



ARTICLE

Improved Cyclic System Based Optimization Algorithm (ICSBO)

Yanjiao Wang and Zewei Nan^{*}

School of Electrical Engineering, Northeast Electric Power University, Jilin, 132012, China

^{*}Corresponding Author: Zewei Nan. Email: nzws4299@126.com

Received: 23 September 2024; Accepted: 12 December 2024; Published: 06 March 2025

ABSTRACT: Cyclic-system-based optimization (CSBO) is an innovative metaheuristic algorithm (MHA) that draws inspiration from the workings of the human blood circulatory system. However, CSBO still faces challenges in solving complex optimization problems, including limited convergence speed and a propensity to get trapped in local optima. To improve the performance of CSBO further, this paper proposes improved cyclic-system-based optimization (ICSBO). First, in venous blood circulation, an adaptive parameter that changes with evolution is introduced to improve the balance between convergence and diversity in this stage and enhance the exploration of search space. Second, the simplex method strategy is incorporated into the systemic and pulmonary circulations, which improves the update formulas. A learning strategy aimed at the optimal individual, combined with a straightforward opposition-based learning approach, is employed to enhance population convergence while preserving diversity. Finally, a novel external archive utilizing a diversity supplementation mechanism is introduced to enhance population diversity, maximize the use of superior genes, and lower the risk of the population being trapped in local optima. Testing on the CEC2017 benchmark set shows that compared with the original CSBO and eight other outstanding MHAs, ICSBO demonstrates remarkable advantages in convergence speed, convergence precision, and stability.

KEYWORDS: Circulatory system-based optimization (CSBO) algorithm; meta-heuristic algorithm; external archives; adaptive learning; individual renewal strategy

1 Introduction

Optimization problems widely exist in production and life, such as optimizing the transportation path to reduce logistics and distribution costs and allocating funds to different investment projects in the financial field to maximize the return under the premise of controllable risk. Traditional optimization methods such as least squares and most rapid descent, generally for structured problems, have a clearer description of the problem and conditions, whereas the meta heuristic algorithms (MHAs) for a more universal description of the problem generally lack structural information. Moreover, MHAs have the characteristics of flexible structure, high stability, and satisfactory robustness, which make them the most effective methods for solving optimization problems.

As optimization problems become more complex, traditional MHAs, including genetic algorithms and differential algorithms, often exhibit limitations in convergence speed and accuracy. To address these challenges and enhance the solution quality for complex optimization tasks, many novel MHAs have been developed. In 2022, Ghasemi et al. proposed a new MHA, namely, circulatory-system-based optimization (CSBO), by modeling the human circulatory system [1]. Extensive experiments have demonstrated that



CSBO outperforms biogeographic optimization algorithms [2], particle swarm optimization (PSO) algorithms, artificial bee colony algorithms [3], and other MHAs in both convergence speed and accuracy. Nevertheless, CSBO continues to face challenges, such as reduced convergence speed and susceptibility to local optima, when addressing high-dimensional, multi-peak complex optimization problems.

To address these challenges, this paper proposes an improved circulatory-system-based optimization (ICSBO) algorithm, with the following motivations and contributions:

(1) In the venous blood circulation mechanism of CSBO, individuals not only learn from themselves and other individuals but are also influenced by perturbations from other individuals. To ensure the evolutionary direction of the entire population, mutual learning should play a guiding role in the evolution of individuals, and the degree of learning through mutual interaction should be higher than that of perturbation-based learning. On this basis, a new venous blood circulation mechanism by adjusting the learning degree of perturbation-based learning is proposed.

(2) Given the advantage of the simplex method in fast convergence, it is incorporated into the updating mechanism of systemic circulation, which accelerates the convergence speed and accuracy of the population in the systemic circulation phase.

(3) Random perturbation in the pulmonary circulation can exhibit randomness, so it is prone to ineffective searches that fail to supplement diversity effectively. Because different search mechanisms can enhance the algorithm's ability to handle various optimization problems, and the simplex method offers fast convergence while opposition-based learning can introduce more population diversity, both search mechanisms are incorporated into the pulmonary circulation strategy. This approach ensures the population's convergence speed while providing greater diversity.

(4) MHAs are generally very susceptible to the situation where the overall diversity of a population is still good but distinct individuals remain unchanged for many consecutive generations during population evolution. Although the regeneration approach helps replenish diversity, it completely abandons the previous evolution of the individual. Individuals that outperform the previous generation in the evolution are stored in an archive. When an individual falls into local stagnation of updating, a historical individual is randomly selected from the archive to replace it, and a mechanism to replenish the diversity of population based on the external archive is proposed.

Experimental results on the CEC2017 test set demonstrate that the proposed ICSBO outperforms eight representative optimization algorithms, showcasing notable advantages in convergence speed, accuracy, and stability.

2 Related Work

MHA is a class of computational intelligence-based mechanism for solving complex optimization problems with optimal or satisfactory solutions, also known as intelligent optimization algorithms. MHA designs an intelligent iterative search by simulating relevant behaviors, functions, and mechanisms in biological, physical, chemical, and other systems or domains. MHA is widely used in many practical engineering fields because it has good global search capability and flexibility, and it does not require the optimization problem to have a functional form such as differentiable and derivable.

Metaheuristic algorithms are generally classified into four main groups according to their foundational principles: (1) Evolution-inspired algorithms, such as Genetic Algorithm (GA) [4], Differential Evolution (DE) [5], and Co-evolutionary Algorithms [6]; (2) Population-based methods, including Ant Colony Optimization (ACO) [7], Grey Wolf Optimizer (GWO) [8], Butterfly Optimization Algorithm (BOA) [9], and Great White Shark Algorithm [10]. (3) Algorithms based on physical phenomena, such as Simulated

Annealing (SA) and Archimedes Optimization Algorithm [11]; (4) Algorithms based on human activities, such as Teaching-Learning-Based Optimization (TLBO) [12], Knowledge Sharing Algorithm (KSA) [13] and Information acquisition optimizer (IAO) [14].

However, MHAs still face challenges, such as slow convergence speed and a tendency to get trapped in local optima, when applied to high-dimensional, multi-peak complex optimization problems. To obtain the optimal solution of this kind of optimization problem, scholars have studied MHAs from two aspects.

On one hand, improving existing traditional heuristic algorithms to further enhance their optimization performance has been a focus of research. In 2023, Yao et al. developed an Enhanced Serpent Optimizer (ESO) [15] by integrating a new dynamic mechanism and an opposition-based learning strategy, aiming to enhance the Serpent Optimizer for solving practical engineering problems. In the same year, Wang et al. introduced an Improved Archimedes Optimization Algorithm (IAOA) [16], which employs the simplex method to adjust individuals with low fitness, enhancing the algorithm's convergence speed and accuracy. Hu et al. (2023) introduced the Improved Orca Predation Algorithm (IOPA) [17], which enhances population diversity and helps the algorithm escape local minima by integrating a dimension-learning strategy with an opposition-based learning method. In 2024, Ye et al. proposed a multi-strategy enhanced dung beetle optimization algorithm (MDBO) [18], incorporating Latin hypercube sampling, lens imaging-based opposition learning, and dimension-wise optimization strategies to address issues such as poor diversity and unsatisfactory convergence speed. Furthermore, Nadimi-Shahraki et al. proposed the MTV-SCA (Multiple Trial Vectors Sine Cosine Algorithm) [19], which integrates multiple trial vectors along with four different search strategies to address the Sine Cosine Algorithm's propensity for becoming stuck in local optima and to better balance exploration and exploitation.

On the other hand, new metaheuristic algorithms have been proposed and their performance improved. Alzoubi et al. introduced the Synergistic Swarm Optimization Algorithm (SSOA) [20], which combines swarm intelligence and cooperative collaboration to effectively search for optimal solutions. In 2021, Abualigah et al. introduced the Aquila Optimizer (AO), inspired by the natural hunting behavior of eagles. In 2022, Abdollahzadeh et al. developed the Mountain Gazelle Optimization Algorithm (MGO) [21], drawing inspiration from the social behavior and hierarchy of wild mountain gazelles. In 2023, Venkata et al. simulated the hunting behavior of the Tyrannosaurus Rex to propose the Tyrannosaurus Rex Optimization Algorithm (TROA). In the same year, Mojtaba et al. combined the Circulatory System-Based Optimization (CSBO) algorithm with the Levy flight mechanism, proposing the Gaussian Bare-Bones Levy Circulatory System Optimization Algorithm, which enhanced its ability to solve complex OPF (Optimal Power Flow) problems. In 2024, Yang et al. introduced adaptive inertia weights, golden sine operators, and chaotic strategies into the CSBO algorithm, creating the Multi-Strategy Enhanced CSBO (MECSBO) [22], which minimized the risk of the CSBO algorithm becoming trapped in local optima. Additionally, Wang et al. merged the CSBO algorithm with other optimization methods to develop a hybrid algorithm, which was used to create an improved prediction model for PM_{2.5} (Particulate Matter) concentration. In 2024, Ji et al. drew inspiration from neuroscience and proposed the Neural Population Dynamics Optimization Algorithm (NPDOA) [23].

In addition, numerous scholars have tried to apply MHAs in the following practical engineering applications. In the field of engineering, cloud heuristic algorithms can be applied to design optimization, production optimization, and logistic optimization. In the field of biological sciences, MHAs can be applied to gene analysis, protein recombination, and other problems. In the field of manufacturing, MHAs can be applied to production planning, quality control, and supply chain management. The common feature of these practical application problems is that their mathematical nature is optimization problems.

3 Cyclic System-Based Optimization (CSBO) Algorithm

The human circulatory system delivers oxygen and other nutrients to various tissues through the blood in the blood vessels, while also removing carbon dioxide and other waste products generated by metabolism, effectively ensuring the body's health. Blood flows through the combined action of the pulmonary and systemic circulation loops. The systemic circulation loop transports oxygen-rich arterial blood throughout the body via arteries and facilitates the exchange of oxygen, nutrients, and metabolic waste through capillaries, converting it into venous blood, which is then returned to the right atrium through the veins. The pulmonary circulation describes the movement of deoxygenated blood from the right ventricle through the pulmonary arteries to the lungs, where it exchanges carbon dioxide for oxygen, before returning oxygenated blood to the left ventricle through the pulmonary veins.

Simulating the above circulatory system, Mojtaba Ghasemi et al. proposed a new heuristic algorithm: CSBO. In the CSBO algorithm, the process includes venous circulation, systemic circulation, and pulmonary circulation. In this algorithm, the better-performing individuals in the population after venous circulation undergo systemic circulation, while the poorer-performing individuals undergo pulmonary circulation. Algorithm 1 displays the pseudocode for the CSBO algorithm, with a summary of the key steps outlined below:

Algorithm 1: CSBO

Input: popsize: N_p ; Dimensionality of the Problem: dim ; Maximum Number of Iterations: $maxit$;

Maximum Number of Evaluations: $maxFE$; Range of the Search Space: U, L

output: Optimal Solution and Its Fitness Value

01. $pop \leftarrow$ Initialize the Population
 02. Calculate the Fitness of Each Individual
 03. While $FE < maxFE$
 04. Update Individuals in pop According to the Venous Blood Circulation in [Section 3.1](#)
 05. Sort the pop Population by Fitness Values in Ascending Order
 06. $pop_nl \leftarrow$ The NI Individuals with Better Fitness in pop
 07. $pop_nr \leftarrow$ The $N_p - NI$ Individuals with Worse Fitness in pop
 08. Update individuals in pop_nl using the body circulation from [Section 3.2](#).
 09. Update individuals in pop_nr using the pulmonary circulation from [Section 3.3](#).
 10. $pop \leftarrow pop_nl \cup pop_nr$
 11. end While
 12. Output the Global Best Individual and Its Corresponding Global Optimal Solution
-

3.1 Venous Circulation

In the venous circulation phase, the CSBO algorithm designs a new individual generation method as shown in [Eq. \(1\)](#). It is crucial to highlight that the generated individual will only replace the original if it shows superior performance; otherwise, the original individual directly participates in the subsequent evolution.

$$X_{pop_i}^{new} = X_{pop_i} + K_{i,1} \times p_i \times (X_{pop_i} - X_{pop_1}) + K_{3,2} \times p_i \times (X_{pop_3} - X_{pop_2}) \quad (1)$$

Here, X_{pop_1} , X_{pop_2} , and X_{pop_3} are three individuals randomly chosen from X_{pop} ; $K_{m,n}$ represents the movement direction between individuals X_{pop_m} and X_{pop_n} , as shown in [Eq. \(2\)](#). The parameter p_i is

randomly generated between $[0, 1]$ in the initial iteration, and its calculation method in subsequent iterations is shown in Eq. (3).

$$K_{m,n} = \frac{fit_n - fit_m}{|fit_m - fit_n| + \varepsilon} \quad (2)$$

Here, ε is a very small value, typically set to 10^{-6} ; fit_m and fit_n correspond to the fitness of individuals X_{pop_m} and X_{pop_n} , respectively.

$$p_i = \begin{cases} \frac{fit_i - fit_{worst}}{fit_{best} - fit_{worst}} & i \in pop_{nl} \\ rand & i \in pop_{nr} \end{cases} \quad (3)$$

In this case, *rand* denotes a randomly generated value within the range $[0, 1]$, while fit_{best} and fit_{worst} correspond to the highest and lowest fitness values in the current population, respectively.

3.2 Systemic Circulation

In the venous circulation phase, the *nl* most excellent individuals in the population form an elite group *pop_{nl}*. Each individual in this group undergoes systemic circulation as follows: First, intermediate individuals $X_{pop_{nl_i}}^{new}$ are generated according to Eq. (4). Then, $X_{pop_{nl_i}}^{new}$ and X_{pop_i} undergo crossover operations dimension by dimension. *rand* is generated between $[0, 1]$; and if *rand* is greater than the crossover probability *cr* ($cr = 0.9$), the gene information from $X_{pop_{nl_i}}^{new}$ is retained; otherwise, the gene information from X_{pop_i} is retained, resulting in a new individual $X_{pop_{nl_i}}^{new}$. Finally, individuals with superior fitness values from both the new and previous generations are selected to proceed to the next stage of evolution.

$$X_{pop_{nl_i}}^{new} = X_{pop_{p_1}} + p_i \times (X_{pop_{p_2}} - X_{pop_{p_3}}) \quad (4)$$

Here, $X_{pop_{p_1}}$, $X_{pop_{p_2}}$, and $X_{pop_{p_3}}$ are three individuals randomly chosen from *pop*; the parameter p_i is calculated according to Eq. (3).

3.3 Pulmonary Circulation

During the venous circulation phase, the *nr* ($nr = np - nl$) poorer individuals in the population enter the pulmonary circulation, generating new individuals based on Eq. (5). If the newly generated individuals outperform the original ones, they replace them and continue in the subsequent iterations.

$$X_{pop_{nr_i}}^{new} = X_{pop_i} + \frac{randn}{it} \times randc(1, dim) \quad (5)$$

In this context, *dim* refers to the problem's dimension; *randn* represents a random value sampled from a normal distribution within the interval $[-0.5, 0.5]$; *it* indicates the current iteration number.

4 Improved CSBO Algorithm

To enhance the convergence rate and precision of the CSBO algorithm, several enhancements were made, resulting in the development of an improved CSBO algorithm. The flowchart is presented in Fig. 1.

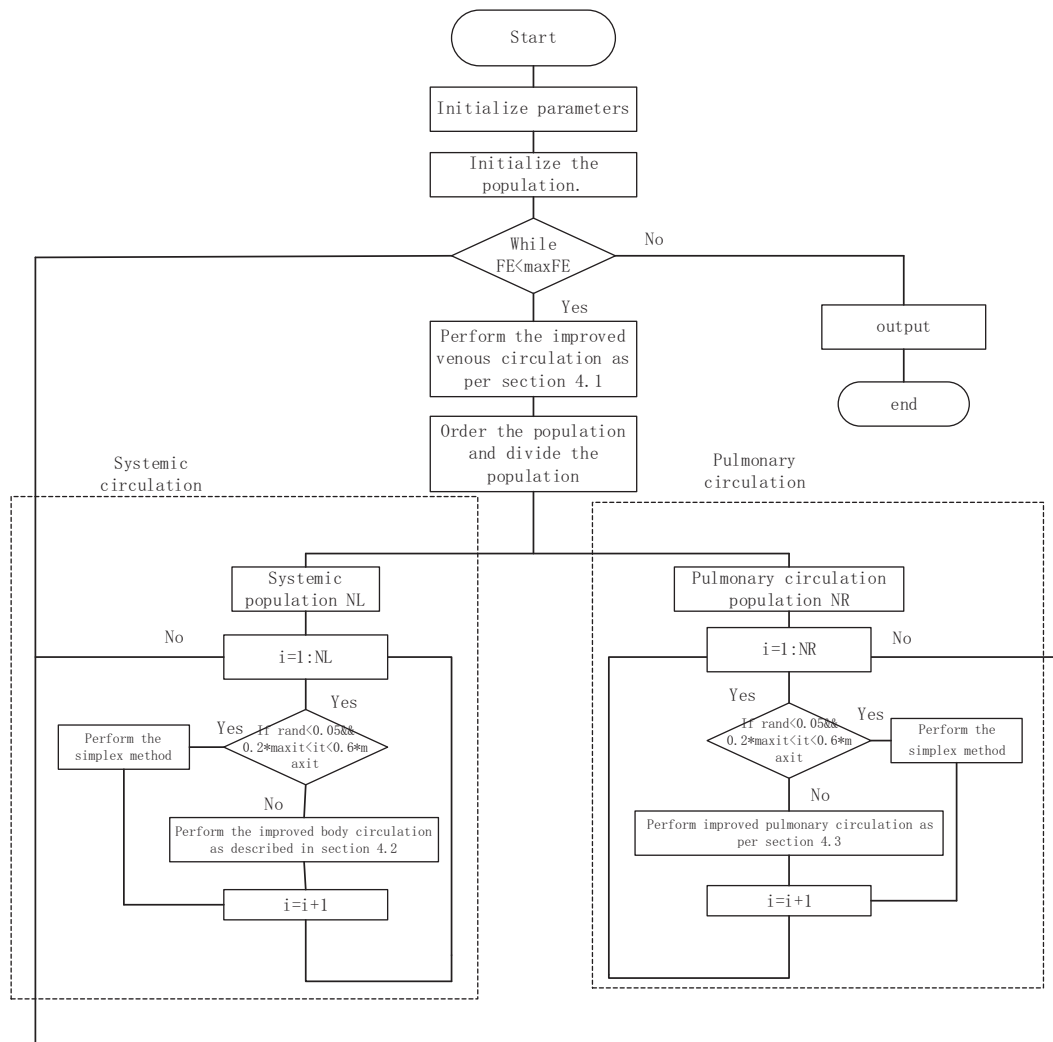


Figure 1: Flowchart of ICSBO algorithm

4.1 Improved Venous Circulation

As shown in Eq. (1) of Section 3.1, in the venous circulation phase of CSBO, the update of individuals relies on “self-learning of the individual + mutual learning between the individual and other individuals + perturbation learning between two random individuals.” Mutual learning and perturbation learning allow the gene information of the three random individuals to flow into the current individual, which helps maintain population diversity. However, the step size before mutual learning and perturbation learning is determined by $K_{m,n} \times p_i$. According to Eq. (2), when individual X_{pop_m} performs better than X_{pop_n} , $K_{m,n} = +1$; conversely, when individual X_{pop_m} performs worse than X_{pop_n} , $K_{m,n} = -1$. Thus, $K_{m,n}$ only determines the direction of mutual learning and perturbation learning, whereas p_i determines the degree of learning from mutual learning and perturbation learning.

According to literature [24], learning in the direction of superior individuals enhances the likelihood of individuals moving toward better positions, which can partially improve the algorithm’s convergence speed. The parameter $K_{m,n}$ can effectively achieve this purpose. To ensure the overall evolutionary direction of the

population, mutual learning should play a dominant guiding role in individual evolution. The learning degree of mutual learning and perturbation learning cannot be exactly the same. Therefore, the direction guidance parameter $K_{m,n}$ is retained before mutual learning and perturbation learning, and different learning degrees are assigned to mutual learning and perturbation learning. A new venous circulation operation is proposed, as defined in Eq. (6).

$$X_{pop_i}^{new} = X_{pop_i} + K_{i,1} \times p_i \times (X_{pop_i} - X_{pop_1}) + K_{2,3} \times f_1 \times (X_{pop_3} - X_{pop_2}) \quad (6)$$

The generation method for f_1 follows LSHADE (Improving the Search Performance of SHADE (Success-History based Adaptive Differential Evolution) Using Linear Population Size Reduction), and it is produced according to Eq. (7). If the generated f_1 value is not within the boundary $[0, H_1]$, it needs to be revised according to Eq. (8).

$$f_1 = randc(uf(h), R) \quad (7)$$

$$f_1 = \begin{cases} randc(uf(h), R) & \text{if } f_1 < 0 \\ H_1 & \text{if } f_1 > H_1 \end{cases} \quad (8)$$

Here, H_1 and R have different values at different evolutionary stages. When it $\leq 0.4 \times \max it$, $H_1 = 0.9$ and $R = 0.1$; otherwise, $H_1 = 1.2$ and $R = 1$. $randc$ represents the Cauchy distribution, and $uf(h)$ denotes a value randomly selected from a D -dimensional array uf . In general, setting $D = 5$ produces satisfactory results, and this value can be adjusted based on the specific problem at hand. Initially, all elements in uf are set to 0.5. As iterations proceed, the elements in uf are updated as follows: First, the individuals that are successfully retained after updating according to Eq. (6) are saved to the set sg , and the corresponding f_1 values of these individuals are saved to the set sf . Because each individual has a different f_1 value, sf can contain up to Np elements or be an empty set. Then, the weights corresponding to the successfully retained individuals are calculated using Eq. (9), and the Lemher mean value for all successfully retained individuals is computed using Eq. (10). Finally, the elements of uf are updated algebraically according to Eq. (11)

$$wg f_i = \frac{\sqrt{\sum_{j=1}^{\dim} \left(NC_{ig} - \left(\sum_{j=1}^{\dim} NC_{ig} / \dim \right) \right)^2 / \dim}}{\sum_{i \in sf} \sqrt{\sum_{j=1}^{\dim} \left(NC_{ig} - \left(\sum_{j=1}^{\dim} NC_{ig} / \dim \right) \right)^2 / \dim}} \quad (9)$$

Here, NC_i represents the i -th individual in the set sg .

$$mean_{lf} = \frac{\sum_{i=1}^{|sf|} wg f_i \times sf_i^2}{\sum_{i=1}^{|sf|} wg f_i \times sf_i} \quad (10)$$

$$uf(t_f) = \begin{cases} mean_{lf} & \text{if } sf \neq \emptyset \\ 0.5 & \text{otherwise} \end{cases} \quad (11)$$

Here, $t_f = (it - 1) \% 5 + 1$.

In summary, compared with the original venous circulation operation shown in Eq. (1), the novel venous circulation operation proposed in this section uses the Cauchy distribution to determine the learning degree

of perturbation learning. The mathematical formulation of the Cauchy distribution suggests that during the initial phases of evolution, the learning intensity of perturbation learning varies within the range $[0, 0.9]$, and smaller scale parameters result in values clustered around 0.5. During the final phases of evolution, the learning degree of perturbation learning ranges from $[0, 1.2]$, and larger scale parameters make values around 0.5 slightly more probable than other values. According to the generation method of P_i in [Section 3.1](#), for better individuals, P_i is a normalized fitness value, whereas for poorer individuals, P_i is randomly generated within $[0, 1]$. Thus, better individuals are more likely to have P_i values greater than 0.5, whereas poorer individuals are more likely to have P_i values less than 0.5. Consequently, in the early stages of evolution, better individuals are more likely to have a learning degree of mutual learning greater than that of perturbation learning, which speeds up the convergence of better individuals. Conversely, for poorer individuals, the learning degree of mutual learning is more likely to be less than that of perturbation learning, which accelerates their shift toward other individuals and thus enhances the evolutionary speed while maintaining population diversity. During the final phases of evolution, the learning degree of perturbation learning is further increased compared with mutual learning, which better maintains the diversity of the population.

4.2 Improved Body Circulation Strategy

A thorough analysis of the systemic circulation strategy proposed by CSBO reveals that its convergence speed is slow, primarily due to the following reasons: First, intermediate individuals are generated through mutual learning among three randomly selected individuals from the venous blood population. While this random selection method helps maintain population diversity, the inherent randomness inevitably slows down the algorithm's convergence speed. Second, when generating new individuals through crossover operations between intermediate individuals and original individuals, a high crossover probability results in the incorporation of only a small amount of gene information from the intermediate individuals into the original individuals. While this approach effectively maintains the existing diversity within the population, it considerably slows down the pace of population evolution.

To achieve an effective balance between convergence speed and population diversity, this section presents an improvement strategy for updating individuals in the elite population. Specifically, when the condition $rand < 0.05 \& 0.2 \times maxit < it < 0.6 \times maxit$ is satisfied, a new individual is created using the simplex method described in [Section 4.2.1](#); otherwise, the approach outlined in [Section 4.2.2](#) is used to generate a new individual. As with the systemic circulation strategy in CSBO, the top-performing individuals, whether newly generated or from the previous population, are selected for further evolution.

4.2.1 Simplex Method

The method for generating new individuals using the simplex method is as follows: Calculate the reflection point X_r of individual X_{pop_i} using [Eq. \(12\)](#). If the fitness value at the reflection point fit_r is greater than that of the current best individual X_{gbest} , generate a new individual using the expansion operation shown in [Eq. \(13\)](#). If the fitness value fit_r is worse than that of individual X_{pop_i} , generate a new individual using the contraction operation shown in [Eq. \(14\)](#). If X_r is worse than that of the current best individual X_{gbest} but better than X_{pop_i} , generate a new individual using the shrinking operation shown in [Eq. \(15\)](#).

$$X_c = \frac{X_{gbest} + X_{cibest}}{2} + b \times \left(\frac{X_{gbest} + X_{cibest}}{2} - X_{pop_i} \right) \quad (12)$$

Here, b is the reflection coefficient, typically set to 1; X_{gbest} and X_{cibest} denote the best and second-best values within the current population, respectively.

$$X_{pop_i}^{new} = X_c + \lambda \times (X_r - X_c) \quad (13)$$

$$X_{pop_i}^{new} = X_c + \eta \times (X_{pop_i} - X_c) \quad (14)$$

$$X_{pop_i}^{new} = X_c - \mu \times (X_{pop_i} - X_c) \quad (15)$$

Here, λ is the expansion coefficient, typically set to 2; η is the compression coefficient, generally set to 0.5; and μ is the shrinking coefficient, generally set to 0.5. $X_c = \frac{X_{gbest} + X_{cibest}}{2}$.

4.2.2 Novel Update Mechanism

The new individual update mechanism is as follows: First, similar to the generation method of f_1 in Section 4.1, use the crossover probability cr as shown in Eq. (16). Then, generate intermediate individuals according to Eq. (17). Finally, perform crossover operations between intermediate individuals and original individuals dimension by dimension with the probability cr to generate new individuals.

$$cr = randc(ucr(h), R) \quad (16)$$

The setting for R is exactly the same as in Section 3.1, and the update method for $ucr(h)$ is similar to that in Section 4.1. It is important to note that when $it \leq 0.4 \times \max it$, if the generated cr is less than 0.5, a new cr is generated according to Eq. (16); if the generated cr is greater than 0.9, cr is directly set to 0.9. When $it > 0.4 \max it$, if the generated cr is less than 0, a new cr is generated according to Eq. (16); if the generated cr is greater than 0.8, cr is directly set to 0.9.

$$X_{pop_nl_i}^{new} = \begin{cases} X_{pop_nl_i} + p_i \times (X_{pop_nl_i} - X_{pop_1}) + 2 \times rand(1, \dim) \times (X_{pop_2} - X_{pop_3}) & \text{if } rand < tf \\ pbest + p_i \times (gbest - X_{pop_1}) & \text{else} \end{cases} \quad (17)$$

Here, X_{pop_1} , X_{pop_2} and X_{pop_3} denote three randomly selected individuals from the population X_{pop} ; $pbest$ represents one individual randomly chosen from the top 3 best-performing individuals; $gbest$ represents the optimal individual in the current population, tf is as shown in Eq. (18).

$$tf = e^{\frac{it - \max it}{\max it}} \quad (18)$$

In conclusion, the novel systemic circulation strategy introduced in this section offers several advantages. First, compared to the update mechanism in Section 3.2, the proposed strategy incorporates an exploration method based on the simplex approach, applied near the global optimal individual to generate new candidates. This method significantly improves the algorithm's convergence speed. Moreover, the low frequency of simplex method execution helps preserve population diversity without causing excessive disruption. Additionally, the simplex method is only employed during the mid-evolution phase, ensuring global search during the early stages and reducing the risk of the algorithm getting trapped in local optima due to insufficient diversity in later stages. Second, as shown in Eq. (17), besides the simplex method, this section introduces two other search strategies: one executed when $rand < tf$, which maintains population diversity through sufficient interaction among individuals, and another improves convergence speed through learning from better and optimal individuals. In Eq. (18), tf gradually increases to 1 during the evolution. In

the early evolution stage, this enhances convergence speed, whereas in the later stage, this better maintains population diversity. Third, the cr generation method proposed in this section varies with the evolution stage. In the early evolution stage, cr ranges from [0.5, 0.9], whereas in the later stage, it ranges from [0, 0.8]. Compared with the originally fixed higher cr values in CSBO, the cr values generated in this section are smaller in the early and later stages of evolution. This approach allows more gene flow into new individuals, which overall benefits the rapid convergence of the population. Additionally, the larger cr value range in the early evolution stage, combined with the crossover and search strategies, ensures improved convergence speed while maintaining a certain level of global search.

4.3 Improvements in the Pulmonary Circulation Stage

In Eq. (5), the update of individuals in the pulmonary circulation involves making slight perturbations to the original individuals to provide more diversity to the population. However, due to the randomness of the perturbations, the probability that the new individual is better than the original is extremely low. New individuals are rarely preserved for subsequent evolution, which leads to ineffective search, which not only hinders providing more population diversity but also reduces the algorithm's convergence speed.

This section introduces an enhanced pulmonary circulation strategy to balance the convergence rate and population diversity: For each individual participating in the pulmonary circulation, if $rand < 0.05 \& 0.2 \times maxit < it < 0.6 \times maxit$, a new individual is generated using the simplex method as described in Section 4.2.1; otherwise, a new individual is generated according to Eq. (19). As with the systemic circulation strategy in CSBO, the individual with the higher fitness, regardless of whether it is new or old, is selected to proceed to the next stage of evolution.

$$X_{pop_nr_i}^{new} = \begin{cases} X_{pop_{a2}} + rand \times (X_{pop_{b2}} - X_{pop_{c2}}) + f_2 \times (X_{pop_{a2}} - X_{pop_nr_i}) & \text{if } rand > cr1 \\ X_{duili} & \text{otherwise} \end{cases} \quad (19)$$

Here, $X_{pop_{b2}}$ and $X_{pop_{c2}}$ denote one individual randomly chosen from the top 10 best-performing and the 10 worst-performing individuals, respectively; $X_{pop_{a2}}$ is an individual randomly chosen from the remaining individuals; f_2 is a number generated using Lévy flight; $cr1$ is typically set to 1/3 for better results, but it can be manually adjusted based on the specific problem; X_{duili} represents the individual generated according to Eq. (20).

$$X_{duili_{i,j}} = \begin{cases} \max wei_j + \min wei_j - X_{pop_nr_{ij}} & \text{if } rand > cr2 \\ X_{pop_nr_{ij}} & \text{otherwise} \end{cases} \quad (20)$$

Here, $\max wei_j$ and $\min wei_j$ denote the highest and lowest values of each dimension in the current population, respectively; $X'_{pop_nr_{ij}} = \max wei_j + \min wei_j - X_{pop_nr_{ij}}$ represents the opposition position of the current individual in dimension j ; $cr2$ is a parameter, typically set to 0.1 for better results, but it can be manually adjusted based on the specific problem.

In summary, the newly proposed pulmonary circulation strategy in this section offers the following advantages: First, Eq. (20) provides two types of search modes. Mode one facilitates interaction among individuals from different regions. Mode two introduces opposition positions in certain dimensions. These two modes explore more of the search space in diverse ways and provide greater diversity to the population. Second, similar to Section 4.2, the combination of Eq. (20) with the simplex method improves the algorithm's convergence speed while maintaining population diversity. Additionally, the three methods for generating individuals further enhance the algorithm's success rate in solving different optimization problems.

4.4 Diversity Supplementation Mechanism Based on External Archives

Similar to other MHAs, CSBO can encounter situations where although the overall population diversity is acceptable, distinct individuals do not change across multiple generations during the solving of complex optimization problems. In such cases, these individuals contribute almost nothing to the evolution of the current population. To improve the optimization results and avoid wasting computational resources, this section establishes an external archive for each individual and designs the following diversity supplementation mechanism based on external archives:

Initially, individuals are directly stored in their respective archives. In subsequent iterations, If the newly generated individual outperforms the original one and its fitness is different from all individuals in the external archive, this new individual is added to the external archive. To prevent the archive from becoming extremely large, every k iteration, checks if the number of individuals exceeds a threshold gd , if it does, gd individuals are randomly chosen from the archive to create a new one. Generally, $gd = 5$ and $K = 10$ can yield satisfactory results, and they can also be modified according to the specific optimization problem. If an individual has not been updated for kt generations, it is replaced with a randomly selected individual from its corresponding archive to participate in subsequent evolution. Typically, $kt = 40$ can achieve satisfactory results, but it can also be set according to specific needs.

4.5 Complexity Analysis

Assume that the population size of the ICSBO algorithm is N ; the individuals in the part of body circulation and the part of pulmonary circulation are nl and nr , respectively, and the problem dimension is D . The problem dimension of the ICSBO algorithm is D .

The CSBO algorithm consists of the following three strategies: venous blood circulation, body circulation and pulmonary circulation. The maximum time complexity for a single run of the algorithm is as follows: $6N$ multiplications and $6N$ additions for venous circulation, $2nl$ multiplications and $4nl$ additions for somatic circulation, and $2nr$ multiplications and nr additions for pulmonary circulation. The worst time complexity of the CSBO algorithm is about $8 \times O(N \times D) + 8 \times O(N)$.

The ICSBO algorithm consists of the following four main strategies: venous blood circulation, body circulation, pulmonary circulation, and external file. In the ICSBO algorithm running generation, the worst time complexity of the above links is as follows: The maximum time complexity for a single run of the algorithm is as follows: venous blood circulation needs to calculate a total of $6N$ multiplications and $6N$ additions; the body circulation needs to calculate a total of $4nl$ multiplications and $5nl$ additions; the lung circulation needs to calculate a total of $3nr$ multiplications and $5nr$ additions, and the external archive needs to calculate a total of $4N$ additions in a single iteration. additions. The worst time complexity of ICSBO is about $10 \times O(N \times D) + 14 \times O(N)$.

5 Experimental Results and Analysis

The performance of the ICSBO algorithm will be assessed through the following experiments: (1) analyzing the sensitivity of the ICSBO algorithm's parameters; (2) evaluating the impact of the four proposed enhancement strategies; (3) comparing the ICSBO algorithm's performance with that of the original CSBO algorithm and eight other representative high-performance evolutionary algorithms in both 30-dimensional and 100-dimensional spaces; (4) conducting comparative experiments between the ICSBO algorithm and the top three algorithms from the comparison set on cooperative beam optimization problems to verify the proposed algorithm's reliability.

For the experiments, the CEC2017 test suite which includes 30 benchmark functions, will be used. Functions F1 to F3 are unimodal, with F2 excluded due to its non-applicability for testing. Functions F4 to F10 are multimodal with local minima, F11 to F20 are mixed functions created by combining three or more CEC2017 benchmark functions via rotation or translation, and F20 to F29 are composite functions formed by combining at least three mixed functions or CEC2017 benchmark functions with rotation and translation. For further details on the CEC2017 test suite, refer to the provided documentation.

To maintain fairness in the comparison of algorithms, all experiments are conducted on a computer equipped with Windows 11, an Intel(R) Core(TM) i5-11400H @ 2.70 GHz CPU, and MATLAB R2021a for programming.

5.1 Parameter Sensitivity Testing

The ICSBO algorithm enhances the CSBO algorithm by introducing the following additional parameters: $f1$, cr , $cr1$, $cr2$, K , kt , and gd . Among these, $f1$ and cr use an adaptive update mechanism with restricted value ranges, while the remaining parameters are directly assigned values for participation in the algorithm's iterative process. An experiment was conducted to examine how these parameters affect the performance of the ICSBO algorithm: in each trial, only one parameter's value or range is modified, ensuring that other parameters of the algorithm remain unchanged. Each algorithm's population size is set to $N = 50$, with a test function dimensionality of $D = 30$ and a maximum function evaluation count of $\text{MaxFES} = 300,000$. Other parameters are set as follows: $\text{maxit} = 3000$, $kt = 40$, $gd = 5$, $K = 10$. To avoid the randomness of single-run operations, each algorithm is executed 30 times independently on each test function, with the mean and variance recorded; specific data are shown in Tables 1 to 4. The parameter settings for each trial are as follows:

(1) To assess the impact of the parameter $f1$ on the algorithm's performance, two sets of experiments were conducted. In the first set, when it $\leq 0.4 \times \text{maxit}$, the parameter values are chosen from the following three groups, $[0, 0.3]$ $[0.3, 0.6]$ $[0.6, 0.9]$; it $> 0.4 \times \text{maxit}$, the range of values of $f1$ is dimensioned as $[0, 1.2]$, and the second group: for it $> 0.4 \times \text{maxit}$, the range of values is taken as $[0, 0.4]$ $[0.4, 0.8]$ $[0.8, 1.2]$, and the second group: when it $> 0.4 \times \text{maxit}$, the range of values is taken as $[0, 0.4]$ $[0.4, 0.8]$ $[0.8, 1.2]$. The second group: when it $> 0.4 \times \text{maxit}$, the range of values is $[0, 0.4]$ $[0.4, 0.8]$ $[0.8, 1.2]$; when it $\leq 0.4 \times \text{maxit}$, the range of values of $f1$ is $[0, 0.8]$. The rest of the parameters of the two experiments are the same, and the values of each parameter are as follows: cr is $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 0.8]$ for it $> 0.4 \times \text{maxit}$, 0.3 for $cr1$, 0.1 for $cr2$, 10 for k , 40 for kt , and 5 for gd ;

(2) To investigate the impact of the parameter cr on the algorithm's performance, two sets of experiments are conducted. When it $\leq 0.4 \times \text{maxit}$, the values of cr are selected from the following three ranges: $[0.3, 0.5]$ $[0.5, 0.7]$ $[0.7, 0.9]$, and when it $> 0.4 \times \text{maxit}$, the range of cr is taken as $[0, 0.8]$; the second group of experiments: when it $> 0.4 \times \text{maxit}$, the range of cr is taken as $[0, 0.3]$ $[0.3, 0.6]$ $[0.6, 0.8]$; and the second group of experiments: when it $> 0.4 \times \text{maxit}$, the range of cr is taken as $[0, 0.3]$ $[0.3, 0.6]$ $[0.6, 0.8]$. The second set of experiments: when it $> 0.4 \times \text{maxit}$, the value range of cr is $[0, 0.3]$ $[0.3, 0.6]$ $[0.6, 0.8]$; when it $\leq 0.4 \times \text{maxit}$, the value range of $f1$ is $[0.5, 0.9]$. The rest of the parameters of the two experiments are the same, and the values of each parameter are as follows: $f1$ is $[0, 0.8]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 1.2]$ for it $> 0.4 \times \text{maxit}$, 0.3 for $cr1$, 0.1 for $cr2$, 10 for k , 40 for kt , and 5 for gd ;

(3) To examine the impact of the parameters $cr1$ and $cr2$ on the algorithm's performance, the following two sets of parameter values were used: 0.1, 0.6, and 0.9; 0.4, 0.7, and 1. Other than that, the other parameters were set as follows: $f1$ in the range of $[0, 0.8]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 1.2]$ for it $> 0.4 \times \text{maxit}$, $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, and $[0, 0.5, 0.9]$ for it $> 0.4 \times \text{maxit}$, and $[0, 0.5, 0.5]$ for it $> 0.4 \times \text{maxit}$, and $[0, 0.5, 0.9]$ for it $> 0.4 \times \text{maxit}$. maxit and $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 0.8]$ for it $> 0.4 \times \text{maxit}$, 10 for K , 40 for kt and 5 for gd ;

Table 1: The impact of parameter f_1 on algorithm performance

Function	[0, 0.3]	[0.3, 0.6]	[0.6, 0.9]	[0, 0.9]	[0.4, 0.8]	[0, 0.9]	[0, 0.9]	[0.8, 1.2]	[0, 0.9]
	[0, 1.2]	[0, 1.2]	[0, 1.2]	[0, 0.4]	[0.4, 0.8]	[0, 0.9]	[0.4, 0.8]	[0.8, 1.2]	[0, 1.2]
F1	4.26E - 15 ± 1.13E - 14	2.37E - 15 ± 5.39E - 15	2.84E - 15 ± 8.67E - 15	9.47E - 16 ± 5.19E - 15	0.00E + 00 ± 0.00E + 00	0.00E + 00 ± 0.00E + 00	0.00E + 00 ± 0.00E + 00	2.37E - 15 ± 6.55E - 15	4.74E - 16 ± 2.59E - 15
F3	3.03E - 14 ± 2.88E - 14	2.27E - 14 ± 2.83E - 14	3.79E - 14 ± 2.73E - 14	2.46E - 14 ± 2.86E - 14	1.89E - 14 ± 2.73E - 14	1.89E - 14 ± 2.73E - 14	1.89E - 14 ± 2.73E - 14	3.03E - 14 ± 2.88E - 14	1.89E - 14 ± 2.73E - 14
F4	9.20E + 00 ± 1.96E + 01	1.16E + 01 ± 2.30E + 01	4.17E + 01 ± 2.75E + 01	2.33E + 01 ± 2.93E + 01	1.84E + 01 ± 2.81E + 01	1.84E + 01 ± 2.81E + 01	1.84E + 01 ± 2.81E + 01	1.21E + 01 ± 2.32E + 01	1.65E + 01 ± 2.64E + 01
F5	3.04E + 01 ± 7.44E + 00	5.83E + 01 ± 3.95E + 01	6.39E + 01 ± 4.50E + 01	4.73E + 01 ± 3.69E + 01	5.26E + 01 ± 3.76E + 01	5.26E + 01 ± 3.76E + 01	5.26E + 01 ± 3.76E + 01	4.63E + 01 ± 3.36E + 01	3.97E + 01 ± 2.21E + 01
F6	3.63E - 01 ± 4.09E - 01	2.75E - 06 ± 6.93E - 06	3.20E - 04 ± 6.60E - 04	4.83E - 06 ± 1.04E - 05	2.79E - 06 ± 6.33E - 06	2.79E - 06 ± 6.33E - 06	2.79E - 06 ± 6.33E - 06	3.92E - 04 ± 2.10E - 03	1.46E - 05 ± 3.18E - 05
F7	5.83E + 01 ± 9.97E + 00	6.94E + 01 ± 1.97E + 01	8.12E + 01 ± 3.55E + 01	6.89E + 01 ± 2.28E + 01	6.98E + 01 ± 2.48E + 01	6.98E + 01 ± 2.48E + 01	6.98E + 01 ± 2.48E + 01	7.21E + 01 ± 2.94E + 01	6.51E + 01 ± 1.20E + 01
F8	3.02E + 01 ± 9.52E + 00	5.05E + 01 ± 3.53E + 01	5.80E + 01 ± 3.97E + 01	4.02E + 01 ± 2.20E + 01	5.83E + 01 ± 4.00E + 01	5.83E + 01 ± 4.00E + 01	5.83E + 01 ± 4.00E + 01	5.76E + 01 ± 4.02E + 01	4.87E + 01 ± 3.40E + 01
F9	3.63E + 01 ± 4.96E + 01	3.69E - 01 ± 4.79E - 01	2.19E + 00 ± 1.76E + 00	4.92E - 01 ± 7.16E - 01	5.23E - 01 ± 6.86E - 01	5.23E - 01 ± 6.86E - 01	5.23E - 01 ± 6.86E - 01	6.04E - 01 ± 7.98E - 01	5.20E - 01 ± 1.01E + 00
F10	2.76E + 03 ± 5.85E + 02	3.00E + 03 ± 9.36E + 02	2.78E + 03 ± 7.06E + 02	2.85E + 03 ± 7.46E + 02	2.80E + 03 ± 7.55E + 02	2.80E + 03 ± 7.55E + 02	2.80E + 03 ± 7.55E + 02	2.98E + 03 ± 1.02E + 03	2.98E + 03 ± 1.02E + 03
F11	4.74E + 01 ± 5.03E + 01	5.75E + 00 ± 2.98E + 00	9.92E + 00 ± 4.52E + 00	7.16E + 00 ± 3.27E + 00	9.46E + 00 ± 1.01E + 01	9.46E + 00 ± 1.01E + 01	9.46E + 00 ± 1.01E + 01	1.10E + 01 ± 7.04E + 00	6.87E + 00 ± 4.34E + 00
F12	7.62E + 03 ± 5.60E + 03	7.68E + 03 ± 8.23E + 03	7.08E + 03 ± 7.77E + 03	5.90E + 03 ± 6.27E + 03	6.00E + 03 ± 5.33E + 03	6.00E + 03 ± 5.33E + 03	6.00E + 03 ± 5.33E + 03	9.29E + 03 ± 8.13E + 03	7.30E + 03 ± 6.65E + 03
F13	1.15E + 02 ± 8.20E + 01	3.66E + 01 ± 1.85E + 01	2.67E + 01 ± 1.07E + 01	2.93E + 01 ± 1.61E + 01	3.05E + 01 ± 2.11E + 01	3.05E + 01 ± 2.11E + 01	3.05E + 01 ± 2.11E + 01	2.60E + 01 ± 1.46E + 01	2.57E + 01 ± 1.35E + 01
F14	6.74E + 01 ± 2.81E + 01	2.14E + 01 ± 1.03E + 01	1.61E + 01 ± 1.02E + 01	2.51E + 01 ± 8.49E + 00	1.65E + 01 ± 1.02E + 01	1.65E + 01 ± 1.02E + 01	1.65E + 01 ± 1.02E + 01	2.07E + 01 ± 1.05E + 01	1.74E + 01 ± 1.06E + 01
F15	8.17E + 01 ± 4.41E + 01	1.55E + 01 ± 1.41E + 01	7.38E + 00 ± 2.89E + 00	8.22E + 00 ± 6.26E + 00	7.36E + 00 ± 4.61E + 00	7.36E + 00 ± 4.61E + 00	7.36E + 00 ± 4.61E + 00	8.37E + 00 ± 5.18E + 00	6.75E + 00 ± 2.84E + 00
F16	3.27E + 02 ± 1.87E + 02	4.10E + 02 ± 1.83E + 02	5.42E + 02 ± 1.80E + 02	3.57E + 02 ± 1.66E + 02	4.85E + 02 ± 2.04E + 02	4.85E + 02 ± 2.04E + 02	4.85E + 02 ± 2.04E + 02	4.66E + 02 ± 1.67E + 02	4.45E + 02 ± 2.24E + 02
F17	5.90E + 01 ± 4.66E + 01	4.74E + 01 ± 2.84E + 01	5.99E + 01 ± 5.21E + 01	5.18E + 01 ± 2.12E + 01	6.31E + 01 ± 3.27E + 01	6.31E + 01 ± 3.27E + 01	6.31E + 01 ± 3.27E + 01	6.79E + 01 ± 5.21E + 01	5.02E + 01 ± 3.61E + 01
F18	7.42E + 02 ± 1.02E + 03	8.79E + 01 ± 4.80E + 01	2.30E + 01 ± 9.69E + 00	4.29E + 01 ± 3.98E + 01	2.90E + 01 ± 1.26E + 01	2.90E + 01 ± 1.26E + 01	2.90E + 01 ± 1.26E + 01	3.25E + 01 ± 1.97E + 01	3.25E + 01 ± 1.55E + 01
F19	5.99E + 01 ± 3.95E + 01	1.07E + 01 ± 7.67E + 00	1.04E + 01 ± 2.92E + 00	9.04E + 00 ± 3.08E + 00	9.37E + 00 ± 2.59E + 00	9.37E + 00 ± 2.59E + 00	9.37E + 00 ± 2.59E + 00	1.06E + 01 ± 3.96E + 00	8.40E + 00 ± 2.81E + 00
F20	9.95E + 01 ± 6.11E + 01	7.25E + 01 ± 6.86E + 01	8.84E + 01 ± 6.58E + 01	8.51E + 01 ± 6.98E + 01	5.81E + 01 ± 5.86E + 01	5.81E + 01 ± 5.86E + 01	5.81E + 01 ± 5.86E + 01	1.02E + 02 ± 6.91E + 01	5.80E + 01 ± 5.83E + 01
F21	2.32E + 02 ± 1.05E + 01	2.85E + 02 ± 5.74E + 01	3.05E + 02 ± 5.61E + 01	2.86E + 02 ± 5.13E + 01	3.10E + 02 ± 4.78E + 01	3.10E + 02 ± 4.78E + 01	3.10E + 02 ± 4.78E + 01	3.14E + 02 ± 5.12E + 01	2.86E + 02 ± 5.68E + 01
F22	1.01E + 02 ± 1.78E + 00	1.00E + 02 ± 7.49E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 0.00E + 00
F23	3.97E + 02 ± 1.26E + 01	3.93E + 02 ± 4.60E + 01	4.23E + 02 ± 5.18E + 01	3.81E + 02 ± 3.11E + 01	3.90E + 02 ± 3.86E + 01	3.90E + 02 ± 3.86E + 01	3.90E + 02 ± 3.86E + 01	3.90E + 02 ± 3.93E + 01	3.83E + 02 ± 3.30E + 01
F24	4.77E + 02 ± 1.53E + 01	4.41E + 02 ± 2.09E + 01	4.55E + 02 ± 3.03E + 01	4.40E + 02 ± 6.97E + 00	4.39E + 02 ± 8.77E + 00	4.39E + 02 ± 8.77E + 00	4.39E + 02 ± 8.77E + 00	4.49E + 02 ± 3.21E + 01	4.37E + 02 ± 8.62E + 00
F25	3.93E + 02 ± 1.45E + 01	3.87E + 02 ± 1.95E + 00	3.87E + 02 ± 1.28E - 01	3.87E + 02 ± 2.06E + 00	3.87E + 02 ± 1.08E + 00	3.87E + 02 ± 1.08E + 00	3.87E + 02 ± 1.08E + 00	3.87E + 02 ± 1.93E + 00	3.87E + 02 ± 1.11E + 00
F26	1.07E + 03 ± 6.98E + 02	1.11E + 03 ± 3.65E + 02	1.28E + 03 ± 3.22E + 02	1.11E + 03 ± 2.72E + 02	1.14E + 03 ± 3.38E + 02	1.14E + 03 ± 3.38E + 02	1.14E + 03 ± 3.38E + 02	1.19E + 03 ± 1.64E + 02	1.11E + 03 ± 2.97E + 02
F27	5.38E + 02 ± 1.91E + 01	4.99E + 02 ± 9.48E + 00	5.00E + 02 ± 7.34E + 00	4.99E + 02 ± 1.09E + 01	4.98E + 02 ± 9.25E + 00	4.98E + 02 ± 9.25E + 00	4.98E + 02 ± 9.25E + 00	4.99E + 02 ± 8.85E + 00	4.98E + 02 ± 9.68E + 00
F28	3.18E + 02 ± 4.00E + 01	3.47E + 02 ± 6.03E + 01	3.22E + 02 ± 4.43E + 01	3.25E + 02 ± 4.58E + 01	3.42E + 02 ± 5.64E + 01	3.42E + 02 ± 5.64E + 01	3.42E + 02 ± 5.64E + 01	3.35E + 02 ± 5.56E + 01	3.31E + 02 ± 4.87E + 01
F29	5.29E + 02 ± 9.25E + 01	4.65E + 02 ± 3.32E + 01	5.37E + 02 ± 7.35E + 01	5.02E + 02 ± 6.40E + 01	4.85E + 02 ± 5.25E + 01	4.85E + 02 ± 5.25E + 01	4.85E + 02 ± 5.25E + 01	4.79E + 02 ± 3.47E + 01	4.69E + 02 ± 3.73E + 01
F30	2.23E + 03 ± 2.81E + 02	2.18E + 03 ± 1.46E + 02	2.19E + 03 ± 1.05E + 02	2.16E + 03 ± 1.21E + 02	2.19E + 03 ± 1.13E + 02	2.19E + 03 ± 1.13E + 02	2.19E + 03 ± 1.13E + 02	2.37E + 03 ± 6.27E + 02	2.16E + 03 ± 1.12E + 02

Table 2: The impact of parameter α on algorithm performance

Function	[0.3, 0.5]	[0.5, 0.7]	[0.7, 0.9]	[0.5, 0.9]	[0.3, 0.6]	[0.5, 0.9]	[0.6, 0.8]	[0.5, 0.9]
	[0, 0.8]	[0, 0.8]	[0, 0.8]	[0, 0.3]	[0.3, 0.6]	[0.5, 0.9]	[0.6, 0.8]	[0.5, 0.9]
F1	4.74E - 16 ± 2.59E - 15	9.47E - 16 ± 3.61E - 15	0.00E + 00 ± 0.00E + 00	0.00E + 00 ± 0.00E + 00	3.32E - 15 ± 6.11E - 15	9.47E - 16 ± 3.61E - 15	4.74E - 16 ± 2.59E - 15	4.74E - 16 ± 2.59E - 15
F3	2.46E - 14 ± 2.86E - 14	2.27E - 14 ± 2.83E - 14	3.41E - 14 ± 2.83E - 14	3.60E - 14 ± 2.79E - 14	3.41E - 14 ± 2.83E - 14	1.52E - 14 ± 2.56E - 14	1.89E - 14 ± 2.73E - 14	1.89E - 14 ± 2.73E - 14
F4	1.77E + 01 ± 2.75E + 01	1.64E + 01 ± 2.74E + 01	2.23E + 01 ± 2.79E + 01	2.60E + 01 ± 2.98E + 01	1.99E + 01 ± 2.81E + 01	1.83E + 01 ± 2.79E + 01	1.65E + 01 ± 2.64E + 01	1.65E + 01 ± 2.64E + 01
F5	3.73E + 01 ± 2.82E + 01	3.75E + 01 ± 2.70E + 01	4.81E + 01 ± 3.56E + 01	2.97E + 01 ± 8.25E + 00	6.83E + 01 ± 4.69E + 01	8.82E + 01 ± 4.93E + 01	3.97E + 01 ± 2.21E + 01	3.97E + 01 ± 2.21E + 01
F6	1.41E - 02 ± 2.63E - 02	1.25E - 04 ± 3.58E - 04	5.56E - 06 ± 1.17E - 05	1.91E - 04 ± 4.91E - 04	9.84E - 06 ± 2.50E - 05	3.06E - 05 ± 9.13E - 05	1.46E - 05 ± 3.18E - 05	1.46E - 05 ± 3.18E - 05
F7	6.06E + 01 ± 8.04E + 00	7.17E + 01 ± 3.70E + 01	6.82E + 01 ± 1.83E + 01	6.34E + 01 ± 1.23E + 01	1.12E + 02 ± 4.94E + 01	1.37E + 02 ± 4.12E + 01	6.51E + 01 ± 1.20E + 01	6.51E + 01 ± 1.20E + 01
F8	2.92E + 01 ± 8.06E + 00	4.56E + 01 ± 4.15E + 01	4.62E + 01 ± 3.77E + 01	3.45E + 01 ± 1.17E + 01	7.73E + 01 ± 5.16E + 01	1.13E + 02 ± 4.00E + 01	4.87E + 01 ± 3.40E + 01	4.87E + 01 ± 3.40E + 01
F9	2.14E + 00 ± 1.87E + 00	9.50E - 01 ± 1.15E + 00	3.75E - 01 ± 6.38E - 01	1.14E + 00 ± 1.67E + 00	1.90E - 01 ± 3.05E - 01	3.44E - 01 ± 5.56E - 01	5.20E - 01 ± 1.01E + 00	5.20E - 01 ± 1.01E + 00
F10	3.83E + 03 ± 1.42E + 03	3.36E + 03 ± 1.22E + 03	2.37E + 03 ± 5.24E + 02	2.64E + 03 ± 4.76E + 02	4.23E + 03 ± 6.95E + 02	4.31E + 03 ± 4.31E + 02	2.64E + 03 ± 6.06E + 02	2.64E + 03 ± 6.06E + 02
F11	8.59E + 00 ± 5.84E + 00	7.53E + 00 ± 5.22E + 00	6.17E + 00 ± 2.53E + 00	5.77E + 00 ± 3.00E + 00	8.73E + 00 ± 8.58E + 00	1.11E + 01 ± 1.08E + 01	6.87E + 00 ± 4.34E + 00	6.87E + 00 ± 4.34E + 00
F12	8.29E + 03 ± 7.34E + 03	5.21E + 03 ± 4.77E + 03	5.42E + 03 ± 4.50E + 03	7.63E + 03 ± 5.85E + 03	6.73E + 03 ± 5.62E + 03	6.71E + 03 ± 8.44E + 03	7.30E + 03 ± 6.65E + 03	7.30E + 03 ± 6.65E + 03
F13	5.19E + 01 ± 3.19E + 01	4.90E + 01 ± 4.38E + 01	2.60E + 01 ± 1.26E + 01	3.70E + 01 ± 2.08E + 01	2.76E + 01 ± 1.69E + 01	2.78E + 01 ± 1.07E + 01	2.57E + 01 ± 1.35E + 01	2.57E + 01 ± 1.35E + 01
F14	2.47E + 01 ± 6.73E + 00	1.96E + 01 ± 1.02E + 01	1.51E + 01 ± 1.01E + 01	2.73E + 01 ± 4.39E + 00	2.20E + 01 ± 1.26E + 01	2.00E + 01 ± 1.26E + 01	1.74E + 01 ± 1.06E + 01	1.74E + 01 ± 1.06E + 01
F15	2.68E + 01 ± 2.44E + 01	2.93E + 01 ± 3.98E + 01	7.91E + 00 ± 7.11E + 00	1.47E + 01 ± 1.16E + 01	6.83E + 00 ± 4.28E + 00	7.77E + 00 ± 5.52E + 00	6.75E + 00 ± 2.84E + 00	6.75E + 00 ± 2.84E + 00
F16	4.30E + 02 ± 2.47E + 02	4.60E + 02 ± 2.65E + 02	4.63E + 02 ± 1.69E + 02	4.26E + 02 ± 2.22E + 02	5.45E + 02 ± 1.98E + 02	5.51E + 02 ± 2.08E + 02	4.45E + 02 ± 2.24E + 02	4.45E + 02 ± 2.24E + 02
F17	7.67E + 01 ± 9.54E + 01	4.48E + 01 ± 2.97E + 01	5.13E + 01 ± 1.87E + 01	6.02E + 01 ± 5.12E + 01	8.54E + 01 ± 4.64E + 01	1.13E + 02 ± 5.71E + 01	5.02E + 01 ± 3.61E + 01	5.02E + 01 ± 3.61E + 01
F18	6.05E + 01 ± 5.31E + 01	4.47E + 01 ± 3.06E + 01	3.01E + 01 ± 1.80E + 01	5.59E + 01 ± 3.39E + 01	4.29E + 01 ± 3.78E + 01	3.53E + 01 ± 1.48E + 01	3.25E + 01 ± 1.55E + 01	3.25E + 01 ± 1.55E + 01
F19	1.55E + 01 ± 1.09E + 01	1.38E + 01 ± 1.62E + 01	8.89E + 00 ± 2.76E + 00	1.37E + 01 ± 9.08E + 00	1.07E + 01 ± 3.78E + 00	8.48E + 00 ± 3.57E + 00	8.40E + 00 ± 2.81E + 00	8.40E + 00 ± 2.81E + 00
F20	9.81E + 01 ± 9.07E + 01	6.60E + 01 ± 5.68E + 01	7.41E + 01 ± 5.62E + 01	8.66E + 01 ± 7.39E + 01	1.26E + 02 ± 9.51E + 01	1.61E + 02 ± 8.48E + 01	5.80E + 01 ± 5.83E + 01	5.80E + 01 ± 5.83E + 01
F21	2.34E + 02 ± 3.63E + 01	2.62E + 02 ± 5.34E + 01	2.80E + 02 ± 4.87E + 01	2.28E + 02 ± 1.21E + 01	3.30E + 02 ± 2.38E + 01	3.21E + 02 ± 2.95E + 01	2.86E + 02 ± 5.68E + 01	2.86E + 02 ± 5.68E + 01
F22	1.00E + 02 ± 7.47E - 01	1.00E + 02 ± 6.29E - 01	1.00E + 02 ± 5.62E + 00	1.00E + 02 ± 8.68E - 01	1.00E + 02 ± 7.07E - 01	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 0.00E + 00
F23	3.75E + 02 ± 1.36E + 01	3.72E + 02 ± 2.45E + 01	4.03E + 02 ± 4.57E + 01	3.79E + 02 ± 3.42E + 01	4.17E + 02 ± 4.98E + 01	4.06E + 02 ± 4.58E + 01	3.83E + 02 ± 3.30E + 01	3.83E + 02 ± 3.30E + 01
F24	4.41E + 02 ± 8.85E + 00	4.40E + 02 ± 9.39E + 00	4.36E + 02 ± 9.72E + 00	4.43E + 02 ± 1.15E + 01	4.44E + 02 ± 8.50E + 00	4.43E + 02 ± 1.15E + 01	4.39E + 02 ± 8.62E + 00	4.39E + 02 ± 8.62E + 00
F25	3.87E + 02 ± 2.10E + 00	3.87E + 02 ± 1.23E + 00	3.88E + 02 ± 6.74E + 00	3.87E + 02 ± 9.31E - 01	3.87E + 02 ± 2.54E - 01	3.87E + 02 ± 6.83E - 01	3.87E + 02 ± 1.11E + 00	3.87E + 02 ± 1.11E + 00
F26	1.23E + 03 ± 1.57E + 02	9.64E + 02 ± 3.45E + 02	1.05E + 03 ± 4.31E + 02	1.06E + 03 ± 2.73E + 02	1.11E + 03 ± 2.82E + 02	1.12E + 03 ± 3.18E + 02	1.11E + 03 ± 2.97E + 02	1.11E + 03 ± 2.97E + 02
F27	5.06E + 02 ± 9.25E + 00	5.00E + 02 ± 8.68E + 00	4.99E + 02 ± 1.03E + 01	5.03E + 02 ± 1.11E + 01	5.03E + 02 ± 9.04E + 00	4.98E + 02 ± 1.03E + 01	4.98E + 02 ± 9.68E + 00	4.98E + 02 ± 9.68E + 00
F28	3.45E + 02 ± 6.22E + 01	3.25E + 02 ± 4.65E + 01	3.43E + 02 ± 5.91E + 01	3.23E + 02 ± 4.72E + 01	3.35E + 02 ± 5.50E + 01	3.28E + 02 ± 4.77E + 01	3.31E + 02 ± 4.87E + 01	3.31E + 02 ± 4.87E + 01
F29	4.95E + 02 ± 6.57E + 01	4.79E + 02 ± 3.67E + 01	4.80E + 02 ± 5.42E + 01	4.71E + 02 ± 2.86E + 01	4.92E + 02 ± 7.36E + 01	4.95E + 02 ± 8.85E + 01	4.69E + 02 ± 3.73E + 01	4.69E + 02 ± 3.73E + 01
F30	2.20E + 03 ± 1.57E + 02	2.20E + 03 ± 1.52E + 02	2.19E + 03 ± 1.80E + 02	2.29E + 03 ± 6.39E + 02	2.38E + 03 ± 7.20E + 02	2.22E + 03 ± 4.27E + 02	2.16E + 03 ± 1.12E + 02	2.16E + 03 ± 1.12E + 02

Table 3: The impact of parameter $cr1$, $cr2$ on algorithm performance

Function	cr1 = 0.1		cr1 = 0.6		cr1 = 0.9		cr1 = 0.3		cr1 = 0.3		cr1 = 0.3	
	cr2 = 0.1		cr2 = 0.1		cr2 = 0.1		cr2 = 0.4		cr2 = 0.7		cr2 = 1	
F1	3.79E - 15 ± 1.12E - 14	9.47E - 16 ± 3.61E - 15	0.00E + 00 ± 0.00E + 00	5.21E - 15 ± 1.85E - 14	4.26E - 15 ± 6.62E - 15	1.42E - 14 ± 7.46E - 15	4.74E - 16 ± 2.59E - 15					
F3	4.55E - 14 ± 2.31E - 14	1.71E - 14 ± 2.65E - 14	5.68E - 15 ± 1.73E - 14	2.08E - 14 ± 2.79E - 14	3.79E - 14 ± 2.73E - 14	5.68E - 14 ± 2.73E - 14	1.89E - 14 ± 2.73E - 14					
F4	9.37E + 00 ± 2.08E + 01	1.35E + 01 ± 2.49E + 01	1.52E + 01 ± 2.58E + 01	4.13E + 00 ± 1.15E + 01	1.59E + 01 ± 2.65E + 01	2.33E + 01 ± 3.00E + 01	1.65E + 01 ± 2.64E + 01					
F5	4.47E + 01 ± 3.12E + 01	5.32E + 01 ± 3.75E + 01	4.71E + 01 ± 3.27E + 01	4.01E + 01 ± 1.91E + 01	3.66E + 01 ± 1.90E + 01	4.96E + 01 ± 3.64E + 01	3.97E + 01 ± 2.21E + 01					
F6	2.51E - 05 ± 6.41E - 05	4.40E - 05 ± 2.03E - 04	5.91E - 06 ± 9.92E - 06	1.08E - 05 ± 2.68E - 05	3.74E - 05 ± 1.44E - 04	4.42E - 05 ± 2.10E - 04	1.46E - 05 ± 3.18E - 05					
F7	6.74E + 01 ± 1.54E + 01	7.13E + 01 ± 3.20E + 01	6.67E + 01 ± 2.17E + 01	6.91E + 01 ± 1.70E + 01	7.49E + 01 ± 3.38E + 01	6.90E + 01 ± 2.16E + 01	6.51E + 01 ± 1.20E + 01					
F8	4.38E + 01 ± 2.82E + 01	6.52E + 01 ± 4.61E + 01	4.62E + 01 ± 3.81E + 01	5.57E + 01 ± 3.89E + 01	5.08E + 01 ± 3.92E + 01	4.32E + 01 ± 2.89E + 01	4.87E + 01 ± 3.40E + 01					
F9	3.60E - 01 ± 4.09E - 01	3.27E - 01 ± 5.07E - 01	3.14E - 01 ± 3.64E - 01	3.54E - 01 ± 6.55E - 01	4.17E - 01 ± 8.75E - 01	9.93E - 01 ± 1.14E + 00	5.20E - 01 ± 1.01E + 00					
F10	2.72E + 03 ± 8.07E + 02	2.74E + 03 ± 5.68E + 02	2.53E + 03 ± 5.09E + 02	2.43E + 03 ± 5.31E + 02	2.72E + 03 ± 6.91E + 02	2.54E + 03 ± 4.76E + 02	2.64E + 03 ± 6.06E + 02					
F11	6.67E + 00 ± 3.81E + 00	6.67E + 00 ± 3.80E + 00	8.56E + 00 ± 1.02E + 01	6.64E + 00 ± 3.65E + 00	8.53E + 00 ± 1.02E + 01	8.87E + 00 ± 5.45E + 00	6.87E + 00 ± 4.34E + 00					
F12	5.19E + 03 ± 3.70E + 03	7.75E + 03 ± 6.83E + 03	5.34E + 03 ± 4.81E + 03	7.63E + 03 ± 8.10E + 03	7.61E + 03 ± 5.45E + 03	5.87E + 03 ± 4.42E + 03	7.30E + 03 ± 6.65E + 03					
F13	4.24E + 01 ± 5.84E + 01	2.30E + 01 ± 9.01E + 00	2.09E + 01 ± 1.07E + 01	3.72E + 01 ± 2.30E + 01	3.34E + 01 ± 1.81E + 01	4.84E + 01 ± 8.19E + 01	2.57E + 01 ± 1.35E + 01					
F14	1.71E + 01 ± 9.56E + 00	1.50E + 01 ± 9.06E + 00	1.80E + 01 ± 1.11E + 01	1.57E + 01 ± 1.02E + 01	1.91E + 01 ± 1.02E + 01	1.86E + 01 ± 1.08E + 01	1.74E + 01 ± 1.06E + 01					
F15	9.70E + 00 ± 9.17E + 00	7.32E + 00 ± 4.52E + 00	7.01E + 00 ± 5.73E + 00	1.14E + 01 ± 1.12E + 01	8.70E + 00 ± 5.18E + 00	9.02E + 00 ± 8.43E + 00	6.75E + 00 ± 2.84E + 00					
F16	3.95E + 02 ± 1.98E + 02	4.74E + 02 ± 2.13E + 02	4.18E + 02 ± 1.94E + 02	4.54E + 02 ± 1.86E + 02	3.83E + 02 ± 1.84E + 02	4.39E + 02 ± 2.06E + 02	4.45E + 02 ± 2.24E + 02					
F17	5.63E + 01 ± 5.09E + 01	4.46E + 01 ± 1.98E + 01	4.77E + 01 ± 3.52E + 01	4.67E + 01 ± 1.79E + 01	5.58E + 01 ± 2.60E + 01	4.97E + 01 ± 2.14E + 01	5.02E + 01 ± 3.61E + 01					
F18	3.60E + 01 ± 1.64E + 01	3.78E + 01 ± 3.19E + 01	3.38E + 01 ± 1.38E + 01	4.41E + 01 ± 4.21E + 01	3.94E + 01 ± 2.04E + 01	3.30E + 01 ± 1.55E + 01	3.25E + 01 ± 1.55E + 01					
F19	9.03E + 00 ± 2.86E + 00	8.49E + 00 ± 2.45E + 00	8.63E + 00 ± 2.59E + 00	8.51E + 00 ± 2.80E + 00	9.90E + 00 ± 3.97E + 00	9.32E + 00 ± 2.88E + 00	8.40E + 00 ± 2.81E + 00					
F20	9.09E + 01 ± 6.23E + 01	7.29E + 01 ± 6.50E + 01	1.07E + 02 ± 6.77E + 01	8.95E + 01 ± 7.99E + 01	8.83E + 01 ± 6.90E + 01	1.01E + 02 ± 7.96E + 01	5.80E + 01 ± 5.83E + 01					
F21	2.90E + 02 ± 5.11E + 01	2.84E + 02 ± 5.11E + 01	3.07E + 02 ± 4.84E + 01	2.85E + 02 ± 5.57E + 01	2.90E + 02 ± 5.41E + 01	2.83E + 02 ± 5.36E + 01	2.86E + 02 ± 5.68E + 01					
F22	1.00E + 02 ± 6.65E - 01	1.00E + 02 ± 6.23E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 6.23E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 0.00E + 00					
F23	3.97E + 02 ± 4.72E + 01	4.00E + 02 ± 4.85E + 01	3.94E + 02 ± 3.80E + 01	3.91E + 02 ± 3.80E + 01	4.00E + 02 ± 4.01E + 01	3.91E + 02 ± 4.49E + 01	3.83E + 02 ± 3.30E + 01					
F24	4.41E + 02 ± 1.09E + 01	4.48E + 02 ± 3.13E + 01	4.36E + 02 ± 8.77E + 00	4.38E + 02 ± 1.06E + 01	4.37E + 02 ± 9.68E + 00	4.44E + 02 ± 2.35E + 01	4.39E + 02 ± 8.62E + 00					
F25	3.87E + 02 ± 1.98E + 00	3.87E + 02 ± 1.97E + 00	3.87E + 02 ± 6.85E - 01	3.87E + 02 ± 1.28E + 00	3.87E + 02 ± 1.92E + 00	3.87E + 02 ± 1.26E + 00	3.87E + 02 ± 1.11E + 00					
F26	1.02E + 03 ± 3.38E + 02	1.17E + 03 ± 1.32E + 02	1.04E + 03 ± 3.77E + 02	1.21E + 03 ± 1.66E + 02	1.07E + 03 ± 3.48E + 02	1.13E + 03 ± 2.92E + 02	1.11E + 03 ± 2.97E + 02					
F27	4.96E + 02 ± 9.54E + 00	5.00E + 02 ± 8.48E + 00	4.98E + 02 ± 1.11E + 01	5.04E + 02 ± 6.90E + 00	4.99E + 02 ± 9.49E + 00	4.96E + 02 ± 9.60E + 00	4.98E + 02 ± 9.68E + 00					
F28	3.30E + 02 ± 5.18E + 01	3.24E + 02 ± 4.51E + 01	3.23E + 02 ± 4.67E + 01	3.40E + 02 ± 5.48E + 01	3.34E + 02 ± 5.43E + 01	3.27E + 02 ± 5.08E + 01	3.31E + 02 ± 4.87E + 01					
F29	4.73E + 02 ± 3.34E + 01	4.70E + 02 ± 3.02E + 01	4.79E + 02 ± 4.96E + 01	4.77E + 02 ± 4.10E + 01	4.78E + 02 ± 3.43E + 01	4.72E + 02 ± 4.59E + 01	4.69E + 02 ± 3.73E + 01					
F30	2.18E + 03 ± 1.26E + 02	2.17E + 03 ± 1.18E + 02	2.17E + 03 ± 1.78E + 02	2.21E + 03 ± 1.53E + 02	2.18E + 03 ± 1.05E + 02	2.19E + 03 ± 1.44E + 02	2.16E + 03 ± 1.12E + 02					

Table 4: The impact of parameter K , kt , gd on algorithm performance

Function	K = 5, kt = 40; gd = 5	K = 20, kt = 40; gd = 5	K = 30, kt = 40; gd = 5	k = 10; kt = 20; gd = 5	k = 10; kt = 60; gd = 5	k = 10; kt = 80; gd = 5	K = 10; kt = 40; gd = 10	K = 10; kt = 40; gd = 20	K = 10; kt = 40; gd = 30	K = 10; kt = 40; gd = 5
F1	1.42E - 15 ± 4.34E - 15	6.16E - 15 ± 7.16E - 15	0.00E + 00 ± 0.00E + 00	4.74E - 16 ± 2.59E - 15	1.89E - 15 ± 6.17E - 15	1.89E - 15 ± 4.91E - 15	4.74E - 16 ± 2.59E - 15	0.00E + 00 ± 0.00E + 00	4.74E - 16 ± 2.59E - 15	4.74E - 16 ± 2.59E - 15
F3	3.22E - 14 ± 2.86E - 14	4.55E - 14 ± 2.31E - 14	1.91E - 14 ± 2.65E - 14	2.08E - 14 ± 2.79E - 14	3.41E - 14 ± 2.83E - 14	4.93E - 14 ± 1.97E - 14	2.65E - 14 ± 2.88E - 14	2.08E - 14 ± 2.79E - 14	5.68E - 15 ± 1.73E - 14	1.89E - 14 ± 2.73E - 14
F4	9.83E + 00 ± 2.12E + 01	1.62E + 01 ± 2.62E + 01	9.51E + 00 ± 2.08E + 01	1.79E + 01 ± 2.72E + 01	1.83E + 01 ± 2.80E + 01	1.17E + 01 ± 2.38E + 01	1.57E + 01 ± 2.59E + 01	1.38E + 01 ± 2.33E + 01	1.31E + 01 ± 2.41E + 01	1.65E + 01 ± 2.64E + 01
F5	5.23E + 01 ± 3.60E + 01	5.45E + 01 ± 3.98E + 01	4.99E + 01 ± 4.00E + 01	9.00E + 01 ± 4.98E + 01	3.83E + 01 ± 2.39E + 01	4.10E + 01 ± 1.04E + 01	5.06E + 01 ± 3.71E + 01	6.92E + 01 ± 5.50E + 01	4.89E + 01 ± 3.37E + 01	3.97E + 01 ± 2.21E + 01
F6	1.23E - 05 ± 5.19E - 05	3.90E - 06 ± 7.71E - 06	1.96E - 05 ± 8.01E - 05	6.19E - 05 ± 2.35E - 04	6.77E - 06 ± 1.66E - 05	5.64E - 06 ± 1.07E - 05	5.74E - 06 ± 1.28E - 05	1.53E - 05 ± 4.91E - 05	9.85E - 06 ± 2.36E - 05	1.46E - 05 ± 3.18E - 05
F7	7.14E + 01 ± 4.41E + 01	6.84E + 01 ± 2.08E + 01	8.12E + 01 ± 4.46E + 01	9.98E + 01 ± 4.94E + 01	7.09E + 01 ± 2.13E + 01	6.76E + 01 ± 1.13E + 01	7.48E + 01 ± 2.74E + 01	9.10E + 01 ± 4.30E + 01	7.85E + 01 ± 3.65E + 01	6.51E + 01 ± 1.20E + 01
F8	4.41E + 01 ± 3.28E + 01	4.96E + 01 ± 3.80E + 01	7.39E + 01 ± 5.24E + 01	6.62E + 01 ± 4.79E + 01	4.40E + 01 ± 1.78E + 01	4.03E + 01 ± 2.11E + 01	4.26E + 01 ± 2.96E + 01	6.47E + 01 ± 5.10E + 01	6.15E + 01 ± 4.50E + 01	4.87E + 01 ± 3.40E + 01
F9	4.63E - 01 ± 8.21E - 01	5.21E - 01 ± 7.48E - 01	5.80E - 01 ± 1.17E + 00	3.99E - 01 ± 4.98E - 01	4.17E - 01 ± 6.24E - 01	4.02E - 01 ± 5.04E - 01	2.24E - 01 ± 5.14E - 01	3.36E - 01 ± 4.79E - 01	5.48E - 01 ± 1.38E + 00	5.20E - 01 ± 1.01E + 00
F10	2.61E + 03 ± 5.07E + 02	2.82E + 03 ± 8.54E + 02	3.17E + 03 ± 1.08E + 03	3.42E + 03 ± 1.02E + 03	2.53E + 03 ± 4.15E + 02	2.40E + 03 ± 4.61E + 02	2.97E + 03 ± 1.14E + 03	3.70E + 03 ± 1.12E + 03	4.03E + 03 ± 1.14E + 03	2.64E + 03 ± 6.06E + 02
F11	8.22E + 00 ± 7.33E + 00	7.83E + 00 ± 3.76E + 00	6.77E + 00 ± 3.59E + 00	7.90E + 00 ± 1.01E + 01	7.00E + 00 ± 3.43E + 00	7.26E + 00 ± 3.75E + 00	6.54E + 00 ± 2.85E + 00	7.66E + 00 ± 3.95E + 00	7.03E + 00 ± 3.63E + 00	6.87E + 00 ± 4.34E + 00
F12	6.40E + 03 ± 5.96E + 03	8.79E + 03 ± 6.61E + 03	4.88E + 03 ± 4.00E + 03	8.75E + 03 ± 7.32E + 03	6.74E + 03 ± 7.09E + 03	5.12E + 03 ± 3.86E + 03	6.97E + 03 ± 6.02E + 03	7.09E + 03 ± 5.95E + 03	7.23E + 03 ± 6.24E + 03	7.30E + 03 ± 6.65E + 03
F13	3.19E + 01 ± 1.84E + 01	3.27E + 01 ± 1.62E + 01	2.41E + 01 ± 8.88E + 00	1.87E + 01 ± 8.03E + 00	2.89E + 01 ± 1.06E + 01	2.93E + 01 ± 1.04E + 01	2.17E + 01 ± 7.60E + 00	2.55E + 01 ± 8.51E + 00	2.61E + 01 ± 1.30E + 01	2.57E + 01 ± 1.35E + 01
F14	1.81E + 01 ± 9.81E + 00	1.77E + 01 ± 9.75E + 00	1.30E + 01 ± 8.45E + 00	1.92E + 01 ± 1.08E + 01	2.16E + 01 ± 1.21E + 01	2.40E + 01 ± 9.67E + 00	1.45E + 01 ± 1.10E + 01	1.24E + 01 ± 8.88E + 00	1.05E + 01 ± 7.15E + 00	1.74E + 01 ± 1.06E + 01
F15	7.13E + 00 ± 3.38E + 00	1.13E + 01 ± 7.84E + 00	8.32E + 00 ± 7.65E + 00	4.31E + 00 ± 2.41E + 00	1.02E + 01 ± 5.85E + 00	1.08E + 01 ± 6.66E + 00	6.95E + 00 ± 3.12E + 00	9.09E + 00 ± 5.04E + 00	9.14E + 00 ± 5.77E + 00	6.75E + 00 ± 2.84E + 00
F16	4.70E + 02 ± 2.01E + 02	4.83E + 02 ± 1.69E + 02	4.59E + 02 ± 2.26E + 02	5.07E + 02 ± 2.10E + 02	4.50E + 02 ± 2.05E + 02	4.56E + 02 ± 2.00E + 02	4.77E + 02 ± 1.99E + 02	4.89E + 02 ± 2.76E + 02	4.61E + 02 ± 1.91E + 02	4.45E + 02 ± 2.24E + 02
F17	6.32E + 01 ± 4.28E + 01	4.30E + 01 ± 2.10E + 01	4.24E + 01 ± 2.39E + 01	7.84E + 01 ± 4.07E + 01	4.94E + 01 ± 2.38E + 01	6.79E + 01 ± 3.76E + 01	5.62E + 01 ± 4.71E + 01	5.64E + 01 ± 5.61E + 01	6.55E + 01 ± 5.78E + 01	5.02E + 01 ± 3.61E + 01
F18	3.17E + 01 ± 1.26E + 01	3.26E + 01 ± 1.78E + 01	2.55E + 01 ± 1.18E + 01	2.89E + 01 ± 8.79E + 00	3.88E + 01 ± 2.33E + 01	3.24E + 01 ± 1.60E + 01	3.10E + 01 ± 2.15E + 01	2.96E + 01 ± 7.96E + 00	2.96E + 01 ± 9.41E + 00	3.25E + 01 ± 1.55E + 01
F19	8.30E + 00 ± 2.58E + 00	9.28E + 00 ± 2.69E + 00	8.61E + 00 ± 2.53E + 00	9.00E + 00 ± 2.88E + 00	9.40E + 00 ± 2.31E + 00	8.97E + 00 ± 2.11E + 00	8.22E + 00 ± 2.58E + 00	8.22E + 00 ± 2.16E + 00	8.44E + 00 ± 3.01E + 00	8.40E + 00 ± 2.81E + 00
F20	8.39E + 01 ± 6.34E + 01	5.87E + 01 ± 5.81E + 01	1.03E + 02 ± 6.79E + 01	9.61E + 01 ± 8.09E + 01	7.55E + 01 ± 6.05E + 01	8.17E + 01 ± 6.39E + 01	8.28E + 01 ± 6.82E + 01	7.74E + 01 ± 7.00E + 01	8.78E + 01 ± 7.62E + 01	5.80E + 01 ± 5.83E + 01
F21	2.93E + 02 ± 5.48E + 01	2.98E + 02 ± 5.41E + 01	2.97E + 02 ± 5.55E + 01	3.33E + 02 ± 3.66E + 01	2.75E + 02 ± 5.26E + 01	2.51E + 02 ± 3.92E + 01	2.91E + 02 ± 5.14E + 01	3.00E + 02 ± 5.77E + 01	2.94E + 02 ± 5.74E + 01	2.86E + 02 ± 5.68E + 01
F22	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 4.49E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 6.23E - 01	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 6.22E - 01	1.00E + 02 ± 1.06E + 00	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 7.40E - 01	1.00E + 02 ± 0.00E + 00
F23	3.89E + 02 ± 6.62E + 01	4.01E + 02 ± 4.55E + 01	3.89E + 02 ± 4.04E + 01	4.26E + 02 ± 5.79E + 01	3.80E + 02 ± 2.17E + 01	3.77E + 02 ± 1.21E + 01	3.94E + 02 ± 4.87E + 01	3.92E + 02 ± 4.35E + 01	3.87E + 02 ± 3.89E + 01	3.83E + 02 ± 3.30E + 01

(Continued)

Table 4 (continued)

Function	K = 5, kt = 40; gd = 5	K = 20, kt = 40; gd = 5	K = 30, kt = 40; gd = 5	k = 10; kt = 20; gd = 5	k = 10; kt = 60; gd = 5	k = 10; kt = 80; gd = 5	K = 10; kt = 40; gd = 10	K = 10; kt = 40; gd = 20	K = 10; kt = 40; gd = 30	K = 10; kt = 40; gd = 5
F24	4.42E + 02 ± 2.13E + 01	4.40E + 02 ± 9.07E + 00	4.39E + 02 ± 9.56E + 00	4.36E + 02 ± 8.51E + 00	4.40E + 02 ± 9.13E + 00	4.39E + 02 ± 7.80E + 00	4.41E + 02 ± 8.80E + 00	4.38E + 02 ± 7.69E + 00	4.41E + 02 ± 1.10E + 01	4.39E + 02 ± 8.62E + 00
F25	3.87E + 02 ±	3.87E + 02 ±	3.87E + 02 ±	3.88E + 02 ±	3.87E + 02 ±	3.87E + 02 ±	3.87E + 02 ±	3.87E + 02 ±	3.87E + 02 ±	3.87E + 02 ±
F26	1.98E + 00 1.14E + 03 ±	2.02E + 00 1.12E + 03 ±	1.12E + 00 1.15E + 03 ±	2.50E + 00 9.53E + 02 ±	1.11E + 00 1.08E + 03 ±	8.91E - 01 1.20E + 03 ±	9.49E - 01 1.10E + 03 ±	2.17E + 00 1.13E + 03 ±	9.80E - 01 1.16E + 03 ±	1.11E + 00 1.11E + 03 ±
F27	2.90E + 02 5.03E + 02 ±	3.05E + 02 5.01E + 02 ±	2.24E + 02 4.99E + 02 ±	2.83E + 02 4.99E + 02 ±	3.61E + 02 5.05E + 02 ±	2.15E + 02 4.99E + 02 ±	2.91E + 02 5.00E + 02 ±	4.04E + 02 4.98E + 02 ±	2.22E + 02 5.01E + 02 ±	2.97E + 02 4.98E + 02 ±
F28	9.11E + 00 3.24E + 02 ±	1.08E + 01 3.45E + 02 ±	1.11E + 01 3.16E + 02 ±	9.86E + 00 3.34E + 02 ±	1.20E + 01 3.32E + 02 ±	1.02E + 01 3.18E + 02 ±	1.16E + 01 3.26E + 02 ±	9.99E + 00 3.31E + 02 ±	9.60E + 00 3.32E + 02 ±	9.68E + 00 3.31E + 02 ±
F29	5.02E + 01 4.81E + 02 ±	6.18E + 01 4.84E + 02 ±	4.25E + 01 4.78E + 02 ±	5.43E + 01 4.82E + 02 ±	4.93E + 01 4.72E + 02 ±	4.16E + 01 4.76E + 02 ±	4.90E + 01 4.77E + 02 ±	5.23E + 01 4.72E + 02 ±	4.99E + 01 4.80E + 02 ±	4.87E + 01 4.69E + 02 ±
F30	3.18E + 01 2.17E + 03 ±	4.91E + 01 2.15E + 03 ±	4.55E + 01 2.16E + 03 ±	5.33E + 01 2.37E + 03 ±	4.57E + 01 2.21E + 03 ±	4.54E + 01 2.18E + 03 ±	5.29E + 01 2.14E + 03 ±	3.43E + 01 2.15E + 03 ±	3.98E + 01 2.16E + 03 ±	3.73E + 01 2.16E + 03 ±
	1.19E + 02	1.45E + 02	1.39E + 02	7.06E + 02	2.34E + 02	1.30E + 02	1.05E + 02	1.21E + 02	1.02E + 02	1.12E + 02
	2	2	7	3	2	5	5	3	2	9

(4) In examining the effect of the parameter K , kt , and gd on the performance of the algorithm, the following three sets of parameters are: 5, 20 and 30; 20, 60 and 80; and 10, 20 and 30. Other than that, the other parameters were set as follows: $f1$ in the range of $[0, 0.8]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 1.2]$ for it $> 0.4 \times \text{maxit}$, cr in the range of $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, and $[0.5, 0.9]$ for it $> 0.4 \times \text{maxit}$, and $[0.5, 0.9]$ for it $> 0.4 \times \text{maxit}$, and $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 0.8]$ for it $> 0.4 \times \text{maxit}$, 0.3 for $cr1$ and 0.1 for $cr2$.

As presented in Tables 1 to 4, the last row shows how many times the algorithm achieved the global optimum across all functions. From the Tables, it is evident that, unless specified otherwise, $f1$ takes the value range of $[0, 0.8]$ for it $\leq 0.4 \times \text{maxit}$, $[0.1, 1.2]$ for it $> 0.4 \times \text{maxit}$, cr takes the value range of $[0.5, 0.9]$ for it $\leq 0.4 \times \text{maxit}$, $[0, 0.8]$ for it $> 0.4 \times \text{maxit}$, and $cr1$ takes the value of 0.5, 0.9, 0.8, and $cr1$ takes the value of 0.5, 0.9 for it $> 0.4 \times \text{maxit}$, and $cr1$ takes the value of 0.0, 0.8. The algorithm achieves better optimization performance when $cr1$ is taken as 0.3, $cr2$ is taken as 0.1, K is taken as 10, kt is taken as 40 and gd is taken as 5.

5.2 Effectiveness Experiments of Various Improvement Strategies

According to Section 4, the ICSBO algorithm introduces four improvements to the CSBO algorithm. To verify the effectiveness of these four improvement strategies, we created four new modified algorithms by removing each corresponding improvement strategy from ICSBO and replacing it with the corresponding strategy from the basic CSBO algorithm. These include: (1) removing the improved venous blood circulation strategy from Section 4.1, (2) removing the improved body circulation strategy from Section 4.2, (3) removing the improved lung circulation strategy from Section 4.3, and (4) removing the external archive-based diversity supplementation mechanism from Section 4.4. For simplicity, these four new modified algorithms are named ICSBO1, ICSBO2, ICSBO3, and ICSBO4, and are compared with the ICSBO algorithm on the CEC2017 test suite.

To ensure a fair comparison, each algorithm is tested with a population size of $N = 50$, a dimension of $D = 30$, and a maximum of $\text{MaxFEs} = 300,000$ function evaluations. The additional parameters are set as follows: $\text{maxit} = 3000$, $kt = 5$, $gd = 5$, and $K = 10$. To mitigate the effects of randomness from a single run, each algorithm is run independently 30 times on each test function.

Table 5 displays the results of the algorithms over 29 test functions, with the “ $\pm$ ” symbol representing the mean and standard deviation of the optimal values from 30 independent runs. To further compare the performance of the ICSBO algorithm with other improved algorithms, Wilcoxon rank-sum and Friedman tests were performed at a 5% significance level, as shown in Tables 6 and 7. In Table 6, a p -value greater than 0.05 indicates no significant difference between the ICSBO algorithm and the respective improved algorithm, marked by the symbol “=”. A p -value less than 0.05, with the improved algorithm showing a better average optimal value from 30 runs, indicates that the improved algorithm significantly outperforms the ICSBO, represented by the symbol “+”. Conversely, if the performance is significantly worse, it is indicated by the symbol “-”. In Table 7, a lower rank mean for an algorithm signifies better overall performance.

Table 5: Running results of each improvement strategy in a 30-dimensional CEC2017 test

	ICSBO	ICSBO1	ICSBO2	ICSBO3	ICSBO4
F1	4.74E - 16 $\pm$ 2.59E - 15	9.47E - 16 $\pm$ 3.61E - 15	1.89E - 15 $\pm$ 4.91E - 15	1.71E - 14 $\pm$ 2.95E - 14	2.23E - 14 $\pm$ 2.44E - 14
F3	1.89E - 14 $\pm$ 2.73E - 14	2.65E - 14 $\pm$ 2.88E - 14	4.55E - 14 $\pm$ 2.31E - 14	7.20E - 14 $\pm$ 2.56E - 14	8.34E - 14 $\pm$ 4.15E - 14
F4	1.65E + 01 $\pm$ 2.64E + 01	1.62E + 01 $\pm$ 2.70E + 01	2.04E + 01 $\pm$ 2.71E + 01	1.85E + 01 $\pm$ 2.72E + 01	1.83E + 01 $\pm$ 2.68E + 01
F5	3.97E + 01 $\pm$ 2.21E + 01	5.78E + 01 $\pm$ 4.21E + 01	9.84E + 01 $\pm$ 8.03E + 00	4.97E + 01 $\pm$ 3.55E + 01	4.87E + 01 $\pm$ 1.43E + 01
F6	1.46E - 05 $\pm$ 3.18E - 05	6.16E - 05 $\pm$ 3.20E - 04	5.14E - 06 $\pm$ 2.77E - 05	3.44E - 05 $\pm$ 1.04E - 04	3.89E - 04 $\pm$ 2.10E - 03
F7	6.51E + 01 $\pm$ 1.20E + 01	8.03E + 01 $\pm$ 4.50E + 01	1.33E + 02 $\pm$ 7.19E + 00	7.87E + 01 $\pm$ 3.44E + 01	7.54E + 01 $\pm$ 1.33E + 01
F8	4.87E + 01 $\pm$ 3.40E + 01	6.67E + 01 $\pm$ 4.53E + 01	9.84E + 01 $\pm$ 9.55E + 00	6.52E + 01 $\pm$ 4.24E + 01	4.59E + 01 $\pm$ 1.13E + 01

(Continued)

Table 5 (continued)

	ICSBO	ICSBO1	ICSBO2	ICSBO3	ICSBO4
F9	5.20E - 01 ± 1.01E + 00	9.36E - 02 ± 2.13E - 01	2.66E - 01 ± 1.01E + 00	8.02E - 01 ± 9.66E - 01	4.17E - 01 ± 7.26E - 01
F10	2.64E + 03 ± 6.06E + 02	2.86E + 03 ± 8.69E + 02	4.92E + 03 ± 2.68E + 02	2.67E + 03 ± 6.54E + 02	2.52E + 03 ± 5.21E + 02
F11	6.87E + 00 ± 4.34E + 00	9.14E + 00 ± 9.17E + 00	4.65E + 01 ± 5.52E + 01	8.58E + 00 ± 7.50E + 00	1.18E + 01 ± 4.99E + 00
F12	7.30E + 03 ± 6.65E + 03	9.67E + 03 ± 9.72E + 03	9.11E + 03 ± 5.59E + 03	7.34E + 03 ± 6.30E + 03	7.84E + 03 ± 6.91E + 03
F13	2.57E + 01 ± 1.35E + 01	3.18E + 01 ± 1.26E + 01	1.54E + 02 ± 6.17E + 02	4.67E + 01 ± 5.02E + 01	3.70E + 01 ± 2.11E + 01
F14	1.74E + 01 ± 1.06E + 01	2.32E + 01 ± 1.07E + 01	2.32E + 01 ± 1.39E + 01	1.84E + 01 ± 1.21E + 01	3.00E + 01 ± 8.06E + 00
F15	6.75E + 00 ± 2.84E + 00	8.16E + 00 ± 6.68E + 00	1.33E + 01 ± 1.49E + 01	1.14E + 01 ± 8.57E + 00	1.24E + 01 ± 5.37E + 00
F16	4.45E + 02 ± 2.24E + 02	4.71E + 02 ± 1.47E + 02	6.82E + 02 ± 1.32E + 02	4.93E + 02 ± 2.19E + 02	4.65E + 02 ± 1.94E + 02
F17	5.02E + 01 ± 3.61E + 01	5.74E + 01 ± 2.58E + 01	1.22E + 02 ± 2.58E + 01	5.98E + 01 ± 3.50E + 01	6.38E + 01 ± 3.19E + 01
F18	3.25E + 01 ± 1.55E + 01	1.04E + 02 ± 1.05E + 02	1.02E + 02 ± 3.04E + 02	3.52E + 01 ± 2.21E + 01	4.77E + 01 ± 3.05E + 01
F19	8.40E + 00 ± 2.81E + 00	8.80E + 00 ± 2.53E + 00	1.28E + 01 ± 4.92E + 00	9.88E + 00 ± 2.31E + 00	1.09E + 01 ± 4.73E + 00
F20	5.80E + 01 ± 8.83E + 01	7.43E + 01 ± 6.16E + 01	1.47E + 02 ± 6.10E + 01	7.69E + 01 ± 6.42E + 01	1.06E + 02 ± 7.01E + 01
F21	2.86E + 02 ± 5.68E + 01	3.07E + 02 ± 4.71E + 01	2.95E + 02 ± 1.01E + 01	3.06E + 02 ± 4.73E + 01	2.44E + 02 ± 9.41E + 00
F22	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 4.49E - 01	1.00E + 02 ± 0.00E + 00	1.00E + 02 ± 4.48E - 01	1.00E + 02 ± 6.23E - 01
F23	3.83E + 02 ± 3.30E + 01	4.26E + 02 ± 5.28E + 01	4.31E + 02 ± 2.30E + 01	4.08E + 02 ± 4.87E + 01	3.93E + 02 ± 1.29E + 01
F24	4.39E + 02 ± 8.62E + 00	4.44E + 02 ± 2.56E + 01	4.43E + 02 ± 2.56E + 01	4.45E + 02 ± 2.28E + 01	4.54E + 02 ± 1.36E + 01
F25	3.87E + 02 ± 1.11E + 00	3.87E + 02 ± 2.05E + 00	3.87E + 02 ± 9.27E - 01	3.89E + 02 ± 1.00E + 01	3.89E + 02 ± 7.02E + 00
F26	1.11E + 03 ± 2.97E + 02	1.15E + 03 ± 2.16E + 02	1.28E + 03 ± 4.89E + 02	1.16E + 03 ± 2.29E + 02	1.29E + 03 ± 3.31E + 02
F27	4.98E + 02 ± 9.68E + 00	4.95E + 02 ± 1.23E + 01	5.04E + 02 ± 1.13E + 01	4.98E + 02 ± 1.05E + 01	5.02E + 02 ± 5.97E + 00
F28	3.31E + 02 ± 4.87E + 01	3.26E + 02 ± 4.78E + 01	3.39E + 02 ± 5.21E + 01	3.43E + 02 ± 5.38E + 01	3.38E + 02 ± 5.52E + 01
F29	4.69E + 02 ± 3.73E + 01	4.87E + 02 ± 6.19E + 01	5.54E + 02 ± 4.80E + 01	4.75E + 02 ± 3.55E + 01	5.00E + 02 ± 5.47E + 01
F30	2.16E + 03 ± 1.12E + 02	2.28E + 03 ± 3.77E + 02	2.29E + 03 ± 3.73E + 02	2.20E + 03 ± 1.88E + 02	2.22E + 03 ± 1.83E + 02

Table 6: Wilcoxon rank sum test results of each improved algorithm and ICSBO algorithm

	<i>p</i> -value (vs. ICSBO)			
	ICSBO1	ICSBO2	ICSBO3	ICSBO4
F1	0.570 (=)	0.169 (=)	0.000 (-)	0.000 (-)
F3	0.300 (=)	0.000 (-)	0.000 (-)	0.000 (-)
F4	0.205 (=)	0.613 (-)	0.832 (=)	0.512 (=)
F5	0.249 (=)	0.000 (-)	0.446 (=)	0.003 (-)
F6	0.158 (=)	0.000 (+)	0.424 (=)	0.283 (=)
F7	0.982 (=)	0.000 (-)	0.181 (=)	0.003 (-)
F8	0.261 (=)	0.000 (-)	0.068 (=)	0.075 (=)
F9	0.005 (-)	0.016 (+)	0.042 (-)	0.586 (=)
F10	0.297 (=)	0.000 (-)	0.877 (=)	0.717 (=)
F11	0.115 (=)	0.000 (-)	0.255 (=)	0.000 (-)
F12	0.540 (=)	0.109 (=)	0.853 (=)	0.947 (=)
F13	0.018 (-)	0.008 (-)	0.002 (-)	0.022 (-)
F14	0.085 (=)	0.096 (=)	0.819 (=)	0.000 (-)
F15	0.739 (=)	0.050 (-)	0.019 (-)	0.000 (-)
F16	0.935 (=)	0.000 (-)	0.579 (=)	0.853 (=)
F17	0.042 (-)	0.000 (-)	0.052 (=)	0.007 (-)
F18	0.000 (-)	0.695 (=)	0.684 (=)	0.072 (=)
F19	0.652 (=)	0.001 (-)	0.027 (-)	0.018 (-)
F20	0.212 (=)	0.000 (-)	0.420 (=)	0.001 (-)
F21	0.130 (=)	0.464 (=)	0.068 (=)	0.158 (=)
F22	0.334 (=)	NaN(=)	0.334 (=)	0.161 (=)
F23	0.016 (-)	0.000 (-)	0.056 (=)	0.000 (-)
F24	0.842 (=)	0.304 (=)	0.112 (=)	0.000 (-)

(Continued)

Table 6 (continued)

	<i>p</i> -value (vs. ICSBO)			
	ICSBO1	ICSBO2	ICSBO3	ICSBO4
F25	0.090 (=)	0.773 (=)	0.363 (=)	0.070 (=)
F26	0.559 (=)	0.007 (–)	0.824 (=)	0.004 (–)
F27	0.284 (=)	0.068 (=)	0.620 (=)	0.258 (=)
F28	0.841 (=)	0.503 (=)	0.284 (=)	0.505 (=)
F29	0.133 (=)	0.000 (–)	0.304 (=)	0.018 (–)
F30	0.663 (=)	0.149 (=)	0.610 (=)	0.404 (=)

Table 7: Friedman test results of each improved algorithm

	ICSBO	ICSBO1	ICSBO2	ICSBO3	ICSBO4
<i>Avg.rank</i>	1.36	2.93	4.05	3.21	3.45
<i>sort</i>	1	2	5	3	4

Table 5 shows that, compared to the ICSBO algorithm, the performance of the four improved algorithms is worse than that of ICSBO in 25, 26, 26, and 25 test functions, respectively. According to the Wilcoxon rank-sum test results in Table 6, ICSBO1 performs worse than ICSBO in 5 test functions, marked by a “–”, while in the remaining 24 functions, the performance is comparable, indicated by “=”. ICSBO2 outperforms ICSBO in 2 test functions, has similar performance in 11, and underperforms in 16. ICSBO3 shows similar performance to ICSBO in 23 test functions and worse performance in 6. ICSBO4 performs similarly to ICSBO in 14 functions and worse in 15. Table 7 shows that excluding any of the improvement strategies from ICSBO results in a decline in the rankings of the modified algorithms, indicating that the four proposed strategies have a significant impact on the overall performance of ICSBO.

In conclusion, the four improvement strategies introduced in this paper show varying degrees of effectiveness. The strategies outlined in Sections 4.2 and 4.4 have a more significant impact on the performance of ICSBO, while the performance improvements from the other two strategies are similar to that of the original ICSBO.

5.3 Comparison of ICSBO Algorithm Performance with Other Algorithms

To evaluate whether the ICSBO algorithm outperforms others in convergence accuracy and speed, this section compares its performance with eight representative evolutionary algorithms using the CEC2017 test suite. These algorithms include the Enhanced Serpentine Optimizer (ESO), which introduces a new dynamic mechanism and an opposition-based learning strategy; the Improved Orca Predation Algorithm (IOPA); the Improved Archimedes Algorithm (IAOA); the Multi-Strategy Enhanced CSBO (MECSBO) algorithm; the Strategy-Enhanced MDBO algorithm; the Multi-Trial Vector Sine Cosine Algorithm (MTV-SCA); the newly proposed Synergistic Swarm Optimization Algorithm (SSOA); and the Information Acquisition Optimization Algorithm. To ensure fairness in comparison, all algorithms will use a population size of $N = 50$ and a maximum function evaluation count of $\text{MaxFEs} = 300,000$. Table 8 presents the parameter settings for each algorithm, with the values for the comparison algorithms matching those used in the original studies.

Table 8: Initialization settings for each algorithm parameter

Algorithm	Initial parameters
ICSBO	Popnumber = 50, dim = 30, maxFEs = 300,000, maxit = 3000, D = 5, limit = [-100, 100]
ESO	Threshold = 0.25, <i>Thresold2</i> = 0.6, <i>T</i> = 3000, <i>vec_flag</i> = [1, -1]
IAOA	<i>C2</i> = 6, <i>C3</i> = 2, <i>C4</i> = 0.5
MECSBO	Popnumber = 50, dim = 30, maxFEs = 300,000, maxit = 3000, D = 5, limit = [-100, 100]
IOPA	<i>p1</i> = <i>q</i> = 0.9, <i>p2</i> = 0.005, <i>F</i> = 2
SSOA	maxFE = 300,000, <i>w</i> = 0.7, <i>c1</i> = 2, <i>c2</i> = 2, <i>k</i> = 0.5
MDBO	<i>yz</i> = $1e - 5$, <i>RDBN_np</i> = 0.2*np, <i>YDBN_np</i> = 0.2*np, <i>DDBN_np</i> = round(0.25*np), <i>TDBN_np</i> = np-RDBN_np-YDBN_np-DDBN_np
IAO	Popnumber = 50, dim = 30, maxFEs = 300,000, maxit = 3000; D = 5, limit = [-100, 100]
MTV-SCA	<i>lambda</i> = 0.25, <i>MaxIter</i> = 3100, <i>MaxFES</i> = 300,000

5.3.1 Comparison of Convergence Accuracy between the ICSBO Algorithm and Other Algorithms

To assess the convergence accuracy of the ICSBO algorithm relative to other algorithms, experiments were conducted using the CEC2017 test suite with $D = 30$. Table 9 displays the average values and standard deviations for each algorithm, based on 30 independent runs in 30 dimensions of the CEC2017 dataset, with the best optimization results highlighted in bold. The table also presents the outcomes of the Wilcoxon rank-sum test, comparing the ICSBO algorithm with the other algorithms. For a more thorough evaluation of the overall performance, Table 10 provides the results from the Friedman test.

Table 9 shows that in 30-dimensional optimization problem, ICSBO achieves global optimality in 18 test functions, including F1, F3, F7, F10, F11, F13, F14, F15, F17, F18, F19, F20, F22, F23, F24, F27, F29, F30; CSBO attains global optimality in 2 functions, including F6, F22; MECSBO obtains global optimum in 6 functions including F4, F9, F22, F25, F27, F28; IAOA achieves global optimum in 6 functions, including F5, F8, F13, F16, F18, F21; IOPA attains global optimum in F12, and MDBO achieves global optimum in F25. The optimization results in Table 10 show that compared with ICSBO, CSBO performs similarly in 7 functions, worse in 21 functions, and better in 1 function; MECSBO performs similarly in 7 functions, worse in 21 functions, and better in 1 function; IAOA has only 2 functions, and the 4 remaining functions have the same performance. Performance is equal, the 4 remaining functions have better performance, and 23 functions have worse performance. MTV-SCA has worse performance in 26 functions, a slightly different performance in 2 functions, and better performance in 1 function. IOPA has equal performance in 3 functions and worse performance in the 26 remaining functions. MDBO has equal performance in 5 functions and worse performance on 24 functions compared with ICSBO. ESO and IAO have 28 functions with worse performance and only 1 function with flat performance. SSOA performs worse than ICSBO in 29 functions. Table 10 illustrates that, for the 30-dimensional test functions, the ranking order is ICSBO > CSBO > MECSBO > MDBO > MTV-SCA > IAOA > IOPA > IAO > ESO > SSOA. In conclusion, the ICSBO algorithm demonstrated a significant enhancement in convergence accuracy compared to other algorithms on the 30-dimensional CEC2017 test suite.

Table 9: Data results of ICSBO and other algorithms on the 30-dimensional CEC2017 test set

	ICSBO	CSBO	IAOA	MECSBO	IOPA	ESO	SSAO	MDBO	MTVSCA	IAO
F1	4.74E-16 ±	1.58E-13 ±	5.71E-01 ±	7.57E-12 ±	1.20E+0	8.35E+06 ±	4.90E+10 ±	1.18E+03 ±	1.58E-05 ±	7.64E+09 ±
F2	2.59E-15	1.61E-13	1.31E+00	3.27E-11	0.2±2.09E+02	8.67E+06	4.40E+10	1.85E+03	3.21E-11	5.69E+09
F3	1.89E-14 ±	5.27E-0	1.77E+01 ±	3.27E-08 ±	1.14E-12 ±	1.45E+04 ±	8.76E+04 ±	2.26E+03	2.77E+00 ±	1.58E+04 ±
F4	2.73E-14	13±1.70E-13	8.97E+01	1.44E-07	1.27E-12	2.91E+03	9.18E+04	4.02E-04	1.26E-02	2.29E+04
F5	1.65E+01 ±	2.19E+01 ±	3.53E+01 ±	1.10E+01 ±	1.87E+01 ±	1.11E+02 ±	1.40E+04 ±	3.06E+01 ±	5.49E+01 ±	5.94E+02 ±
F6	2.64E+01	2.96E+01	3.59E+01	2.15E+01	2.73E+01	2.26E+01	1.76E+04	3.26E+01	7.31E+01	6.39E+02
F7	3.97E+01 ±	6.69E+01 ±	3.02E+01 ±	9.36E+01 ±	7.68E+01 ±	1.02E+02 ±	4.38E+02 ±	4.55E+01 ±	5.07E+01 ±	1.70E+02 ±
F8	2.21E+01	6.13E+00	1.29E+01	8.90E+00	2.26E+01	2.22E+01	4.47E+02	1.17E+01	4.38E+01	1.91E+02
F9	1.46E-05 ±	2.77E-13 ±	2.04E+00 ±	2.67E-06 ±	2.21E+00 ±	1.63E+01 ±	9.61E+01 ±	5.29E-01 ±	3.02E-01 ±	3.63E+01 ±
F10	3.18E-05	6.46E-14	1.84E+00	3.56E-06	2.12E+00	4.67E+02	8.70E+01	5.26E-01	1.49E+00	4.17E+01
F11	6.51E+01 ±	1.03E+02 ±	6.55E+01 ±	1.33E+02	1.08E+02 ±	2.80E+02	7.15E+02 ±	7.99E+01 ±	9.61E+01 ±	2.89E+02 ±
F12	1.20E+01	7.43E+00	1.47E+01	8.84E+00	2.16E+01	2.81E+01	6.69E+02	1.49E+01	9.17E+01	3.22E+02
F13	4.87E+01 ±	6.79E+01 ±	2.90E+01 ±	9.72E+01 ±	7.46E+01 ±	7.94E+01 ±	3.64E+02 ±	4.03E+01 ±	4.84E+01 ±	1.54E+02 ±
F14	3.40E+01	7.72E+00	8.52E+00	7.26E+00	1.83E+01	2.12E+01	3.82E+02	1.31E+01	8.26E+01	1.75E+02
F15	5.20E-01 ±	2.41E-02 ±	1.77E+01 ±	1.44E-13 ±	2.11E+02 ±	2.07E+03 ±	1.12E+04 ±	8.83E+01 ±	5.73E+01 ±	2.61E+03 ±
F16	1.01E+00	8.57E-02	2.99E+01	5.11E-14	2.22E+02	8.84E+02	9.75E+03	5.74E+01	1.46E+01	1.35E+03
F17	2.64E+03 ±	3.65E+03 ±	3.88E+03 ±	4.09E+03 ±	3.50E+03 ±	5.16E+03 ±	7.91E+03 ±	3.35E+03 ±	3.60E+03 ±	4.02E+03 ±
F18	6.06E+02	3.10E+02	5.98E+02	2.72E+02	5.42E+02	1.91E+03	8.40E+03	5.94E+02	4.24E+03	4.19E+03
F19	6.87E+00 ±	2.63E+01 ±	2.41E+02 ±	2.80E+01 ±	2.40E+02 ±	4.51E+02 ±	8.80E+03 ±	1.35E+02 ±	1.27E+02 ±	2.24E+02 ±
F20	4.34E+00	3.93E+01	2.95E+02	1.48E+01	1.31E+02	1.53E+02	7.59E+03	8.57E+01	2.87E+02	1.72E+02
F21	7.30E+03 ±	1.47E+04 ±	3.51E+04 ±	1.89E+04 ±	7.16E+03 ±	6.64E+06 ±	1.36E+10 ±	1.74E+04 ±	1.31E+04 ±	7.95E+05 ±
F22	6.65E+03	1.03E+04	7.97E+04	8.82E+03	4.62E+03	6.49E+06	1.31E+10	1.13E+04	1.36E+04	1.66E+05
F23	2.57E+01 ±	7.04E+01 ±	1.26E+04 ±	5.83E+01 ±	8.62E+03 ±	3.59E+05 ±	1.24E+10 ±	6.61E+03 ±	2.79E+03 ±	4.32E+03 ±
F24	1.35E+01	3.74E+01	1.50E+04	2.67E+01	1.05E+04	4.06E+05	1.21E+10	7.55E+03	1.80E+02	7.65E+03
F25	1.74E+01 ±	3.45E+01 ±	6.62E+04 ±	3.54E+01 ±	1.58E+02 ±	1.11E+04 ±	3.08E+06 ±	3.15E+02 ±	1.19E+02 ±	9.01E+01 ±
F26	1.06E+01	3.78E+00	3.07E+05	7.21E+00	4.71E+01	1.19E+04	3.54E+06	4.72E+02	8.23E+01	9.35E+01
F27	6.75E+00 ±	1.33E+01 ±	4.92E+03 ±	1.30E+01 ±	1.33E+03 ±	3.20E+04 ±	3.60E+08 ±	4.27E+03 ±	4.91E+02 ±	5.54E+02 ±
F28	2.84E+00	1.29E+01	9.62E+03	3.23E+02	1.66E+03	4.51E+04	3.14E+08	6.64E+03	4.36E+01	3.66E+02
F29	4.45E+02 ±	5.15E+02 ±	3.19E+02 ±	6.69E+02 ±	6.96E+02 ±	8.11E+02 ±	4.51E+03 ±	3.39E+02 ±	5.15E+02 ±	8.91E+02 ±
F30	2.24E+02	1.43E+02	1.43E+02	9.98E+01	2.09E+02	2.30E+02	4.19E+03	2.43E+02	5.94E+02	1.06E+03
F31	5.02E+01 ±	8.64E+01 ±	1.14E+02 ±	1.15E+02 ±	3.06E+02 ±	3.69E+02 ±	2.74E+03 ±	8.33E+01 ±	8.76E+01 ±	1.95E+02 ±
F32	3.61E+01	1.70E+01	5.41E+01	1.65E+01	1.54E+02	2.09E+02	2.29E+03	6.31E+01	1.80E+02	2.02E+02
F33	3.25E+01 ±	1.09E+02 ±	3.09E+04 ±	4.42E+01 ±	2.30E+03 ±	1.89E+05 ±	6.72E+07 ±	2.30E+04 ±	9.30E+03 ±	2.81E+02 ±
F34	1.55E+01	1.05E+02	5.01E+04	1.96E+01	2.81E+03	1.37E+05	2.60E+07	1.71E+04	3.59E+03	1.34E+02
F35	8.40E+00 ±	1.27E+01 ±	4.36E+03 ±	1.61E+01 ±	1.73E+02 ±	3.44E+05 ±	6.60E+08 ±	2.79E+03 ±	7.97E+01 ±	9.53E+01 ±
F36	2.81E+00	2.08E+00	6.01E+03	1.53E+01	7.75E+01	7.56E+05	7.36E+08	3.18E+03	7.13E+01	4.13E+01
F37	5.80E+01 ±	1.19E+02 ±	6.20E+02 ±	1.44E+02 ±	2.91E+02 ±	3.46E+02 ±	1.16E+03 ±	1.23E+02 ±	1.46E+02 ±	2.69E+02 ±
F38	5.52E+01	5.52E+01	2.79E+02	4.75E+01	1.36E+02	1.09E+02	1.23E+03	7.86E+01	1.56E+02	2.41E+02
F39	2.86E+02 ±	2.65E+02 ±	2.29E+02 ±	2.93E+02 ±	2.68E+02 ±	2.76E+02 ±	6.76E+02 ±	2.38E+02 ±	2.43E+02 ±	3.15E+02 ±
F40	5.68E+01	7.79E+00	7.98E+00	6.76E+00	2.41E+01	2.05E+01	7.06E+02	1.09E+01	2.44E+02	3.17E+02
F41	1.00E+02 ±	1.00E+02 ±	3.63E+03 ±	1.00E+02 ±	1.01E+02 ±	1.21E+02 ±	7.40E+03 ±	1.01E+02 ±	5.47E+02 ±	9.87E+02 ±
F42	0.00E+00	4.71E-13	1.21E+03	5.11E-13	1.40E+00	7.09E+00	8.60E+03	2.03E+00	3.17E+03	1.65E+03
F43	3.83E+02 ±	4.12E+02 ±	3.84E+02 ±	4.36E+02 ±	4.28E+02 ±	4.38E+02 ±	1.43E+03 ±	4.15E+02 ±	4.00E+02 ±	4.96E+02 ±
F44	3.30E+01	7.20E+00	1.37E+01	9.40E+00	2.41E+01	2.46E+01	1.17E+03	2.24E+01	3.95E+02	4.88E+02
F45	4.39E+02 ±	4.73E+02 ±	4.75E+02 ±	5.09E+02 ±	4.81E+02 ±	4.98E+02 ±	1.60E+03 ±	4.74E+02 ±	4.65E+02 ±	5.85E+02 ±
F46	8.62E+00	1.13E+01	1.40E+01	8.24E+00	1.86E+01	5.24E+01	1.43E+03	2.43E+01	4.72E+02	5.62E+02
F47	3.87E+02 ±	3.87E+02 ±	3.93E+02 ±	3.86E+02 ±	3.91E+02 ±	4.26E+02 ±	2.59E+03 ±	3.90E+02 ±	3.90E+02 ±	5.97E+02 ±
F48	1.11E+00	1.24E+00	1.08E+01	1.83E+00	1.43E+01	2.26E+01	2.53E+03	9.85E+00	3.94E+02	5.30E+02
F49	1.11E+03 ±	1.51E+03 ±	1.31E+03 ±	1.40E+03 ±	1.78E+03 ±	2.98E+03 ±	8.55E+03 ±	1.38E+03 ±	1.65E+03 ±	2.25E+03 ±
F50	2.97E+02	1.07E+02	2.34E+02	6.96E+02	6.97E+02	8.27E+02	8.18E+03	6.62E+02	1.30E+03	3.10E+03
F51	4.98E+02 ±	5.01E+02 ±	5.94E+02 ±	4.98E+02 ±	5.37E+02 ±	5.61E+02 ±	2.23E+03 ±	5.44E+02 ±	5.32E+02 ±	5.45E+02 ±
F52	9.68E+00	1.27E+01	5.68E+01	1.27E+01	1.84E+01	1.62E+01	3.01E+03	1.80E+01	5.27E+02	5.48E+02

(Continued)

Table 9 (continued)

	ICsBO	CSBO	IAOA	MECSBO	IOPA	ESO	SSAO	MDBO	MTVSCA	IAO
F28	3.31E + 02 ± 4.87E + 01	3.18E + 02 ± 4.27E + 01	5.76E + 02 ± 7.31E + 02	3.14E + 02 ± 3.67E + 01	3.33E + 02 ± 5.59E + 01	4.40E + 02 ± 2.85E + 01	4.28E + 03 ± 5.30E + 03	3.29E + 02 ± 5.47E + 01	3.47E + 02 ± 3.00E + 02	7.48E + 02 ± 9.29E + 02
F29	4.69E + 02 ± 3.73E + 01	5.21E + 02 ± 2.47E + 01	6.16E + 02 ± 1.83E + 02	5.91E + 02 ± 3.13E + 01	7.97E + 02 ± 1.57E + 02	1.13E + 03 ± 2.52E + 02	5.07E + 03 ± 5.76E + 03	5.98E + 02 ± 1.22E + 02	5.91E + 02 ± 6.40E + 02	9.30E + 02 ± 8.81E + 02
F30	2.16E + 03 ± 1.12E + 02	2.45E + 03 ± 4.45E + 02	3.10E + 05 ± 1.55E + 06	2.32E + 03 ± 2.46E + 02	2.47E + 03 ± 4.53E + 02	9.25E + 05 ± 1.70E + 06	1.58E + 09 ± 1.35E + 09	4.76E + 03 ± 1.93E + 03	4.43E + 03 ± 4.53E + 03	7.63E + 03 ± 7.33E + 03
+/-/=		1/21/7	4/23/2	1/21/7	0/26/3	0/28/1	0/29/0	0/25/4	1/26/2	0/28/1

Table 10: Friedman test of each algorithm

	ICSBO	CSBO	IAOA	MECSBO	IOPA	ESO	SSAO	MDBO	MTVSCA	IAO
Avg.rank	1.79	3.16	5.66	3.98	5.69	8.17	10	4.31	4.59	7.69
sort	1	2	6	3	7	9	10	4	5	8

Table 11 presents the mean and standard deviation of the algorithms from 30 independent experiments conducted on the 100-dimensional CEC2017 dataset, along with the results of the Wilcoxon rank-sum test. The function that achieves the best optimization result for each case is highlighted in bold.

Table 11 shows that ICSBO can take the global optimum in 17 functions such as F1, F5, F7, F9, F11, F12, F14 F16, F17, F21, F23, F24, F26, F28, F29, F30, CSBO can take the global optimum in 3 functions such as F3, F4, F18, MECSBO can take the global optimum in F6, F13, F15, F18, F26, MTVSCA can take the function optimum in F20, F22, IAOA can take the global optimum in F25, and IOPA can take the global optimum in F10. Compared with ICSBO, IAOA performs better in 2 functions, worse in 26 functions, and equally in 1 function. MECSBO outperforms ICSBO in 3 functions, worse in 21 functions, and performs equally in five functions. IOPA outperforms ICSBO in 2 functions, worse in 20 functions, and performs equally in 7 functions. ESO and SSOA are worse than ICSBO. IAO is worse than ICSBO in 28 functions and performs equally in only 1 function. MDBO outperforms ICSBO on 3 functions, worse in 19 functions, and performs equally on 5 functions. The MDBO algorithm outperforms the ICSBO algorithm in 3 functions, worse in 19 functions, and is equal in 7 functions. MTVSCA outperforms ICSBO in 5 functions, worse in 22 functions, and performs equally in 2 functions.

Tables 9 and **11** show that as the dimensionality increases, the algorithm's performance changes, with the impact on single-peak functions being particularly pronounced. The ICSBO algorithm proposed in this paper performs well in 100 dimensions, achieving the global optimum for nearly 17 functions. When compared to the results from the 30-dimensional tests on multi-peak, composite, and combined functions, ICSBO's performance remains stable. This consistency further demonstrates that ICSBO maintains good convergence accuracy and faster convergence speed.

Table II: Data results of ICSBO and other algorithms on the 100-dimensional CEC2017 test set

	ICSBO	CSBO	IAOA	MECSBO	IOPA	ESO	SSAO	MDBO	MTVSCA	IAO
F1	4.41E+03 ± 5.72E+03	5.03E+03 ± 4.78E+03	1.22E+04 ± 1.81E+04	6.97E+03 ± 5.11E+03	1.01E+04 ± 1.25E+04	2.25E+08 ± 7.84E+07	2.65E+11 ± 6.83E+09	1.30E+04 ± 1.45E+04	9.09E+06 ± 2.42E+07	1.85E+11 ± 1.35E+10
F3	1.20E+04 ± 4.78E+03	7.57E+03 ± 2.64E+03	2.26E+05 ± 1.06E+05	3.42E+04 ± 8.41E+03	1.70E+04 ± 6.23E+03	3.16E+05 ± 1.61E+04	3.56E+05 ± 1.93E+04	4.54E+04 ± 1.14E+04	1.24E+05 ± 2.37E+04	2.28E+05 ± 1.95E+04
F4	2.45E+02 ± 4.12E+01	2.31E+02 ± 6.30E+01	2.69E+02 ± 4.42E+01	2.53E+02 ± 6.22E+01	2.95E+02 ± 5.13E+01	6.05E+02 ± 5.75E+01	9.55E+04 ± 1.64E+03	2.59E+02 ± 4.18E+01	3.59E+02 ± 6.08E+01	3.94E+04 ± 7.33E+03
F5	1.97E+02 ± 1.64E+02	8.04E+02 ± 2.40E+01	3.73E+02 ± 1.53E+02	9.04E+02 ± 3.02E+01	6.05E+02 ± 8.40E+01	7.64E+02 ± 7.44E+01	1.64E+03 ± 3.46E+01	3.89E+02 ± 4.02E+01	4.74E+02 ± 6.67E+01	1.24E+03 ± 4.99E+01
F6	3.60E-01 ± 3.29E-01	4.35E-03 ± 7.84E-03	2.32E+01 ± 4.27E+00	6.85E-04 ± 9.68E-04	2.82E+01 ± 6.66E+00	5.77E+01 ± 7.69E+00	1.16E+02 ± 3.14E+00	2.16E+01 ± 3.46E+00	2.76E+01 ± 4.71E+00	8.16E+01 ± 5.89E+00
F7	2.55E+02 ± 2.61E+01	9.44E+02 ± 4.72E+01	7.06E+02 ± 2.44E+02	1.02E+03 ± 4.45E+01	1.10E+03 ± 2.02E+02	1.83E+03 ± 1.54E+02	3.31E+03 ± 5.26E+01	8.98E+02 ± 1.46E+02	1.25E+03 ± 2.06E+02	2.50E+03 ± 1.76E+02
F8	1.51E+02 ± 2.56E+01	8.02E+02 ± 2.98E+01	3.52E+02 ± 1.65E+02	8.88E+02 ± 3.59E+01	5.82E+02 ± 1.22E+02	7.54E+02 ± 8.39E+01	1.82E+03 ± 4.11E+01	4.02E+02 ± 3.82E+01	4.92E+02 ± 7.70E+01	1.34E+03 ± 8.34E+01
F9	2.45E+02 ± 1.04E+02	9.91E+02 ± 1.16E+03	8.72E+03 ± 7.34E+03	2.02E+03 ± 2.74E+03	1.20E+04 ± 3.65E+03	5.35E+04 ± 6.75E+03	8.52E+04 ± 3.53E+03	8.14E+03 ± 2.15E+03	1.47E+04 ± 6.40E+03	4.57E+04 ± 3.72E+03
F10	2.00E+04 ± 6.91E+03	2.60E+04 ± 5.10E+02	2.56E+04 ± 1.58E+03	2.64E+04 ± 5.94E+02	1.43E+04 ± 1.34E+03	2.87E+04 ± 9.36E+02	3.21E+04 ± 6.68E+02	1.47E+04 ± 1.02E+03	1.53E+04 ± 3.39E+03	2.27E+04 ± 1.29E+03
F11	3.34E+02 ± 7.55E+01	7.01E+02 ± 5.44E+02	8.35E+04 ± 2.58E+05	2.52E+03 ± 9.00E+02	2.40E+03 ± 6.58E+02	1.24E+05 ± 2.67E+04	2.52E+05 ± 6.84E+04	7.51E+02 ± 1.20E+02	1.37E+03 ± 5.36E+02	8.32E+04 ± 3.32E+04
F12	3.68E+05 ± 1.38E+05	8.54E+05 ± 2.82E+05	5.97E+06 ± 1.69E+07	1.54E+06 ± 5.57E+05	1.49E+06 ± 7.49E+05	6.51E+08 ± 2.36E+08	1.89E+11 ± 1.78E+10	2.30E+06 ± 1.05E+06	5.13E+06 ± 3.01E+06	7.24E+10 ± 2.08E+10
F13	3.55E+03 ± 3.30E+03	3.80E+03 ± 2.96E+03	1.01E+04 ± 8.10E+03	3.50E+03 ± 2.14E+03	5.18E+03 ± 3.67E+03	4.25E+06 ± 2.98E+06	4.48E+10 ± 3.43E+09	5.62E+03 ± 6.80E+03	6.67E+03 ± 3.92E+03	6.64E+09 ± 2.86E+09
F14	1.98E+04 ± 1.18E+04	4.04E+04 ± 2.88E+04	7.94E+04 ± 5.89E+04	4.32E+04 ± 2.92E+04	4.80E+04 ± 3.05E+04	1.91E+06 ± 4.92E+05	1.08E+08 ± 4.15E+07	2.14E+05 ± 8.37E+04	9.12E+04 ± 4.89E+04	7.14E+04 ± 6.42E+04
F15	2.04E+03 ± 3.51E+03	1.48E+03 ± 2.11E+03	4.36E+03 ± 4.85E+03	1.27E+03 ± 1.11E+03	2.65E+03 ± 3.13E+03	2.07E+06 ± 1.79E+06	2.32E+10 ± 2.92E+09	3.39E+03 ± 3.78E+03	2.69E+03 ± 2.86E+03	7.04E+07 ± 9.08E+07
F16	2.77E+03 ± 8.24E+02	5.05E+03 ± 3.51E+02	4.07E+03 ± 8.37E+02	6.43E+03 ± 3.03E+02	3.84E+03 ± 5.72E+02	5.60E+03 ± 9.95E+02	2.18E+04 ± 2.53E+03	2.94E+03 ± 5.92E+02	3.43E+03 ± 6.68E+02	7.64E+03 ± 1.06E+03
F17	2.75E+03 ± 9.15E+02	3.10E+03 ± 2.69E+02	3.32E+03 ± 5.77E+02	3.88E+03 ± 2.34E+02	3.02E+03 ± 5.78E+02	4.47E+03 ± 7.21E+02	7.28E+06 ± 3.58E+06	2.13E+03 ± 5.10E+02	2.92E+03 ± 4.28E+02	4.39E+03 ± 1.02E+03
F18	1.20E+05 ± 4.96E+04	8.36E+04 ± 3.92E+04	2.63E+05 ± 1.82E+05	1.95E+05 ± 9.89E+04	1.28E+05 ± 6.78E+04	2.81E+06 ± 1.05E+06	2.58E+08 ± 8.62E+07	7.35E+05 ± 3.31E+05	4.29E+05 ± 2.85E+05	3.55E+05 ± 3.59E+05
F19	2.06E+03 ± 2.76E+03	1.75E+03 ± 2.03E+03	4.62E+03 ± 5.96E+03	7.18E+02 ± 6.87E+02	2.91E+03 ± 4.16E+03	6.53E+06 ± 6.57E+06	2.30E+10 ± 3.57E+09	2.23E+03 ± 2.76E+03	2.80E+03 ± 2.48E+03	2.24E+08 ± 2.95E+08
F20	2.57E+03 ± 7.39E+02	3.03E+03 ± 2.52E+02	3.12E+03 ± 4.84E+02	3.56E+03 ± 2.96E+02	2.96E+03 ± 4.57E+02	4.87E+03 ± 4.12E+02	6.13E+03 ± 3.57E+02	2.71E+03 ± 4.51E+02	2.47E+03 ± 4.12E+02	3.00E+03 ± 2.95E+02
F21	3.67E+02 ± 2.83E+01	9.89E+02 ± 2.79E+01	5.50E+02 ± 1.29E+02	1.07E+03 ± 2.27E+01	7.45E+02 ± 8.53E+01	9.16E+02 ± 9.34E+01	2.98E+03 ± 2.12E+02	6.14E+02 ± 4.89E+01	6.36E+02 ± 6.40E+01	1.54E+03 ± 1.14E+02
F22	1.99E+04 ± 7.03E+03	2.63E+04 ± 4.99E+03	2.60E+04 ± 1.25E+03	2.24E+04 ± 1.08E+04	1.49E+04 ± 3.23E+03	3.03E+04 ± 1.67E+03	3.36E+04 ± 7.21E+02	1.57E+04 ± 1.47E+03	1.67E+04 ± 3.28E+03	2.50E+04 ± 8.66E+02
F23	7.12E+02 ± 3.70E+01	1.22E+03 ± 2.90E+01	8.45E+02 ± 1.17E+02	1.35E+03 ± 2.18E+01	1.03E+03 ± 1.01E+02	1.13E+03 ± 8.00E+01	5.78E+03 ± 6.23E+02	1.12E+03 ± 8.00E+01	1.06E+03 ± 5.67E+01	1.93E+03 ± 1.03E+02
F24	7.12E+02 ± 3.70E+01	1.60E+03 ± 4.90E+01	1.33E+03 ± 1.95E+02	1.68E+03 ± 2.78E+01	1.57E+03 ± 1.55E+02	1.89E+03 ± 1.70E+02	1.06E+04 ± 7.79E+02	1.79E+03 ± 1.50E+02	1.55E+03 ± 1.47E+02	3.04E+03 ± 3.01E+02

(Continued)

Table II (continued)

	ICSBO	CSBO	IAOA	MECSBO	IOPA	ESO	SSAO	MDBO	MTVSCA	IAO
F25	1.03E + 03 ± 4.39E + 01	8.15E + 02 ± 4.45E + 01	7.95E + 02 ± 6.93E + 01	8.09E + 02 ± 4.30E + 01	8.19E + 02 ± 5.83E + 01	1.15E + 03 ± 6.72E + 01	2.39E + 04 ± 1.27E + 03	7.95E + 02 ± 6.41E + 01	8.99E + 02 ± 4.77E + 01	1.50E + 04 ± 2.29E + 03
F26	7.92E + 02 ± 7.43E + 01	1.14E + 04 ± 9.62E + 02	9.92E + 03 ± 3.02E + 03	1.14E + 04 ± 2.13E + 03	1.57E + 04 ± 4.65E + 03	2.54E + 04 ± 2.57E + 03	5.16E + 04 ± 1.84E + 03	1.35E + 04 ± 2.13E + 03	1.24E + 04 ± 2.37E + 03	3.37E + 04 ± 3.55E + 03
F27	4.81E + 03 ± 6.88E + 02	8.19E + 02 ± 5.25E + 01	1.03E + 03 ± 1.60E + 02	7.38E + 02 ± 3.29E + 01	1.11E + 03 ± 1.60E + 02	1.98E + 03 ± 3.18E + 02	1.27E + 04 ± 1.36E + 03	1.22E + 03 ± 1.82E + 02	1.21E + 03 ± 9.79E + 01	1.80E + 03 ± 2.55E + 02
F28	5.73E + 02 ± 3.50E + 01	5.99E + 02 ± 4.31E + 01	2.73E + 03 ± 8.07E + 03	6.19E + 02 ± 3.78E + 01	6.04E + 02 ± 3.60E + 01	8.93E + 02 ± 4.75E + 01	3.15E + 04 ± 1.23E + 03	6.17E + 02 ± 4.43E + 01	7.62E + 02 ± 5.51E + 01	1.75E + 04 ± 2.29E + 03
F29	1.95E + 03 ± 4.64E + 02	3.68E + 03 ± 3.48E + 02	3.52E + 03 ± 7.98E + 02	4.72E + 03 ± 2.81E + 02	4.06E + 03 ± 4.80E + 02	8.92E + 03 ± 1.53E + 03	5.39E + 05 ± 2.34E + 05	3.34E + 03 ± 5.09E + 02	4.11E + 03 ± 6.21E + 02	1.08E + 04 ± 2.07E + 03
F30	4.23E + 03 ± 1.78E + 03	5.49E + 03 ± 2.09E + 03	8.60E + 04 ± 2.01E + 05	8.23E + 03 ± 2.31E + 03	7.67E + 03 ± 4.04E + 03	6.43E + 07 ± 3.77E + 07	3.90E + 10 ± 5.62E + 09	6.47E + 04 ± 8.35E + 04	1.22E + 05 ± 1.87E + 05	1.51E + 09 ± 1.27E + 09
+/-/=		5/17/7	2/26/1	3/21/5	2/20/7	0/29/0	0/29/0	3/19/7	5/22/2	0/28/1

5.3.2 Comparison of Convergence Speed between the ICSBO Algorithm and Other Algorithms

Fig. 2 provides a visual comparison of the convergence speeds for each algorithm, displaying the convergence curves across the 29 test functions in the 30-dimensional CEC2017 test suite. The x -axis represents the number of evaluations, while the y -axis displays the logarithmic values of the fitness scores. (The subscripts of each picture in Fig. 2 indicate the tested function, and because the F2 function is not tested, it is not shown).

Fig. 2 reveals that in unimodal functions (F1 and F3), ICSBO achieves the optimal value and the fastest convergence speed. Specifically, in F1, ICSBO substantially accelerates during the middle stages of evolution compared with the other algorithms and stabilizes in the later stages until reaching the optimal value. In F3, ICSBO exhibits a clear advantage during the early and middle stages of evolution, and is second only to IAOA. In multimodal functions (F4–F10), as shown in Fig. 2, ICSBO exhibits a noticeable advantage in convergence speed in F6 and F9. In F4, the convergence speed of ICSBO is comparable with that of other algorithms in the early stages, but its performance is inferior to that of IOPA, MDBO, CSBO, and MECSBO in the later stages. In F5 and F8, ICSBO accelerates during the middle stages, with smooth convergence in the early and later stages. In F5, the performance of ICSBO is close to that of MDBO but inferior to that of IAOA. In F8, the performance of ICSBO is similar to that of MDBO and MECSBO but behind that of MECSBO. In F7, ICSBO speeds up during the middle stages and stabilizes later on; it performs similarly to IOPA and MDBO and outperforms other algorithms. In F10, the performance of ICSBO is comparable with that of other algorithms in the early and middle stages but accelerates in the later stages, although its overall performance is still inferior to that of MDBO. In hybrid functions (F11–F20), as shown in Fig. 2, ICSBO achieves the global optimum and demonstrates satisfactory convergence speed in functions F11, F12, F14–F17, and F19. In F13, F18, the convergence speed of ICSBO is similar to that of MECSBO and CSBO. In F20, ICSBO accelerates its convergence speed in the early and middle stages, and levels off its speed in the later stages; it performs similarly to MTVSCA but is overall slightly inferior to MTVSCA. In composite functions (F21–F30), ICSBO achieves the global optimum in functions F24, F28, and F29. In functions F22, F23, F25, F27, and F30, the convergence speed and accuracy of ICSBO are comparable with those of CSBO, MECSBO, IAOA, and MDBO. However, in F21, ICSBO performs poorly; only outperforms SSOA, ESO, and IAO; and performs worse than the five other algorithms. In F26, ICSBO performs worse than MDBO but better than the other algorithms. In summary, compared with the eight other algorithms, ICSBO shows a certain advantage in convergence speed.

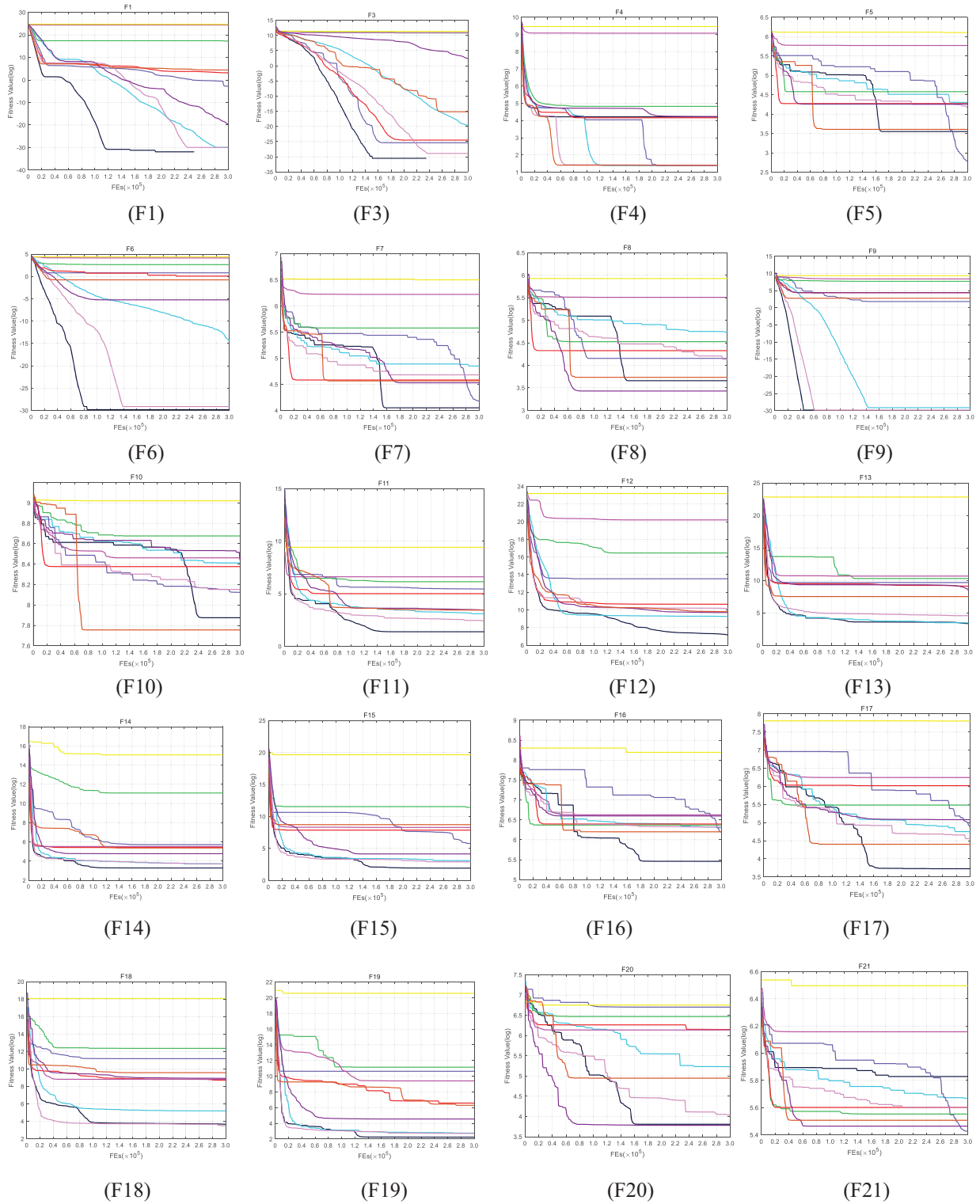


Figure 2: (Continued)

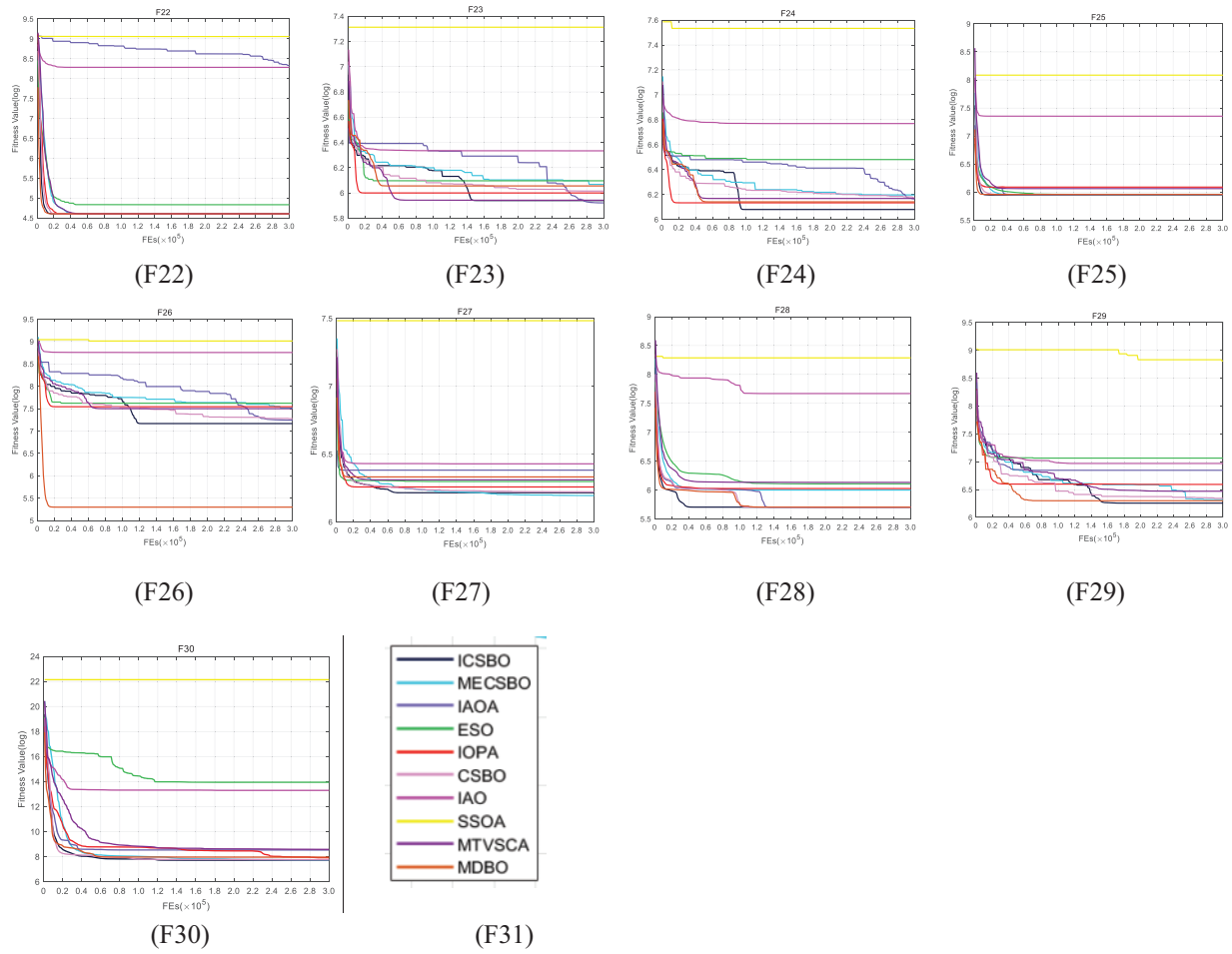


Figure 2: Convergence curves of 29 functions on CEC2017 test set between ICSBO algorithm and other 8 comparison algorithms (The different algorithms represented by the different color curves are shown in the final figure F31)

5.4 Comparison of Engineering Application Effects

To further evaluate the effectiveness of ICSBO in engineering applications, this section investigates its performance on the cooperative beamforming optimization problem. This problem, commonly encountered in antenna arrays, involves optimizing the amplitude ($\xi \in [0, 1]$) and phase ($\alpha \in [-\pi, \pi]$) of the transmit signal weights at each cooperative node. These parameters are considered as decision variables during the optimization process, with the goal of minimizing the PSL (peak sidelobe level), as specified in Eq. (21).

$$PSL = 20 \log_{10} \frac{\max |AF(\theta_{SL}, w)|}{AF(\varphi, w)} \quad (21)$$

In this context, $AF(\theta, w)$ represents the array factor, as given in Eq. (22). The value φ corresponds to the direction of the main beam, while θ_{SL} indicates the direction of the side lobes in the array factor, excluding the main beam peaks, within the range of $\theta \in [-\pi, \varphi) \cup (\varphi, \pi]$. The denominator, $AF(\varphi, w)$, refers to the power of the main beam, while the numerator, $\max |AF(\theta_{SL}, w)|$, represents the maximum power of the side lobes.

$$AF(\theta, w) = \sum_{k=1}^k w_k e^{j(2\pi/\lambda)r_k[\cos(\theta-\psi_k)]} \quad (22)$$

In this study, the ICSBO algorithm is compared with MECSBO, MDBO, and MTVSCA algorithms on the cooperative beam optimization problem to assess its effectiveness. To ensure a fair comparison, the problem size (N) in this experiment is set to 20, with a maximum of 50 function evaluations (maxit = 50), 4 nodes ($k = 4$), a pole radius of 1 ($r = 1$), and a dimension of 2 ($\dim = 2$). Additional parameter settings for each algorithm are provided in Table 5. Fig. 3 presents the beam pattern of the ICSBO algorithm alongside the comparison algorithms in the Cartesian coordinate system, with the corresponding PSL values for each algorithm annotated within the figure.

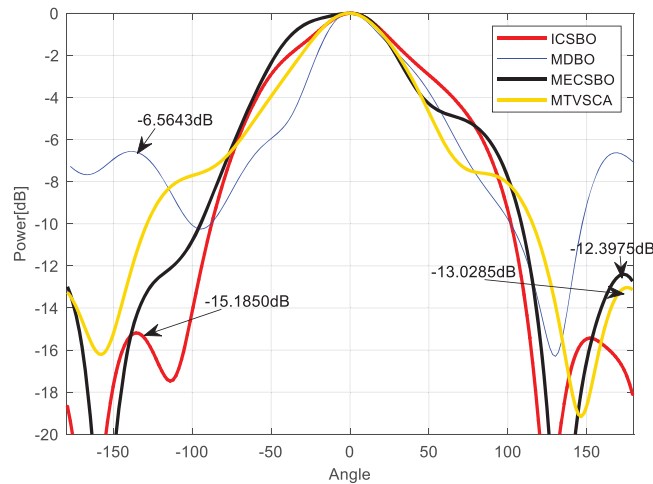


Figure 3: Beamplot of ICSBO vs. other algorithms in the Cartesian coordinate system

As shown in Fig. 3, the ICSBO algorithm achieves an optimal PSL value of -15.1850 dB, while the MECSBO, MDBO, and MTVSCA algorithms yield values of -12.3975 , -6.5643 , and -13.0285 dB, respectively. Among these, ICSBO delivers the lowest PSL, demonstrating the best performance in cooperative beam optimization. Overall, the ICSBO algorithm exhibits outstanding effectiveness in engineering applications.

6 Conclusion

This paper proposes ICSBO, which further enhances the convergence speed and local optima escape capability of CSBO. Initially, corresponding improvement strategies are introduced for the three components of CSBO: venous blood circulation, body circulation, and pulmonary circulation. These improvements enrich population diversity and reduce the likelihood of becoming trapped in local optima during evolution. By dividing the evolutionary process and improving the parameters and updating methods, the convergence speed of the population is accelerated in the early stages of evolution. Additionally, the introduction of the simplex method in the body and pulmonary circulation further speeds up convergence. Finally, to increase population diversity and reduce gene loss, an archive updating mechanism is designed to enhance the utilization probability of high-quality genes. The simulation results across multiple experiments on the CEC2017 benchmark set indicate that the proposed improvements significantly boost the overall performance of ICSBO, demonstrating significant improvements in both convergence rate and solution precision compared to other algorithms.

While the ICSBO algorithm presented in this paper offers significant benefits in convergence accuracy, its time complexity tends to rise. Future studies will aim to minimize the time complexity of ICSBO. Furthermore, the application of ICSBO in practical engineering problems will be explored in greater depth.

Acknowledgment: The authors would like to thank the anonymous reviewers and the editor for their valuable suggestions, which greatly contributed to the improved quality of this article.

Funding Statement: This work is partially supported by the Project of Scientific and Technological Innovation Development of Jilin in China under Grant 20210103090.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Yanjiao Wang, Zewei Nan; data collection: Zewei Nan; analysis and interpretation of results: Yanjiao Wang, Zewei Nan; draft manuscript preparation: Yanjiao Wang, Zewei Nan. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The data that support the findings of this study are openly available at <https://gitcode.com/open-source-toolkit/456e4> (accessed on 11 October 2024).

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Ghasemi M, Akbari MA, Jun C, Bateni SM, Zare M, Zahedi A, et al. Circulatory System Based Optimization (CSBO): an expert multilevel biologically inspired meta-heuristic algorithm. *Eng Appl Comput Fluid Mech*. 2022;16(1):1483–525. doi:10.1080/19942060.2022.2098826.
2. Garg H. An efficient biogeography based optimization algorithm for solving reliability optimization problems. *Swarm Evol Comput*. 2015;24:1–10. doi:10.1016/j.swevo.2015.05.001.
3. Karaboga D, Celal O. A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Appl Soft Comput*. 2011;11(1):652–7. doi:10.1016/j.asoc.2009.12.025.
4. Mirjalili S, Mirjalili S. Genetic algorithm. *Evol Algorithms Neural Netw: Theory Appl*. 2019;780:43–55. doi:10.1007/978-3-319-93025-1_4.
5. Mallipeddi R, Suganthan PN, Pan QK, Tasgetiren MF. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl Soft Comput*. 2011;11(2):1679–96. doi:10.1016/j.asoc.2010.04.024.
6. Cai X, Zhao H, Shang S, Zhou Y, Deng W, Chen H, et al. An improved quantum-inspired cooperative co-evolution algorithm with multi-strategy and its application. *Expert Syst Appl*. 2021;1(1):114629. doi:10.1016/j.eswa.2021.114629.
7. Dorigo M, Birattari M, Stutzle T. Ant colony optimization. *IEEE Comput Intell Mag*. 2007;1(4):28–39. doi:10.1109/MCI.2006.329691.
8. Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. *Adv Eng Softw*. 2014;69:46–61. doi:10.1016/j.advengsoft.2013.12.007.
9. Arora S, Singh S. Butterfly optimization algorithm: a novel approach for global optimization. *Soft Comput*. 2018;23:715–34. doi:10.1007/s00500-018-3102-4.
10. Braik M, Hammouri A, Atwan J, Al-Betar MA, Awadallah MA. White shark optimizer: a novel bio-inspired meta-heuristic algorithm for global optimization problems. *Knowl Based Syst*. 2022;243:108457. doi:10.1016/j.knosys.2022.108457.
11. Hashim FA, Hussain K, Houssein EH, Mabrouk MS, Al-Atabany W. Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems. *Appl Intell*. 2021;51:1531–51. doi:10.1007/s10489-020-01893-z.
12. Rao RV, Savsani VJ, Vakharia DP. Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput-Aided Design*. 2011;43(3):303–15. doi:10.1016/j.cad.2010.12.015.

13. Mohamed AW, Hadi AA, Mohamed AK. Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm. *Int J Mach Learn Cybern.* 2020;11(7):1501–29. doi:10.1007/s13042-019-01053-x.
14. Wu X, Li S, Jiang X, Zhou Y. Information acquisition optimizer: a new efficient algorithm for solving numerical and constrained engineering optimization problems. *J Supercomput.* 2024;80(18):25736–91. doi:10.1007/s11227-024-06384-3.
15. Yao L, Yuan P, Tsai CY. ESO: an enhanced snake optimizer for real-world engineering problems. *Expert Syst Appl.* 2023;230:120594. doi:10.1016/j.eswa.2023.120594.
16. Wang YJ, Chen MC, Ku CS. An improved archimedes optimization algorithm (IAOA). *J Netw Intell.* 2023;8(3):693–709.
17. Hu G, Wei G, Abbas M. Opposition-based learning boosted orca predation algorithm with dimension learning: a case study of multi-degree reduction for NURBS curves. *J Comput Des Eng.* 2023;10(2):722–57. doi:10.1093/jcde/qwad017.
18. Ye M, Zhou H, Yang H, Hu B, Wang X. Multi-strategy improved dung beetle optimization algorithm and its applications. *Biomimetics.* 2024;9(5):291. doi:10.3390/biomimetics9050291.
19. Nadimi-Shahraki MH, Taghian S, Javaheri D, Sadiq AS, Khodadadi N, Mirjalili S. MTV-SCA: multi-trial vector-based sine cosine algorithm. *Cluster Comput.* 2024;27:1–45. doi:10.1007/s10586-024-04602-4.
20. Alzoubi S, Sharaf L, Sharaf M, Daoud MS, Khodadadi N, Jia H. Synergistic swarm optimization algorithm. *CMES-Comput Model Eng Sci.* 2024;139(3):2557–604. doi:10.32604/cmes.2023.045170.
21. Abdollahzadeh B, Gharehchopogh FS, Khodadadi N, Mirjalili S. Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. *Adv Eng Softw.* 2022;174:103282. doi:10.1016/j.advengsoft.2022.103282.
22. Yang S, Guo C, Meng D, Guo Y, Guo Y, Pan L, et al. MECSBO: multi-strategy enhanced circulatory system based optimisation algorithm for global optimisation and reliability-based design optimisation problems. *IET Collab Intell Manuf.* 2024;6(2):e12097. doi:10.1049/cim2.12097.
23. Ji J, Wu T, Yang C. Neural population dynamics optimization algorithm: a novel brain-inspired meta-heuristic method. *Knowl Based Syst.* 2024;300:112194. doi:10.1016/j.knosys.2024.112194.
24. Wang F, Zhang H, Li K, Lin Z, Yang J, Shen XL. A hybrid particle swarm optimization algorithm using adaptive learning strategy. *Inf Sci.* 2018;436:162–77. doi:10.1016/j.ins.2018.01.027.