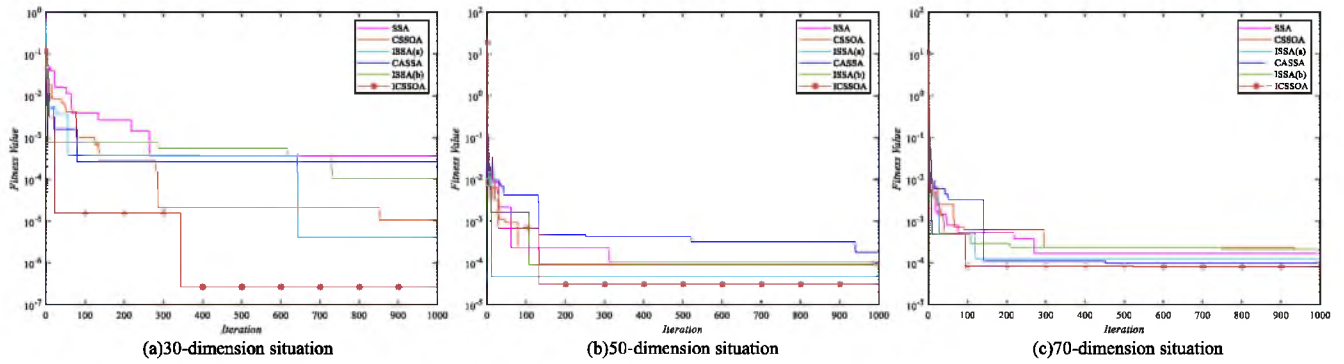


FIGURE 11. Performance comparison on F10.

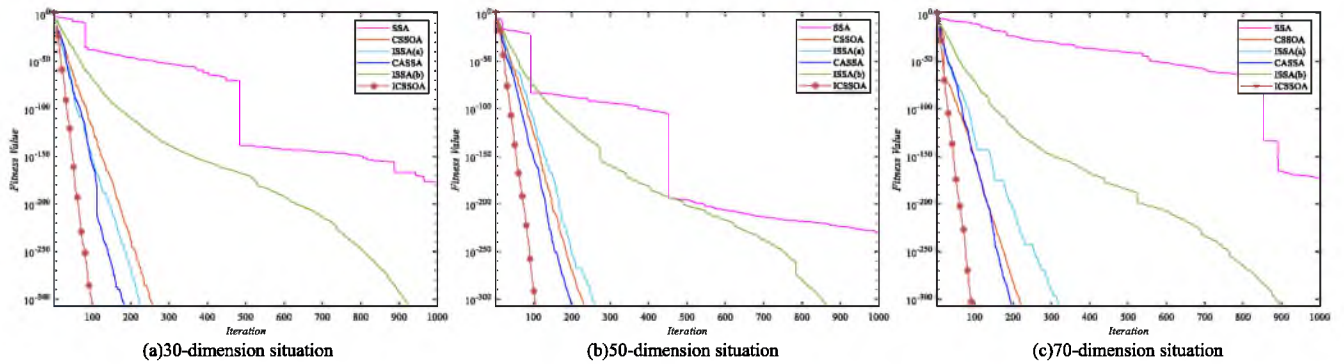


FIGURE 12. Performance comparison on F11.

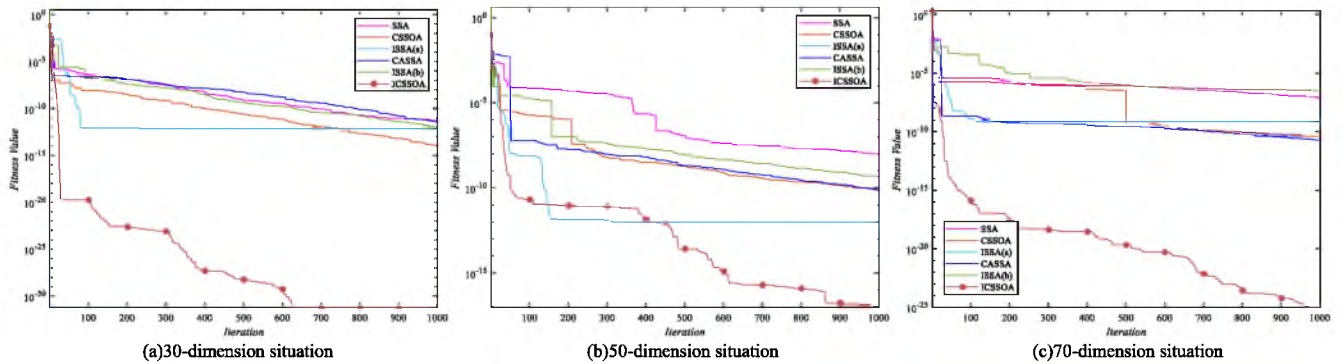


FIGURE 13. Performance comparison on F12.

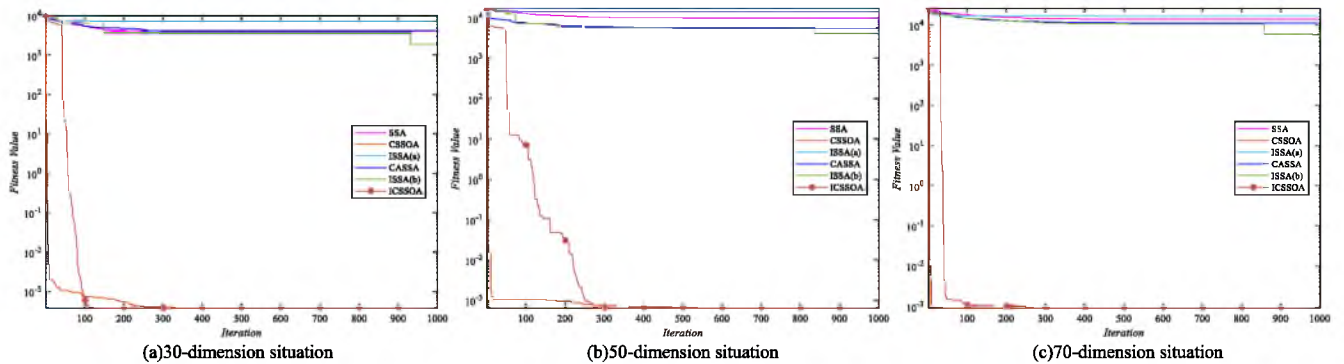


FIGURE 14. Performance comparison on F13.

Table 5 because the algorithm converges too quickly. Moreover, five folded including red asterisk solid line, purple solid line, orange solid line, cyan solid line, and green solid

line, represent the ICSSOA, GWO, PSO, WOA, and ACO algorithms in the horizontal comparison experimental results, respectively.

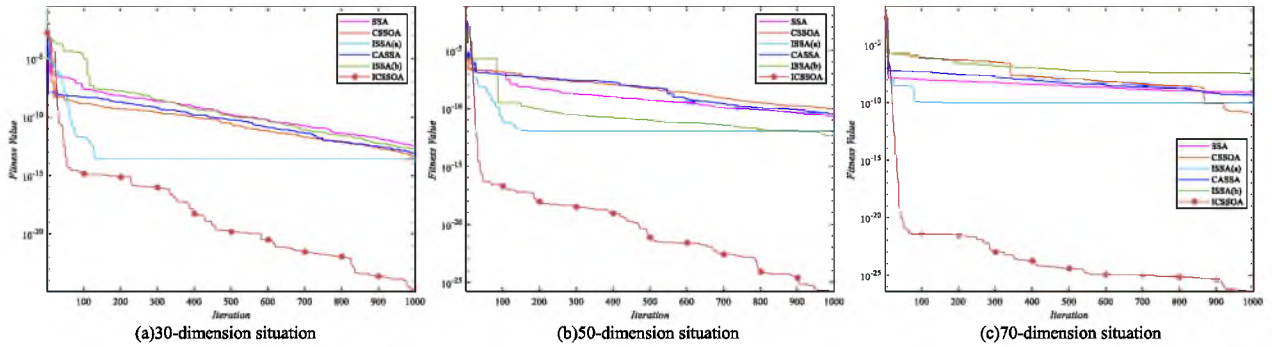


FIGURE 15. Performance comparison on F14.

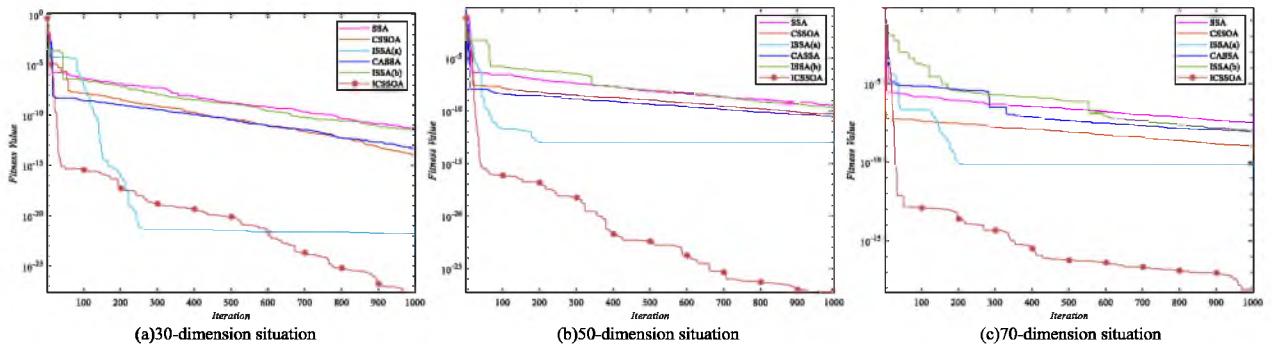


FIGURE 16. Performance comparison on F15.

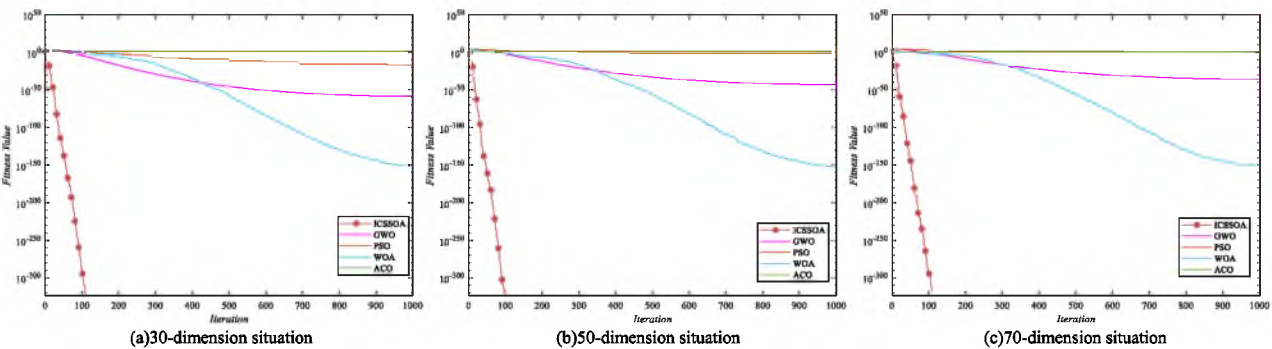


FIGURE 17. Performance comparison on F1.

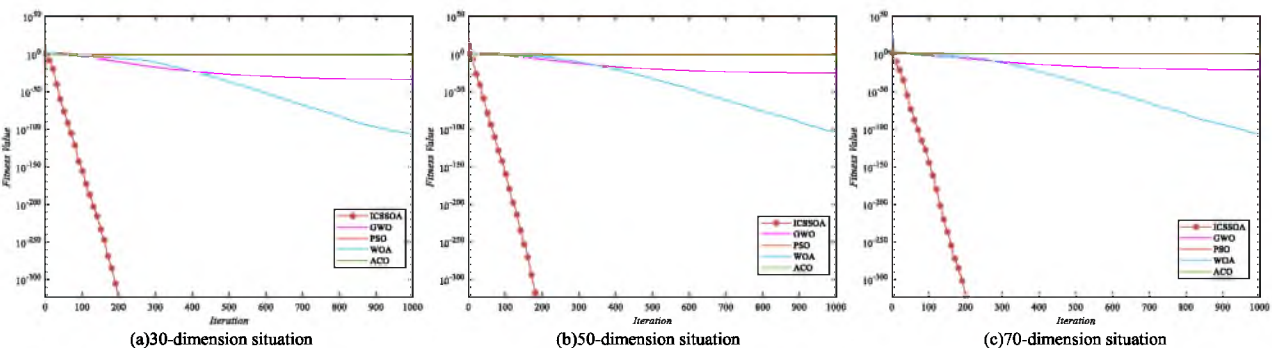


FIGURE 18. Performance comparison on F2.

Table 5 uses mean and variance statistics to further analyze the performance of the five algorithms in the horizontally experiments. The results of the algorithms are obtained by running each algorithm 30 times independently in 30, 50, and 70 dimensions. The related results are recorded in

Table 5, and the bolded font shows the optimal values in the algorithms.

Longitudinal experiments are compared using the standard SSA algorithm and four SSA variants for 20 test functions, which were run 30 times under the same test environment,

TABLE 4. Longitudinal experimental results of fixed dimensions.

Dim	SSA	CSSOA	ISSA(a)	CASSA	ISSA(b)	ICSSOA
	MEAN±SD	MEAN±SD	MEAN±SD	MEAN±SD	MEAN±SD	MEAN±SD
F1	30	9.63E-70±4.32E-139	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	9.63E-90±1.55E-177	0.00E+00±0.00E+00	8.73E-120±3.81E-161	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	2.63E-75±4.87E-148	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F2	30	3.81E-44±1.94E-86	0.00E+00±0.00E+00	5.77E-119±1.00E-235	0.00E+00±0.00E+00	4.57E-182±0.00E+00
	50	1.57E-38±1.17E-74	0.00E+00±0.00E+00	4.61E-49±1.06E-95	0.00E+00±0.00E+00	1.30E-187±0.00E+00
	70	3.79E-37±1.00E-71	0.00E+00±0.00E+00	1.25E-50±1.10E-38	0.00E+00±0.00E+00	7.43E-182±0.00E+00
F3	30	1.34E-62±5.44E-123	0.00E+00±0.00E+00	4.44E-267±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	3.41E-58±5.81E-114	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	9.03E-61±5.71E-119	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F4	30	2.21E-41±7.91E-81	0.00E+00±0.00E+00	3.54E-319±0.00E+00	0.00E+00±0.00E+00	2.18E-188±0.00E+00
	50	1.16E-37±6.77E-73	0.00E+00±0.00E+00	1.86E-296±0.00E+00	0.00E+00±0.00E+00	1.33E-180±0.00E+00
	70	1.32E-37±1.22E-72	0.00E+00±0.00E+00	4.13E-57±1.19E-17	0.00E+00±0.00E+00	9.60E-189±0.00E+00
F5	30	9.12E-06±3.55E-10	6.08E-07±2.53E-10	1.79E-09±3.86E-17	5.71E-07±2.95E-10	6.76E-06±6.09E-11
	50	2.08E-05±2.06E-09	7.58E-06±1.39E-10	2.26E-07±2.35E-12	6.11E-06±1.97E-10	6.49E-06±3.22E-08
	70	2.19E-05±4.40E-09	8.77E-06±5.79E-10	2.41E-06±2.83E-10	1.89E-05±6.49E-09	3.22E-06±6.60E-09
F6	30	1.32E-11±6.14E-22	4.56E-12±1.45E-22	6.52E-09±2.81E-16	9.64E-12±5.48E-22	5.92E-11±1.11E-20
	50	1.23E-08±1.87E-15	1.11E-09±6.45E-18	1.71E-09±1.43E-14	2.79E-09±9.28E-17	3.59E-09±1.32E-14
	70	2.04E-08±2.99E-15	1.23E-08±3.52E-15	1.18E-08±1.60E-14	1.31E-08±3.47E-15	2.25E-07±6.74E-13
F7	30	1.15E-77±3.99E-153	0.00E+00±0.00E+00	1.19E-265±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	1.41E-86±6.67E-171	0.00E+00±0.00E+00	8.29E-173±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	5.03E-76±1.62E-149	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F8	30	9.83E-83±2.89E-163	0.00E+00±0.00E+00	1.38E-175±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	8.80E-72±3.87E-141	0.00E+00±0.00E+00	7.49E-78±2.81E-153	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	4.54E-69±1.44E-135	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F9	30	2.42E-44±1.75E-86	0.00E+00±0.00E+00	4.62E-74±6.42E-146	0.00E+00±0.00E+00	1.91E-307±0.00E+00
	50	7.72E-46±2.95E-89	0.00E+00±0.00E+00	4.26E-135±9.08E-268	0.00E+00±0.00E+00	4.18E-292±0.00E+00
	70	5.70E-34±2.11E-65	0.00E+00±0.00E+00	1.61E-76±1.82E-150	0.00E+00±0.00E+00	2.78E-294±0.00E+00
F10	30	1.79E-04±3.49E-08	1.05E-04±5.03E-09	8.86E-05±5.43E-09	2.15E-04±3.24E-08	1.06E-04±8.64E-09
	50	2.36E-04±1.86E-07	7.19E-05±1.18E-08	4.53E-05±5.30E-09	9.29E-05±1.69E-08	9.70E-05±1.67E-08
	70	2.29E-04±3.63E-08	1.10E-04±1.04E-08	9.27E-05±6.11E-09	1.37E-04±1.74E-08	1.23E-04±9.66E-09
F11	30	6.55E-57±9.22E-112	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	3.28E-318±0.00E+00
	50	6.80E-65±2.30E-127	0.00E+00±0.00E+00	5.25E-320±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	1.42E-77±1.42E-152	0.00E+00±0.00E+00	1.30E-168±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F12	30	9.50E-12±2.35E-22	2.27E-12±1.88E-23	1.23E-12±3.02E-15	2.88E-12±5.01E-21	1.03E-12±1.23E-19
	50	3.66E-09±1.56E-16	1.36E-09±1.05E-17	4.88E-10±4.09E-18	3.52E-09±2.03E-16	6.49E-10±1.94E-16
	70	1.87E-08±3.78E-15	9.90E-09±8.50E-16	2.34E-10±3.35E-18	1.03E-08±1.16E-15	9.80E-09±9.14E-14
F13	30	4.68E+03±5.00E+03	<b>3.8E-04±2.26E-16</b>	4.449E+03±3.80E+06	3.26E+03±1.33E+06	2.07E+03±3.52E+06
	50	4.89E+03±1.68E+07	<b>4.47E-04±3.38E-07</b>	3.84E+03±3.58E+07	3.40E+03±1.23E+07	2.07E+03±9.31E+06
	70	4.89E+03±3.28E+07	<b>9.48E-04±1.39E-05</b>	6.88E+03±7.00E+07	3.47E+03±2.11E+07	2.42E+03±1.60E+07
F14	30	7.65E-13±1.80E-24	1.69E-13±1.61E-25	7.77E-10±1.42E-17	1.37E-13±6.36E-25	8.06E-13±1.55E-24
	50	3.59E-11±7.29E-21	5.49E-12±2.43E-20	2.07E-12±1.67E-18	9.41E-12±6.24E-20	8.84E-12±2.45E-20
	70	8.30E-10±2.16E-17	1.32E-10±1.71E-19	8.22E-11±1.45E-19	6.72E-11±1.01E-19	8.23E-10±1.65E-17
F15	30	1.80E-11±6.56E-21	4.26E-12±2.77E-22	1.31E-12±2.43E-17	1.89E-12±9.81E-24	7.91E-11±6.25E-20
	50	1.58E-09±2.59E-17	6.36E-10±3.60E-18	1.23E-10±4.06E-19	2.08E-10±4.75E-17	1.33E-10±1.04E-15
	70	7.06E-09±4.27E-16	3.91E-09±2.36E-16	1.69E-09±5.14E-17	2.60E-09±4.68E-17	2.85E-09±4.55E-15
F16	30	-1.17E+03±2.44E-14	-1.17E+03±4.66E-15	-1.17E+03±1.40E-05	-1.17E+03±2.52E-15	-1.17E+03±7.00E-14
	50	-1.17E+03±9.39E+05	-1.17E+03±9.39E+05	-1.17E+03±9.39E+05	-1.17E+03±9.39E+05	-1.17E+03±9.39E+05
	70	-1.17E+03±1.86E+06	-1.17E+03±1.86E+06	-1.17E+03±1.86E+06	-1.17E+03±1.86E+06	-1.17E+03±1.86E+06
F17	30	8.88E-16±0.00E+00	8.88E-16±0.00E+00	8.88E-16±0.00E+00	8.88E-16±0.00E+00	8.88E-16±0.00E+00
	50	5.32E-16±1.93E-31	5.32E-16±1.93E-31	5.32E-16±1.93E-31	5.32E-16±1.93E-31	5.32E-16±1.93E-31
	70	3.80E-16±1.95E-31	3.80E-16±1.95E-31	3.80E-16±1.95E-31	3.80E-16±1.95E-31	3.80E-16±1.95E-31
F18	30	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F19	30	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	50	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
	70	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00	0.00E+00±0.00E+00
F20	30	-8.12E+03±2.74E+05	-1.25E+04±2.99E-16	-9.54E+03±3.38E+06	-8.99E+03±1.57E+06	-1.04E+04±3.39E+06
	50	-4.70E+03±4.08E+07	-1.25E+04±1.07E+07	-8.57E+03±2.55E+07	-8.86E+03±5.61E+07	-1.04E+04±8.06E+07
	70	-7.52E+03±7.70E+07	-1.25E+04±2.13E+07	-8.58E+03±5.45E+07	-9.52E+03±1.30E+07	-1.02E+05±1.48E+07

respectively, and the specific results are analyzed and illustrated in Figures 2 and 16 and Table 4. The results further indicate that the ICSSOA is much faster and more stable

than the other four algorithms in solving the F1-F4 function test functions. It can be seen from Figures 6 and 7 that the other five algorithms all fall into the problem of local

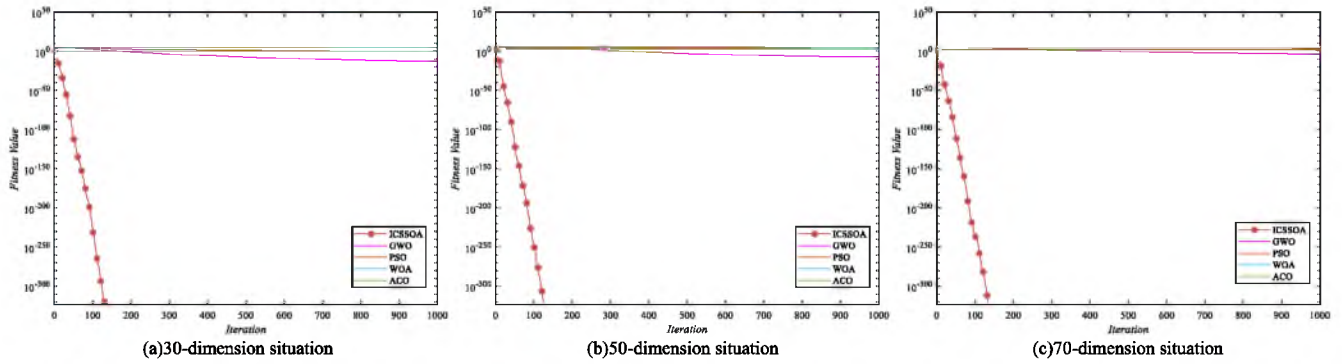


FIGURE 19. Performance comparison on F3.

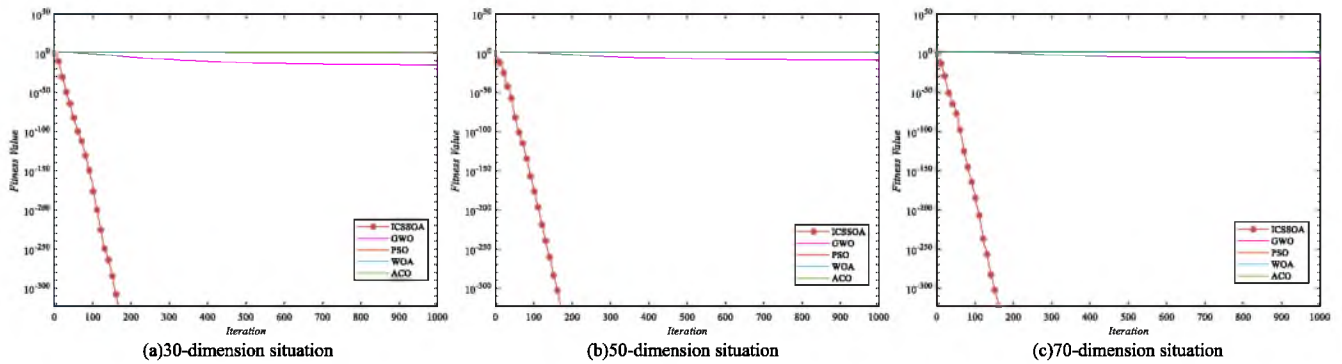


FIGURE 20. Performance comparison on F4.

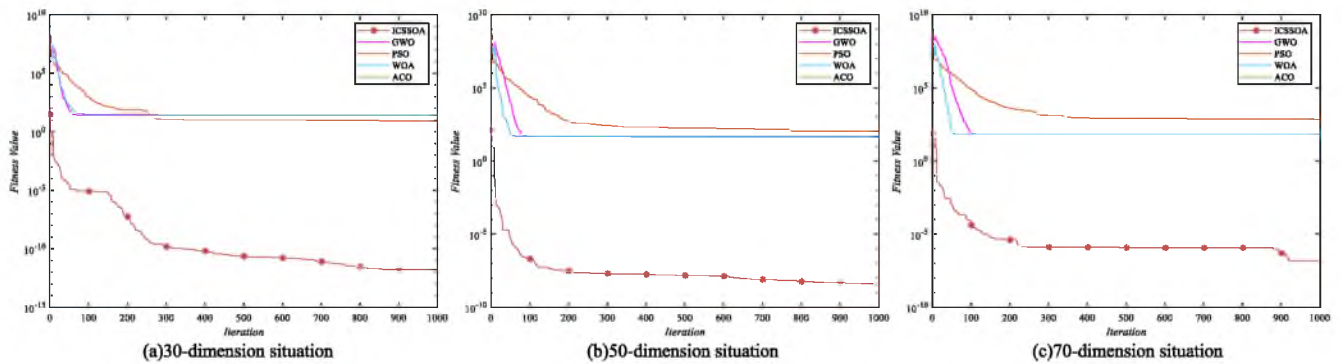


FIGURE 21. Performance comparison on F5.

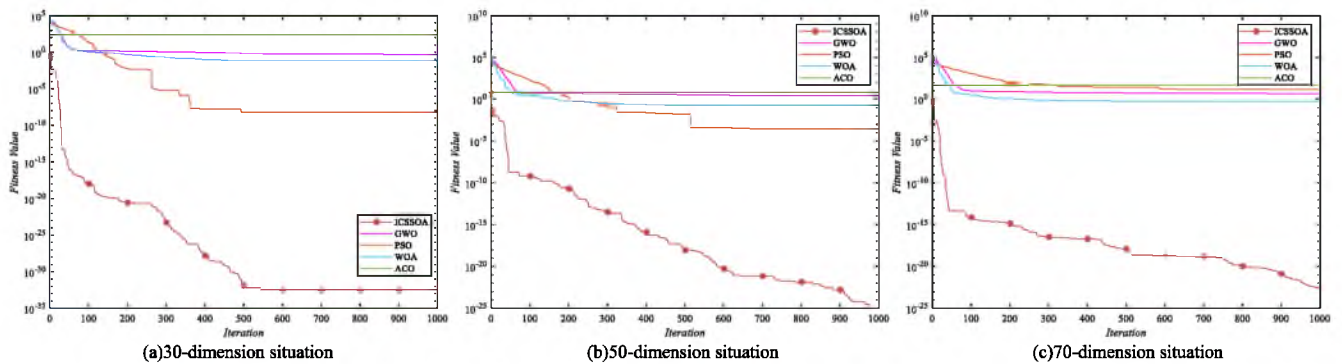


FIGURE 22. Performance comparison on F6.

optimum in the process of solving F5-F6 functions, but the ICSSOA algorithm can better jump out of the local optimum dilemma through the mixed strategy perturbation, and the

solution accuracy is better than the other algorithms. In the F7-F9, the ICSSOA outperforms the other five algorithms both in speed and stability, and in the low-dimensional F10,

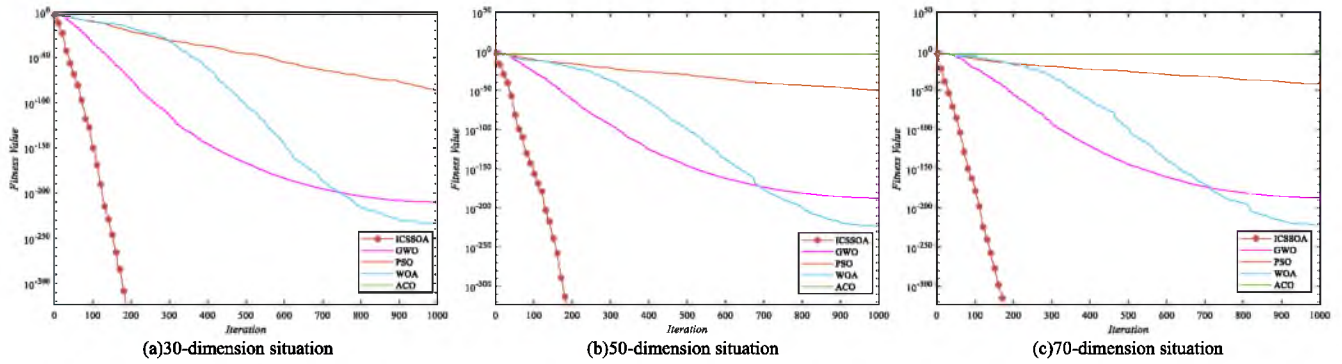


FIGURE 23. Performance comparison on F7.

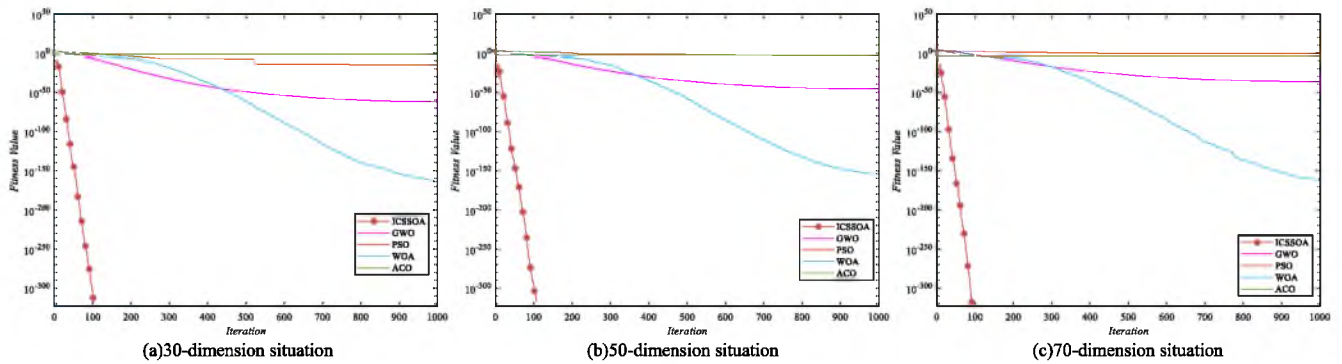


FIGURE 24. Performance comparison on F8.

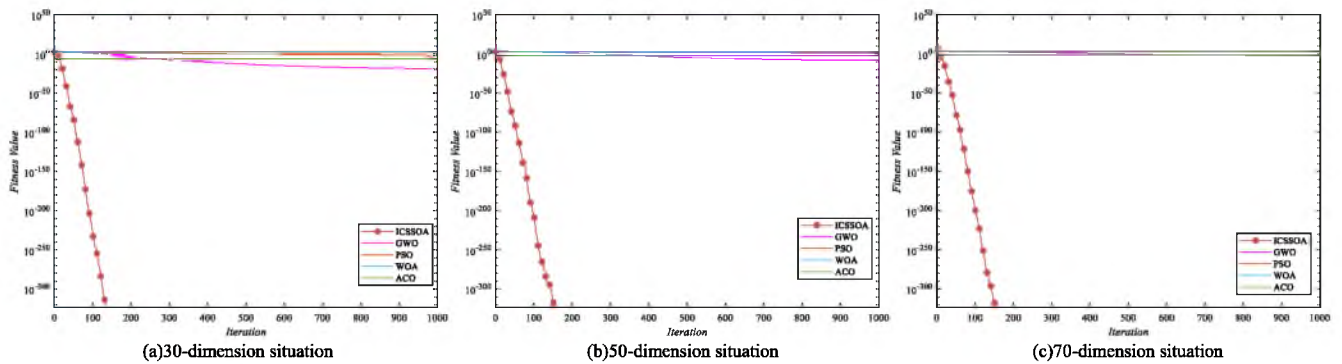


FIGURE 25. Performance comparison on F9.

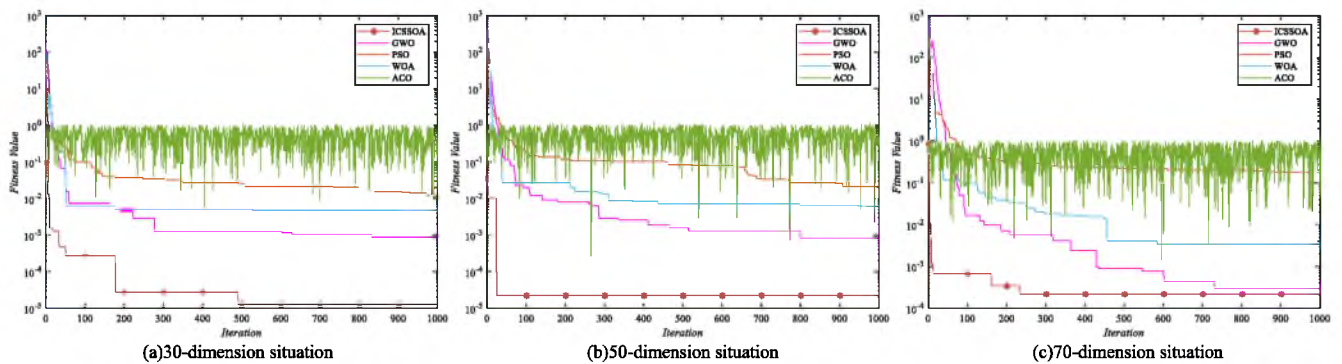


FIGURE 26. Performance comparison on F10.

the ICSSOA also shows better convergence accuracy than other algorithms, but in the high dimension, the ICSSOA's advantage is not obvious. The performance of ICSSOA is

slightly lower than that of CSSOA in the F13. In the F14-F20 function test, the convergence accuracy of ICSSOA is better than that of the other five algorithms. The ICSSOA has a

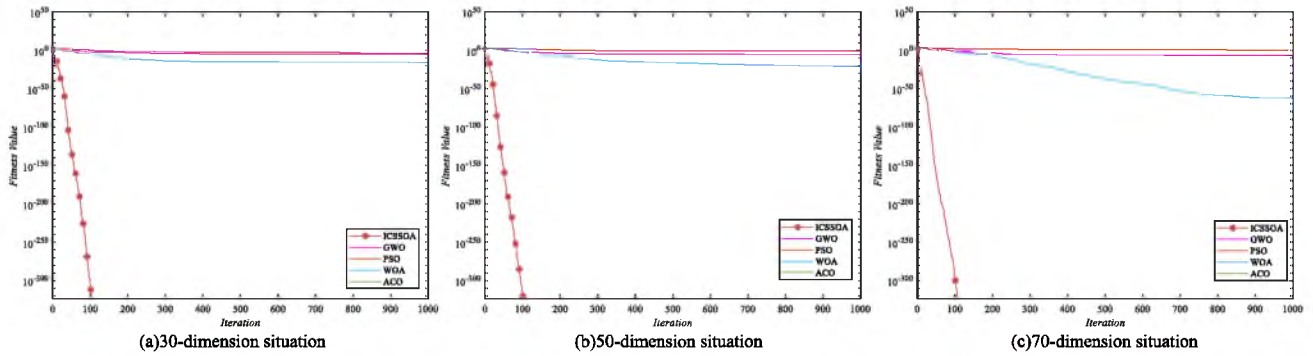


FIGURE 27. Performance comparison on F11.

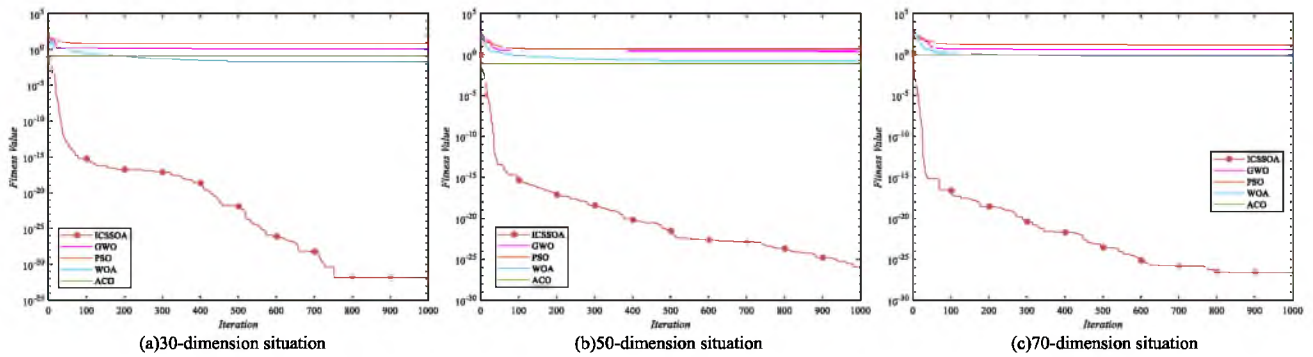


FIGURE 28. Performance comparison on F12.

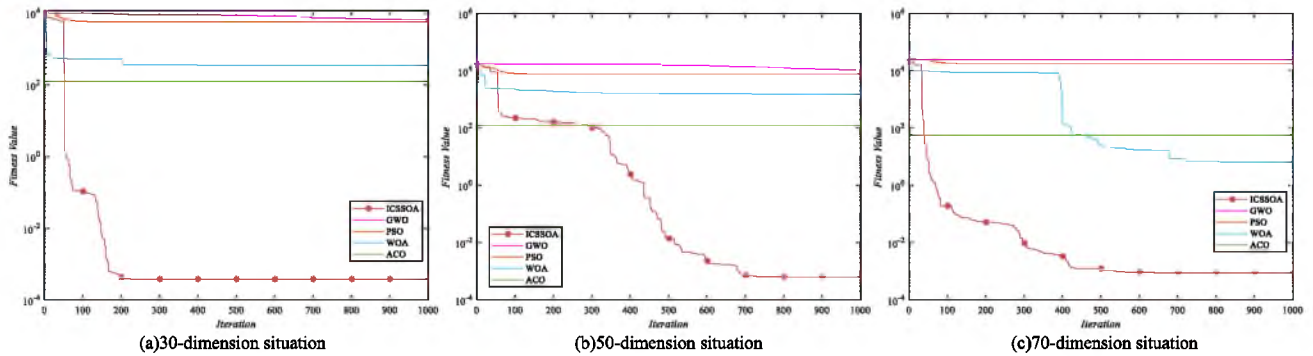


FIGURE 29. Performance comparison on F13.

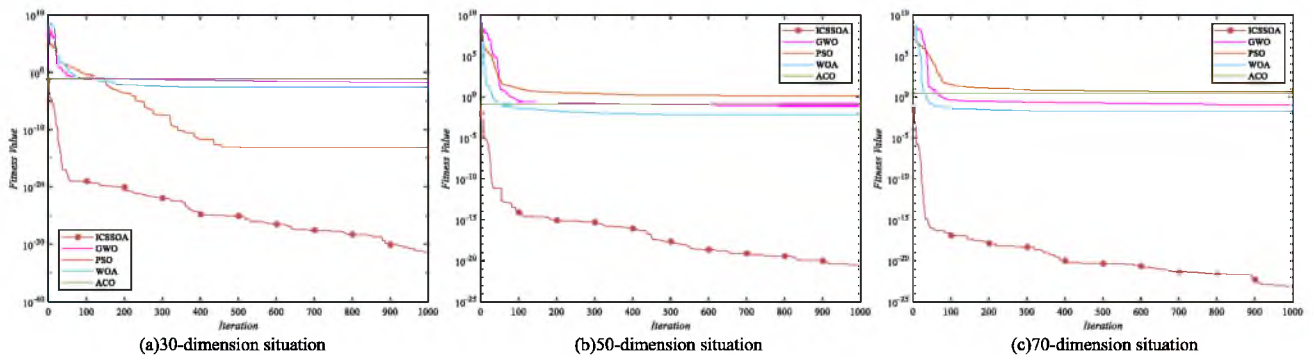


FIGURE 30. Performance comparison on F14.

strong adaptive capability in optimizing single-peak functions or multi-peak functions and can use hybrid strategies to make the algorithm jump out of the current local optimal solution's dilemma to obtain higher convergence accuracy.

The cross-sectional experiments are compared using four different SI algorithms for 20 test functions, which are run 30 times under the same test environment, respectively, and the specific results are analyzed and obtained in



TABLE 5. Horizontally experimental results of fixed dimensions.

Dim	ICSSOA	GWO	PSO	WOA	ACO	
	MEAN±SD	MEAN±SD	MEAN±SD	MEAN±SD	MEAN±SD	
F1	30	0.00E+00±0.00E+00	9.21E-59±2.70E-116	1.77E-05±9.13E-09	1.41E-147±5.69E-293	3.11E+01±2.11E+03
	50	0.00E+00±0.00E+00	5.21E-44±1.79E-86	7.64E-02±4.64E-02	3.45E-147±5.97E-292	2.60E+01±2.63E+03
	70	0.00E+00±0.00E+00	1.91E-36±1.88E-71	6.44E+00±3.22E+02	5.93E-146±2.13E-28	1.50E+01±7.76E+02
F2	30	0.00E+00±0.00E+00	1.20E-34±3.04E-68	7.63E-02±1.14E-02	1.00E-102±1.50E-203	7.39E-01±6.36E-01
	50	0.00E+00±0.00E+00	1.79E-26±3.97E-52	8.46E-01±1.01E+00	1.44E-104±7.94E-207	8.98E-01±3.90E-01
	70	0.00E+00±0.00E+00	3.08E-22±2.03E-43	2.01E+00±7.72E+00	2.86E-99±5.74E-196	2.01E+01±9.15E+02
F3	30	0.00E+00±0.00E+00	2.34E-14±6.18E-27	7.70E+00±1.79E+02	2.01E+04±1.77E+08	1.67E+01±2.82E+02
	50	0.00E+00±0.00E+00	1.04E-06±8.36E-12	5.14E+02±3.33E+05	7.72E+04±4.78E+09	1.64E+01±6.14E+02
	70	0.00E+00±0.00E+00	1.07E-02±2.25E-03	1.07E-02±2.25E-03	1.42E+05±2.87E+10	2.01E+01±9.15E+02
F4	30	0.00E+00±0.00E+00	2.03E-14±1.05E-27	5.76E-01±1.06E-01	4.63E+01±9.72E+02	3.86E+00±9.78E+00
	50	0.00E+00±0.00E+00	1.15E-09±8.91E-18	3.50E+00±8.91E+00	4.05E+01±1.51E-03	5.08E+00±2.36E+01
	70	0.00E+00±0.00E+00	2.88E-06±1.85E-10	4.76E+00±3.22E+01	2.96E+01±1.44E-03	4.25E+00±1.20E+00
F5	30	1.22E-10±4.41E-19	2.64E+01±7.21E-01	3.79E+01±6.59E+02	2.70E+01±1.76E-01	2.43E+02±1.44E+03
	50	1.82E-08±1.63E-14	2.83E+01±5.46E+02	1.14E+02±2.99E+04	2.85E+01±5.52E+02	4.02E+02±2.36E+03
	70	2.27E-06±3.40E-10	2.88E+01±1.12E+03	2.32E+02±1.00E+05	2.89E+01±1.13E+03	4.80E+02±4.08E+03
F6	30	6.16E-34±3.53E-66	6.51E-01±2.12E-01	4.45E-07±4.55E-12	5.69E-02±4.49E-03	3.07E+01±2.59E+03
	50	9.88E-19±3.98E-35	1.42E+00±1.59E+00	8.69E-02±9.26E-02	2.51E-01±8.14E-02	2.45E+00±2.48E+03
	70	4.35E-17±1.19E-31	2.05E+00±5.86E+00	4.55E+00±1.35E+02	3.87E-01±2.83E-01	1.62E+01±4.10E+02
F7	30	0.00E+00±0.00E+00	5.92E-194±0.00E+00	1.91E-83±6.46E-165	4.69E-215±0.00E+00	1.76E-03±7.33E-06
	50	0.00E+00±0.00E+00	5.69E-187±0.00E+00	1.00E-47±5.00E-93	1.83E-216±0.00E+00	2.27E-03±1.50E-05
	70	0.00E+00±0.00E+00	2.02E-172±0.00E+00	1.90E-42±1.91E-82	9.12E-214±0.00E+00	1.85E-03±8.29E-06
F8	30	0.00E+00±0.00E+00	8.09E-61±1.56E-120	8.17E-09±1.61E-15	1.61E-151±7.70E-301	8.51E-02±2.81E-02
	50	0.00E+00±0.00E+00	2.85E-45±3.71E-89	6.24E-02±4.88E-02	6.64E-150±2.20E-297	6.62E-02±7.02E-03
	70	0.00E+00±0.00E+00	2.61E-37±3.74E-73	4.61E-01±1.86E+00	4.36E-148±1.32E-293	7.08E-02±7.17E-03
F9	30	0.00E+00±0.00E+00	3.14E-19±4.10E-37	5.55E-02±1.80E-02	4.51E+02±6.89E-03	2.15E-01±1.89E-01
	50	0.00E+00±0.00E+00	5.95E-08±4.85E-14	1.50E+01±2.62E+02	5.18E+02±1.94E+05	7.34E-02±1.21E-02
	70	0.00E+00±0.00E+00	8.53E-04±7.19E-06	9.91E+01±1.52E+04	5.13E+02±3.70E+05	1.76E-01±7.38E-02
F10	30	4.87E-06±6.04E-09	6.78E-04±1.46E-07	1.90E-02±9.83E-05	1.70E-03±5.01E-06	1.31E-01±4.27E-03
	50	2.51E-06±1.59E-09	8.58E-04±9.26E-07	3.69E-02±1.24E-03	1.33E-03±5.63E-06	1.25E-01±4.21E-03
	70	2.97E-06±7.82E-10	8.04E-04±1.63E-07	1.88E-02±6.00E-05	1.89E-03±8.98E-06	1.46E-01±4.08E-03
F11	30	0.00E+00±0.00E+00	4.26E-06±1.87E-11	2.01E-03±2.88E-05	7.15E-07±2.37E-12	2.40E+03±8.87E+05
	50	0.00E+00±0.00E+00	4.47E-06±7.69E-11	1.05E-01±6.87E-02	5.69E-09±1.58E-15	4.51E+03±2.05E+07
	70	0.00E+00±0.00E+00	4.90E-06±1.26E-10	9.88E-01±5.03E+00	1.17E-22±8.05E-43	5.90E+03±6.53E+07
F12	30	5.19E-30±7.89E-58	1.13E+00±3.40E-02	2.23E+00±2.68E+00	1.76E-01±4.50E-02	5.62E+02±9.56E-03
	50	3.61E-17±6.53E-32	1.49E+00±1.54E+00	4.20E+00±1.78E+01	2.99E-01±9.93E-02	1.06E+01±2.94E-02
	70	1.72E-12±2.04E-22	1.73E+00±4.09E+00	4.85E+00±3.96E+01	3.13E-01±2.20E-01	1.58E+01±7.27E-02
F13	30	4.55E+02±3.01E+06	6.34E+03±4.50E+05	6.06E+03±6.95E+05	1.11E+03±2.44E+06	4.62E+03±5.28E+06
	50	9.86E+02±9.34E+06	7.02E+03±3.42E+07	6.06E+03±2.60E+07	1.13E+03±4.74E+06	5.65E+03±2.44E+07
	70	1.53E+03±1.96E+07	7.34E+03±7.40E+07	6.43E+03±5.70E+07	1.15E+03±6.15E+06	7.45E+03±5.34E+07
F14	30	1.03E-20±3.17E-39	4.43E-02±4.35E-04	1.24E-01±5.59E-02	7.64E-03±5.11E-05	8.37E+00±3.52E+01
	50	1.78E-16±1.59E-30	4.74E-02±2.14E-03	5.14E-01±8.13E-01	6.01E-03±3.87E-05	7.67E+00±2.99E+00
	70	5.77E-17±2.03E-31	6.16E-02±6.04E-03	1.17E+00±2.72E+00	6.37E-03±1.33E-04	7.96E+00±7.13E+00
F15	30	9.66E-15±2.80E-27	4.87E-01±2.65E-02	1.68E-01±2.21E-01	1.78E-01±2.56E-02	3.57E-01±3.32E-01
	50	2.70E-14±3.66E-26	1.06E+00±8.55E-01	5.52E+00±4.55E+01	3.40E-01±1.30E-01	9.71E-01±2.43E+00
	70	9.21E-15±5.69E-27	1.51E+00±3.16E+00	1.43E+01±4.13E+02	4.06E-01±3.35E-01	9.19E+02±2.40E+07
F16	30	-1.17E+03±1.27E-20	-1.04E+03±2.52E+03	-1.03E+03±7.63E+02	-1.17E+03±1.76E-06	-4.33E+01±5.22E-29
	50	-1.17E+03±9.39+05	-9.84E+02±6.62E+05	-9.79E+02±6.65E+05	-1.17E+03±9.39E+05	-4.33E+01±5.22E-29
	70	-1.17E+03±1.86E+06	-9.28E+02±1.17E+06	-8.80E+02±1.13E+06	-1.17E+03±1.86E+06	-4.33E+01±8.96E-04
F17	30	8.88E-16±0.00E+00	1.50E-14±6.09E-30	1.70E+00±3.76E-01	3.25E-15±4.64E-30	5.89E+00±1.26E+01
	50	5.32E-16±1.93E-31	2.02E-14±2.89E-28	1.70E+00±2.11E+00	2.30E-15±7.79E-30	5.81E+00±7.15E+00
	70	3.80E-16±1.95E-31	2.27E-14±7.17E-28	1.59E+00±3.64E+00	2.10E-15±9.19E-30	5.12E+00±1.57E+01
F18	30	0.00E+00±0.00E+00	1.17E-03±2.52E-05	1.92E-02±7.62E-04	1.37E-02±8.80E-04	9.83E-01±5.70E-01
	50	0.00E+00±0.00E+00	2.49E-03±6.19E-05	1.06E-01±1.88E-02	5.52E-03±4.92E-04	8.14E-01±6.72E-01
	70	0.00E+00±0.00E+00	1.45E-04±1.48E-06	3.84E-01±2.79E-01	2.20E-03±3.39E-04	7.06E-01±4.32E-01
F19	30	0.00E+00±0.00E+00	3.65E-01±1.93E+00	5.21E+01±2.97E+02	0.00E+00±0.00E+00	2.07E+02±3.43E+00
	50	0.00E+00±0.00E+00	5.07E-01±1.88E+00	5.60E+01±2.50E+00	0.00E+00±0.00E+00	2.14E+02±2.89E+00
	70	0.00E+00±0.00E+00	3.14E-01±1.63E+00	5.65E+01±4.49E+03	3.24E-15±7.38E-28	1.55E+02±1.71E+00
F20	30	-1.26E+04±1.41E+06	-6.31E+03±6.18E+05	-6.39E+03±9.08E+05	-1.07E+04±3.09E+06	-2.12E+02±1.19E+03
	50	-1.28E+04±1.03E+07	-5.53E+03±2.18E+07	-6.25E+03±2.77E+07	-1.18E+04±9.74E+07	-2.13E+02±2.72E+03
	70	-1.28E+05±2.01E+07	-5.05E+03±3.56E+07	-6.30E+03±5.52E+07	-1.12E+04±1.76E+08	-2.21E+02±4.57E+03

Figures 17 and 31 and Table 5. In the F1-F4, F7-F9, and F11 functions, the ICSSOA outperforms the other four algorithms in terms of convergence accuracy and convergence speed and

can find the optimal solution. In the F5-F6, all algorithms fall into the local optimum at the early stage, but the ICSSOA can get rid of the local optimum dilemma by the hybrid policy

TABLE 6. Wilcoxon test results of the experimental results fixed dimensions.

Dim	SSA		CSSOA		ISSA(a)		CASSA		ISSA(b)		ICSSOA		GWO		PSO		WOA		ACO		
	W	R	W	R	W	R	W	R	W	R	W	R	W	R	W	R	W	R	W	R	
F1	30	+	7	=	1	=	1	=	1	=	1	1	+	8	+	9	+	6	+	10	
	50	+	7	=	1	+	6	=	1	=	1	1	+	8	+	9	+	5	+	10	
	70	+	7	=	1	=	1	=	1	=	1	1	+	8	+	9	+	6	+	10	
F2	30	+	7	=	1	+	5	=	1	+	4	1	+	8	+	9	+	6	+	10	
	50	+	7	=	1	+	6	=	1	+	4	1	+	8	+	9	+	5	+	10	
	70	+	7	=	1	+	5	=	1	+	4	1	+	8	+	9	+	6	+	10	
F3	30	+	6	=	1	+	5	=	1	=	1	1	+	7	+	8	+	9	+	10	
	50	+	6	=	1	=	1	=	1	=	1	1	+	7	+	8	+	10	+	9	
	70	+	6	=	1	=	1	=	1	=	1	1	+	7	+	8	+	10	+	9	
F4	30	+	6	=	1	+	4	=	1	+	5	1	+	7	+	8	+	10	+	9	
	50	+	6	=	1	+	4	=	1	+	5	1	+	7	+	8	+	9	+	10	
	70	+	6	=	1	+	5	=	1	+	4	1	+	7	+	9	+	10	+	8	
F5	30	+	6	+	4	+	2	=	3	+	5	1	+	7	+	9	+	8	+	10	
	50	+	6	+	5	+	2	+	3	+	4	1	+	7	+	9	+	8	+	10	
	70	+	6	+	4	+	2	+	5	+	3	1	+	7	+	9	+	8	+	10	
F6	30	+	4	+	2	+	6	+	3	+	5	1	+	9	+	7	+	8	+	10	
	50	+	6	+	2	+	5	+	3	+	4	1	+	9	+	7	+	8	+	10	
	70	+	5	+	3	+	2	+	4	+	6	1	+	8	+	9	+	7	+	10	
F7	30	+	9	=	1	+	5	=	1	=	1	1	+	7	+	8	+	6	+	10	
	50	+	8	=	1	+	7	=	1	=	1	1	+	6	+	9	+	5	+	10	
	70	+	8	=	1	=	1	=	1	=	1	1	+	7	+	9	+	6	+	10	
F8	30	+	7	=	1	+	5	=	1	=	1	1	+	8	+	9	+	6	+	10	
	50	+	7	=	1	+	6	=	1	=	1	1	+	8	+	10	+	5	+	9	
	70	+	7	=	1	=	1	=	1	=	1	1	+	8	+	9	+	6	+	10	
F9	30	+	6	=	1	+	5	=	1	+	4	1	+	7	+	8	+	10	+	9	
	50	+	6	=	1	+	5	=	1	+	4	1	+	7	+	8	+	10	+	9	
	70	+	6	=	1	+	5	=	1	+	4	1	+	7	+	8	+	10	+	9	
F10	30	+	6	+	4	+	2	+	4	+	5	1	+	7	+	9	+	8	+	10	
	50	+	6	+	3	+	2	+	4	+	5	1	+	7	+	9	+	8	+	10	
	70	+	6	+	3	+	2	+	5	+	4	1	+	7	+	9	+	8	+	10	
F11	30	+	6	=	1	=	1	=	1	+	5	1	+	8	+	9	+	7	+	10	
	50	+	6	=	1	+	5	=	1	=	1	1	+	8	+	9	+	7	+	10	
	70	+	6	=	1	+	5	=	1	=	1	1	+	8	+	9	+	7	+	10	
F12	30	+	6	+	4	+	3	+	5	+	2	1	+	8	+	9	+	7	+	10	
	50	+	6	+	4	+	2	+	5	+	3	1	+	8	+	9	+	7	+	10	
	70	+	6	+	4	+	2	+	5	+	3	1	+	8	+	9	+	7	+	10	
F13	30	+	7	-	1	+	6	+	5	+	4	2	+	10	+	9	+	3	+	8	
	50	+	7	-	1	+	6	+	5	+	4	2	+	10	+	9	+	3	+	8	
	70	+	6	-	1	+	7	+	5	+	4	3	+	9	+	8	-	2	+	10	
F14	30	+	4	+	3	+	6	+	2	+	5	1	+	8	+	9	+	7	+	10	
	50	+	6	+	3	+	2	+	5	+	4	1	+	8	+	9	+	7	+	10	
	70	+	6	+	4	+	3	+	2	+	5	1	+	8	+	9	+	7	+	10	
F15	30	+	5	+	4	+	2	+	3	+	6	1	+	9	+	7	+	8	+	10	
	50	+	6	+	5	+	2	+	4	+	3	1	+	9	+	10	+	7	+	8	
	70	+	6	+	5	+	2	+	3	+	4	1	+	8	+	9	+	7	+	10	
F16	30	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	=	1	+	10	
	50	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	=	1	+	10	
	70	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	=	1	+	10	
F17	30	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	+	7	+	10	
	50	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	+	7	+	10	
	70	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	+	7	+	10	
F18	30	=	1	=	1	=	1	=	1	=	1	1	+	7	+	9	+	8	+	10	
	50	=	1	=	1	=	1	=	1	=	1	1	+	7	+	9	+	8	+	10	
	70	=	1	=	1	=	1	=	1	=	1	1	+	7	+	9	+	8	+	10	
F19	30	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	=	1	+	10	
	50	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	=	1	+	10	
	70	=	1	=	1	=	1	=	1	=	1	1	+	8	+	9	+	7	+	10	
F20	30	+	7	+	2	+	5	+	6	+	4	1	+	9	+	8	+	3	+	10	
	50	+	7	+	2	+	6	+	5	+	4	1	+	9	+	8	+	3	+	10	
	70	+	7	+	2	+	6	+	5	+	4	1	+	9	+	8	+	3	+	10	
Ave		5.25		1.85		3.2		2.25		2.81		1.06		7.83		8.72		6.43		9.75	
Final		6		2		5		3		4		1		8		9		7		10	

(PS: Ave is the average ranking of accuracy. Final is the final ranking)

perturbation, and the convergence accuracy of the solution is more than 15 orders of magnitude higher than the other four algorithms. In the 70-dimensional test of the F13, the convergence accuracy of the ICSSOA algorithm is slightly lower than that of the WOA algorithm. In the F14-F20, the convergence accuracy of ICSSOA is much higher than the

other four algorithms in all cases. The ICSSOA is much better than the other four algorithms in terms of convergence speed and accuracy and shows excellent performance in 30, 50, and 70 dimensions.

The function convergence diagram shows that ICSSOA can perform a wide range of exploration and increase the

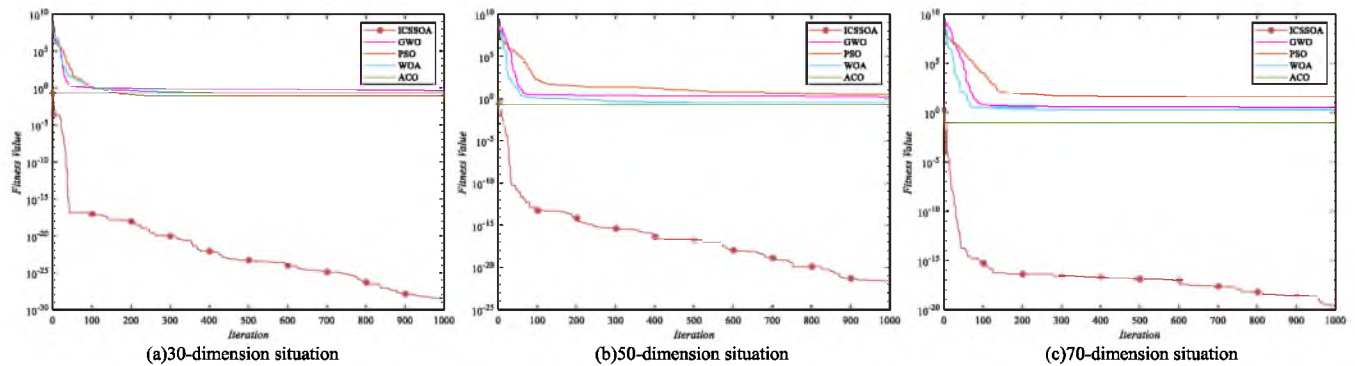


FIGURE 31. Performance comparison on F15.

diversity of optional solutions by adaptive weighting factors in the early iteration. As the number of iterations increases, ICSSOA gradually reduces the search scope and conducts regional small-scale exploitation to improve the solution accuracy. At the same time, in order to prevent the algorithm from falling into the dilemma of local optimum, ICSSOA is able to jump out of the local optimum better through the hybrid strategy, and thus find the optimal solution.

Comparing the test data analysis of the nine algorithms in the longitudinal experiment and the cross-sectional experiment, the ICSSOA outperforms the other algorithms in terms of overall convergence speed and accuracy of the solution in low dimensions and high latitudes. Based on the function convergence graphs, it can be observed that the proposed hybrid strategy can better improve the ability of the algorithm to jump out of the local optimum.

The Wilcoxon test was used to further analyze the experimental results. '+' means that the proposed algorithm outperforms the selected algorithm, '-' means that it is the opposite, '=' means that both algorithms get the same results, and the Rank column is the accuracy ranking of their average solutions. The specific results are analyzed as shown in Table 6.

From the data analysis in Table 6, the performance of ICSSOA is only slightly lower than CSSOA and WOA in the test of the F13 function, and it ranks first in the test of other functions, and it ranks first in the overall ranking of 20 test functions, which further indicates that the performance of ICSSOA is significantly better than the other nine algorithms. In summary, it can be seen that the ICSSOA is better than the other nine algorithms in terms of search capability, and has a strong adaptive capability and robustness.

## V. CONCLUSION

In this paper, an improved chaos sparrow search optimization algorithm, namely ICSSOA, is proposed to compensate for the shortcomings of the standard SSA of insufficient population diversity, weak local searchability, and easily falling into

local optimum, and ICSSOA significantly improves the optimization performance based on the following three points. First, Cubic chaotic mapping is used in the population initialization phase to enrich the population diversity and reduce the risk of the algorithm falling into the local optimum. Secondly, an adaptive inertia weighting strategy is used in the discoverer location update stage to expand the global search step in the first stage and shorten the local exploitation step in the second stage to balance the global search and local exploitation capabilities. Finally, after the population location update, a Levy flight and reverse learning hybrid strategy are used to perturb the population through a stochastic strategy. A greedy strategy is used to select individuals with higher fitness to improve the ability of the algorithm to escape from the local optimum. The experimental results and related statistical analysis show that the ICSSOA algorithm has significant advantages over the other five SSA algorithms and four different SI optimization algorithms in terms of optimal seeking ability, solution accuracy, and convergence speed. In the future, we will try to apply ICSSOA to industrial problems to improve the versatility and applicability of this algorithm.

## REFERENCES

- [1] S.-Q. Ye, K.-Q. Zhou, C.-X. Zhang, A. M. Zain, and Y. Ou, "An improved multi-objective cuckoo search approach by exploring the balance between development and exploration," *Electronics*, vol. 11, no. 5, p. 704, Feb. 2022.
- [2] S. Nouyan, A. Campo, and M. Dorigo, "Path formation in a robot swarm," *Swarm Intell.*, vol. 2, no. 1, pp. 1–23, Mar. 2008.
- [3] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile search algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Syst. Appl.*, vol. 191, Apr. 2022, Art. no. 116158.
- [4] W. Sun, M. Tang, L. Zhang, Z. Huo, and L. Shu, "A survey of using swarm intelligence algorithms in IoT," *Sensors*, vol. 20, no. 5, p. 1420, Mar. 2020.
- [5] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, Jan. 2020.
- [6] Y. Zhu and N. Yousefi, "Optimal parameter identification of PEMFC stacks using adaptive sparrow search algorithm," *Int. J. Hydrogen Energy*, vol. 46, no. 14, pp. 9541–9552, Feb. 2021.
- [7] T. Liu, Z. Yuan, L. Wu, and B. Badami, "An optimal brain tumor detection by convolutional neural network and enhanced sparrow search algorithm," *Proc. Inst. Mech. Eng., H. J. Eng. Med.*, vol. 235, no. 4, pp. 459–469, Apr. 2021.

- [8] T. Andi, H. A. N. Tong, X. U. Dengwu, and X. I. E. Lei, "Path planning method of unmanned aerial vehicle based on chaos sparrow search algorithm," *J. Comput. Appl.*, vol. 41, no. 7, p. 2128, Jul. 2021.
- [9] X. Lv, X.-D. Mu, and J. Zhang, "Multi-threshold image segmentation based on improved sparrow search algorithm," *Syst. Eng. Electron.*, vol. 43, no. 2, pp. 318–327, Feb. 2021.
- [10] W. Tuerxun, X. Chang, G. Hongyu, J. Zhijie, and Z. Huajian, "Fault diagnosis of wind turbines based on a support vector machine optimized by the sparrow search algorithm," *IEEE Access*, vol. 9, pp. 69307–69315, 2021.
- [11] J. Zhou and D. Chen, "Carbon price forecasting based on improved CEEMDAN and extreme learning machine optimized by sparrow search algorithm," *Sustainability*, vol. 13, no. 9, p. 4896, Apr. 2021.
- [12] C. Zhang and S. Ding, "A stochastic configuration network based on chaotic sparrow search algorithm," *Knowl.-Based Syst.*, vol. 220, May 2021, Art. no. 106924.
- [13] C. Ouyang, D. Zhu, and F. Wang, "A learning sparrow search algorithm," *Comput. Intell. Neurosci.*, vol. 2021, pp. 1–23, Aug. 2021.
- [14] P. Wang, Y. Zhang, and H. Yang, "Research on economic optimization of microgrid cluster based on chaos sparrow search algorithm," *Comput. Intell. Neurosci.*, vol. 2021, Mar. 2021, Art. no. e5556780.
- [15] J. Zhang, K. Xia, Z. He, Z. Yin, and S. Wang, "Semi-supervised ensemble classifier with improved sparrow search algorithm and its application in pulmonary nodule detection," *Math. Problems Eng.*, vol. 2021, Feb. 2021, Art. no. e6622935.
- [16] C. Wu, X. Fu, J. Pei, and Z. Dong, "A novel sparrow search algorithm for the traveling salesman problem," *IEEE Access*, vol. 9, pp. 153456–153471, 2021.
- [17] H. Wu, A. Zhang, Y. Han, J. Nan, and K. Li, "Fast stochastic configuration network based on an improved sparrow search algorithm for fire flame recognition," *Knowl.-Based Syst.*, vol. 245, Jun. 2022, Art. no. 108626.
- [18] P. Singh and S. S. Bose, "A quantum-clustering optimization method for COVID-19 CT scan image segmentation," *Expert Syst. Appl.*, vol. 185, Dec. 2021, Art. no. 115637.
- [19] P. Singh, "FQTSFM: A fuzzy-quantum time series forecasting model," *Inf. Sci.*, vol. 566, pp. 57–79, Aug. 2021.
- [20] P. Singh and Y.-P. Huang, "A new hybrid time series forecasting model based on the neutrosophic set and quantum optimization algorithm," *Comput. Ind.*, vol. 111, pp. 121–139, Oct. 2019.
- [21] Y. Wang, S. Ding, L. Wang, and S. Du, "A manifold  $p$ -spectral clustering with sparrow search algorithm," *Soft Comput.*, vol. 26, no. 4, pp. 1765–1777, Feb. 2022.
- [22] X. Li, J. Gu, X. Sun, J. Li, and S. Tang, "Parameter identification of robot manipulators with unknown payloads using an improved chaotic sparrow search algorithm," *Int. J. Speech Technol.*, vol. 52, no. 9, pp. 10341–10351, Jan. 2022.
- [23] X. Lv, X. D. Mu, J. Zhang, and Z. Wang, "Chaos sparrow search optimization algorithm," *J. Beijing Univ. Aeronaut. Astronaut.*, vol. 12, pp. 1–10, Aug. 2020.
- [24] Q. H. Mao and Q. Zhang, "Improved sparrow algorithm combining Cauchy mutation and opposition-based learning," *J. Frontiers Comput. Sci. Technol.*, vol. 15, no. 6, pp. 1155–1164, 2021.
- [25] 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, Feb. 2021.
- [26] J. Yuan, Z. Zhao, Y. Liu, B. He, L. Wang, B. Xie, and Y. Gao, "DMPPT control of photovoltaic microgrid based on improved sparrow search algorithm," *IEEE Access*, vol. 9, pp. 16623–16629, 2021.
- [27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.
- [28] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Jun. 2007.
- [29] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, Feb. 2016.
- [30] S. Mirjalili, "Ant colony optimisation," in *Evolutionary Algorithms and Neural Networks: Theory and Applications*, S. Mirjalili, Ed. Cham, Switzerland: Springer, 2019, pp. 33–42.
- [31] P. Lalbakhsh, B. Zaeri, and A. Lalbakhsh, "An improved model of ant colony optimization using a novel pheromone update strategy," *IEICE Trans. Inf. Syst.*, vol. 96, no. 11, pp. 2309–2318, 2013.
- [32] D. Angus and C. Woodward, "Multiple objective ant colony optimisation," *Swarm Intell.*, vol. 3, no. 1, pp. 69–85, Mar. 2009.
- [33] A. Palacios, "Cycling chaos in one-dimensional coupled iterated maps," *Int. J. Bifurcation Chaos*, vol. 12, no. 8, pp. 1859–1868, Aug. 2002.
- [34] J. Feng, J. Zhang, X. Zhu, and W. Lian, "A novel chaos optimization algorithm," *Multimedia Tools Appl.*, vol. 76, no. 16, pp. 17405–17436, Aug. 2017.
- [35] M. J. Zhang, D. Y. Long, D. D. Li, X. Wang, T. Qin, and J. Yang, "A novel chaotic grey wolf optimisation for high-dimensional and numerical optimisation," *Int. J. Comput. Appl. Technol.*, vol. 67, nos. 2–3, pp. 194–203, Jan. 2021.
- [36] C.-X. Zhang, K.-Q. Zhou, S.-Q. Ye, and A. M. Zain, "An improved cuckoo search algorithm utilizing nonlinear inertia weight and differential evolution for function optimization problem," *IEEE Access*, vol. 9, pp. 161352–161373, 2021.
- [37] H. Haki and H. Uğuz, "A novel particle swarm optimization algorithm with Levy flight," *Appl. Soft Comput.*, vol. 23, pp. 333–345, Oct. 2014.



**XUAN-YU ZHANG** was born in Changsha, China, in 1997. He received the B.S. degree in communication engineering from Jishou University, in 2019. He is currently pursuing the master's degree in electronic information engineering with Jishou University. His main research interests include soft computing techniques and fuzzy Petri net and its applications.



**KAI-QING ZHOU** was born in Changsha, China, in 1984. He received the B.S. degree in computer science and technology from Jishou University, in 2006, the M.S. degree in computer applied techniques from the Changsha University of Science and Technology, in 2011, and the Ph.D. degree in computer science from Universiti Teknologi Malaysia, in 2016. He was a Postdoctoral Fellow with the College of Information and Engineering, Central South University, from 2016 to 2018.

He is currently an Associate Professor with the College of Communication and Electronic Engineering, Jishou University. His main research interests include fuzzy Petri net and its applications, clinical decision support systems, and soft computing techniques.



**PENG-CHENG LI** was born in Tianmen, China, in 1998. He received the B.S. degree in network engineering from Liaoning Technical University, in 2019. He is currently pursuing the master's degree in electronic information engineering with Jishou University. His main research interests include soft computing techniques and natural language processing.



**YIN-HONG XIANG** was born in Changde, China, in 1998. He received the B.S. degree in network engineering from Jishou University, in 2020, where he is currently pursuing the master's degree in electronic information. His main research interests include fuzzy Petri net and its applications.



**AZLAN MOHD ZAIN** (Member, IEEE) received the master's degree in science (productivity and quality improvement) from Universiti Kebangsaan Malaysia (UKM) and the Ph.D. degree in computer science from Universiti Teknologi Malaysia (UTM), in 2010. He is currently a Professor with the Faculty of Engineering, School of Computing, UTM. He is also the Director of the UTM Big Data Research Centre. As an academic staff, he has successfully supervised more than 25 postgraduate students and received more than 20 research grant funding to support research students. He has published more than 100 research papers. He has been invited as keynote speaker at over five international conferences, serves on numerous committees, and has served on editorial board for three international journals.



**AREZOO SARKHEYLI-HÄGELE** received the first Ph.D. degree in soft computing and computer science by focusing on machine learning and optimization algorithms for intelligent systems and the second Ph.D. degree in engineering by focusing on cognitive systems and supervision of human operators. She is currently a Senior Lecturer with the Department of Computer Science and Media Technology, Faculty of Technology and Society, Malmö University. Her main research interests include knowledge representation, situation recognition and reasoning in cognitive systems, artificial intelligence, machine learning, and optimization in dynamical environments and the Internet of Things.

\*\*\*