

Article

IAOO: An Improved Animated Oat Optimization Algorithm with Adaptive Multi-Strategy Search for UAV Path Planning

Xingxing Zhang ^{1,2}, Cankun Xie ² and Shaobo Li ^{2,*}

¹ College of Data Science, Guizhou Institute of Technology, Guiyang 550025, China; gs.xxzhang21@gzu.edu.cn

² Guizhou Provincial Laboratory of Big Data, State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China; gs.ckxie23@gzu.edu.cn

* Correspondence: lishaobo@gzu.edu.cn

Abstract: The recently proposed Animated Oat Optimization (AOO) algorithm exhibits competitive search behavior, but its fixed branching rules and limited use of inter-individual information may cause diversity loss and premature stagnation. This study proposes an Improved Animated Oat Optimization algorithm (IAOO) that integrates the original AOO operator, a hybrid DE/rand/1-DE/best/1 operator with a decreasing scale factor, and an elite neighborhood-directed local search within a feedback-driven framework. Strategy probabilities are updated according to normalized successful fitness gains, enabling search effort to adapt to the current optimization state. IAOO was evaluated through 30 independent runs on the CEC2017 (dim = 30/100), CEC2020, and CEC2022 suites and achieved Friedman mean ranks of 1.50, 1.37, 2.70, and 1.83, respectively, achieving competitive Friedman mean ranks among the compared algorithms and demonstrating statistically supported advantages on most benchmark suites. In three-dimensional UAV reference-path planning, IAOO reduced the mean path cost from 406.26 for AOO to 298.94, corresponding to a 26.4% reduction, while the standard deviation decreased from 67.01 to 40.73. Its runtime increased only from 26.99 s to 27.13 s. These results indicate that feedback-based operator cooperation improves solution quality and robustness with limited computational overhead. The current UAV model produces geometrically feasible and kinematically constrained reference paths; full six-degree-of-freedom tracking validation remains future work.

Keywords: Animated Oat Optimization; multi-strategy adaptive mechanism; differential evolution; meta-heuristic algorithm; global optimization; UAV path planning

MSC: 68W50

Academic Editor: Shih-Wei Lin

Received: 7 July 2026

Revised: 29 July 2026

Accepted: 4 August 2026

Published: 6 August 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

With the continuous advancement of intelligent systems and complex engineering applications, optimization problems involving high dimensionality, nonlinearity, multiple constraints, and multimodal landscapes are becoming increasingly common in engineering design, energy scheduling, machine learning, and UAV autonomous navigation [1–4]. Such problems often present irregular search spaces, a large number of local optima, and objective functions that are difficult or even impossible to differentiate. Under these conditions, conventional gradient-based methods usually rely heavily on initial solutions and may easily converge to unsatisfactory local optima, which limits their applicability in many real-world

engineering scenarios [5,6]. Therefore, metaheuristic algorithms have attracted growing attention because they can perform global search effectively without requiring strict assumptions about the mathematical properties of the optimization problem.

In recent years, researchers have developed numerous metaheuristic algorithms by drawing inspiration from natural phenomena, swarm intelligence, and evolutionary processes [7,8]. Typical examples include the Genetic Algorithm (GA), which imitates natural selection and genetic evolution [9]; Particle Swarm Optimization (PSO), which is based on the collective movement patterns of birds and fish [10]; the Grey Wolf Optimizer (GWO), which reflects the leadership hierarchy and cooperative hunting mechanism of grey wolves [11]; Differential Evolution (DE), which employs evolutionary variation and selection mechanisms related to the idea of survival of the fittest [12]; the Whale Optimization Algorithm (WOA), which simulates the bubble-net hunting strategy of humpback whales [13]; and the Great Wall Construction Algorithm (GWCA), which is designed according to the competitive selection and elimination mechanisms associated with the historical construction process of the Great Wall [14].

Meanwhile, several newly developed metaheuristic algorithms have further enriched existing optimization frameworks. For instance, the Marine Predators Algorithm (MPA) is designed by imitating the foraging strategies of marine predators [15]; The Graduate Student Evolutionary Algorithm (GSEA) abstracts common processes in postgraduate research, including topic selection, supervisor guidance, and career planning, into an optimization model [16]; The Secretary Bird Optimization Algorithm (SBOA) is established based on the adaptive survival behaviors of secretary birds in different environments [17]; The Cuckoo Catfish Optimizer (CCO) simulates the exploration, predation, and parasitic reproductive behaviors of cuckoo catfish [18]. Through population collaboration, information exchange, and random disturbance mechanisms, these algorithms can show promising global search ability in complex optimization tasks. Nevertheless, as indicated by the No Free Lunch (NFL) theorem, an optimizer that performs best on all kinds of problems does not exist [19]. Therefore, many metaheuristic algorithms may still experience reduced search efficiency, insufficient population diversity, or premature convergence, especially when solving high-dimensional and strongly multimodal problems.

UAV path planning is an important practical application of intelligent optimization methods and has been widely used in reconnaissance, civil surveying, logistics transportation, and emergency rescue tasks [20,21]. In three-dimensional environments, UAV path planning aims to determine a feasible and high-quality flight trajectory under complex spatial conditions. During this process, several requirements must be considered simultaneously, including path length, flight altitude stability, path smoothness, and obstacle avoidance. Therefore, this problem can be regarded as a high-dimensional optimization task with multiple constraints and objectives. Conventional path planning methods, such as A* and Dijkstra's algorithm [22,23], usually perform well in relatively simple or structured environments, but their adaptability may decrease when the search space becomes complex or dynamically changing. By contrast, metaheuristic algorithms have been increasingly applied to UAV path planning because of their global search capability and flexibility in handling complex constraints [6,24]. However, practical flight environments often involve uneven terrain, irregular obstacle distributions, and changing constraints, which require higher optimization accuracy, faster convergence, and stronger stability and robustness. Many existing algorithms still find it difficult to maintain a good balance among these aspects, thereby restricting their performance in real engineering scenarios.

Recent studies have increasingly emphasized real-time decision-making and vehicle constraints in autonomous navigation systems. Wang et al. proposed an improved soft actor–critic framework incorporating flexible steering for real-time obstacle avoidance of autonomous underwater vehicles, demonstrating the effectiveness of reinforcement

learning in dynamic navigation environments [25]. Meng et al. developed a constraint-embedded model predictive control approach for real-time trajectory planning of unmanned surface vehicles, highlighting the importance of explicitly integrating vehicle constraints into the planning process [26]. Although these studies mainly focus on underwater and surface vehicles rather than aerial platforms, they provide valuable insights into constraint-aware autonomous navigation.

For aerial systems, Li et al. developed a control-based beamforming and trajectory optimization framework for UAV-enabled integrated sensing and communication systems. Their approach explicitly considered both three-degree-of-freedom (3-DOF) and six-degree-of-freedom (6-DOF) UAV dynamic models, demonstrating that geometrically feasible paths may not necessarily be dynamically executable [27]. Tran et al. jointly investigated trajectory and control parameter optimization for robust quadrotor tracking, establishing an important connection between reference trajectory generation and closed-loop flight control [28]. These studies indicate that practical UAV navigation requires a clear distinction among global path planning, dynamically feasible trajectory generation, and low-level tracking control.

Metaheuristic-based UAV planners have also attracted considerable attention in recent years. Recent approaches, including multi-strategy RIME [29] and Red-Billed Blue Magpie Optimizer [30], have demonstrated competitive performance in both two-dimensional and three-dimensional UAV path planning problems. However, most existing methods combine search operators using fixed probabilities or predefined iteration stages [31]. An operator that is effective during the early exploration phase may become disruptive near convergence, while local exploitation operators may perform inefficiently when the population remains widely dispersed. Furthermore, improvements in population diversity are often indirectly inferred from final fitness values rather than being quantitatively measured through explicit diversity metrics.

Despite recent progress, four limitations remain insufficiently addressed. First, many hybrid optimizers use fixed strategy probabilities that cannot respond to landscape-dependent search requirements. Second, the accumulated improvement of an operator is frequently affected by how often it is selected, producing biased credit assignment. Third, claims concerning improved population diversity are often unsupported by explicit diversity measurements. Fourth, many UAV path-planning studies generate geometrically feasible paths without clarifying their relationship to aircraft dynamics and closed-loop tracking. These limitations motivate the feedback-driven operator allocation, normalized strategy reward, diversity analysis, and dynamics-scope discussion developed in this study.

Through the above qualitative review of mainstream metaheuristic algorithms and UAV path optimization approaches, it can be observed that existing optimizers differ significantly in core search mechanisms, applicable test environments, optimization strengths and inherent defects. To systematically organize and intuitively reveal the differences among existing studies, their core methodologies, experimental test systems, core contributions, inherent limitations, and unresolved research gaps are summarized in a unified comparative table, as shown in Table 1. This table provides a concise horizontal comparison of all representative optimization algorithms cited in this work, facilitating a clear overview of the state-of-the-art and highlighting the bottlenecks of current optimization techniques applied to high-dimensional multimodal problems and three-dimensional UAV trajectory planning tasks.

Table 1. Comparative summary of recent optimization and autonomous-navigation studies.

Study (Ref.)	Methodology	Advantages	Limitations	Identified Research Gaps
GA [9]	Swarm evolutionary algorithm inspired by natural selection, crossover & mutation operators	First mainstream evolutionary optimizer; flexible constraint handling; easy implementation	Slow convergence on high-dimensional problems; prone to population homogenization; relies heavily on crossover hyperparameters	Lack adaptive operator probability adjustment; poor performance for complex 3D UAV optimization
PSO [10]	Swarm intelligence simulating bird flocking; velocity-position update rule with cognitive/social weights	Simple structure, low computational cost, fast early exploration	Fixed inertia weight; premature convergence on multimodal landscapes; weak local exploitation	No dynamic balance of exploration/exploitation; fails in terrain-constrained 3D UAV environments
GWO [11]	Hierarchical wolf hunting model with alpha/beta/delta guide individuals	Clear leadership guidance mechanism; strong local search potential	Fixed convergence factor decay; ignores inter-individual differential information; poor high-dimensional adaptability	No multi-strategy fusion; cannot dynamically adjust search focus in complex multimodal spaces
DE [12]	Evolutionary framework with mutation/crossover/selection; utilizes individual difference vectors	Rich population diversity via differential vectors; high precision for unimodal problems	Fixed scaling factor; single mutation strategy; slow late-stage refinement	Lack adaptive F adjustment; no elite-guided local search module
WOA [13]	Humpback whale bubble-net predation & spiral search mechanism	Dual exploration-exploitation mode with spiral exploitation	Fixed probability switching; insufficient use of elite individual information	Poor performance on high-dimensional composite functions; unfit for dynamic UAV terrain constraints
GWCA [14]	Simulation of ancient wall competitive construction and elimination mechanism	Novel engineering-inspired search logic; lightweight computation	Single search mode, weak global exploration ability; no adaptive strategy switching	No fusion of evolutionary operators; unvalidated in 3D autonomous navigation tasks
MPA [15]	Marine predator-prey foraging and velocity matching strategy	Multi-stage velocity update matching prey distribution	Fixed stage division; ignores elite neighborhood fine search	Cannot adapt search rhythm according to real-time fitness gain
GSEA [16]	Abstract postgraduate research process (topic selection, supervisor guidance) as search operators	Human-behavior inspired novel search paradigm	Over-complex operator logic; high computation overhead	No application research on UAV path planning
SBOA [17]	Adaptive survival and foraging behavior of secretary birds	Balanced random search and directional movement	Fixed strategy selection probability; weak exploitation around elite solutions	Lacks adaptive multi-strategy feedback mechanism; limited 3D flight verification
CCO [18]	Simulate parasitic reproduction, predation and exploration of cuckoo catfish	Dual parasitic-global search modes	Slow convergence on high-dimensional multimodal problems	No differential information utilization; untested on complex terrain UAV tasks
AOO [32]	Imitate oat seed rolling, wind dispersal, obstacle ejection search	Novel plant-inspired metaheuristic with Lévy flight exploration	Fixed branching rules; no inter-individual difference utilization; single update framework; diversity loss & premature stagnation; fixed-probability operator switching	No feedback-based multi-strategy adaptive mechanism; missing hybrid DE and elite local search modules; insufficient balance of exploration-exploitation in high-dimensional multimodal UAV environments (This paper fills this gap via IAOO)

The Animated Oat Optimization (AOO) algorithm [32] is a recently developed nature-inspired metaheuristic that constructs its search process by imitating the rolling diffusion, hygroscopic movement, and obstacle-induced ejection behaviors of animated oat seeds in natural environments. In AOO, Lévy flight and random perturbation are used to expand the search range, while the current best solution provides directional guidance for population updating [33]. Although these mechanisms enable AOO to coordinate exploration and exploitation to a certain degree, its update process still mainly depends on a relatively simple search mode and fixed-probability switching rules. Consequently, when solving complex multimodal or high-dimensional problems, the algorithm may have difficulty maintaining sufficient global exploration while performing accurate local refinement. This limitation can cause search stagnation and reduce the quality of the obtained solutions.

To address these limitations, this study develops an enhanced Animated Oat Optimization framework, named IAOO. The proposed method retains the basic search mechanism of AOO while introducing adaptive multi-strategy cooperation to improve exploration capability, exploitation accuracy, and search robustness. The main purpose of

IAOO is to strengthen the optimization ability of AOO in complex search environments while retaining its original biological inspiration. In the proposed method, a multi-strategy adaptive update framework is designed by combining the basic AOO update rule, a hybrid differential evolution mechanism, and a directed local search strategy. During the iterative process, the selection probability of each strategy is adjusted according to its actual contribution to fitness improvement. In this way, IAOO can choose more appropriate search behaviors at different stages of optimization and maintain a more flexible balance between global exploration and local exploitation.

The main contributions of this paper are summarized as follows:

- (1) A feedback-driven multi-strategy enhancement framework is developed for AOO, where operator allocation is adaptively adjusted according to the optimization feedback. This mechanism improves the flexibility of strategy utilization without changing the fundamental optimization paradigm of AOO.
- (2) The original AOO operator, a decreasing-scale hybrid DE operator, and an elite neighborhood-directed local search operator are integrated according to clearly differentiated search roles. The AOO branch preserves Lévy-assisted stochastic exploration, the DE branch utilizes inter-individual differences to maintain population diversity, and the directed local search branch improves refinement accuracy around promising regions.
- (3) A probability-floor mechanism is introduced to prevent an operator from being permanently eliminated because of temporary stochastic failure. Together with greedy replacement, this mechanism improves the stability of strategy adaptation while preserving the monotonic best-so-far property.
- (4) The individual and cooperative contributions of the proposed mechanisms are evaluated through ablation, parameter-sensitivity, strategy-probability, population-diversity, computational-complexity, and statistical analyses.
- (5) IAOO is evaluated on the CEC2017, CEC2020, and CEC2022 benchmark suites and is further applied to constrained three-dimensional UAV reference-path planning. Its reliability is assessed using independent runs, boxplots, Wilcoxon tests and Friedman rankings.

The remainder of this paper is organized as follows. Section 2 introduces the basic principle of the original AOO algorithm and describes the proposed IAOO algorithm in detail, including the multi-strategy adaptive update mechanism, the hybrid differential evolution strategy, and the directed local search strategy. Section 3 presents the experimental settings and analyzes the performance of IAOO on the CEC2017, CEC2020, and CEC2022 benchmark suites. Section 4 applies IAOO to the three-dimensional UAV path planning problem and discusses its performance in generating feasible and high-quality flight paths. Finally, Section 5 summarizes the main findings of this study and outlines several possible directions for future research.

2. Animated Oat Optimization and the Proposed Methodology

2.1. Animated Oat Optimization

Animated Oat Optimization (AOO) is a recently proposed nature-inspired metaheuristic optimization algorithm motivated by the seed dispersal behavior of animated oat plants in natural environments [32]. By mimicking the rolling diffusion, hygroscopic motion, and obstacle-triggered ejection mechanisms of animated oat seeds, AOO constructs a search framework that combines global exploration and local exploitation to solve complex optimization problems.

The AOO algorithm models the physical rolling and ejection processes involved in seed dispersal as a two-stage optimization framework consisting of exploration and

exploitation phases, as illustrated in Figure 1. The dispersal process can be summarized by three typical mechanisms [32]:

- (1) Seed dispersion driven by external factors such as wind, water, or animals;
- (2) Rolling motion induced purely by hygroscopic movement in the absence of obstacles;
- (3) Projectile motion caused by the sudden release of stored elastic energy when obstacles are encountered.

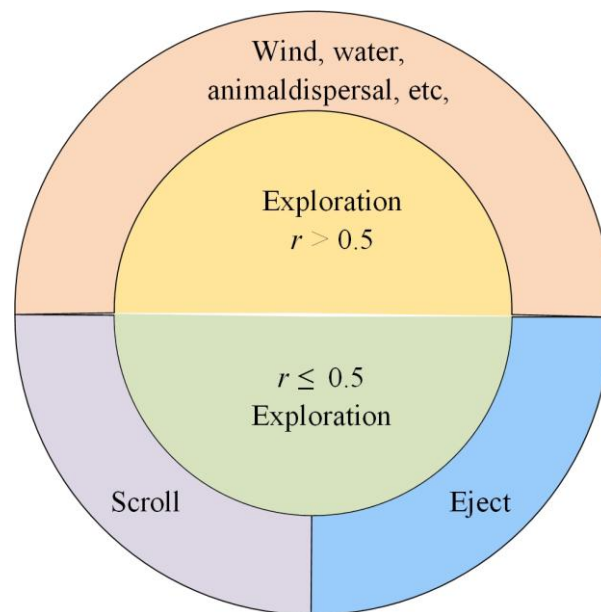


Figure 1. Exploration and exploitation phase.

From an optimization perspective, AOO consists of three main phases: the initialization phase, the exploration phase, and the exploitation phase. The corresponding mathematical models are described as follows.

(1) Initialization phase

Similar to other population-based metaheuristic algorithms, AOO adopts a random initialization strategy. The initial population is generated according to:

$$X(i) = r \cdot (UB - LB) + LB, \quad i = 1, 2, \dots, Np \quad (1)$$

where r denotes a uniformly distributed random number in the interval $[0,1]$; UB and LB represent the upper and lower bounds of the search space, respectively; and Np denotes the population size.

(2) Exploration phase

In natural environments, detached oat seeds may be dispersed by wind, water flow, or animal activity, resulting in a random spatial distribution that facilitates global exploration of the search space. This behavior is modeled in AOO through the following position update equations [32,33]:

$$\begin{cases} X_{t+1}(i) = \frac{1}{N} \sum_{i=1}^N X_t(i) + W, & \text{if } \text{mod}\left(i, \frac{Np}{10}\right) = 0, \\ X_{t+1}(i) = X_{best} + W, & \text{if } \text{mod}\left(i, \frac{Np}{10}\right) = 1, \\ X_{t+1}(i) = X_t(i) + W, & \text{otherwise.} \end{cases} \quad (2)$$

$$W = \frac{c}{\pi} \times (2 \times r_{dim} - 1) \otimes UB \quad (3)$$

where $X_t(i)$ denotes the position of the i -th individual at iteration t , X_{best} is the best solution found so far, and UB represents the upper bound of the search space.

This exploration mechanism enables individuals to move toward promising regions while maintaining sufficient randomness to enhance global search diversity.

(3) Exploitation phase

During the exploitation phase, the remaining oat seeds undergo two different diffusion strategies depending on whether obstacles are encountered, with both cases occurring with equal probability [32,33].

(a) Obstacle-free rolling diffusion (hygroscopic rolling):

When no obstacle is encountered, seed motion is primarily driven by hygroscopic rolling, which is governed by hydraulic stress gradients. The corresponding mathematical formulation is given by Equations (4)–(9).

$$A = UB - \left| \frac{UB \times t \times \sin(2\pi r)}{T} \right| \quad (4)$$

$$\begin{cases} m = 0.5 \times \frac{r}{dim} \\ L = N \times \frac{r}{dim} \\ e = 0.5 \times \frac{r}{dim} \end{cases} \quad (5)$$

$$R = (m \times e + L^2) \times r_{dim} \frac{(-A, A)}{dim} \quad (6)$$

$$\text{Levy}(dim) = 0.01 \times \frac{\mu \times \sigma}{|v|^{1/\beta}} D \quad (7)$$

$$\sigma = \left(\frac{\Gamma(1 + \beta) \times \sin\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1 + \beta}{2}\right) \times \beta \times 2^{(\beta-1)/2}} \right) \quad (8)$$

$$X_t(i) = X_{best} + R + \left[1 - \left(\frac{t}{T} \right)^3 \right] \times \text{Levy}(dim) \otimes X_{best} \quad (9)$$

where t denotes the current iteration number, T is the maximum number of iterations; m represents the mass of the oat seed; L denotes the length of the main awn; e is the eccentric rotation coefficient; and $r_{dim}(-A, A)_{dim}$ is a random matrix uniformly distributed within $[-A, A]$, with the same dimensionality as the problem. The Lévy-flight term provides nonlocal stochastic exploration in the original AOO branch. Owing to its heavy-tailed distribution, Lévy flight generates mostly small displacements and occasional long jumps. These long jumps increase the probability of escaping a locally attractive basin.

In the Lévy flight mechanism, μ is the expected value (typically in the range $[0,1]$), σ is the scale parameter controlling the step length distribution, v denotes the current velocity, β is the stability parameter (set to 1.5 in this study), and $\Gamma(\cdot)$ denotes the Gamma function.

(b) Obstacle-triggered projectile diffusion (ejection motion):

When an obstacle is encountered, the main awn releases stored elastic energy, causing the seed to undergo projectile motion. The corresponding dynamic model is expressed as Equations (10)–(13) [32,33]:

$$X_t(i) = X_{best} + R + \left[1 - \left(\frac{t}{T} \right)^3 \right] \times \text{Levy}(dim) \otimes X_{best} \quad (10)$$

$$J = \frac{2 \times k \times x^2 \times \sin(2\theta)}{mg} \times r_{dim} \frac{(-B, B)}{dim} \times (1 - \alpha) \quad (11)$$

$$B = UB - \left| \frac{UB \times t \times \cos(2\pi r)}{T} \right| \quad (12)$$

$$\begin{cases} k = 0.5 + 0.5 \times r \\ x = 3 \times \frac{r}{dim} \\ \theta = \pi \times r \\ \alpha = \frac{1}{\pi} \times e^{\frac{r'}{T}} \end{cases} \quad (13)$$

where k is the elastic coefficient of the main awn, x represents the deformation during the energy storage phase, θ denotes the projection angle, α is the air resistance coefficient, and $r \in [0, 1]$ $r' \in [0, T]$.

2.2. Proposed Improved Animated Oat Optimization

Although AOO demonstrates competitive performance on certain optimization problems, its reliance on a single update mechanism and fixed-probability branching limits its effectiveness when addressing high-dimensional or highly multimodal search spaces. To overcome the limitations of the original AOO algorithm, an Improved Animated Oat Optimization (IAOO) algorithm is proposed. IAOO enhances the adaptability, diversity, and exploitation accuracy of AOO by introducing a multi-strategy self-adaptive update framework.

2.2.1. Multi-Strategy Self-Adaptive Update Mechanism

In the standard AOO algorithm, individual position updates primarily rely on a single update framework, with exploration and exploitation controlled through random branching. While this mechanism is effective for low-dimensional or relatively simple problems, it often fails to simultaneously ensure sufficient global exploration and efficient local exploitation in complex, multimodal, or high-dimensional optimization problems.

To address this issue, IAOO introduces a multi-strategy self-adaptive update mechanism, in which the solution generation process is expanded into a pool of candidate strategies with complementary search characteristics. Specifically, a strategy pool consisting of three distinct strategies is constructed:

- (1) The original AOO update strategy;
- (2) A hybrid differential evolution strategy;
- (3) A directed local search strategy.

At each iteration, each individual selects one strategy from the pool using a roulette-wheel selection mechanism based on a strategy probability vector $p = [p_1, p_2, p_3]$. This design allows the population to dynamically balance exploration and exploitation during the search process.

To avoid search bias caused by fixed strategy probabilities, an adaptive adjustment mechanism based on actual fitness improvement is further introduced. When a strategy successfully generates a better solution, its corresponding fitness improvement is accumulated as the strategy's success score. The strategy selection probabilities are then updated using a smoothing-based rule:

$$p_s^{(t+1)} = (1 - \alpha) \cdot p_s^{(t)} + \alpha \cdot \frac{c_s}{\sum_{k=1}^s c_k} \quad (14)$$

where c_s denotes the selection probability of the s -th strategy, c_s represents its accumulated improvement, k is the number of strategies, and α is the probability smoothing factor, set to 0.2 in this study.

Through this adaptive mechanism, IAEO dynamically adjusts the utilization frequency of each strategy according to real-time search feedback, enabling more effective strategies to be preferentially selected in subsequent iterations and significantly enhancing robustness and adaptability.

2.2.2. Differential Evolution Hybrid Strategy

Although the original AEO incorporates random perturbations and Lévy flight to enhance search capability, its update direction is primarily influenced by the current individual and the global best solution, without fully utilizing inter-individual difference information. This limitation may lead to insufficient exploration or slow convergence in complex search spaces.

To enhance population diversity and fully exploit difference information among individuals, IAEO integrates differential evolution (DE) operators into the multi-strategy framework, adopting two complementary mutation strategies: DE/rand/1 and DE/best/1 [34].

For the DE/rand/1 strategy, three mutually distinct individuals are randomly selected to generate the mutant vector. In contrast, the DE/best/1 strategy uses the current global best individual as the base vector. Both strategies are executed with equal probability:

$$F(t) = F_{min} + \frac{(F_{max} - F_{min})}{1 + \exp(0.02 \times (t - T/2))} \quad (15)$$

$$V = \begin{cases} X_{r_1} + F(t) \cdot (X_{r_2} - X_{r_3}), & \text{if } rand > 0.5 \\ X_{best} + F(t) \cdot (X_{r_1} - X_{r_2}), & \text{otherwise} \end{cases} \quad (16)$$

where $rand$ is a random number between 0 and 1. F is a random scaling factor controlling the amplitude of the differential vector, and $F_{max} = 2$ and $F_{min} = 0.2$ denote its upper and lower bounds, respectively. The bounds are set to $F_{max} = 2$ and $F_{min} = 0.2$. The upper value permits relatively large differential moves during the early search stage, thereby increasing population dispersion and the probability of moving between attraction basins. The lower value avoids a complete disappearance of the differential component and supports fine-scale refinement near convergence. Equation (15) defines a monotonically decreasing logistic schedule. When $t = T/2$, $F(t) = \frac{F_{max} + F_{min}}{2} = 1.1$.

As illustrated in Figure 2, the adaptive scaling mechanism enables larger step sizes during early iterations to enhance exploration, while gradually reducing the step size in later stages to improve exploitation accuracy.

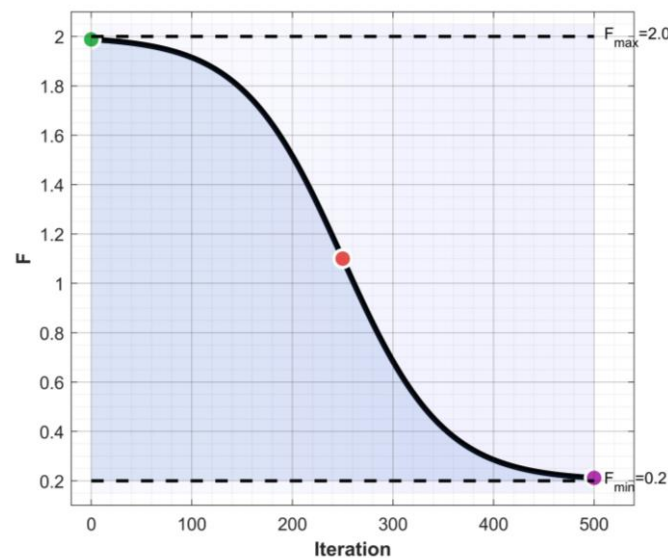


Figure 2. Scaling factor variation curve.

After mutation, a binomial crossover operator is applied to generate the trial solution:

$$U_{i,j} = \begin{cases} V_{i,j} & \text{if } rand < CR \text{ or } j = j_{rand} \\ X_{i,j}, & \text{otherwise} \end{cases} \quad (17)$$

where $CR = 0.2$ denotes the crossover probability, and j_{rand} is a randomly selected dimension index that guarantees at least one dimension undergoes mutation.

By embedding differential evolution operators into the AOO framework, IAOO effectively leverages population diversity and global-best guidance, thereby improving convergence speed and global search capability.

2.2.3. Directed Local Search Strategy

In the middle and late stages of optimization, solutions typically approach promising regions of the search space. At this stage, excessive randomness may cause oscillations around the optimal region, degrading convergence precision. The original AOO lacks a refined local exploitation mechanism, limiting its ability to fully exploit high-quality solutions.

To enhance local exploitation capability, a directed local search strategy is introduced. This strategy analyzes the spatial distribution of high-quality individuals and constructs a directionally guided perturbation mechanism based on neighborhood difference information, enabling fine-grained exploration near the global optimum while maintaining search stability.

First, individuals are sorted in ascending order according to fitness values, and the top 50% are selected as elite candidates for local search. The number of participating individuals is defined as [35]:

$$Np_{local} = \max\left(1, \left\lfloor \frac{Np}{2} \right\rfloor\right) \quad (18)$$

For each elite individual $current_x$, the Euclidean distance to all other individuals is calculated:

$$distances(m) = \sqrt{\sum_{j=1}^{dim} (X(m,j) - current_x(j))^2} \quad (19)$$

where $X(m, j)$ represents the position of the m -th individual in the j -th dimension, and dim is the dimensionality of the problem.

Subsequently, the distance vectors are sorted, and the current individual itself is excluded. The two nearest neighboring individuals are then selected and denoted as X_{nbest1} and X_{nbest2} , respectively. Based on the positional difference information between these two neighboring individuals, a directional perturbation is constructed to guide the current high-quality individual toward a more promising direction within its local neighborhood, thereby generating a candidate solution:

$$X(t+1) = current_X(t) + 2 \times rand \times (X_{nbest1}(t) - X_{nbest2}(t)) \quad (20)$$

where $rand$ denotes a dimension-wise random perturbation coefficient uniformly distributed in $[0,1]$. This coefficient is used to regulate the search step size and to avoid determinism in the search trajectory.

After applying boundary constraint handling to the generated candidate solution, its fitness value is evaluated. If the fitness of the candidate solution is superior to that of the current high-quality individual, the positional information of the individual is updated accordingly. Finally, the updated high-quality individuals are written back into the population according to their original sorting indices, thereby completing the closed-loop update process of the directional local search.

The proposed directional local search strategy effectively exploits the spatial differential information within the neighborhood of high-quality individuals to provide explicit guidance for the search direction, thus avoiding the blindness inherent in purely random local search mechanisms. Without introducing any additional gradient information, this strategy significantly enhances the local exploitation capability of the AOO algorithm in the vicinity of the global optimum, reduces the risk of premature convergence to local optima, and further improves the convergence accuracy and overall optimization efficiency of the algorithm.

Based on the preceding analysis, the complete pseudocode of the Improved Animated Oat Optimization Algorithm (IAOO) is summarized in Algorithm 1.

Algorithm 1. The Pseudocode Representation of Feedback-Driven Improved Animated Oat Optimization

Input: $N, T, D, lb, ub, f(\cdot), \alpha, p_{min}, F_{min}, F_{max}, CR, \rho$.

Output: X_{best}, f_{best} .

- 1: Initialize population X randomly within $[ub, lb]$ by Equation (1).
 - 2: Evaluate fitness of each individual and set the best solution X_{best} .
 - 3: Define strategy pool $S = \{S_1, S_2, S_3\}$.
 - 4: Set $K = 3$ and initialize and initialize strategy probability vector $\mathbf{p} = [1/3, 1/3, 1/3]$.
 - 5: **while** $t = 1:T$ **do**
 - 6: Sort the population by fitness and define the top $[\rho N]$ solutions as the elite set.
 - 7: Initialize strategy rewards $r_s = 0$ and selection counts $n_s = 0$.
 - 8: Calculate the adaptive DE scaling factor $F(t)$.
 - 9: **for** $i = 1:N$
 - 10: select S_1, S_2 , or S_3 using $\mathbf{p}$.
 - 11: **if** $S_k = S_1$ **then**
 - 12: Update position using original AOO mechanism by Equations (2)–(13).
 - 13: **else if** $S_k = S_2$ **then**
 - 14: Update position using differential evolution hybrid strategy by Equations (15)–(17).
 - 15: **else if** $S_k = S_3$ **then**
 - 16: Update position using directed local search strategy by Equations (18)–(20).
 - 17: **end if**
 - 18: Evaluate $f(U_i)$ and calculate and accumulate the normalized positive fitness gain.
 - 19: **if** $f(U_i) < f(X_i)$, replace X_i with U_i .
-

```

20:   end for
21:   Update strategy success  $c(k) = c(k) + (f_i^{old} - f_i^{new})$ .
22:   Update strategy probabilities using success accumulator by Equation (14).
23:   Update the best solution found so far  $X_{best}$ .
24: end while
25: Return  $X_{best}$  and  $f_{best}$ .

```

2.3. Computational Complexity Analysis

Let N , D , T , and C_f denote the population size, problem dimension, number of iterations, and cost of one objective-function evaluation, respectively. The original AOO requires $O(TN(D + C_f))$ time and $O(ND)$ memory.

In IAOO, roulette-wheel selection and probability updating require $O(NK)$ and $O(K)$ operations per iteration, respectively, where $K = 3$. The original AOO and DE candidate-generation strategies require $O(D)$ operations per individual.

The directed local search is the dominant additional component. For ρN elite individuals, calculating their distances to the remaining population requires $O(\rho N^2 D)$.

If the complete distance vector is sorted, an additional $O(\rho N^2 \log N)$ term is introduced. Because only the two nearest neighbors are needed, a linear two-minimum scan can eliminate this sorting term. Therefore, the optimized IAOO implementation has the time complexity $O(T[NC_f + ND + \rho N^2 D])$.

IAOO stores the population, fitness values, trial vectors, one temporary distance vector, and three strategy statistics. When distances are calculated on demand, the memory complexity is $O(ND + N + K) = O(ND)$, which is asymptotically identical to that of AOO. A complete $N \times N$ distance matrix is unnecessary and is not stored.

3. Experimental Results and Analysis

3.1. Comparison Methods and Parameter Specification

This section investigates the effectiveness of the proposed IAOO algorithm on three widely recognized and highly demanding numerical optimization benchmark sets, namely CEC2017 [36], CEC2020 [37] and CEC2022 [38], and is evaluated in comparison with several alternative algorithms. The selected methods for comparison include: Velocity Pausing Particle Swarm Optimization (VPPSO) [39], Advanced Differential Evolution (ADE) [40], Improved multi-strategy adaptive Grey Wolf Optimization (IAGWO) [41], Runge Kutta optimizer (RUN) [42], Sparrow Search Algorithm (SSA) [43], Artificial Lemming Algorithm (ALA) [44], Dung Beetle Optimizer (DBO) [45], Secretary Bird Optimization Algorithm (SBOA) [17], and Animated Oat Optimization (AOO) [32]. The parameter settings and implementation details of all evaluated algorithms are summarized in Table 2.

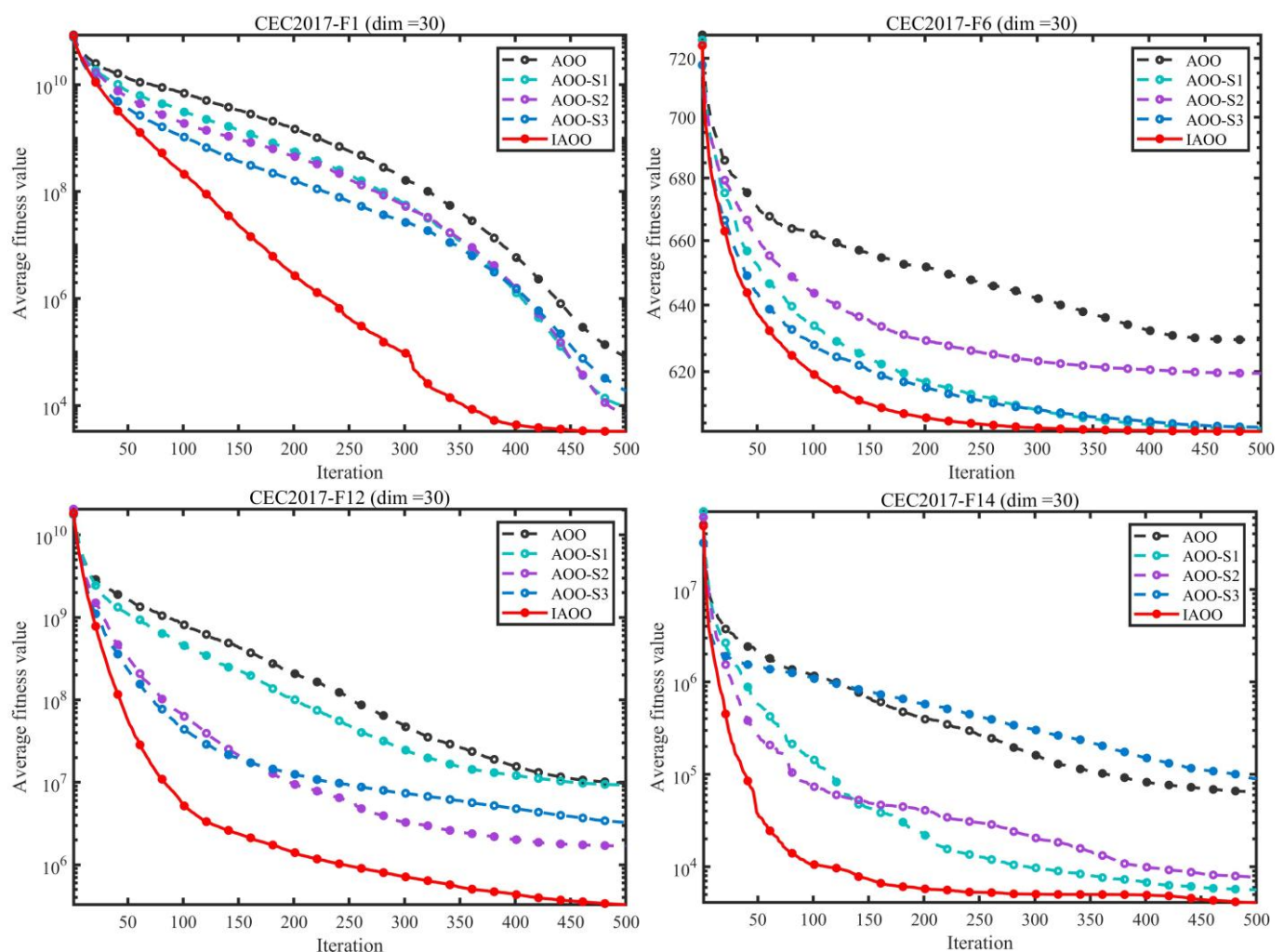
Table 2. Experimental parameter settings for all compared algorithms.

Algorithms	Name of the Parameter	Value of the Parameter
VPPSO	$c_1, c_2, w, \alpha, N_1, N_2$	2, 2, 0.8, [1,0], 0.15, 0.15
ADE	$pop_{min}, pop_{init}, p_{CR}$	4, 28, 0.2
IAGWO	$v_{rand}(t), a, \omega, \theta$	[-20,20], [0,2], [0.3,0.9], 0.5
RUN	a, b	20, 12
SSA	SD, ST	0.1, 0.8
ALA	$Prob, CF, I$	0.3, [0,2], {1, 2}
DBO	$P_{percent}$	0.2
SBOA	CF, K, R_1, R_2	[0,1], {1, 2}, [0, 1], [0, 1]
AOO	K, θ	[0.5,1.5], [0, π]
IAOO	K, θ, CR	[0.5,1.5], [0, π], 0.2

To maintain consistency across experiments and reduce stochastic effects, all algorithms were evaluated using identical control parameters, including a population size of 30 and a maximum of 500 iterations. Each algorithm was executed independently for 30 trials. The experimental results are statistically summarized using the mean (Ave) and standard deviation (Std), with superior results highlighted in bold typeface. The experimental platform comprised a Windows 11 operating system, an AMD Ryzen 7 9700X eight-core CPU operating at 3.80 GHz, 48 GB RAM, and MATLAB 2024b.

3.2. Analysis of Ablation Experiments

To analyze the individual effects and the coupling behavior of the three proposed strategies, ablation experiments were conducted by selectively disabling each strategy. Specifically, AOO-S1 incorporates only the DE/rand/1 strategy, AOO-S2 employs solely the DE/best/1 strategy, and AOO-S3 integrates only the Directed Local Search strategy. Each variant was independently executed 30 times on the CEC2017 benchmark functions, and the corresponding results are presented in Figures 3 and 4.



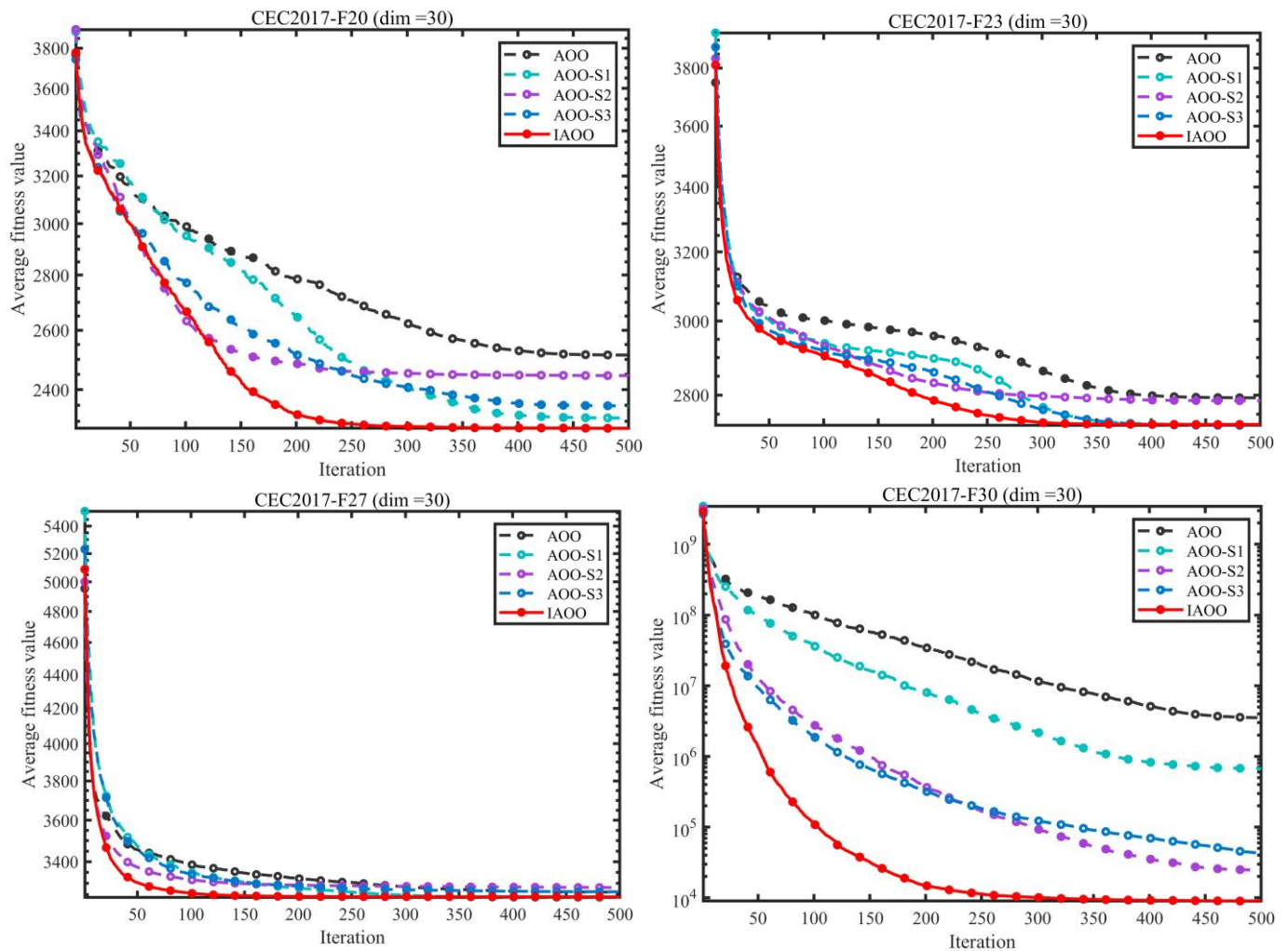


Figure 3. Fitness value curves of different strategy-improved AOO algorithms.

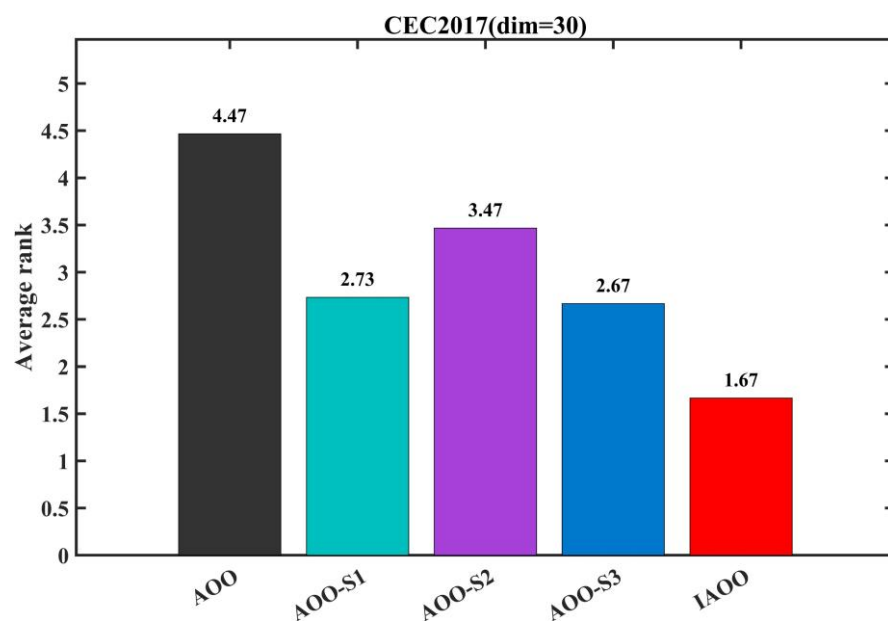


Figure 4. The average ranking of different strategy-improved AOO algorithms.

Figures 3 and 4 illustrate the performance of the AOO algorithm enhanced by different strategies in the ablation study on the 30-dimensional CEC2017 benchmark set, clearly revealing the individual contributions of each strategy as well as the synergistic effects

arising from multi-strategy coupling. As observed from the convergence curves in Figure 4, AOO-S1, which incorporates only the DE/rand/1 strategy, exhibits a faster convergence speed in the early iterations. In particular, it rapidly reduces the fitness values on functions such as F1 and F14, confirming the positive role of differential evolution in enhancing population diversity and improving global exploration capability. In contrast, AOO-S2, equipped solely with the DE/best/1 strategy, demonstrates a more stable descending trend during the middle stage of iterations. Benefiting from guidance by the global best solution, its optimization accuracy is slightly superior to that of AOO-S1, highlighting the targeted advantage of this strategy in balancing exploration and exploitation. Meanwhile, AOO-S3, which only integrates the directional local search strategy, shows outstanding local refinement ability in the later iterations. On complex functions such as F12 and F30, its convergence curves are smoother and achieve lower final fitness values, indicating its effectiveness in improving exploitation precision around promising solutions.

Compared with these three single-strategy variants, IAOO generally achieves lower fitness values and faster convergence trends on most test functions compared with the single-strategy variants. These improvements indicate that the adaptive multi-strategy framework can effectively enhance the optimization capability of AOO. Notably, on high-dimensional and complex functions such as F12 and F30, its convergence superiority persists throughout the entire evolutionary process, fully demonstrating the complementary and cooperative effects among different strategies. The average ranking results in Figure 5 further quantify the effectiveness of each strategy: IAOO achieves an average rank of 1.67, significantly outperforming all single-strategy variants (AOO-S1: 3.47, AOO-S2: 4.47, AOO-S3: 2.73) as well as the original AOO (4.67). These results indicate that the adaptive multi-strategy mechanism can dynamically balance global exploration and local exploitation, effectively compensating for the performance limitations of individual strategies in complex search spaces, and providing direct experimental evidence for the superior optimization capability of IAOO.

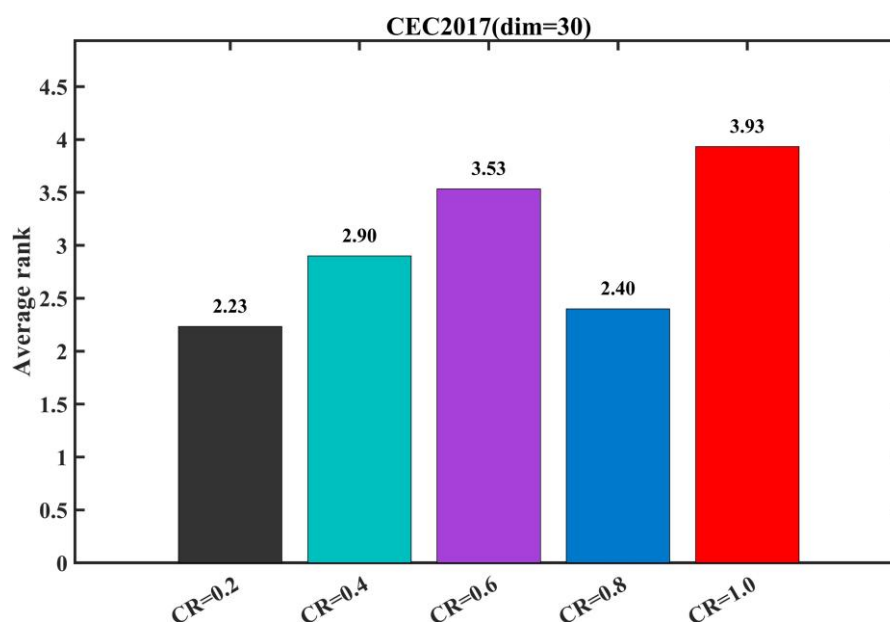


Figure 5. Average ranking of IAOO across different CR values.

3.3. Parameter Sensitivity Analysis

To determine the optimal value of the crossover probability CR , experiments were conducted on the CEC2017 benchmark suite with a dimensionality of 30, and the average rankings of IAOO under different CR settings were statistically analyzed, as shown in Figure 5.

Figure 5 presents the average rankings of IA00 with different crossover probability (CR) values on the 30-dimensional CEC2017 test set, providing an intuitive illustration of the impact of CR on the optimization performance. When $CR = 0.2$, IA00 achieves the best average ranking of 2.23, which is significantly superior to those obtained with other settings ($CR = 0.4$: 2.40, $CR = 0.6$: 2.90, $CR = 0.8$: 3.53, and $CR = 1.0$: 3.93). This indicates that a moderate crossover probability is more effective in balancing global exploration and local exploitation within the differential evolution strategies. As CR increases from 0.2 to 1.0, the average ranking exhibits a monotonically increasing trend, suggesting that an excessively large crossover probability may introduce excessive population diversity, disrupt the continuity of high-quality solutions, and consequently reduce local refinement efficiency. In contrast, setting $CR = 0.2$ allows the algorithm to introduce sufficient individual variation through limited crossover to maintain exploration capability, while avoiding excessive perturbations to promising regions already identified. As a result, the algorithm converges more rapidly toward the vicinity of the global optimum during the evolutionary process. These findings validate the rationality of setting CR to 0.2 in this study and provide clear experimental evidence for the parameter configuration of IA00, ensuring its stable and superior performance on complex high-dimensional optimization problems.

3.4. Population Diversity Analysis

Population diversity is a key quantitative indicator for evaluating the global exploration capability and premature convergence risk of metaheuristic algorithms. A higher diversity value indicates that individuals are more widely distributed throughout the solution space, thereby providing broader search coverage and a greater potential to escape from local optima. Conversely, if population diversity decreases rapidly and remains at an extremely low level for a prolonged period, the population tends to become homogeneous, substantially increasing the risk of premature stagnation. In this section, six representative 30-dimensional benchmark functions—F1, F5, F14, F18, F25, and F29—from the CEC2017 test suite are selected. The population diversity curves of the original A00 and the proposed IA00 over the entire iterative process are presented in Figure 6. By comparing their diversity evolution during the early, middle, and late stages of optimization, the effectiveness of the proposed adaptive multi-strategy framework in maintaining and dynamically regulating population diversity is investigated.

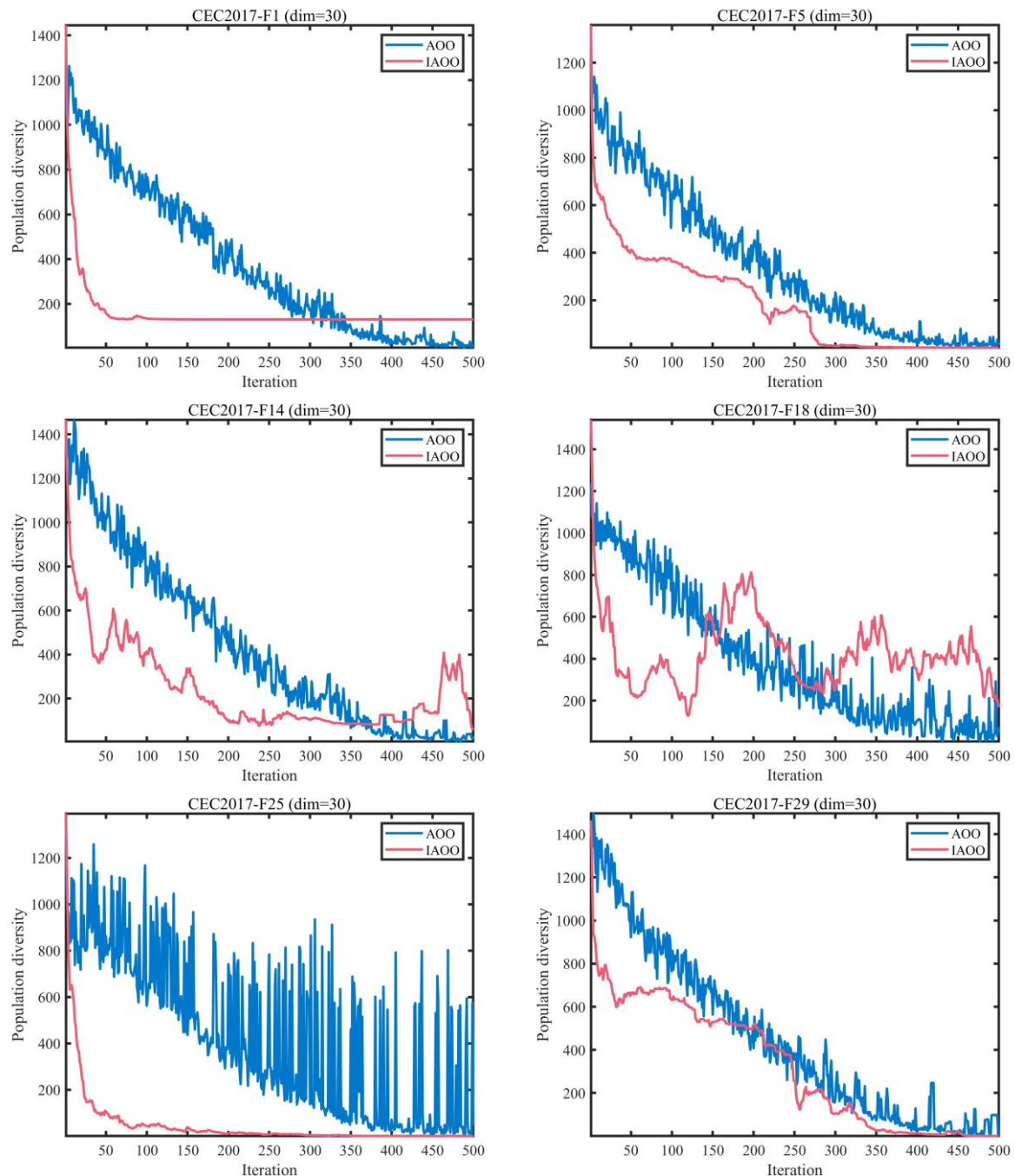


Figure 6. Population diversity evolution curves of AOO and IAEO on selected CEC2017 benchmark functions (dim = 30).

As can be intuitively observed from the trends in the subplots of Figure 6, the original AOO, represented by the blue curves, exhibits persistent large-amplitude fluctuations across all test functions. Although its population diversity decreases relatively slowly, it generally remains at a high level. This phenomenon arises because AOO switches between rolling and launching search modes according to fixed probabilities and lacks directional guidance for population updating. Consequently, individuals remain widely dispersed throughout the iterative process, while the algorithm is unable to adaptively regulate the diversity level according to the current search stage. During the early iterations, excessive random perturbations lead to redundant global exploration. In the middle and late stages, the population cannot actively contract its search range to conduct refined local exploitation. Therefore, AOO fails to achieve a dynamic balance between exploration and

exploitation, which is also a major reason for its limited convergence accuracy and persistent oscillations during the later iterations.

In contrast, the proposed IAEO, represented by the red curves, exhibits a desirable stage-wise adaptive diversity regulation pattern. During the early stage, the hybrid differential evolution strategy is selected with a high probability. By exploiting differential information among individuals to perturb the population, IAEO rapidly conducts extensive exploration of the entire search space. Consequently, a relatively high initial diversity level is maintained, enabling the algorithm to investigate a larger number of potentially promising regions. As the optimization enters the middle stage, the adaptive strategy selection mechanism gradually reduces the selection probability of differential evolution according to its contribution to fitness improvement, while increasing the application frequency of the directed local search strategy. Population diversity therefore decreases smoothly and progressively, and individuals gradually aggregate around promising regions, thereby preventing excessive dispersion from slowing convergence. During the late stage, the diversity of IAEO remains within a low but nonzero range and exhibits slight periodic fluctuations. This indicates that the population does not completely lose individual differences. Instead, the differential operator continues to introduce a small number of diverse candidate solutions during local exploitation, effectively preventing complete population homogenization and the resulting premature convergence.

The diversity differences between the two algorithms are particularly evident on the complex multimodal functions F14, F18, and F29. The diversity curves of the original AEO fluctuate sharply throughout the optimization process, indicating that the population cannot converge stably toward promising regions and instead moves irregularly among multiple local optima. By contrast, IAEO dynamically coordinates the global and local search operators through its adaptive strategy pool. Its diversity curves decline smoothly without abrupt fluctuations. The differential evolution strategy maintains an appropriate level of individual variation, allowing the population to escape from complex multimodal traps, whereas the directed local search strategy continuously narrows the search range for refined exploitation. This cooperation substantially reduces the probability of becoming trapped in local optima. IAEO also demonstrates clear advantages on the unimodal functions F1, F5, and F25. While AEO maintains excessively high diversity for an extended period and consequently wastes considerable computational resources, IAEO rapidly reduces population dispersion and concentrates the search on promising regions, thereby accelerating convergence while preserving both search efficiency and solution accuracy.

Overall, the population diversity results demonstrate that the proposed adaptive multi-strategy update mechanism and hybrid differential evolution strategy generate a synergistic effect. The differential evolution operators enrich individual differences and preserve global exploration capability during the early stage. The adaptive probability adjustment mechanism dynamically regulates population diversity across different optimization stages, while the directed local search strategy performs refined exploitation after the diversity has been appropriately reduced. Compared with the original AEO, which relies on a relatively limited search framework, IAEO can adaptively regulate the population dispersion according to the optimization progress and effectively balance global exploration and local exploitation. From a mechanistic perspective, this capability alleviates the inherent limitations of conventional AEO associated with imbalanced population diversity and premature convergence. These findings also provide an underlying explanation for the performance improvements observed in the preceding ablation experiments and convergence analyses.

3.5. Performance Assessment on the CEC Benchmark Test Suites

In this section, the performance of IAEO is examined on three commonly used benchmark suites, namely CEC2017, CEC2020, and CEC2022, and compared with several representative optimization algorithms. The benchmark functions cover different types of optimization landscapes, including unimodal, multimodal, hybrid, and composition functions. Among them, unimodal functions are mainly used to assess the exploitation ability of an algorithm because they contain only one global optimum. Multimodal functions, which include many local optima, are more suitable for evaluating global exploration and the ability to escape local traps. Hybrid and composition functions combine different landscape characteristics and therefore provide a more difficult test environment for examining robustness, convergence behavior, and resistance to premature convergence.

For a fair comparison, all algorithms were implemented under the same experimental conditions. The population size was set to 30, and the maximum number of iterations was fixed at 500. To reduce the influence of randomness, each algorithm was independently run 30 times on each test function. The optimization results were reported using the average value (Ave) and standard deviation (Std), and the best results were marked in bold. All experiments were conducted in MATLAB 2024b on a Windows 11 computer equipped with an AMD Ryzen 7 9700X eight-core processor running at 3.80 GHz and 48 GB of RAM. In addition, the convergence curves and box plots are provided to further illustrate the convergence behavior and distribution stability of the compared algorithms.

3.5.1. Evaluation Results and Discussion for the CEC2017 Test Suite

Tables 3 and 4 report the average fitness values (Ave) and standard deviations (Std) obtained by IAEO and nine comparison algorithms on the 30- and 100-dimensional CEC2017 benchmark suite. These results provide a detailed comparison of the algorithms on different types of test functions, including unimodal, multimodal, hybrid, and composition functions.

Table 3. Summary of Optimization Results for the CEC2017 Functions (dim = 30).

Function	Metric	VPPSO	ADE	IAGWO	RUN	SSA	ALA	DBO	SBOA	AOO	IAEO
F1	Ave	4.2545×10^6	5.0979×10^6	1.4445×10^6	1.5165×10^4	3.3525×10^3	3.0092×10^6	2.9040×10^8	2.3851×10^4	8.0757×10^4	4.4798×10^3
	Std	1.2340×10^7	3.1052×10^6	5.1212×10^5	7.6408×10^3	4.3035×10^3	1.8157×10^6	2.0378×10^8	3.8859×10^4	4.4082×10^4	4.8780×10^3
F2	Ave	4.0313×10^{26}	1.1975×10^{32}	4.7818×10^{34}	8.0680×10^{22}	4.0278×10^{23}	1.6404×10^{24}	3.8624×10^{35}	1.4251×10^{17}	6.8877×10^{16}	6.3266×10^{24}
	Std	2.2056×10^{27}	4.0315×10^{32}	1.9989×10^{35}	4.0777×10^{23}	1.7499×10^{24}	7.7367×10^{24}	2.0544×10^{36}	2.8380×10^{17}	1.8918×10^{17}	2.7655×10^{25}
F3	Ave	4.9416×10^4	1.4081×10^5	6.2409×10^4	9.2137×10^3	7.9458×10^4	2.9128×10^4	9.6897×10^4	2.5676×10^4	2.4278×10^4	9.6661×10^3
	Std	1.2981×10^4	3.5446×10^4	2.8329×10^4	3.4964×10^3	2.2813×10^4	9.0869×10^3	4.0361×10^4	7.3472×10^3	1.0914×10^4	8.1612×10^3
F4	Ave	5.2811×10^2	5.1113×10^2	6.2597×10^2	5.1245×10^2	5.2032×10^2	5.2504×10^2	7.2934×10^2	5.0778×10^2	5.0546×10^2	4.9431×10^2
	Std	2.1751×10^1	1.1543×10^1	1.6676×10^2	1.7283×10^1	2.5792×10^1	3.6056×10^1	1.5659×10^2	2.4420×10^1	1.7352×10^1	2.2630×10^1
F5	Ave	6.5344×10^2	6.9601×10^2	6.5077×10^2	6.9424×10^2	6.6457×10^2	6.0562×10^2	7.4772×10^2	5.8722×10^2	6.2267×10^2	5.6673×10^2
	Std	3.4020×10^1	1.3021×10^1	2.7618×10^1	3.4505×10^1	5.4108×10^1	3.6089×10^1	5.2364×10^1	2.1767×10^1	2.7990×10^1	1.9908×10^1
F6	Ave	6.4010×10^2	6.0089×10^2	6.2202×10^2	6.4779×10^2	6.5710×10^2	6.1242×10^2	6.4569×10^2	6.0258×10^2	6.2894×10^2	6.0206×10^2
	Std	8.9619×10^0	2.8222×10^{-1}	7.6439×10^0	6.0230×10^0	1.5032×10^1	4.5433×10^0	1.2071×10^1	2.2221×10^0	8.9131×10^0	1.7638×10^0
F7	Ave	9.4472×10^2	9.5628×10^2	9.2838×10^2	1.0804×10^3	9.5172×10^2	8.9551×10^2	1.0126×10^3	8.6826×10^2	8.6388×10^2	8.1298×10^2
	Std	5.0320×10^1	1.3480×10^1	3.1377×10^1	6.1922×10^1	8.7486×10^1	3.9574×10^1	9.2146×10^1	5.3299×10^1	3.9242×10^1	2.6391×10^1
F8	Ave	9.2471×10^2	9.9500×10^2	9.2050×10^2	9.4749×10^2	9.5917×10^2	9.1001×10^2	1.0215×10^3	8.7258×10^2	9.0236×10^2	8.5695×10^2
	Std	3.7127×10^1	1.3565×10^1	2.0574×10^1	2.0484×10^1	3.2973×10^1	2.4923×10^1	4.6464×10^1	1.9635×10^1	2.3596×10^1	1.8433×10^1
F9	Ave	3.4475×10^3	2.3128×10^3	4.0982×10^3	3.9038×10^3	5.6512×10^3	2.2599×10^3	7.0317×10^3	1.3621×10^3	3.8252×10^3	1.0523×10^3
	Std	1.0663×10^3	6.7727×10^2	9.3445×10^2	8.1694×10^2	1.6629×10^3	8.7703×10^2	2.5823×10^3	4.9602×10^2	1.5732×10^3	1.3864×10^2
F10	Ave	4.6927×10^3	7.5537×10^3	4.5092×10^3	4.9536×10^3	5.1846×10^3	5.8900×10^3	7.0742×10^3	4.2269×10^3	5.0092×10^3	4.3679×10^3
	Std	5.8062×10^2	4.2519×10^2	6.1104×10^2	1.1703×10^3	7.1610×10^2	6.3852×10^2	1.5086×10^3	6.3573×10^2	7.4073×10^2	6.1382×10^2
F11	Ave	1.4195×10^3	2.1805×10^3	1.9091×10^3	1.2300×10^3	1.4255×10^3	1.2768×10^3	1.8497×10^3	1.2301×10^3	1.2857×10^3	1.1889×10^3
	Std	1.6841×10^2	8.2626×10^2	8.4743×10^2	3.4849×10^1	1.1408×10^2	4.4896×10^1	5.6713×10^2	4.8097×10^1	4.8228×10^1	5.0059×10^1
F12	Ave	2.3440×10^7	2.4332×10^7	2.6397×10^7	6.1681×10^6	3.6735×10^7	3.1399×10^6	8.5387×10^7	1.1070×10^6	1.0292×10^7	1.9858×10^5
	Std	1.6093×10^7	1.9013×10^7	4.1559×10^7	4.1306×10^6	3.9844×10^7	2.4194×10^6	1.6096×10^8	9.6927×10^5	1.0000×10^7	1.9424×10^5
F13	Ave	9.0466×10^4	2.0006×10^6	2.4818×10^5	2.7492×10^4	1.3754×10^5	4.0958×10^4	1.0145×10^7	2.7602×10^4	1.0434×10^5	1.2846×10^4

F14	Std	4.7725 × 10 ⁴	3.2403 × 10 ⁶	1.2053 × 10 ⁶	1.3756 × 10⁴	9.7750 × 10 ⁴	3.2181 × 10 ⁴	2.0329 × 10 ⁷	1.9980 × 10 ⁴	5.2454 × 10 ⁴	1.6075 × 10 ⁴
	Ave	1.3197 × 10 ⁵	5.2486 × 10 ⁵	5.7837 × 10 ⁵	2.1842 × 10 ⁴	9.9716 × 10 ⁴	1.8607 × 10³	4.0265 × 10 ⁵	4.8935 × 10 ⁴	5.5743 × 10 ⁴	5.1569 × 10 ³
F15	Std	9.3117 × 10 ⁴	3.9118 × 10 ⁵	6.4632 × 10 ⁵	1.4680 × 10 ⁴	7.8596 × 10 ⁴	3.1524 × 10²	4.3328 × 10 ⁵	4.2554 × 10 ⁴	3.9426 × 10 ⁴	9.0966 × 10 ³
	Ave	1.0512 × 10 ⁵	8.1203 × 10 ⁵	5.8571 × 10³	1.3689 × 10 ⁴	6.4918 × 10 ⁴	2.4115 × 10 ⁴	5.7900 × 10 ⁴	1.1921 × 10 ⁴	4.6925 × 10 ⁴	7.9820 × 10 ³
F16	Std	3.3958 × 10 ⁵	7.3603 × 10 ⁵	5.0765 × 10 ³	2.5578 × 10³	3.7771 × 10 ⁴	1.4562 × 10 ⁴	3.8197 × 10 ⁴	1.2341 × 10 ⁴	2.9802 × 10 ⁴	7.7809 × 10 ³
	Ave	2.8478 × 10 ³	2.8974 × 10 ³	3.0809 × 10 ³	2.8740 × 10 ³	3.0319 × 10 ³	2.7722 × 10 ³	3.4023 × 10 ³	2.3702 × 10 ³	2.6929 × 10 ³	2.2860 × 10³
F17	Std	4.0475 × 10 ²	2.4201 × 10 ²	3.3470 × 10 ²	3.5664 × 10 ²	4.0766 × 10 ²	2.7920 × 10 ²	4.5137 × 10 ²	3.2829 × 10 ²	3.0335 × 10 ²	2.2834 × 10²
	Ave	2.1924 × 10 ³	2.2395 × 10 ³	2.3771 × 10 ³	2.1750 × 10 ³	2.3790 × 10 ³	2.1900 × 10 ³	2.6386 × 10 ³	1.9671 × 10³	2.1721 × 10 ³	1.9683 × 10 ³
F18	Std	1.6388 × 10 ²	1.2492 × 10 ²	2.4422 × 10 ²	2.2229 × 10 ²	2.4214 × 10 ²	2.0763 × 10 ²	3.1559 × 10 ²	1.1185 × 10²	1.9633 × 10 ²	1.4156 × 10 ²
	Ave	1.3549 × 10 ⁶	3.3045 × 10 ⁶	2.2364 × 10 ⁶	1.3266 × 10 ⁵	2.7510 × 10 ⁶	9.0357 × 10⁴	4.1234 × 10 ⁶	6.1666 × 10 ⁵	6.9608 × 10 ⁵	9.2927 × 10 ⁴
F19	Std	1.6000 × 10 ⁶	1.4468 × 10 ⁶	2.3374 × 10 ⁶	8.0324 × 10 ⁴	3.3137 × 10 ⁶	3.9951 × 10⁴	5.0364 × 10 ⁶	5.8346 × 10 ⁵	8.5661 × 10 ⁵	7.4642 × 10 ⁴
	Ave	2.1594 × 10 ⁶	4.8693 × 10 ⁵	8.6722 × 10 ³	3.1292 × 10 ⁴	4.6418 × 10 ⁶	1.9814 × 10 ⁴	4.6254 × 10 ⁶	9.0168 × 10 ³	4.9011 × 10 ⁵	7.4202 × 10³
F20	Std	1.4756 × 10 ⁶	4.2702 × 10 ⁵	1.0226 × 10 ⁴	5.5266 × 10 ⁴	3.3632 × 10 ⁶	1.6238 × 10 ⁴	1.3397 × 10 ⁷	1.1010 × 10 ⁴	6.5292 × 10 ⁵	6.0618 × 10³
	Ave	2.4888 × 10 ³	2.5635 × 10 ³	2.5700 × 10 ³	2.5194 × 10 ³	2.7427 × 10 ³	2.5304 × 10 ³	2.7785 × 10 ³	2.2320 × 10³	2.4782 × 10 ³	2.3038 × 10 ³
F21	Std	1.8056 × 10 ²	1.1430 × 10 ²	1.4645 × 10 ²	1.7637 × 10 ²	2.0136 × 10 ²	1.7790 × 10 ²	2.7742 × 10 ²	9.3484 × 10¹	1.7260 × 10 ²	1.5438 × 10 ²
	Ave	2.4321 × 10 ³	2.4925 × 10 ³	2.4521 × 10 ³	2.4435 × 10 ³	2.4637 × 10 ³	2.4274 × 10 ³	2.5345 × 10 ³	2.3658 × 10 ³	2.4114 × 10 ³	2.3578 × 10³
F22	Std	2.8857 × 10 ¹	1.3137 × 10¹	2.5396 × 10 ¹	3.6906 × 10 ¹	4.0890 × 10 ¹	2.5306 × 10 ¹	4.0591 × 10 ¹	2.0349 × 10 ¹	2.8063 × 10 ¹	1.7856 × 10 ¹
	Ave	3.4380 × 10 ³	8.6467 × 10 ³	3.5481 × 10 ³	4.3023 × 10 ³	4.9043 × 10 ³	6.8115 × 10 ³	5.0642 × 10 ³	2.5245 × 10³	4.9853 × 10 ³	2.5672 × 10 ³
F23	Std	1.8785 × 10 ³	9.1450 × 10 ²	2.1644 × 10 ³	2.2384 × 10 ³	2.1901 × 10 ³	2.2013 × 10 ³	2.4815 × 10 ³	8.6696 × 10²	2.1354 × 10 ³	1.0119 × 10 ³
	Ave	2.8165 × 10 ³	2.8402 × 10 ³	3.1316 × 10 ³	2.7994 × 10 ³	2.8203 × 10 ³	2.7842 × 10 ³	3.0045 × 10 ³	2.7277 × 10 ³	2.7967 × 10 ³	2.7149 × 10³
F24	Std	4.1534 × 10 ¹	1.5435 × 10¹	1.8933 × 10 ²	3.9855 × 10 ¹	6.0310 × 10 ¹	3.7420 × 10 ¹	9.0672 × 10 ¹	2.6201 × 10 ¹	5.1786 × 10 ¹	1.8505 × 10 ¹
	Ave	2.9622 × 10 ³	3.0448 × 10 ³	3.3317 × 10 ³	2.9576 × 10 ³	2.9379 × 10 ³	2.9600 × 10 ³	3.2175 × 10 ³	2.8946 × 10³	2.9518 × 10 ³	2.8969 × 10 ³
F25	Std	3.6257 × 10 ¹	1.0873 × 10¹	1.3307 × 10 ²	3.5919 × 10 ¹	3.2823 × 10 ¹	3.5606 × 10 ¹	9.0871 × 10 ¹	2.8546 × 10 ¹	3.4305 × 10 ¹	2.9792 × 10 ¹
	Ave	2.9432 × 10 ³	2.9026 × 10 ³	2.9586 × 10 ³	2.9191 × 10 ³	2.9404 × 10 ³	2.9192 × 10 ³	3.0022 × 10 ³	2.9039 × 10 ³	2.9153 × 10 ³	2.9021 × 10³
F26	Std	2.6681 × 10 ¹	7.4203 × 10⁰	3.0136 × 10 ¹	2.5615 × 10 ¹	3.4332 × 10 ¹	2.5200 × 10 ¹	7.3744 × 10 ¹	1.9562 × 10 ¹	2.8051 × 10 ¹	1.8525 × 10 ¹
	Ave	4.8222 × 10 ³	5.5935 × 10 ³	5.1060 × 10 ³	6.4410 × 10 ³	5.2717 × 10 ³	5.0767 × 10 ³	6.9511 × 10 ³	4.1075 × 10³	4.6855 × 10 ³	4.2105 × 10 ³
F27	Std	1.3230 × 10 ³	1.7581 × 10²	1.8568 × 10 ³	1.3195 × 10 ³	6.4954 × 10 ²	3.4239 × 10 ²	8.1346 × 10 ²	7.7955 × 10 ²	1.0225 × 10 ³	8.8778 × 10 ²
	Ave	3.3017 × 10 ³	3.2291 × 10 ³	3.3093 × 10 ³	3.2859 × 10 ³	3.2759 × 10 ³	3.2518 × 10 ³	3.3164 × 10 ³	3.2194 × 10³	3.2569 × 10 ³	3.2484 × 10 ³
F28	Std	4.5501 × 10 ¹	5.1900 × 10⁰	2.7913 × 10 ²	2.8997 × 10 ¹	3.7066 × 10 ¹	3.7686 × 10 ¹	4.5266 × 10 ¹	1.2338 × 10 ¹	2.5027 × 10 ¹	2.0913 × 10 ¹
	Ave	3.3222 × 10 ³	3.3057 × 10 ³	3.4106 × 10 ³	3.2569 × 10 ³	3.3023 × 10 ³	4.1639 × 10 ³	3.5586 × 10 ³	3.2507 × 10 ³	3.2487 × 10 ³	3.2192 × 10³
F29	Std	3.0302 × 10 ¹	2.7999 × 10 ¹	2.3267 × 10 ²	2.2512 × 10 ¹	3.1538 × 10 ¹	1.4738 × 10 ³	4.7402 × 10 ²	2.4532 × 10 ¹	2.3298 × 10 ¹	1.9257 × 10¹
	Ave	4.2842 × 10 ³	4.1698 × 10 ³	4.0161 × 10 ³	4.2698 × 10 ³	4.4146 × 10 ³	3.8988 × 10 ³	4.4478 × 10 ³	3.6494 × 10³	4.0899 × 10 ³	3.6747 × 10 ³
F30	Std	2.7752 × 10 ²	1.5517 × 10²	3.0326 × 10 ²	2.9934 × 10 ²	2.9864 × 10 ²	2.2966 × 10 ²	3.4437 × 10 ²	1.5878 × 10 ²	2.5544 × 10 ²	1.7045 × 10 ²
	Ave	9.7906 × 10 ⁶	3.9901 × 10 ⁵	1.1122 × 10 ⁵	4.3573 × 10 ⁵	1.1781 × 10 ⁷	9.6359 × 10 ⁴	3.1570 × 10 ⁶	9.8963 × 10 ⁴	3.7874 × 10 ⁶	8.5369 × 10³
	Std	6.1319 × 10 ⁶	3.1147 × 10 ⁵	5.8101 × 10 ⁵	4.0948 × 10 ⁵	8.5997 × 10 ⁶	1.2761 × 10 ⁵	6.4346 × 10 ⁶	3.8308 × 10 ⁵	1.9147 × 10 ⁶	1.7305 × 10³

Bold values indicate the best-performing result in that row or column.

Table 4. Summary of Optimization Results for the CEC2017 Functions (dim = 100).

Function	Metric	VPPSO	ADE	IAGWO	RUN	SSA	ALA	DBO	SBOA	AOO	IAOO
F1	Ave	1.7534×10^{10}	8.0834×10^9	3.2899×10^8	3.1531×10^9	1.4095×10^{10}	2.3039×10^{10}	7.3254×10^{10}	1.3594×10^{10}	4.3360×10^8	2.3899×10^7
	Std	4.2076×10^9	1.4277×10^9	1.0292×10^8	1.4615×10^9	4.7464×10^9	6.0869×10^9	6.4037×10^{10}	6.6876×10^9	1.6564×10^8	8.7823×10^6
F2	Ave	5.4083×10^{145}	1.0552×10^{156}	1.4663×10^{158}	1.2177×10^{140}	3.9167×10^{139}	2.9113×10^{136}	8.6948×10^{159}	9.8122×10^{119}	9.0021×10^{118}	2.2118×10^{155}
	Std	2.6443×10^{146}	6.5535×10^4	6.5535×10^4	4.7097×10^{140}	2.1452×10^{140}	1.5279×10^{137}	6.5535×10^4	5.3744×10^{120}	4.8772×10^{119}	6.5535×10^4
F3	Ave	3.7909×10^5	8.2509×10^5	4.5881×10^5	3.0303×10^5	6.6288×10^5	3.1388×10^5	6.9417×10^5	3.3500×10^5	5.6616×10^5	3.4075×10^5
	Std	4.7338×10^4	1.6250×10^5	5.4886×10^4	5.8794×10^4	2.2868×10^5	2.8310×10^4	2.8851×10^5	2.5574×10^4	8.0415×10^4	5.3355×10^4
F4	Ave	3.0913×10^3	1.6740×10^3	2.3760×10^3	1.7091×10^3	2.0494×10^3	3.0627×10^3	1.5425×10^4	1.7568×10^3	1.0570×10^3	9.3376×10^2
	Std	5.3589×10^2	1.7795×10^2	2.0720×10^3	2.5244×10^2	3.6047×10^2	8.6672×10^2	1.3853×10^4	3.5350×10^2	8.7732×10^1	7.1973×10^1
F5	Ave	1.3524×10^3	1.6569×10^3	1.2705×10^3	1.3265×10^3	1.5093×10^3	1.4250×10^3	1.7006×10^3	1.1773×10^3	1.2781×10^3	9.3132×10^2
	Std	8.0858×10^1	3.2174×10^1	6.1517×10^1	4.0779×10^1	1.1096×10^2	9.8651×10^1	2.3407×10^2	6.3711×10^1	1.0194×10^2	6.0307×10^1
F6	Ave	6.6456×10^2	6.2200×10^2	6.4877×10^2	6.6576×10^2	6.7692×10^2	6.5934×10^2	6.7718×10^2	6.3848×10^2	6.6350×10^2	6.2609×10^2
	Std	4.9112×10^0	1.8437×10^0	5.8713×10^0	1.9039×10^0	5.2772×10^0	8.2621×10^0	1.2911×10^1	4.4393×10^0	7.8495×10^0	4.9385×10^0
F7	Ave	2.5734×10^3	2.2385×10^3	2.2248×10^3	2.9796×10^3	2.8939×10^3	2.5747×10^3	2.9633×10^3	2.3590×10^3	1.9610×10^3	1.6102×10^3
	Std	3.1802×10^2	7.3222×10^1	1.8784×10^2	1.2232×10^2	3.2266×10^2	1.7887×10^2	2.4701×10^2	2.3803×10^2	2.4241×10^2	2.0651×10^2
F8	Ave	1.6874×10^3	1.9326×10^3	1.5974×10^3	1.7666×10^3	1.9352×10^3	1.7601×10^3	2.1007×10^3	1.5096×10^3	1.5704×10^3	1.2310×10^3
	Std	1.0153×10^2	4.8005×10^1	7.5809×10^1	7.4822×10^1	1.4465×10^2	1.1080×10^2	2.1633×10^2	9.2128×10^1	7.3867×10^1	4.7934×10^1
F9	Ave	2.6771×10^4	5.3526×10^4	4.0733×10^4	2.6689×10^4	4.1657×10^4	4.6067×10^4	7.6363×10^4	2.6885×10^4	3.7744×10^4	1.2980×10^4
	Std	3.3389×10^3	1.0855×10^4	6.2770×10^3	2.0688×10^3	5.2582×10^3	1.0912×10^4	9.2750×10^3	5.2431×10^3	7.0748×10^3	3.5141×10^3
F10	Ave	1.7152×10^4	3.1861×10^4	1.6295×10^4	1.9069×10^4	1.8657×10^4	2.7929×10^4	2.9587×10^4	1.7813×10^4	1.7022×10^4	1.4565×10^4
	Std	1.2727×10^3	6.7454×10^2	2.5049×10^3	4.7268×10^3	1.4121×10^3	2.1385×10^3	3.9940×10^3	1.3847×10^3	1.5477×10^3	1.2541×10^3
F11	Ave	7.4541×10^4	2.2831×10^5	9.8052×10^4	1.2180×10^4	1.5132×10^5	6.9442×10^4	2.4274×10^5	3.6986×10^4	3.0892×10^4	1.1313×10^4
	Std	1.4858×10^4	4.9544×10^4	5.7904×10^4	2.7946×10^3	3.7059×10^4	1.9026×10^4	6.2783×10^4	1.0957×10^4	9.4551×10^3	5.8138×10^3
F12	Ave	1.3867×10^9	3.0795×10^9	2.7489×10^9	6.0210×10^8	1.6796×10^9	1.5427×10^9	6.6413×10^9	4.3566×10^8	5.6987×10^8	8.3571×10^7
	Std	5.2329×10^8	8.6580×10^8	8.9562×10^9	1.7393×10^8	8.1350×10^8	5.0126×10^8	1.9321×10^9	5.2775×10^8	2.6414×10^8	4.3370×10^7
F13	Ave	6.0578×10^4	2.3911×10^5	3.4414×10^5	3.8172×10^4	6.5752×10^5	3.5473×10^6	3.1372×10^8	5.2300×10^4	7.9218×10^4	6.6254×10^3

	Std	2.4224×10^4	9.2828×10^5	1.1885×10^6	8.0928×10^3	2.5951×10^6	3.4646×10^6	2.2721×10^8	2.2954×10^4	2.6541×10^4	2.9768×10^3
F14	Ave	6.6435×10^6	5.2608×10^7	8.2838×10^6	7.1940×10^5	9.7573×10^6	3.1951×10^6	1.9601×10^7	3.5481×10^6	3.6481×10^6	6.0906×10^5
	Std	2.5056×10^6	1.4795×10^7	5.5840×10^6	2.2926×10^5	6.3436×10^6	1.7243×10^6	1.6143×10^7	1.8382×10^6	2.0211×10^6	3.3375×10^5
F15	Ave	5.5175×10^4	1.0295×10^6	1.1953×10^5	2.3069×10^4	9.5782×10^4	2.1462×10^6	6.7818×10^7	5.9346×10^3	5.7481×10^4	4.4397×10^3
	Std	2.6181×10^4	1.8429×10^6	4.6083×10^5	5.7821×10^3	4.0057×10^4	6.3191×10^6	7.4074×10^7	2.1771×10^3	2.1589×10^4	2.3543×10^3
F16	Ave	8.1349×10^3	1.1520×10^4	7.0124×10^3	7.7591×10^3	7.9624×10^3	8.0915×10^3	8.9465×10^3	5.7037×10^3	6.5675×10^3	5.2560×10^3
	Std	9.6826×10^2	3.3184×10^2	1.5839×10^3	9.3795×10^2	8.5746×10^2	8.6664×10^2	1.3999×10^3	7.8664×10^2	6.7175×10^2	6.5239×10^2
F17	Ave	5.4750×10^3	8.1254×10^3	3.4663×10^4	5.6941×10^3	5.9113×10^3	6.3582×10^3	9.3911×10^3	5.0244×10^3	5.3906×10^3	4.4745×10^3
	Std	6.2125×10^2	3.0582×10^2	1.5659×10^5	6.5537×10^2	5.7704×10^2	5.8918×10^2	1.3789×10^3	5.5494×10^2	4.7553×10^2	4.8501×10^2
F18	Ave	4.8894×10^6	7.7631×10^7	1.2658×10^7	1.2781×10^6	9.8868×10^6	3.0351×10^6	2.9249×10^7	4.5833×10^6	5.4089×10^6	1.5423×10^6
	Std	2.1417×10^6	2.5169×10^7	2.5457×10^7	4.3213×10^5	5.9881×10^6	1.1479×10^6	1.7958×10^7	2.6190×10^6	2.8312×10^6	8.6055×10^5
F19	Ave	9.4460×10^6	3.5252×10^6	5.2925×10^6	9.5472×10^5	2.7858×10^7	6.9302×10^5	8.8897×10^7	6.8876×10^3	5.3564×10^6	3.5764×10^3
	Std	8.6067×10^6	6.4056×10^6	2.8624×10^7	4.6777×10^5	1.5175×10^7	1.8315×10^6	9.3536×10^7	4.0348×10^3	2.7963×10^6	1.6515×10^3
F20	Ave	5.2486×10^3	7.6298×10^3	5.4793×10^3	5.1597×10^3	5.6644×10^3	6.6793×10^3	7.2359×10^3	4.8008×10^3	5.2421×10^3	4.5727×10^3
	Std	4.4607×10^2	3.3098×10^2	5.2137×10^2	5.3658×10^2	5.9801×10^2	7.0698×10^2	6.9766×10^2	6.9747×10^2	5.5438×10^2	5.8494×10^2
F21	Ave	3.1918×10^3	3.4786×10^3	4.1762×10^3	3.1251×10^3	3.3948×10^3	3.2560×10^3	4.0486×10^3	2.9583×10^3	3.1306×10^3	2.7549×10^3
	Std	9.2068×10^1	3.5013×10^1	6.2244×10^2	1.0984×10^2	1.7952×10^2	1.1059×10^2	1.9407×10^2	7.4491×10^1	9.8322×10^1	6.0817×10^1
F22	Ave	2.1447×10^4	3.3833×10^4	2.0135×10^4	2.2445×10^4	2.2251×10^4	2.9343×10^4	2.9308×10^4	2.0637×10^4	1.9908×10^4	1.7065×10^4
	Std	1.8511×10^3	5.3114×10^2	2.4737×10^3	3.8058×10^3	1.6537×10^3	2.6226×10^3	5.1701×10^3	1.8372×10^3	3.4676×10^3	3.1343×10^3
F23	Ave	4.0138×10^3	3.8049×10^3	6.0277×10^3	3.6746×10^3	3.8222×10^3	3.7505×10^3	4.7752×10^3	3.3615×10^3	3.7788×10^3	3.4322×10^3
	Std	1.7651×10^2	3.0848×10^1	4.1316×10^2	1.0985×10^2	1.6613×10^2	1.7091×10^2	1.9212×10^2	7.1415×10^1	1.2044×10^2	1.1842×10^2
F24	Ave	4.6462×10^3	4.3549×10^3	7.0329×10^3	4.7687×10^3	4.4800×10^3	4.5654×10^3	6.0483×10^3	4.0858×10^3	4.5015×10^3	3.8813×10^3
	Std	2.1944×10^2	3.9452×10^1	1.3110×10^3	1.8320×10^2	1.4540×10^2	2.4305×10^2	5.6564×10^2	1.0411×10^2	1.7768×10^2	1.3406×10^2
F25	Ave	5.1023×10^3	5.0789×10^3	5.0606×10^3	4.0186×10^3	5.0755×10^3	5.7134×10^3	7.1639×10^3	4.3825×10^3	3.7771×10^3	3.6278×10^3
	Std	3.6920×10^2	3.5534×10^2	1.2813×10^3	1.0760×10^2	4.0546×10^2	4.5163×10^2	2.7634×10^3	3.9682×10^2	1.0869×10^2	5.4532×10^1
F26	Ave	2.2407×10^4	1.7140×10^4	1.9444×10^4	2.9091×10^4	1.9996×10^4	1.7683×10^4	2.5957×10^4	1.7885×10^4	1.6398×10^4	1.6509×10^4
	Std	3.0078×10^3	4.7654×10^2	5.8135×10^3	2.2319×10^3	3.4096×10^3	2.1818×10^3	3.5865×10^3	3.4243×10^3	3.2545×10^3	4.3981×10^3
F27	Ave	4.5006×10^3	4.1970×10^3	5.6411×10^3	4.7549×10^3	4.2905×10^3	3.7708×10^3	4.6202×10^3	3.7391×10^3	4.0340×10^3	3.8701×10^3
	Std	2.4302×10^2	1.6816×10^2	2.4162×10^3	3.4488×10^2	2.5080×10^2	1.2161×10^2	4.9522×10^2	1.1994×10^2	1.6398×10^2	1.9704×10^2
F28	Ave	6.0897×10^3	1.3797×10^4	6.1299×10^3	4.5494×10^3	6.8673×10^3	9.4766×10^3	1.6654×10^4	4.9774×10^3	3.9232×10^3	3.7255×10^3
	Std	6.1221×10^2	1.7241×10^3	2.2555×10^3	2.1008×10^2	1.6928×10^3	2.9479×10^3	5.8029×10^3	6.5996×10^3	1.2923×10^2	7.5754×10^1
F29	Ave	1.0331×10^4	1.0433×10^4	8.0980×10^3	1.1206×10^4	1.0793×10^4	8.8473×10^3	1.1639×10^4	7.0960×10^3	8.9064×10^3	6.7348×10^3
	Std	8.5494×10^2	6.3637×10^2	2.4578×10^3	1.0397×10^3	1.0550×10^3	8.8979×10^2	4.5289×10^3	4.8555×10^2	8.8178×10^2	5.4070×10^2
F30	Ave	3.5317×10^8	6.1822×10^6	5.1315×10^8	5.5279×10^7	4.2319×10^8	5.6357×10^6	2.7204×10^8	1.1513×10^6	1.0275×10^8	9.7542×10^4
	Std	1.5091×10^8	3.7028×10^6	2.3752×10^9	3.0751×10^7	2.2077×10^8	3.4315×10^6	1.6602×10^8	1.0843×10^6	4.3448×10^7	7.1542×10^4

Bold values indicate the best-performing result in that row or column.

Under the 30-dimensional setting (Table 3), IA00 achieves the best mean objective values on the vast majority of unimodal, multimodal, hybrid, and composition functions. For the unimodal functions F1 and F3, IA00 obtains mean values of 4.4798×10^3 and 9.6661×10^3 , respectively, which are substantially lower than the corresponding values of 8.0757×10^4 and 2.4278×10^4 achieved by the original A00. IA00 also outperforms advanced comparison algorithms such as DBO and IAGWO by several orders of magnitude. These results demonstrate that the hybrid differential evolution strategy significantly strengthens the global exploration capability and enables the algorithm to rapidly approach the global optimum. The multimodal and composition functions F5, F9, F12, and F30 contain numerous local optima and therefore serve as critical test cases for assessing the ability of an algorithm to escape from local optima. On these functions, the superiority of IA00 becomes even more pronounced. Taking the composition function F30 as an example, IA00 achieves a mean value of only 8.5369×10^3 , whereas the original A00 obtains 3.7874×10^6 , representing a difference of nearly three orders of magnitude. For F12, a highly nonlinear and complex function, IA00 achieves a mean value of 1.9858×10^5 , compared with 1.0292×10^7 for A00, corresponding to an accuracy improvement of more than 50 times. This result indicates that the adaptive multi-strategy framework, combined with the directed local search strategy, can effectively prevent premature convergence in complex multimodal landscapes.

When the dimensionality is increased to 100, as reported in Table 4, the performance of most comparison algorithms deteriorates sharply. In particular, the mean objective values obtained by VPPSO, DBO, and ADE increase substantially, whereas IA00 continues

to maintain a significant performance advantage. For F1, the optimization difficulty increases markedly with dimensionality. Nevertheless, IA00 achieves a mean value of 2.3899×10^7 , whereas the original A00 obtains 4.3360×10^8 , indicating an approximately 20-fold improvement in solution accuracy. For the high-dimensional composition function F30, IA00 achieves a mean value of 9.7542×10^4 , which is considerably lower than those obtained by all comparison algorithms. These results suggest that the adaptive strategy pool can dynamically adjust the relative contributions of global exploration and local exploitation as the problem dimensionality changes. Consequently, IA00 effectively mitigates the deterioration in optimization performance caused by the curse of dimensionality and exhibits substantially stronger high-dimensional adaptability than conventional metaheuristic algorithms and the original A00.

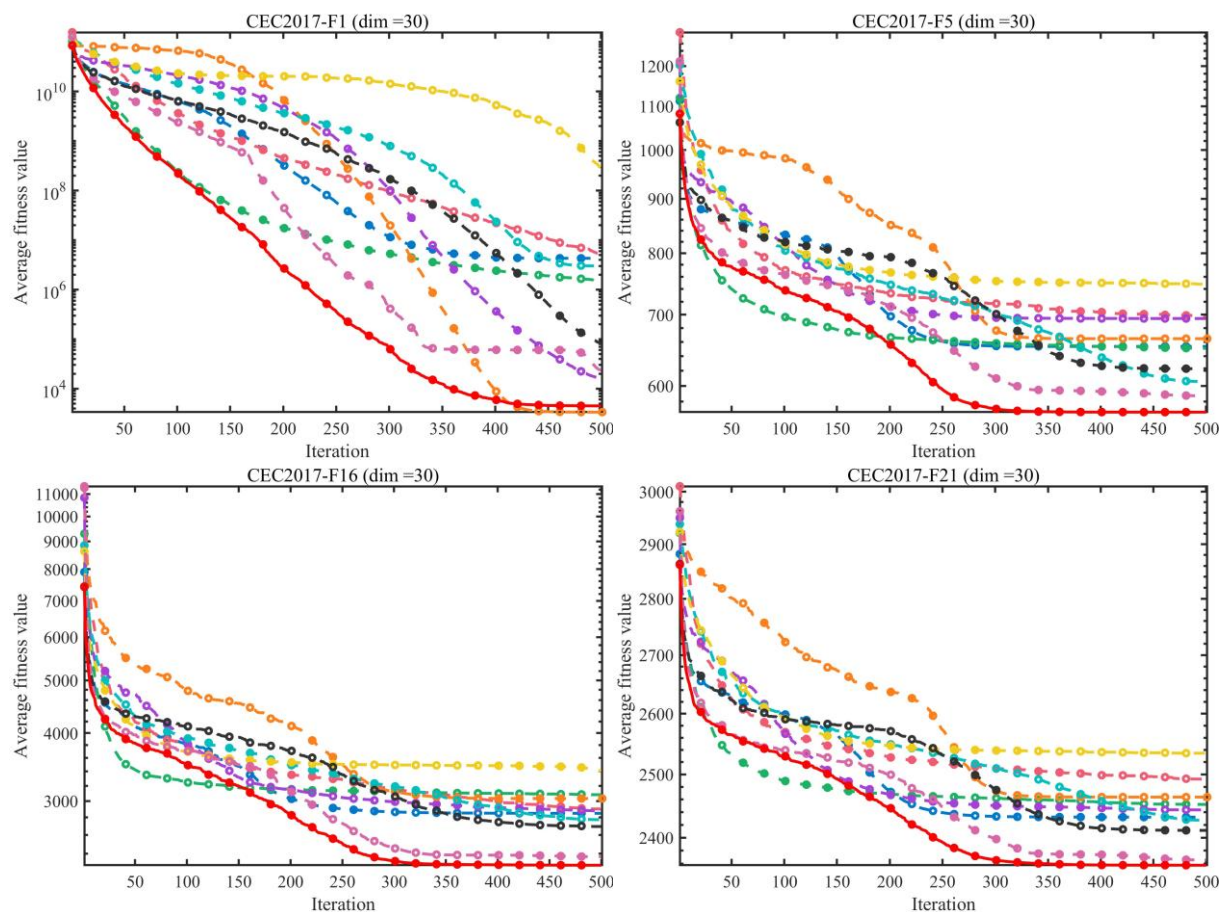
The standard deviation (Std) reflects the dispersion of the results obtained over 30 independent runs. A smaller standard deviation indicates lower run-to-run variability and, therefore, greater reliability in practical applications. Across the 30-dimensional functions, IA00 generally produces smaller standard deviations than the comparison algorithms. For example, its standard deviation on F9 is 1.3864×10^2 , whereas that of the original A00 is 1.5732×10^3 . For the composition function F23, IA00 achieves a standard deviation of only 1.8505×10^1 , outperforming representative algorithms such as SBO and A00. The difference becomes even more pronounced in the 100-dimensional setting. For the high-dimensional multimodal function F19, the standard deviation of IA00 is only 1.6515×10^3 , whereas that of A00 reaches 2.7963×10^6 . The original A00 relies on fixed switching rules, leading to poorly controlled search randomness and considerable variation among independent runs. By contrast, the adaptive strategy selection mechanism of IA00 continuously accumulates the contribution weights of effective search strategies, thereby reducing ineffective random perturbations. Moreover, the interaction between differential evolution and directed local search constrains excessive search fluctuations and substantially narrows the variation range of the final solutions. Consequently, IA00 demonstrates strong robustness in both medium- and high-dimensional optimization scenarios.

Nevertheless, IA00 does not achieve the best result on every function. It is clearly inferior to A00 on F2 and is not the best-performing algorithm on several other functions. These results indicate that the proposed operator combination is not universally advantageous, which is consistent with the No Free Lunch theorem. Therefore, the results support improved overall robustness rather than universal dominance.

Figure 7 presents the convergence curves of representative unimodal, multimodal, hybrid, and composition functions from the CEC2017 benchmark suite under the 30- and 100-dimensional settings. These curves illustrate the dynamic changes in the mean fitness values of the algorithms over 500 iterations. During the early stage, corresponding to iterations 0–200, the IA00 curves decrease considerably faster than those of all comparison algorithms. This behavior can be attributed to the increased selection probability of the differential evolution strategy during the initial iterations. By exploiting differential information among individuals, IA00 rapidly explores a broad region of the solution space and identifies promising areas that potentially contain the global optimum. In contrast, the original A00 relies only on the two fundamental search behaviors of oat-seed rolling and launching. Its global search process therefore exhibits stronger randomness, resulting in a relatively slow reduction in fitness during the early stage and the expenditure of considerable computational resources in unpromising regions. Conventional algorithms such as VPPSO and RUN also tend to aggregate prematurely, leading to lower early-stage convergence efficiency than IA00.

During the middle and late stages, corresponding to iterations 200–500, the comparison algorithms generally exhibit two major limitations. The convergence curves of some algorithms become prematurely flat, indicating that they are trapped in local optima and

cannot generate further improvements. Other algorithms exhibit persistent and substantial fluctuations, suggesting that their populations cannot stably converge toward promising regions. In contrast, the convergence curves of IAEO continue to decrease smoothly and exhibit no pronounced fluctuations after stabilization. This behavior results from the adaptive mechanism progressively increasing the weight of the directed local search strategy during the middle and late iterations, thereby enabling refined exploitation of the identified promising regions. Meanwhile, differential operators are retained with a relatively low probability to preserve a basic level of population diversity, preventing premature convergence caused by complete population homogenization. When the dimensionality increases to 100, the convergence curves of all comparison algorithms shift upward and their rates of descent decrease considerably. Nevertheless, IAEO maintains a steep downward trend and continues to reduce the fitness value even during the late iterations, further confirming the effectiveness of the proposed mechanisms in alleviating the curse of dimensionality. The convergence behavior on representative functions, including F1, F5, F16, F21, and F30, demonstrates that IAEO can achieve rapid exploration followed by stable and refined exploitation on both relatively simple and highly complex fitness landscapes, without exhibiting evident deficiencies in either search stage.



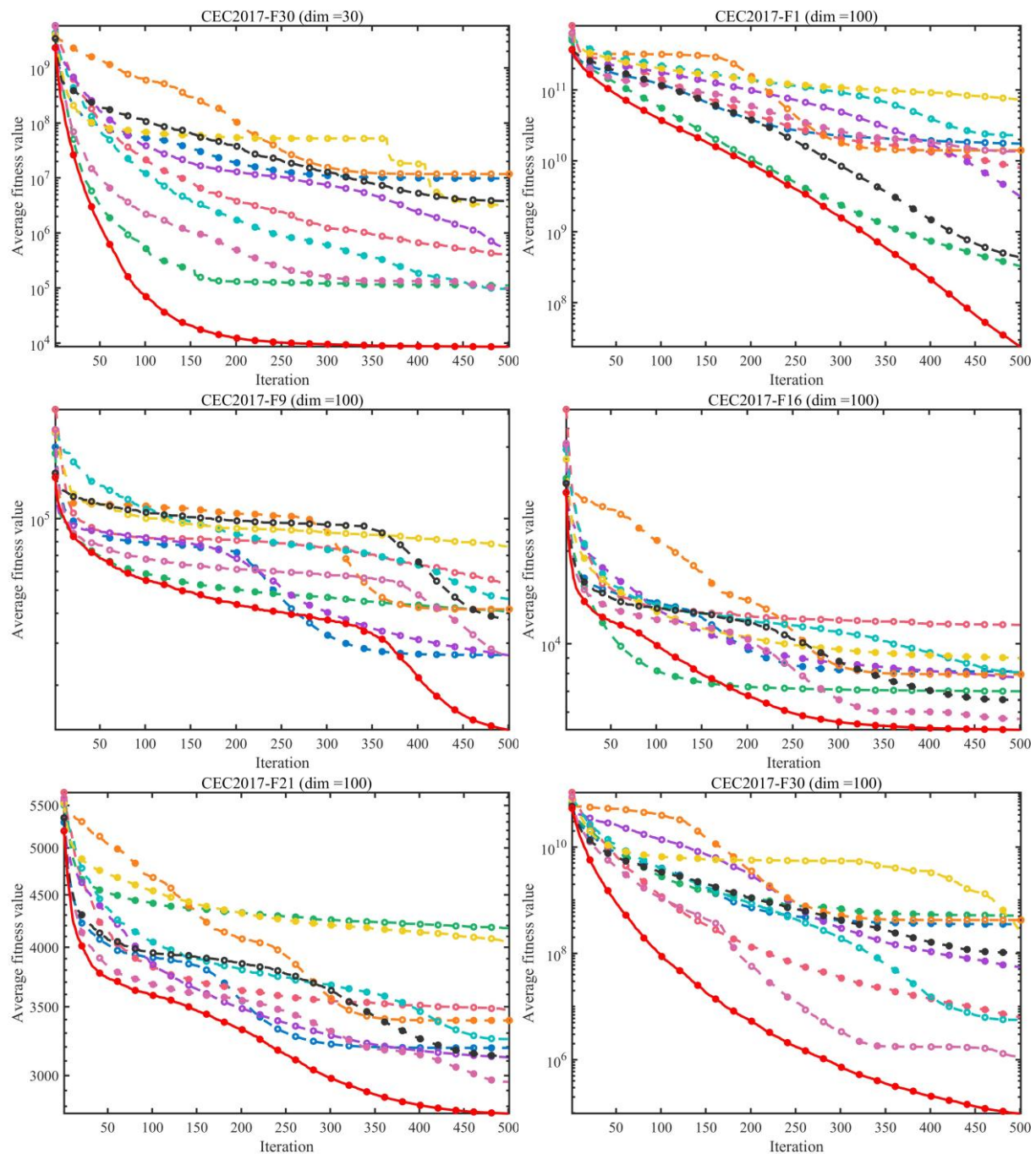


Figure 7. Evaluation of convergence trends of competing algorithms using the CEC2017 test set.

Taken together, the quantitative results reported in Tables 3 and 4 and the convergence curves illustrated in Figure 7 provide comprehensive evidence of the performance of IAOO. Under both the 30- and 100-dimensional CEC2017 test conditions, IAOO achieves faster convergence, higher optimization accuracy, and stronger robustness than the original AOO and several representative classical and recently developed metaheuristic algorithms. The original AOO is constrained by its fixed and relatively limited search framework, which cannot effectively balance global exploration and local exploitation throughout the iterative process. As a result, it commonly suffers from slow convergence, premature stagnation, and performance degradation in high-dimensional problems. By contrast, the adaptive multi-strategy update framework, hybrid differential evolution strategy, and directed local search strategy proposed in IAOO form a coordinated search system. Differential evolution expands the search range and enriches population diversity

during the early iterations; the adaptive probability adjustment mechanism dynamically allocates the application frequencies of different search strategies according to their real-time optimization performance; and directed local search concentrates on promising regions to achieve high-precision exploitation. Through their coordinated interaction, these mechanisms improve the overall search process of the algorithm and enable IAEO to exhibit comprehensive performance advantages in high-dimensional, nonlinear, and multi-modal numerical optimization problems.

3.5.2. Evaluation Results and Discussion for the CEC2020 Test Suite

Table 5 summarizes the average fitness values (Ave) and standard deviations (Std) obtained by IAEO and the nine comparison algorithms on the 20-dimensional CEC2020 benchmark suite. The results provide a comparative evaluation of the algorithms on different types of medium-dimensional optimization problems, including unimodal, multi-modal, hybrid, and composition functions.

From the Ave results, IAEO shows strong competitiveness on the CEC2020 test suite. For F1, IAEO obtains the best mean fitness value of 2.2899×10^2 , which is lower than that of RUN (9.6779×10^2) and much better than the original AEO (7.3463×10^3). Compared with DBO (3.5338×10^7), the result of IAEO is reduced by more than five orders of magnitude, indicating its strong exploitation ability on this function. In addition, IAEO also achieves the best results on several representative functions, such as F3, F4, F8, and F9, showing that the proposed improvement strategies can effectively enhance the search accuracy of AEO.

For F5, IAEO obtains an Ave value of 3.6996×10^4 , which is clearly lower than those of AEO (2.0721×10^5) and ADE (1.8815×10^6). Although ALA achieves a better result on this function, IAEO still shows a significant improvement over the original AEO, suggesting that the introduced hybrid differential evolution strategy helps strengthen the global search ability of the algorithm. Similarly, on F7, IAEO achieves an Ave value of 1.2353×10^4 , representing an 83.6% reduction compared with AEO (7.5373×10^4). This result further demonstrates that IAEO can obtain competitive solutions in complex search spaces.

The Std results further reflect the stability of IAEO on several representative functions. For F1, the Std value of IAEO is 2.4382×10^2 , which is much lower than that of AEO (4.3113×10^3). On F8, IAEO achieves a very small Std value of 1.0464×10^0 , which is also clearly lower than those of AEO (1.2188×10^3) and VPPSO (5.2592×10^3). In addition, for F4, where the average fitness values of different algorithms are relatively close, IAEO still obtains the lowest Std value of 1.1058×10^0 . These results indicate that the proposed multi-strategy adaptive mechanism can reduce search fluctuation and improve the consistency of the optimization results. Overall, IAEO performs well on the CEC2020 benchmark suite, especially in terms of solution accuracy and stability compared with the original AEO.

Table 5. Summary of Optimization Results for the CEC2020 Functions (dim = 20).

Function	Metric	VPPSO	ADE	IAGWO	RUN	SSA	ALA	DBO	SBOA	AOO	IAOO
F1	Ave	2.5171×10^3	7.7313×10^4	2.6666×10^5	9.6779×10^2	3.8901×10^3	5.2711×10^3	3.5338×10^7	4.6362×10^3	7.3463×10^3	2.2899×10^2
	Std	2.9126×10^3	4.1762×10^4	1.2501×10^5	1.0574×10^3	3.2389×10^3	3.3723×10^3	4.2480×10^7	3.7045×10^3	4.3113×10^3	2.4382×10^2
F2	Ave	3.2069×10^3	2.9581×10^3	1.8838×10^3	2.3901×10^3	3.3464×10^3	3.0507×10^3	3.4576×10^3	2.0714×10^3	2.9667×10^3	2.0697×10^3
	Std	4.5661×10^2	2.3820×10^2	2.9572×10^2	4.6423×10^2	5.1690×10^2	4.7395×10^2	5.0989×10^2	3.2239×10^2	4.8258×10^2	4.3231×10^2
F3	Ave	8.2077×10^2	7.8389×10^2	7.7745×10^2	8.5662×10^2	8.2603×10^2	7.8516×10^2	8.4480×10^2	7.7176×10^2	7.8991×10^2	7.5112×10^2
	Std	3.0000×10^1	6.0866×10^0	1.5885×10^1	3.0803×10^1	3.3995×10^1	2.8896×10^1	4.0010×10^1	2.1026×10^1	2.2355×10^1	1.0969×10^1
F4	Ave	1.9057×10^3	1.9084×10^3	1.9105×10^3	1.9182×10^3	1.9070×10^3	1.9051×10^3	1.9341×10^3	1.9061×10^3	1.9057×10^3	1.9036×10^3
	Std	2.2868×10^0	1.2100×10^0	4.5939×10^0	5.8670×10^0	2.6565×10^0	1.7355×10^0	4.7058×10^1	2.1004×10^0	1.5918×10^0	1.1058×10^0
F5	Ave	2.3509×10^5	1.8815×10^6	7.5583×10^5	4.1611×10^4	6.6882×10^5	8.5575×10^3	1.4883×10^6	1.7476×10^5	2.0721×10^5	3.6996×10^4
	Std	1.9309×10^5	9.7898×10^5	5.6062×10^5	2.8335×10^4	7.1622×10^5	5.0957×10^3	1.3529×10^6	1.4110×10^5	1.3392×10^5	5.6599×10^4
F6	Ave	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3	1.9037×10^3
	Std	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2	2.3860×10^2
F7	Ave	1.3788×10^5	5.2297×10^5	3.8611×10^5	3.2359×10^4	3.1642×10^5	4.2952×10^3	3.0485×10^5	9.7511×10^4	7.5373×10^4	1.2353×10^4
	Std	1.0743×10^5	3.5915×10^5	3.2605×10^5	1.9626×10^4	2.6587×10^5	8.0078×10^2	2.8244×10^5	1.0745×10^5	5.0825×10^4	2.7785×10^4

F8	Ave	2.4374×10^3	4.6112×10^3	2.7630×10^3	2.4688×10^3	3.7504×10^3	3.3174×10^3	3.0117×10^3	2.3420×10^3	3.3952×10^3	2.3011×10^3
	Std	5.2592×10^2	1.3241×10^3	1.0520×10^3	6.5122×10^2	1.4693×10^3	1.2202×10^3	1.2147×10^3	2.2427×10^2	1.2188×10^3	1.0464×10^0
F9	Ave	2.8709×10^3	2.9034×10^3	3.0156×10^3	2.8528×10^3	2.8651×10^3	2.8636×10^3	2.9956×10^3	2.8413×10^3	2.8727×10^3	2.8319×10^3
	Std	2.3684×10^1	1.4655×10^1	1.2652×10^2	2.2031×10^1	2.2483×10^1	1.8701×10^1	5.7197×10^1	1.9311×10^1	3.2636×10^1	1.7655×10^1
F10	Ave	2.9732×10^3	2.9170×10^3	2.9917×10^3	2.9882×10^3	2.9504×10^3	2.9463×10^3	3.0018×10^3	2.9519×10^3	2.9399×10^3	2.9615×10^3
	Std	3.3111×10^1	1.3361×10^1	2.2376×10^1	2.7714×10^1	3.2668×10^1	3.2602×10^1	6.2887×10^1	3.4521×10^1	3.2207×10^1	3.3167×10^1

Bold values indicate the best-performing result in that row or column.

The convergence curves in Figure 8 further illustrate the convergence behavior of IA00 on the CEC2020 benchmark suite. For F1, IA00 shows a sharp decrease in fitness during the early iterations and reaches a relatively stable state after about 100 iterations. Compared with algorithms such as DBO and IAGWO, whose curves remain at higher fitness levels for a longer period, IA00 exhibits faster convergence and better final solution quality. This result is consistent with the statistical results in Table 3, where IA00 obtains the best mean fitness value on F1.

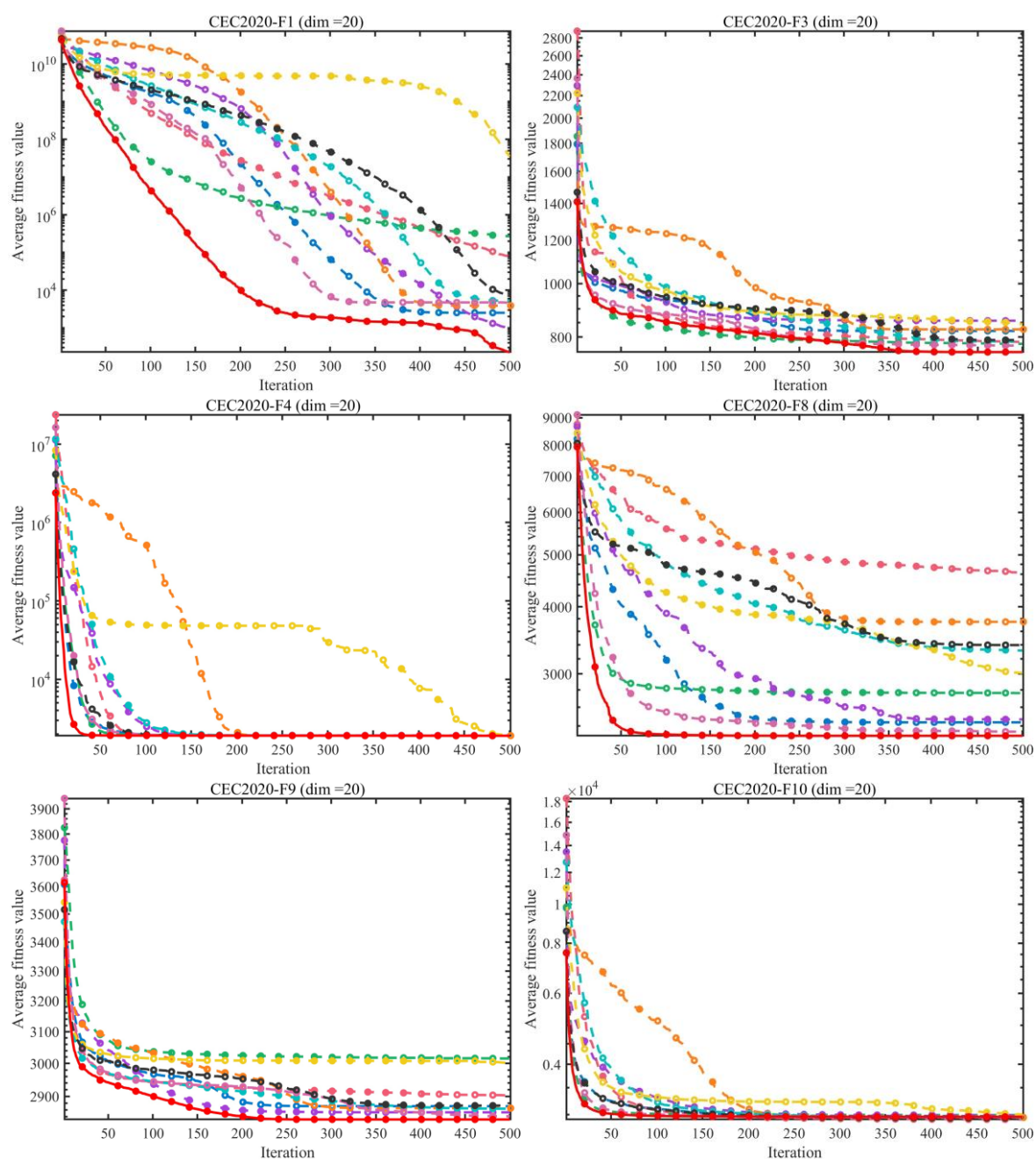


Figure 8. Evaluation of convergence trends of competing algorithms using the CEC2020 test set.

For F3, IAOO generally maintains a lower convergence curve than most comparison algorithms throughout the optimization process, and no obvious late-stage oscillation is observed. This indicates that the proposed directed local search strategy can effectively refine promising solutions and improve the exploitation accuracy of the algorithm. Meanwhile, the stable convergence trend also suggests that IAOO can reduce unnecessary fluctuations during the later search stage.

For the composition function F8, IAOO approaches a stable fitness value after approximately 200 iterations, while several algorithms, such as AOO and SSA, still show noticeable fluctuations during the search process. This phenomenon suggests that the hybrid differential evolution strategy helps maintain population diversity and provides useful search directions, thereby improving both convergence efficiency and search stability in complex landscapes.

By combining the statistical results in Table 5 with the convergence trends in Figure 7, it can be observed that IAOO achieves a good balance among optimization accuracy, convergence speed, and stability on the CEC2020 test suite. Compared with the original AOO, IAOO shows more stable convergence behavior and better solution quality on several representative functions. These improvements further demonstrate the effectiveness of integrating the multi-strategy adaptive update mechanism, hybrid differential evolution strategy, and directed local search strategy.

3.5.3. Evaluation Results and Discussion for the CEC2022 Test Suite

Table 6 presents the average fitness values (Ave) and standard deviations (Std) obtained by IAOO and nine comparison algorithms on the 20-dimensional CEC2022 benchmark suite. This benchmark contains different types of test functions, including unimodal, multimodal, hybrid, and composition functions, and provides a challenging environment for evaluating the optimization accuracy, stability, and robustness of different algorithms.

From the Ave results, IAOO shows strong competitiveness on several representative functions. For F3, IAOO obtains a mean fitness value of 6.0065×10^2 , which is very close to the best result achieved by ADE (6.0007×10^2) and lower than those of RUN (6.3545×10^2), SSA (6.3880×10^2), AOO (6.1751×10^2), and most other comparison algorithms. This indicates that IAOO can maintain good exploitation accuracy on this function. For F5, IAOO achieves the best mean fitness value of 9.3490×10^2 among all algorithms. Compared with the original AOO (1.4658×10^3) and the second-best SBOA (1.0844×10^3), IAOO reduces the Ave value by 36.2% and 13.8%, respectively, suggesting that the proposed strategies are effective in improving global exploration and avoiding local optima. In addition, for F10, IAOO obtains the lowest mean fitness value of 2.5904×10^3 , which is 26.9% lower than AOO (3.5446×10^3) and 8.5% lower than ADE (2.8306×10^3). These results show that IAOO has good adaptability when dealing with complex composition functions.

The Std results further reflect the stability of IAOO on several test functions. For example, the Std value of IAOO on F9 is only 1.0778×10^{-4} , which is much lower than that of AOO (2.7597×10^0), indicating that IAOO produces highly consistent results on this function. On F11, IAOO obtains a Std value of 3.8066×10^{-1} , which is also significantly smaller than that of AOO (2.8584×10^2). Similarly, for F12, the Std value of IAOO is 3.0572×10^0 , accounting for only 18.4% of AOO's Std value (1.6597×10^1). These results suggest that the multi-strategy adaptive mechanism can effectively reduce search fluctuation and improve the repeatability of the optimization results. It should also be noted that IAOO does not achieve the best Ave value on every function, such as F6. Nevertheless, it still shows competitive overall performance on the CEC2022 suite, indicating a favorable balance between solution accuracy and search stability.

Table 6. Summary of Optimization Results for the CEC2022 Functions (dim = 20).

Function	Metric	VPPSO	ADE	IAGWO	RUN	SSA	ALA	DBO	SBOA	AOO	IAOO
F1	Ave	6.3789×10^3	4.2403×10^4	1.1867×10^4	3.0397×10^2	1.1773×10^4	3.4083×10^3	3.4690×10^4	4.1726×10^3	5.6090×10^2	3.8495×10^2
	Std	3.3262×10^3	8.8750×10^3	4.2817×10^3	4.6983×10^0	7.4066×10^3	1.7967×10^3	1.1006×10^4	2.6632×10^3	2.7137×10^2	2.2480×10^2
F2	Ave	4.7691×10^2	4.4920×10^2	5.1353×10^2	4.5714×10^2	4.6367×10^2	4.5413×10^2	5.0579×10^2	4.5920×10^2	4.5574×10^2	4.5162×10^2
	Std	3.3301×10^1	1.3967×10^{-1}	3.9757×10^1	1.2015×10^1	1.9177×10^1	1.5823×10^1	6.0863×10^1	2.5879×10^1	1.6111×10^1	1.8325×10^1
F3	Ave	6.2493×10^2	6.0007×10^2	6.1089×10^2	6.3545×10^2	6.3880×10^2	6.0546×10^2	6.3675×10^2	6.0151×10^2	6.1751×10^2	6.0065×10^2
	Std	9.8182×10^0	2.8389×10^{-2}	5.7501×10^0	1.0382×10^1	1.6580×10^1	3.2263×10^0	1.1915×10^1	1.5326×10^0	8.8202×10^0	1.1157×10^0
F4	Ave	8.5483×10^2	9.1676×10^2	8.7466×10^2	8.8055×10^2	8.8018×10^2	8.5487×10^2	9.1997×10^2	8.4478×10^2	8.6170×10^2	8.3108×10^2
	Std	1.3094×10^1	1.2263×10^1	1.7058×10^1	1.1510×10^1	2.4138×10^1	2.0693×10^1	2.4920×10^1	1.5285×10^1	1.8520×10^1	1.2625×10^1
F5	Ave	1.6448×10^3	1.1951×10^3	2.0061×10^3	1.8877×10^3	2.2837×10^3	1.0994×10^3	2.1390×10^3	1.0844×10^3	1.4658×10^3	9.3490×10^2
	Std	3.8171×10^2	1.5704×10^2	4.3915×10^2	2.5149×10^2	6.6492×10^2	1.4033×10^2	6.3112×10^2	2.0958×10^2	4.7530×10^2	2.3758×10^1
F6	Ave	3.9027×10^3	4.6764×10^6	9.6814×10^3	3.5876×10^3	7.3129×10^3	1.5099×10^4	8.0358×10^5	6.4584×10^3	7.9416×10^3	4.7420×10^3
	Std	2.4578×10^3	3.8463×10^6	2.2533×10^4	7.4832×10^2	6.3527×10^3	8.2208×10^3	2.7327×10^6	5.1591×10^3	5.9229×10^3	3.4463×10^3
F7	Ave	2.1053×10^3	2.0663×10^3	2.0773×10^3	2.1017×10^3	2.1315×10^3	2.0879×10^3	2.1419×10^3	2.0470×10^3	2.1035×10^3	2.0474×10^3
	Std	5.0150×10^1	1.3694×10^1	4.2153×10^1	2.7093×10^1	5.1449×10^1	4.3287×10^1	4.5644×10^1	1.6438×10^1	3.5144×10^1	2.4346×10^1
F8	Ave	2.2803×10^3	2.2313×10^3	2.2610×10^3	2.2328×10^3	2.2912×10^3	2.2335×10^3	2.3087×10^3	2.2290×10^3	2.2522×10^3	2.2351×10^3
	Std	6.4772×10^1	2.0092×10^0	5.6023×10^1	2.1998×10^1	6.1767×10^1	4.4565×10^0	8.0671×10^1	5.7232×10^0	4.5432×10^1	3.5998×10^1
F9	Ave	2.5167×10^3	2.4808×10^3	2.5085×10^3	2.4813×10^3	2.5191×10^3	2.4808×10^3	2.5172×10^3	2.4808×10^3	2.4830×10^3	2.4808×10^3
	Std	4.0832×10^1	2.4184×10^{-2}	3.3303×10^1	3.6586×10^{-1}	4.4706×10^1	3.5669×10^{-2}	2.8967×10^1	9.1850×10^{-2}	2.7597×10^0	1.0778×10^{-4}
F10	Ave	3.1644×10^3	2.8306×10^3	2.7540×10^3	2.7004×10^3	3.9558×10^3	3.5620×10^3	3.2443×10^3	2.8351×10^3	3.5446×10^3	2.5904×10^3
	Std	9.4389×10^2	4.4451×10^2	1.9853×10^2	3.3057×10^2	1.1872×10^3	9.4565×10^2	1.0590×10^3	4.4472×10^2	7.6940×10^2	1.2487×10^2
F11	Ave	2.9436×10^3	2.9033×10^3	2.9216×10^3	2.9173×10^3	2.9422×10^3	2.9787×10^3	3.3181×10^3	2.9178×10^3	2.9554×10^3	2.9006×10^3
	Std	1.0502×10^2	6.6867×10^1	1.1322×10^2	9.8224×10^1	1.6551×10^2	2.2568×10^2	9.1710×10^2	9.9245×10^1	2.8584×10^2	3.8066×10^{-1}
F12	Ave	2.9806×10^3	2.9702×10^3	2.9192×10^3	2.9703×10^3	2.9858×10^3	2.9687×10^3	3.0594×10^3	2.9496×10^3	2.9706×10^3	2.9464×10^3
	Std	2.9895×10^1	3.0325×10^1	5.0016×10^1	1.8815×10^1	3.7270×10^1	2.5244×10^1	7.2259×10^1	1.5870×10^1	1.6597×10^1	3.0572×10^0

Bold values indicate the best-performing result in that row or column.

The convergence curves in Figure 9 further illustrate the convergence behavior of IAOO on the CEC2022 benchmark suite. For F1, IAOO shows a rapid decrease in fitness during the early search stage and gradually reaches a stable state after about 200 iterations. Compared with algorithms such as DBO, VPPSO, and the original AOO, IAOO obtains a lower final fitness value and exhibits a more stable convergence trend. This result indicates that the proposed improvement strategies can effectively enhance the convergence efficiency of AOO on relatively simple optimization landscapes.

For F5, which is a multimodal function, IAOO maintains a lower convergence curve than most comparison algorithms during the optimization process and does not show obvious late-stage oscillations. This suggests that IAOO can locate promising regions more effectively and conduct stable local refinement after the exploration stage. For the composition function F10, IAOO reaches a stable fitness level after approximately 150 iterations, while some algorithms, such as SSA and RUN, still show noticeable fluctuations. This behavior indicates that the hybrid differential evolution strategy helps preserve population diversity and provides useful search directions, while the directed local search strategy further improves solution refinement near high-quality regions.

By combining the statistical results in Table 6 with the convergence trends in Figure 8, it can be observed that IAOO achieves competitive performance in terms of optimization accuracy, convergence speed, and search stability on the CEC2022 benchmark suite. Compared with the original AOO, IAOO shows better convergence behavior and stronger adaptability on several representative functions, which further verifies the effectiveness of the proposed multi-strategy adaptive update framework.

Furthermore, the experimental results on CEC2017, CEC2020, and CEC2022 demonstrate the overall effectiveness and robustness of the proposed IAOO algorithm. By integrating the multi-strategy adaptive update mechanism, hybrid differential evolution strategy, and directed local search strategy, IAOO can better coordinate global exploration and local exploitation across different types of benchmark functions, including unimodal, multimodal, hybrid, and composition functions.

In terms of optimization accuracy, IA00 achieves notable improvements over the original A00 on several representative functions, such as F9, F12, and F30 in CEC2017. On the CEC2020 benchmark suite, IA00 obtains the best or highly competitive results on functions such as F1, F5, and F7. On the CEC222 benchmark suite, IA00 also shows strong performance on functions such as F5 and F10. These results indicate that IA00 has good adaptability to optimization problems with different dimensions and landscape characteristics.

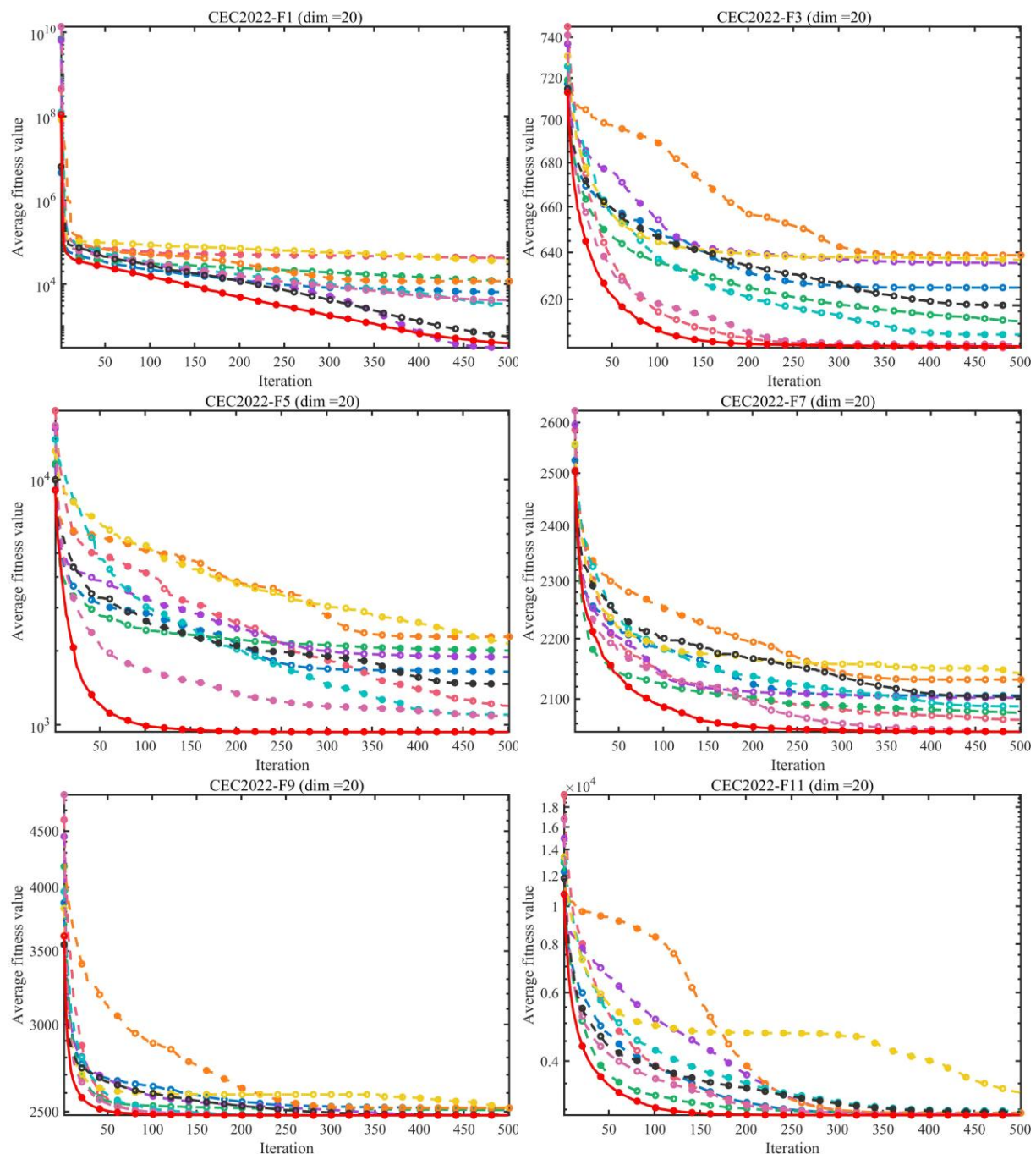


Figure 9. Evaluation of convergence trends of competing algorithms using the CEC2022 test set.

With respect to stability, IA00 obtains relatively small standard deviations on many representative functions, suggesting that the proposed adaptive strategy regulation mechanism can reduce search fluctuation and improve the consistency of optimization results. In terms of convergence speed, the convergence curves show that IA00 usually descends

rapidly in the early iterations and remains relatively stable in the later stage. Therefore, IAOO can approach promising regions more efficiently than the original AOO and several comparison algorithms, while maintaining a favorable balance between convergence accuracy and search stability.

3.6. Stability Analysis

Figure 10 presents box plots for eight representative 30-dimensional functions from the CEC2017 benchmark suite, namely F1, F5, F7, F9, F16, F21, F25, and F30. The plots illustrate the fitness-value distributions, medians, interquartile ranges, whiskers, and outliers obtained by VPPSO, ADE, IAGWO, RUN, SSA, ALA, DBO, SBOA, AOO, and IAOO over 30 independent runs. These statistical characteristics provide an intuitive representation of the dispersion and robustness of the results produced by each algorithm. In general, a narrower box, shorter whiskers, and fewer outliers indicate lower run-to-run variability and stronger algorithmic stability.

As can be observed from all subplots, IAOO exhibits the narrowest overall boxes across all test functions, while its median lines are located at the lowest positions among the compared algorithms. This indicates that IAOO not only obtains higher-quality best solutions but also maintains significantly better median performance over repeated runs. Taking the unimodal function F1 as an example, DBO exhibits an extremely wide box accompanied by numerous high-value outliers, while AOO and VPPSO also present relatively broad interquartile ranges. These characteristics reveal considerable differences among the optimization results obtained in individual runs. By contrast, the box corresponding to IAOO is extremely narrow, and almost no exceptionally poor solutions are observed, demonstrating excellent consistency across independent experiments.

For highly complex functions such as F9 and F30, most comparison algorithms exhibit long whiskers and multiple red outlier markers. This suggests that these algorithms frequently become trapped in inferior local optima during some runs, consequently producing poor-quality solutions. The original AOO also exhibits considerable numerical dispersion, with substantial differences between its best and worst results across repeated experiments. In contrast, IAOO presents not only compact interquartile ranges but also tightly controlled differences between the upper and lower extremes, together with very few outliers. Therefore, even when dealing with complex search spaces characterized by numerous local optima and strong nonlinearity, IAOO consistently produces high-quality solutions and rarely suffers from search failure.

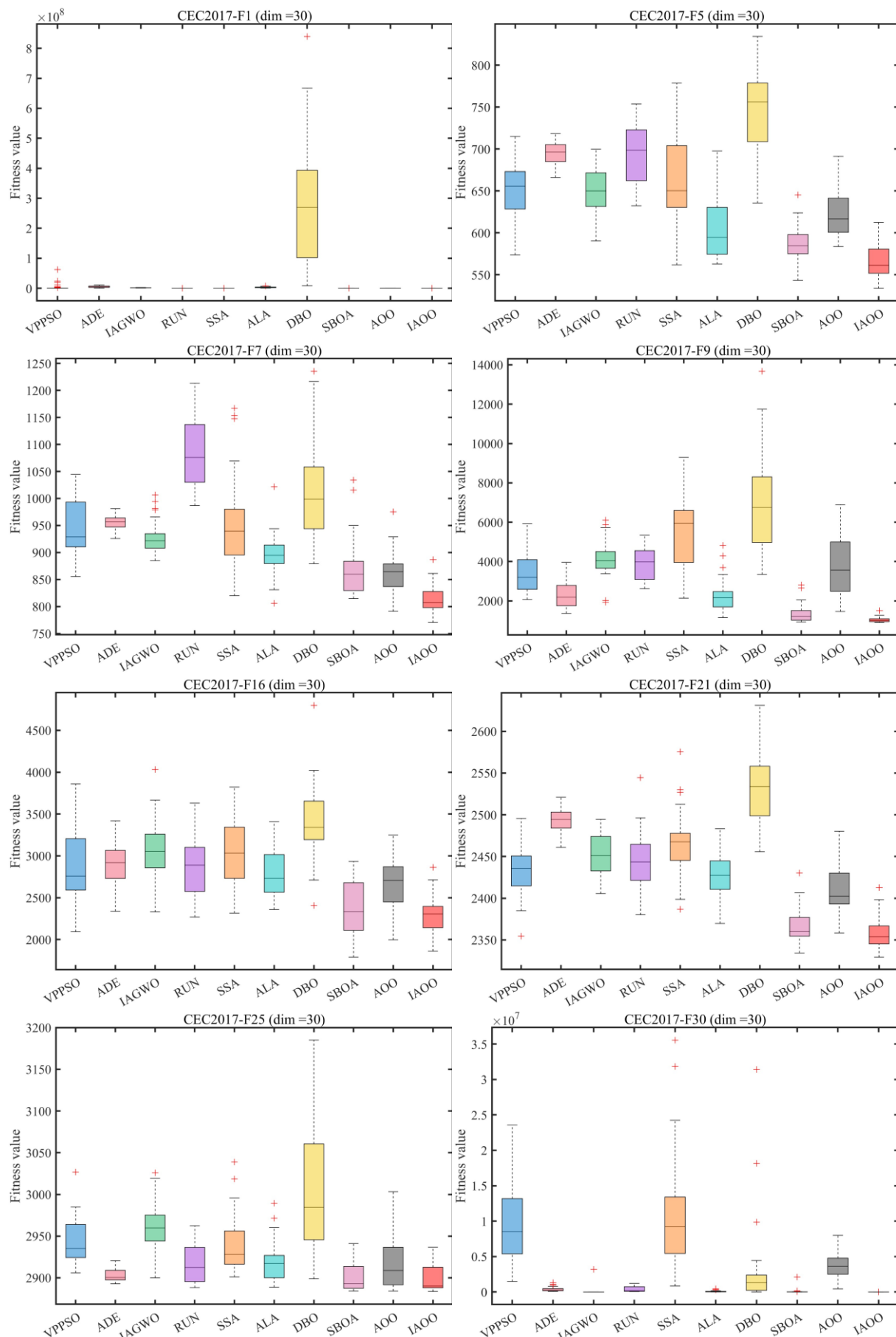


Figure 10. Box-plot comparison of the fitness values obtained by different algorithms on representative CEC2017 benchmark functions (dim = 30). + represents outliers.

For multimodal functions of moderate complexity, such as F5, F7, and F16, SBOA and IAGWO achieve relatively competitive overall performance. Nevertheless, their box widths remain larger than those of IA00, and their median values are also higher. These results demonstrate the effectiveness of the adaptive multi-strategy mechanism in constraining search randomness. The original A00 relies solely on fixed switching between rolling and launching search modes and lacks a dynamic regulation mechanism. Consequently, its search directions and ranges fluctuate considerably during the iterative process, leading to widely dispersed fitness distributions over repeated runs. By contrast, IA00 employs a strategy-probability update rule based on fitness improvements, continuously assigning higher selection probabilities to differential evolution and local search operators that produce greater optimization gains. This mechanism substantially reduces ineffective random perturbations and narrows the distribution range of the results obtained across independent runs.

Overall, the box-plot characteristics demonstrate that IA00 achieves simultaneous advantages in solution quality and result stability across CEC2017 functions with different landscape characteristics and levels of complexity. The proposed multi-strategy collaborative architecture effectively alleviates the large run-to-run fluctuations and poor outlier solutions commonly observed in conventional metaheuristic algorithms. These results further confirm that the proposed improvement strategies significantly enhance the robustness of IA00 and enable it to reliably solve complex numerical optimization problems.

3.7. Statistical Analysis

3.7.1. Wilcoxon Rank Sum Test

In this subsection, the Wilcoxon rank-sum test [46] is used to examine whether the performance differences between IA00 and the comparison algorithms are statistically significant. As a nonparametric statistical method, the Wilcoxon rank-sum test does not require the data to follow a specific distribution. Compared with the traditional t-test, it is more suitable for optimization results that may contain non-normal distributions, unequal variances, or abnormal values [47,48].

The significance level was set to 0.05 in this study. The null hypothesis (H_0) assumes that there is no significant difference between IA00 and the compared algorithm. If the obtained (p)-value is less than 0.05, (H_0) is rejected, indicating that the performance difference is statistically significant. Otherwise, (H_0) is accepted, suggesting that no statistically significant difference can be observed between the two algorithms.

Table 7 reports the Wilcoxon rank-sum test results between IA00 and nine comparison algorithms on the CEC2017, CEC2020, and CEC2022 benchmark suites. The results are expressed in the form of “+/-”, where “+” indicates that IA00 performs significantly better than the corresponding algorithm, “=” means that there is no significant difference, and “-” indicates that IA00 performs significantly worse.

Table 7. Results for various algorithms.

Algorithm	CEC2017 dim = 30 (+/-)	CEC2017 dim = 100 (+/-)	CEC2020 dim = 20 (+/-)	CEC2022 dim = 20 (+/-)
VPPSO	(29/0/1)	(30/0/0)	(7/0/3)	(10/0/2)
ADE	(29/0/1)	(29/0/1)	(10/0/0)	(10/0/2)
IAGWO	(26/0/4)	(28/0/2)	(7/0/3)	(10/0/2)
RUN	(29/0/1)	(28/0/2)	(8/0/2)	(9/0/3)
SSA	(29/0/1)	(30/0/0)	(8/0/2)	(11/0/1)
ALA	(27/0/3)	(28/0/2)	(8/0/2)	(10/0/2)
DBO	(30/0/0)	(30/0/0)	(9/0/1)	(11/0/1)
SBOA	(16/0/14)	(26/0/4)	(7/0/3)	(8/0/4)
A00	(28/0/2)	(28/0/2)	(8/0/2)	(11/0/1)

The results obtained on the two CEC2017 settings are first analyzed. Under the 30-dimensional setting, IAOO significantly outperforms DBO on all 30 test functions, with no function exhibiting inferior performance. Compared with VPPSO, ADE, RUN, and SSA, IAOO performs slightly worse on only one function in each comparison while demonstrating statistically significant superiority on the remaining 29 functions. When compared with IAGWO and ALA, IAOO achieves significant advantages on 26 and 27 functions, respectively, and is inferior only on a small number of functions with relatively simple landscapes. In comparison with SBOA, 14 “–” outcomes are observed, indicating that SBOA remains competitive on some low-complexity unimodal problems. Nevertheless, IAOO retains clear performance advantages on the remaining complex multimodal, hybrid, and composition functions. Compared with the original AOO, IAOO achieves statistically significant improvements on 28 functions, while only a few test cases fail to exhibit performance gains. This result directly demonstrates that the adaptive multi-strategy framework effectively alleviates the exploration–exploitation imbalance and premature convergence problems of the original AOO.

When the dimensionality is increased to 100, the curse of dimensionality substantially increases the optimization difficulty and leads to considerable performance degradation in most comparison algorithms. Consequently, the statistical superiority of IAOO becomes even more pronounced. IAOO significantly outperforms VPPSO, SSA, and DBO on all 30 functions, with no inferior cases. Compared with ADE, IAGWO, RUN, ALA, SBOA, and AOO, IAOO performs worse on only two functions in each comparison and demonstrates statistically significant advantages on the vast majority of high-dimensional complex functions. These findings indicate that the collaborative framework combining hybrid differential evolution with directed local search can effectively alleviate population homogenization in high-dimensional search spaces and is well suited to large-scale optimization tasks.

For the 20-dimensional CEC2020 benchmark, which comprises 10 test functions with different landscape characteristics, IAOO significantly outperforms ADE on all 10 functions. Compared with DBO, nine functions are marked with “+”, while IAOO is inferior on only one function. Against VPPSO, RUN, SSA, ALA, SBOA, and AOO, IAOO achieves significant superiority on seven or eight functions. Although some comparison algorithms perform slightly better on a limited number of simple unimodal functions, IAOO still exhibits a pronounced overall advantage.

On the 20-dimensional CEC2022 benchmark, IAOO significantly outperforms SSA, DBO, and the original AOO on 11 functions, with “–” outcomes occurring only on a small number of relatively simple functions. Compared with VPPSO, ADE, IAGWO, RUN, ALA, and SBOA, IAOO achieves significant advantages on 8–10 functions and fails to surpass the competing algorithms only on a few low-complexity cases. The limited number of “–” outcomes is consistent with the No Free Lunch theorem, according to which no metaheuristic algorithm can remain optimal across all optimization scenarios. Across all benchmark settings, the cases in which IAOO is inferior to other algorithms are mainly concentrated on structurally simple unimodal functions without local optima. In contrast, IAOO exhibits almost no statistical disadvantage on multimodal, hybrid, and composition functions containing numerous local optima and more closely resembling practical engineering problems.

A cross-comparison of the four benchmark settings reveals a consistent statistical pattern. Whether applied to low- and medium-dimensional small-scale problems or high-dimensional complex optimization scenarios with 100 variables, the number of functions on which IAOO significantly outperforms the comparison algorithms consistently constitutes the clear majority. Moreover, no “=” outcomes are reported in Table 6, indicating that the performance differences between IAOO and the competing algorithms are

generally distinguishable through statistical testing. The CEC2017 results obtained at both dimensionalities further demonstrate that the improvements produced by the adaptive multi-strategy update mechanism, hybrid differential evolution, and directed local search are not caused by random experimental fluctuations but represent statistically significant and consistent performance gains over the original AOO.

In summary, the Wilcoxon rank-sum test results confirm that, after accounting for the stochastic variability of repeated experiments, the proposed IAEO achieves reliable and statistically significant overall performance advantages across benchmark functions with different dimensionalities and levels of complexity. These statistical findings provide strong support for the effectiveness of the proposed improvement strategies.

3.7.2. Friedman Mean Rank Test

Within the present segment, a rank-based comparative procedure is utilized to derive a global ordering of the proposed approach relative to alternative techniques. This distribution-free strategy examines central tendency contrasts among multiple dependent samples. Such a framework proves well suited for scenarios involving iterative trials or structured experimental blocks, providing a dependable replacement for variance-based analyses whenever distributional prerequisites fail. The corresponding test metric adopts the formulation presented in Formula (21) [47,49,50].

$$Q = \frac{12}{nk(k+1)} \sum_{j=1}^k R_j^2 - 3n(k+1), \quad (21)$$

Here, the symbol n denotes the total count of experimental units, k represents the quantity of compared categories, and R_j corresponds to the aggregated rank associated with the j -th category. Under conditions where both n and k take sufficiently large values, the statistic Q can be reasonably approximated by a chi-square distribution with $k - 1$ degrees of freedom.

Table 8 reports the Friedman mean ranks and overall rankings of the ten algorithms under four experimental settings: CEC2017 with 30 and 100 dimensions, CEC2020 with 20 dimensions, and CEC2022 with 20 dimensions. Figure 11 further visualizes the rank distributions of the algorithms across all benchmark functions using polar radar charts. As a nonparametric statistical method for comparing multiple related samples, the Friedman test provides a unified ranking of all algorithms by integrating their optimization results across the entire benchmark suite. A lower mean rank indicates stronger overall performance. Therefore, this test reduces the potential bias arising from evaluation on individual functions and quantitatively assesses the overall performance differences between IAEO and the competing algorithms. It also complements the pairwise comparisons conducted using the Wilcoxon rank-sum test.

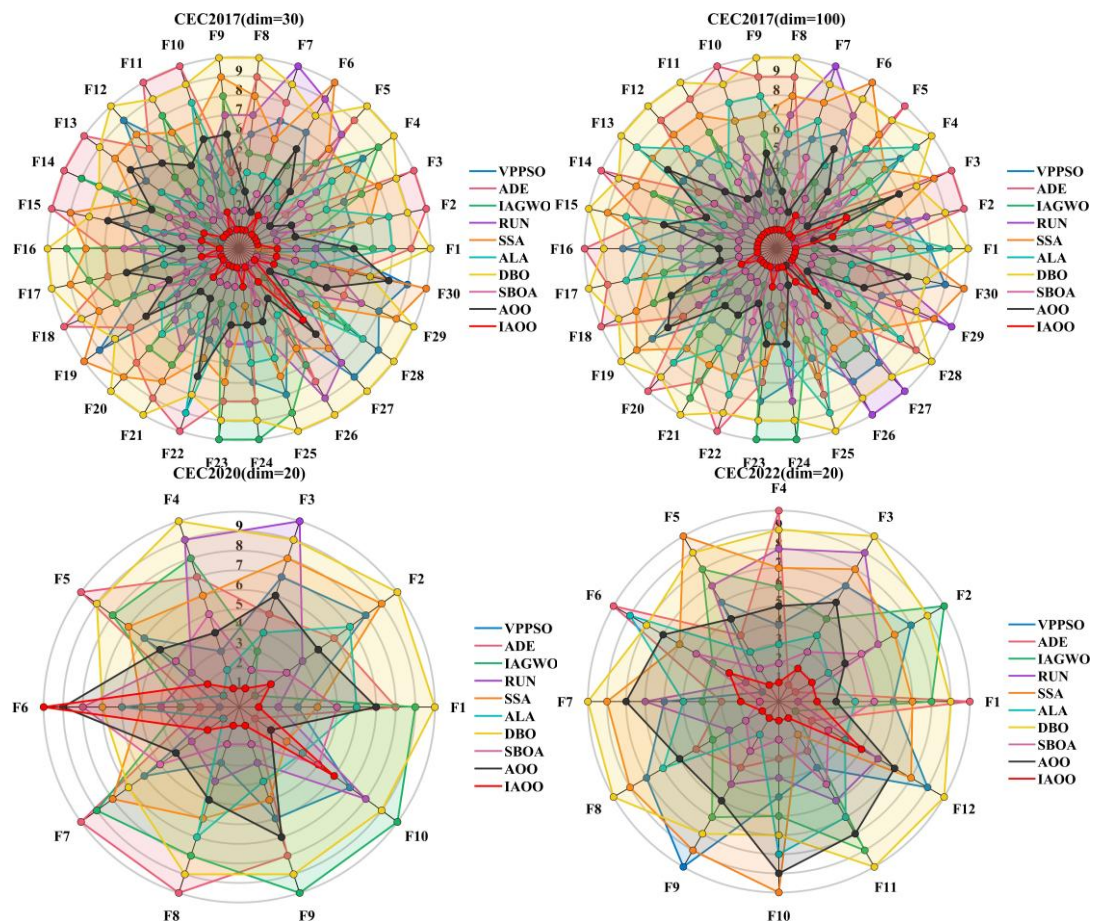


Figure 11. Distribution of rankings of different algorithms.

Table 8. Summary of algorithm rankings using the Friedman method.

Suites	CEC2017				CEC2020		CEC2022	
Dimensions	30	100			20		20	
Algorithms	M. R	M. R	M. R	T. R	M. R	T. R	M. R	T. R
VPPSO	6.43	7	6.20	7	5.00	4	6.17	7
ADE	7.50	9	7.37	8	6.70	8	4.67	3
IAGWO	5.90	6	5.00	4	6.90	9	6.00	6
RUN	5.27	5	5.10	5	5.00	4	5.33	5
SSA	7.33	8	7.47	9	6.10	7	7.67	9
ALA	4.77	4	6.13	6	4.20	3	4.92	4
DBO	9.27	10	9.33	10	8.90	10	9.08	10
SBOA	2.40	2	2.97	2	4.10	2	3.17	2
AOO	4.63	3	4.07	3	5.40	6	6.17	7
IAOO	1.50	1	1.37	1	2.70	1	1.83	1

As shown by the mean-rank results in Table 8, IA OO achieves mean ranks of 1.50, 1.37, 2.70, and 1.83 on the 30-dimensional CEC2017, 100-dimensional CEC2017, 20-dimensional CEC2020, and 20-dimensional CEC2022 benchmarks, respectively. These values are the lowest among all ten algorithms, placing IA OO first in the overall rankings under all four experimental settings. This finding demonstrates that IA OO provides the strongest comprehensive optimization performance across unimodal, multimodal, hybrid, and composition functions.

Under the two CEC2017 dimensional settings, SBOA obtains mean ranks of 2.40 and 2.97, respectively, and consistently ranks second. Nevertheless, a clear rank difference remains between SBOA and IA OO. The original AOO achieves a mean rank of 4.63 under the 30-dimensional setting and ranks third. When the dimensionality increases to 100, its

mean rank changes to 4.07, but its overall performance still remains clearly inferior to that of IA00. By contrast, the mean rank of IA00 remains consistently low and even improves slightly from 1.50 to 1.37, indicating that it preserves its overall advantage in high-dimensional environments.

The mean ranks of DBO range from 8.90 to 9.37 across the benchmark suites, placing it near the bottom of the overall rankings. This result suggests that DBO has relatively limited adaptability to high-dimensional search spaces and landscapes containing numerous local optima. Algorithms such as SSA and RUN also generally obtain relatively high mean ranks, indicating weaker overall optimization performance than IA00, SBOA, and IAGWO. On the lower-dimensional CEC2020 and CEC2022 benchmarks, IA00 continues to achieve the lowest mean ranks. The corresponding mean ranks of the second-ranked SBOA are 4.10 and 3.17, respectively, revealing a clear performance gap between the two algorithms. Even on the more recent and challenging CEC2022 benchmark functions, IA00 maintains the best overall ranking, demonstrating the strong generalization capability of the proposed framework.

The rank comparison between the original A00 and IA00 further demonstrates that integrating adaptive multi-strategy selection, hybrid differential evolution, and directed local search substantially improves the original algorithm. Specifically, these mechanisms enhance diversity regulation and alleviate the imbalance between global exploration and local exploitation, resulting in a marked improvement in the overall performance of IA00 across different fitness landscapes.

The four polar radar charts in Figure 11 treat each benchmark function as a radial axis, with the distance from the center representing the rank obtained by an algorithm on the corresponding function. A line closer to the center indicates a better rank and, therefore, stronger optimization performance. Across the four radar charts, the red line representing IA00 remains close to the central region on nearly all radial axes. Only slight outward deviations are observed on a few relatively simple unimodal functions, whereas IA00 occupies the best-ranking region on the majority of multimodal, hybrid, and composition functions.

In contrast, the lines representing the other algorithms are generally located closer to the outer regions of the radar charts. DBO, SSA, and ADE exhibit considerably poorer ranks on numerous complex functions, as indicated by their pronounced outward extensions. The black line representing the original A00 remains relatively close to the center on several functions under the 30-dimensional CEC2017 setting. However, it shifts outward over a larger number of axes under the 100-dimensional CEC2017, CEC2020, and CEC2022 settings, indicating weaker adaptability as the dimensionality and landscape complexity increase.

As the second-ranked algorithm, SBOA exhibits a purple line that is generally closer to the center than those of most comparison algorithms. Nevertheless, a noticeable gap remains between SBOA and IA00 on many highly complex composition functions. IAGWO and ALA approach the best-ranking region on only a limited number of relatively simple functions, while their rankings deteriorate under more complex conditions. The radar-chart distributions therefore visually support the ranking results reported in Table 7. IA00 demonstrates pronounced advantages on most challenging hybrid and composition functions that more closely resemble complex engineering optimization problems, while exhibiting only slight performance degradation on a few structurally simple unimodal functions. This observation is consistent with the No Free Lunch theorem, which states that no metaheuristic algorithm can perform optimally across all optimization scenarios. It also suggests that the proposed strategies are particularly effective for high-dimensional and multimodal optimization problems relevant to practical engineering applications.

Combining the quantitative ranking results in Table 7 with the visual rank distributions in Figure 11 provides a comprehensive chain of evidence. The Friedman rankings and radar-chart distributions consistently demonstrate, from a global statistical perspective covering multiple functions and dimensionalities, that IA00 achieves superior overall optimization performance compared with VPPSO, ADE, IAGWO, RUN, SSA, ALA, DBO, SBOA, and the original AOO. The adaptive strategy pool dynamically balances global exploration and local exploitation, differential evolution helps preserve appropriate population diversity, and directed local search enhances convergence accuracy. Through the coordinated effects of these mechanisms, IA00 consistently maintains the best overall ranking across different benchmark suites and dimensional settings. These findings further support the performance advantages previously demonstrated by the convergence curves, means, standard deviations, and Wilcoxon rank-sum tests, confirming the general effectiveness and reliability of the proposed improvement mechanisms.

3.8. Running Time Analysis

Figure 12 presents the total running time required by each compared algorithm to complete the optimization tasks on all CEC2017 benchmark functions under the 30-dimensional setting. The horizontal axis represents the total computational time, and the horizontal bar chart provides an intuitive comparison of the computational cost among different algorithms. Beyond optimization accuracy, this analysis further evaluates the engineering applicability of IA00 from the perspective of computational efficiency.

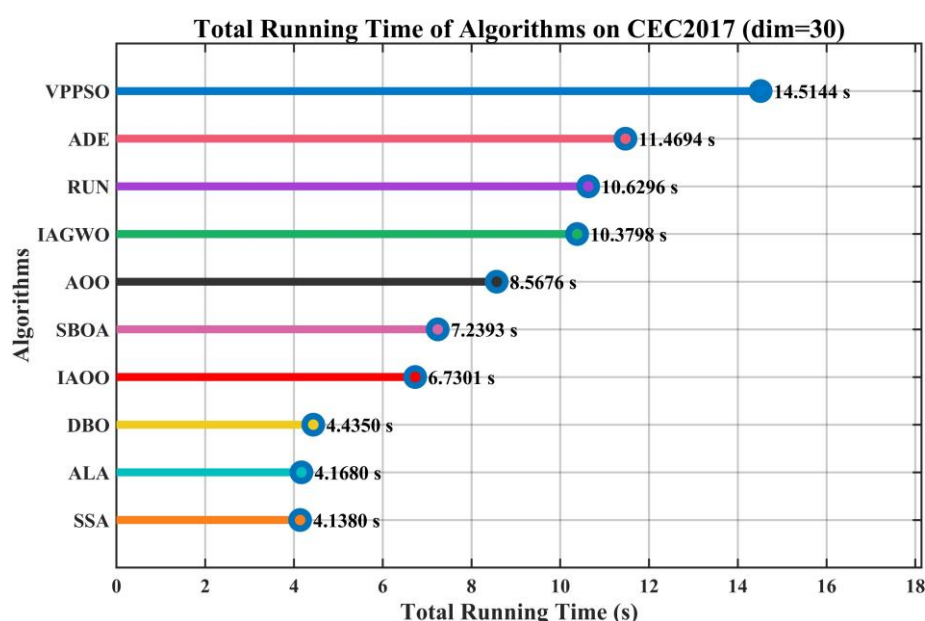


Figure 12. Comparison of the total running time of different algorithms on CEC2017 (dim = 30).

As observed from the results, VPPSO exhibits the highest computational cost, with a total running time of 14.5144 s. ADE, RUN, and IAGWO also require relatively high computational resources, with execution times exceeding 10 s. The original AOO achieves a moderate computational efficiency, with a total running time of 8.5676 s. In contrast, the proposed IA00 requires only 6.7301 s, which is lower than that of the original AOO and most competing algorithms, including VPPSO, ADE, RUN, IAGWO, and SBOA. It is slightly slower only than SSA, ALA, and DBO, which employ relatively simpler algorithmic structures.

Although the proposed improvement strategies introduce additional computational operations, such as adaptive strategy selection and hybrid differential evolution operators, which theoretically increase the computational burden of individual update

processes, IAEO effectively compensates for this overhead through a more efficient search mechanism. By reducing redundant random exploration and improving the utilization efficiency of population information, IAEO avoids excessive ineffective iterations and does not result in a substantial increase in overall computational cost. Compared with the original AEO, IAEO significantly improves optimization accuracy, population diversity regulation capability, and convergence performance while simultaneously reducing the total execution time. This demonstrates that the proposed improvement framework enhances optimization performance without sacrificing computational efficiency.

Although SSA, ALA, and DBO achieve shorter running times due to their relatively simple structures, the convergence curves, Friedman test results, and boxplot analyses presented in previous sections indicate that these algorithms exhibit limited global search capabilities and are more likely to be trapped in local optima when solving high-dimensional and multimodal optimization problems.

By comprehensively considering both optimization performance and computational cost, IAEO achieves an effective balance between solution quality and execution efficiency. Compared with computationally expensive algorithms such as VPPSO and ADE, IAEO provides superior optimization capability while significantly reducing computational overhead. Meanwhile, compared with lightweight and faster algorithms, IAEO incurs only a small additional time cost but obtains considerable improvements in optimization accuracy, stability, and global exploration ability. These results demonstrate that IAEO can simultaneously achieve high-quality solutions and efficient computation, indicating its promising potential for engineering optimization applications where computational efficiency and solution reliability are both required.

4. Three-Dimensional UAV Path Planning

4.1. Problem Description and Mathematical Modeling

In realistic flight missions, unmanned aerial vehicles (UAVs) are required to autonomously generate a safe, efficient, and dynamically feasible trajectory connecting a given start position to a predefined destination within a constrained three-dimensional environment. Let the initial and terminal locations of the UAV be denoted as [17,30]:

$$P_s = (x_s, y_s, z_s), P_e = (x_e, y_e, z_e) \quad (22)$$

To ensure trajectory continuity and maneuverability, the flight path is discretized into a finite set of g waypoints, including the start point, end point, and $(g - 2)$ intermediate control points. A smooth spatial curve is subsequently reconstructed using cubic spline interpolation, yielding an ordered waypoint sequence [17,30]:

$$P = \{p_1, p_2, \dots, p_g\}, p_i = (x_i, y_i, z_i) \quad (23)$$

From an optimization perspective, UAV path planning can be formulated as a constrained multi-objective problem. The overall cost function is defined as a weighted combination of three critical performance indicators [17,30]:

$$F = w_1 \cdot F_{len} + w_2 \cdot F_{alt} + w_3 \cdot F_{smooth} \quad (24)$$

where w_1 , w_2 , and w_3 are non-negative weighting coefficients satisfying:

$$w_1 + w_2 + w_3 = 1 \quad (25)$$

Each cost component reflects a practical flight requirement, as described below.

(1) Path Length Cost

Path length directly influences flight time and energy consumption. The cumulative distance of the UAV trajectory is computed as [17,30]:

$$F_{len} = \sum_{i=1}^{g-1} \|p_{i+1} - p_i\|_2 \quad (26)$$

where $\|\cdot\|_2$ denotes the Euclidean norm.

(2) Altitude Stability Cost

Excessive altitude fluctuations increase control complexity and may degrade flight safety. To suppress abrupt vertical oscillations, the standard deviation of altitude values along the trajectory is adopted [17,30]:

$$F_{alt} = \sqrt{\frac{1}{g} \sum_{i=1}^g (z_i - \bar{z})^2} \quad (27)$$

where $\bar{z}$ represents the mean altitude of all waypoints.

(3) Trajectory Smoothness Cost

The UAV trajectory must satisfy multiple constraints, including:

- Obstacle avoidance: all waypoints must lie outside obstacle volumes;
- Terrain clearance: altitude must remain above the ground surface;
- Flight envelope limits: altitude and maneuverability constraints.

These conditions jointly define the feasible flight space L , leading to the final constrained optimization formulation [17,30]:

$$\min_{P \in L} F \quad (28)$$

In this study, both the ground and obstacle sets are mathematically formulated using Equation (29) [17].

$$z = \sin(y + 1) + \sin(x) + \cos(x^2 + y^2) + 2 \times \cos(y) + \sin(x^2 + y^2) \quad (29)$$

Finally, a map as shown in Figure 13 is obtained.

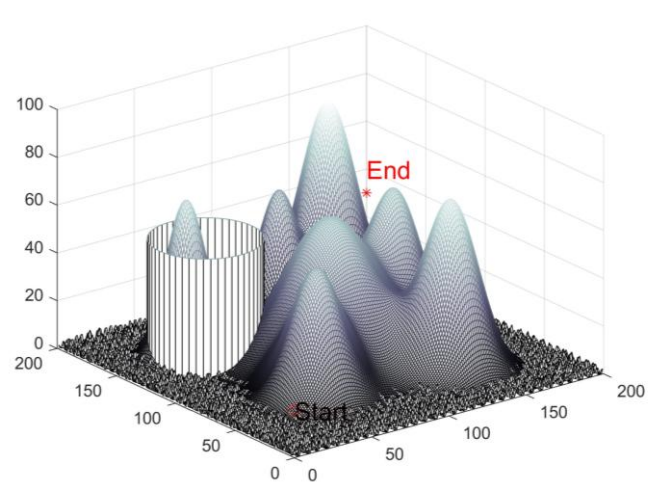


Figure 13. Drone flight map.

4.2. Relationship to UAV Dynamics and Tracking Control

The present model focuses on global reference-path generation and incorporates geometric and first-order kinematic constraints. It does not directly incorporate the complete translational and rotational dynamics of a quadrotor into the IAOO decision vector. This distinction is important because a collision-free and smooth spline is not necessarily

executable under thrust, attitude-rate, actuator-saturation, wind-disturbance, and model-uncertainty constraints [28].

Li et al. [27] incorporated three-degree-of-freedom and six-degree-of-freedom UAV dynamic models into joint trajectory and beamforming optimization. Tran et al. [28] investigated trajectory and parameter optimization for robust quadrotor tracking and demonstrated the importance of coordinating reference-trajectory design with lower-level flight control. These studies provide a complementary control-oriented perspective to the global path-planning problem considered here.

Therefore, the path generated by IAOO should be interpreted as a smooth and kinematically constrained reference path for a downstream trajectory generator and flight controller. A complete planning-and-control framework would pass the IAOO path to a dynamically feasible trajectory optimizer and subsequently evaluate closed-loop tracking error, control effort, actuator saturation, and disturbance rejection. Such integration is beyond the current scope and will be investigated in future work.

4.3. Experimental Configuration and Parameter Settings

To validate the practical effectiveness of the proposed IAOO algorithm, a three-dimensional UAV navigation scenario was constructed. The UAV departs from the initial position (0,0,20) and is required to reach the destination (200,200,30).

Following engineering considerations, the weighting coefficients were set as:

$$w_1 = 0.5, w_2 = 0.3, w_3 = 0.2 \quad (30)$$

emphasizing path efficiency while maintaining altitude stability and smoothness.

All algorithms under comparison were evaluated under identical environmental conditions and parameter configurations to ensure fairness. Each algorithm was independently executed multiple times to reduce randomness, and cubic spline interpolation was consistently employed for trajectory reconstruction.

In the UAV application, AOO does not directly model physical aircraft parameters. Each optimizer individual represents the coordinates of the intermediate waypoints: $\mathbf{x}_i = [x_2, y_2, z_2, \dots, x_{g-1}, y_{g-1}, z_{g-1}]$, $D = 3(g - 2)$.

The upper and lower bounds define the admissible three-dimensional map region. After an individual is updated, it is decoded into a waypoint sequence, reconstructed using cubic-spline interpolation, checked for boundary and collision violations, and evaluated using the UAV path-cost function. Therefore, the population size NNN, iteration number T , Lévy step, rolling displacement, and ejection displacement are optimization variables controlling candidate waypoint vectors. They should not be interpreted as UAV mass, thrust, velocity, or aerodynamic parameters. UAV-related requirements are introduced through the objective function, waypoint bounds, clearance constraints, and maneuverability penalties defined in Section 4.

4.4. Simulation Analysis

Table 9 presents the simulation statistical results of IAOO and nine comparative algorithms in a three-dimensional unmanned aerial vehicle (UAV) path planning task. The reported metrics include the mean path cost (Mean), standard deviation (Std), best value (Best), worst value (Worst), median (Median), runtime (RunTime), and statistical test indicators, thereby providing a comprehensive evaluation of the optimization performance of each algorithm in a practical application scenario. In addition, Figures 14 and 15 illustrate the flight paths obtained by different algorithms and their corresponding fitness convergence curves, respectively.

Table 9. Statistics of drone path planning simulation results.

Algorithm	Mean	Std	Best	Worst	Median	RunTime	Friedman	FriedmanRank	<i>p</i> -Value
VPPSO	331.06	91.88	228.57	419.20	377.01	37.19	4.10	4	8.4460×10^{-2}
ADE	382.09	72.58	228.59	439.81	417.01	26.74	6.37	6	6.7362×10^{-6}
IAGWO	363.76	77.01	228.88	412.77	412.76	45.79	3.53	2	1.2212×10^{-2}
RUN	423.86	75.32	228.78	439.75	439.70	43.32	8.27	10	7.1186×10^{-9}
SSA	507.88	164.62	228.76	827.61	433.57	27.97	7.07	8	4.3531×10^{-5}
ALA	371.63	81.45	228.58	439.69	412.85	26.86	5.00	5	7.6973×10^{-4}
DBO	382.46	80.11	228.59	459.21	420.25	25.16	6.67	7	1.6813×10^{-4}
SBOA	334.80	88.92	228.56	416.01	377.92	50.44	3.67	3	4.0354×10^{-1}
AOO	406.26	67.01	228.86	572.65	419.38	26.99	7.17	9	6.0104×10^{-8}
IAOO	298.94	40.73	228.56	380.45	231.00	27.13	3.17	1	\

Bold values indicate the best-performing result in that row or column.

Table 9 quantitatively summarizes the performance of IAOO and nine comparative algorithms in a three-dimensional UAV path planning task from multiple perspectives. A detailed analysis of the key performance metrics is provided as follows.

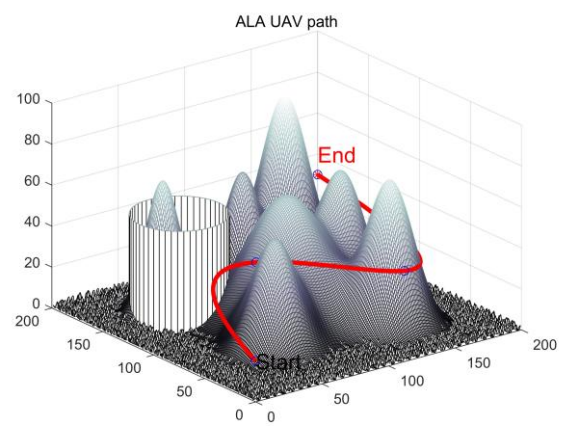
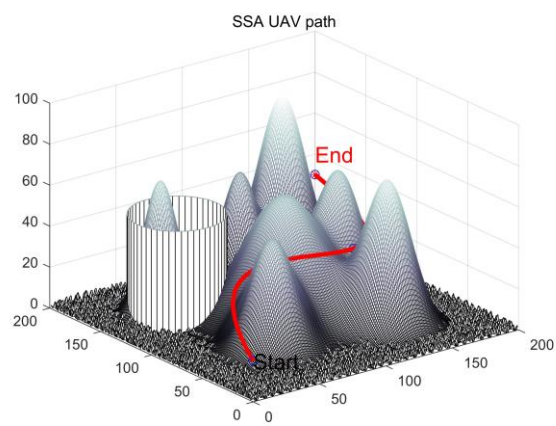
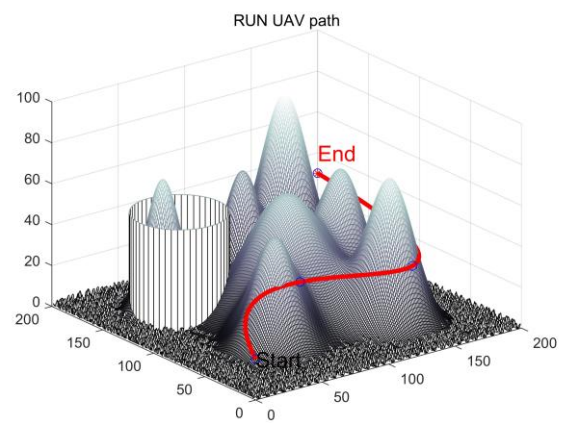
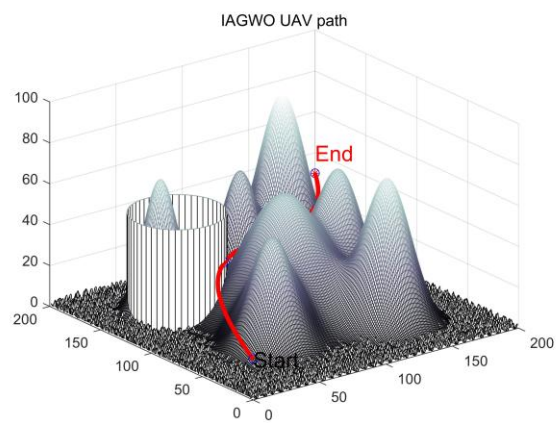
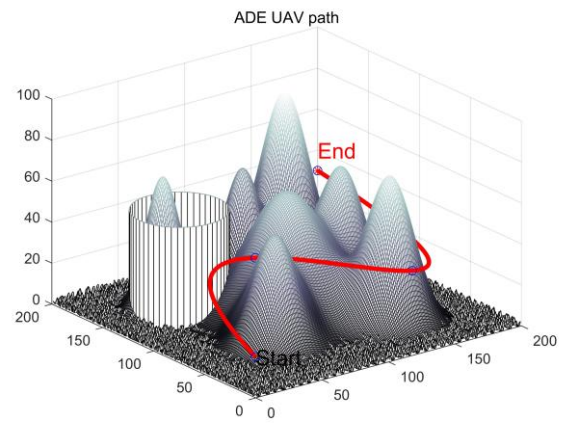
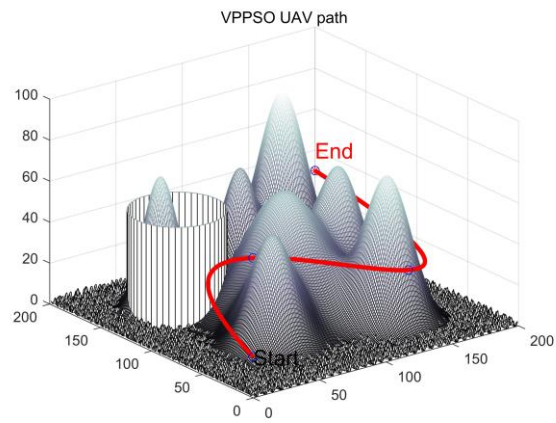
IAOO achieves the lowest mean path cost of 298.94 among all algorithms, representing a reduction of 26.4% compared with the original AOO (406.26), 17.8% compared with the second-best algorithm IAGWO (363.76), and 41.1% compared with the relatively poor-performing SSA (507.88). These results indicate that, while jointly considering path length, altitude stability, and smoothness, IAOO produces significantly more optimal flight paths than the other algorithms, effectively reducing flight energy consumption and time cost.

In terms of stability, IAOO records a standard deviation of only 40.73, which is substantially lower than those of SSA (164.62), VPPSO (91.88), and ALA (81.45), and is only 60.8% of that of the original AOO (67.01). The low standard deviation implies that, over 30 independent runs, IAOO exhibits minimal fluctuations in path cost, reflecting strong robustness in complex three-dimensional environments and the ability to consistently generate high-quality paths while avoiding severe performance degradation caused by stochastic search behavior.

Regarding extreme values, the best path costs obtained by all algorithms are concentrated within the range of 228.56–228.88, indicating that all methods are capable of approaching the theoretical optimum under ideal conditions. However, the worst path cost produced by IAOO (380.45) is markedly lower than those of AOO (572.65), RUN (439.75), and DBO (459.21), with a maximum difference of 192.2. This demonstrates that IAOO possesses stronger resistance to environmental complexity and a reduced tendency to become trapped in poor local optima. In terms of the median metric, IAOO achieves the lowest value of 231.00, further confirming that the majority of its optimization outcomes are of high quality.

With respect to computational efficiency, IAOO exhibits an average runtime of 27.13, which is comparable to those of AOO (26.99), ADE (26.74), and ALA (26.86). This indicates that the introduction of multiple improvement strategies does not incur a significant increase in computational cost, thereby maintaining a favorable balance between optimization performance and execution efficiency.

The Friedman test results further show that IAOO attains the lowest Friedman value of 3.17 and ranks first overall, identifying it as the algorithm with the best comprehensive performance. Moreover, the associated *p*-values confirm that the performance differences between IAOO and most comparative algorithms are statistically significant. These findings collectively demonstrate the superior effectiveness, robustness, and practical applicability of IAOO in three-dimensional UAV path planning tasks.



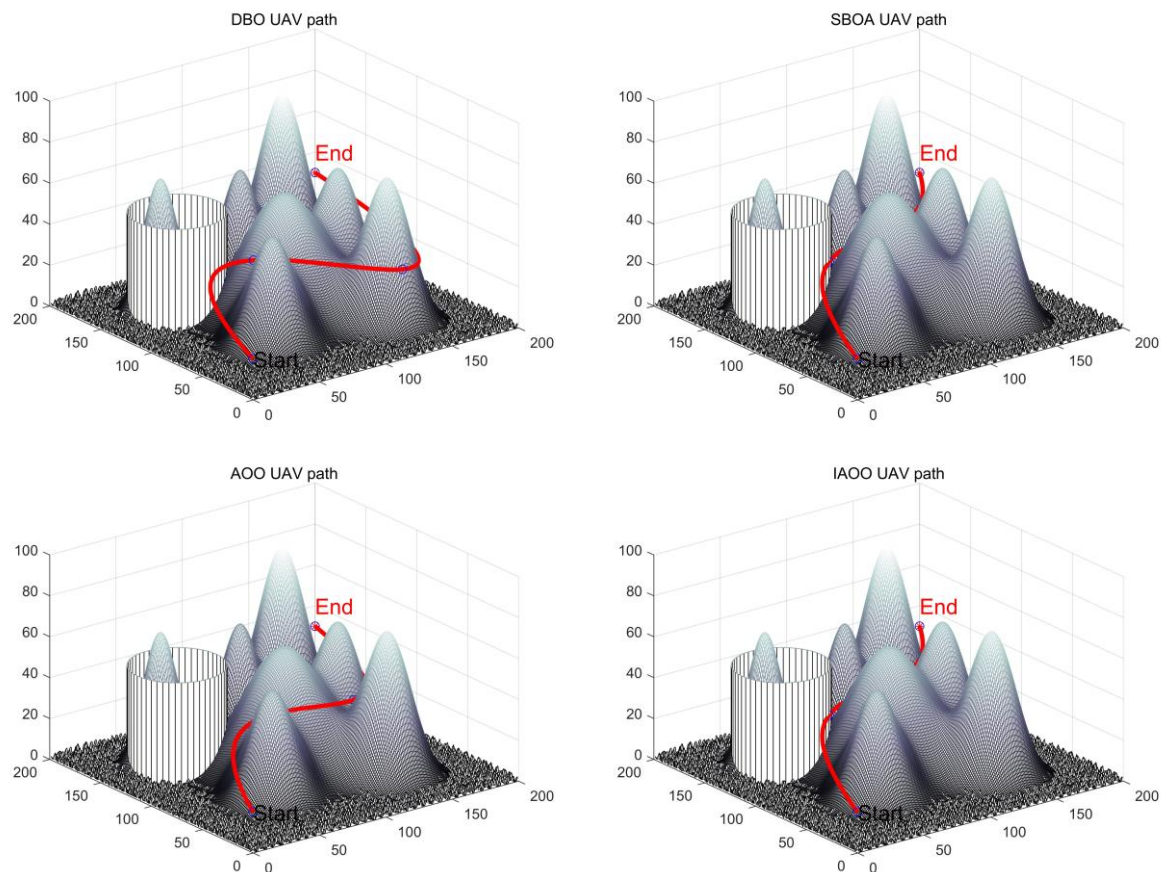


Figure 14. Drone flight paths obtained by different algorithms.

The UAV flight paths illustrated in Figure 14 visually demonstrate the differences in path planning performance among the algorithms. The path generated by IAIO from the start point (0,0,20) to the target point (200,200,30) follows a trajectory that is much closer to a straight line. When traversing complex terrain, IAIO smoothly avoids obstacles with minimal altitude fluctuations, strong trajectory continuity, and no apparent detours or abrupt turns. In contrast, the path planned by the original AOO contains multiple unnecessary deviations, exhibits pronounced altitude oscillations, and approaches obstacle boundaries in certain segments, resulting in reduced safety and efficiency. The paths generated by RUN, SSA, and similar algorithms are even more tortuous, containing numerous redundant segments that significantly increase the overall path length. Compared with relatively competitive algorithms such as SBOA and IAGWO, IAIO further shortens the flight distance while maintaining path smoothness, and its waypoint distribution is more uniform, which better satisfies the dynamic and kinematic constraints of practical UAV flight.

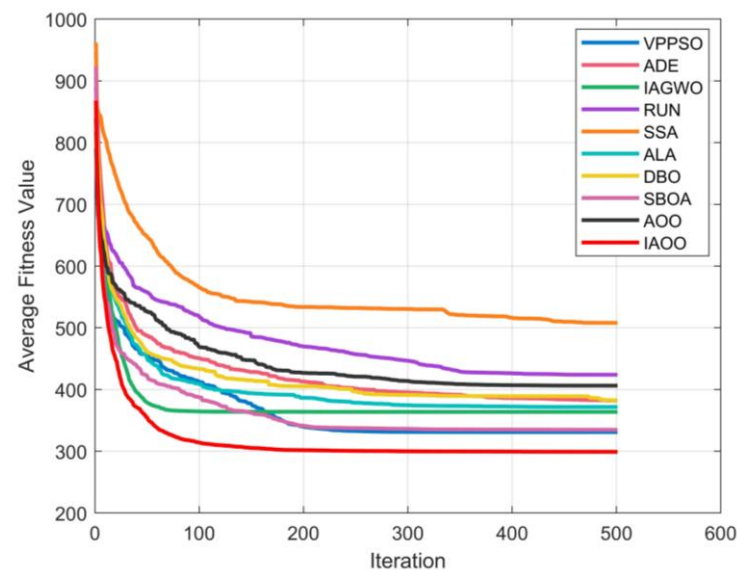


Figure 15. Fitness curve variation.

The fitness convergence curves shown in Figure 15 further reveal the convergence characteristics of IAIO. The fitness curve of IAIO decreases rapidly during the early iterations and becomes stable after approximately 100 iterations, with a final convergence value that is significantly lower than those of all other algorithms. This indicates that IAIO is able to quickly locate the globally optimal path region. In contrast, the convergence curves of AOO, SSA, and similar algorithms exhibit noticeable oscillations, with large fluctuations persisting even in the later stages of iteration, suggesting unstable search behavior and a tendency to stagnate in local optima. Algorithms such as VPPSO and DBO converge more slowly, requiring nearly 300 iterations to reach a stable state, and their final fitness values remain substantially higher than that of IAIO. Throughout the optimization process, the convergence curve of IAIO consistently remains at a lower level without significant oscillations, highlighting the synergistic effect of the differential evolution hybridization strategy and the directional local search strategy. The former enhances population diversity and facilitates rapid global exploration, while the latter enables fine-grained exploitation of promising regions, thereby achieving simultaneous improvements in convergence speed and solution quality.

By jointly analyzing the statistical results in Table 9, the flight path visualization in Figure 14, and the convergence behavior in Figure 15, it can be concluded that IAIO demonstrates significant comprehensive advantages in three-dimensional UAV path planning tasks. While ensuring path safety and smoothness, IAIO effectively shortens the flight distance and improves flight efficiency, while also exhibiting strong stability and favorable convergence characteristics. These results indicate that IAIO provides an efficient and reliable solution for practical UAV path planning applications.

5. Conclusions and Future Work

This study addresses the limitations of the original Animated Oat Optimization Algorithm (AOO) in solving complex optimization problems, including the imbalance between exploration and exploitation, insufficient convergence accuracy, and limited search stability. To overcome these drawbacks, an improved Animated Oat Optimization Algorithm (IAIO) is proposed. By introducing a multi-strategy adaptive updating mechanism and organically integrating differential evolution hybridization with a directional local search strategy, the adaptive capability of the algorithm at different search stages is significantly enhanced.

In terms of numerical optimization performance, IA00 demonstrates stable and statistically significant superiority on three international benchmark suites: CEC2017, CEC2020, and CEC2022. According to the Friedman mean rank test, IA00 achieves average ranks of 1.50, 1.37, 2.70, and 1.83 on the CEC2017 (dim = 30/100), CEC2020, and CEC2022 benchmark sets, respectively, consistently ranking first among all compared algorithms. These results indicate that IA00 maintains strong global exploration and local exploitation capabilities even in dynamic environments with multiple optima, thereby fully validating its superior comprehensive optimization performance and robustness.

In terms of engineering applications, IA00 is successfully applied to a three-dimensional UAV path planning problem. Under identical environmental conditions and parameter configurations, the average path cost obtained by IA00 is only 298.94, which is significantly lower than that of the original A00 (406.26) and other comparative algorithms. Meanwhile, its standard deviation is reduced to 40.73, indicating that IA00 not only produces better solutions but also exhibits higher solution stability under complex constraints. Path simulation results further confirm the comprehensive advantages of IA00 in terms of path length, flight smoothness, and operational safety.

Despite the observed improvements, IA00 has several limitations. First, its contribution is a feedback-coordinated hybrid framework rather than a completely new elementary search operator. Its advantage may therefore diminish on landscapes where the introduced operators generate redundant search directions. Second, the elite neighborhood search introduces an $O(\rho N^2 D)$ computational term. Although the measured runtime overhead is small for the population size used in this study, approximate-neighbor search or adaptive local-search frequency may be required for very large populations. Third, the probability update is reward-driven but not explicitly landscape-aware, and the influence of algorithmic parameters cannot be completely eliminated. Fourth, the current UAV case considers a static environment and generates a geometrically feasible and kinematically constrained reference path. Moving obstacles, wind disturbances, sensor uncertainty, energy dynamics, actuator saturation, and closed-loop six-degree-of-freedom tracking have not yet been investigated.

Future work will focus on lightweight neighborhood structures, landscape-feature-based strategy selection, constrained multi-objective IA00, dynamic and multi-UAV path planning, and integration with model-predictive or robust flight control. Hardware-in-the-loop simulation and real-flight experiments will also be conducted to further evaluate the practical applicability of IA00.

Author Contributions: Conceptualization, X.Z. and C.X.; methodology, X.Z. and C.X.; software, X.Z. and C.X.; validation, X.Z. and C.X.; formal analysis, X.Z. and C.X.; investigation, X.Z. and C.X.; resources, X.Z. and C.X.; data curation, X.Z. and S.L.; writing—original draft preparation, X.Z. and S.L.; writing—review and editing, X.Z. and S.L.; visualization, X.Z. and S.L.; supervision, X.Z. and S.L.; funding acquisition, X.Z. and S.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Guizhou Provincial Science and Technology Plan Project (QianKeHe JiChu QN [2026] 012-10) and the Guizhou Provincial Science and Technology Plan Project (QianKeHe PingTai RSSYS [2025] Major 003).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: The authors would like to express their sincere gratitude to all those who contributed to the completion of this work.

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

References

1. Abdulkareem, M.; Aghdasi, H.S.; Salehpour, P.; Zolfy, M. Binary Secretary Bird Optimization Clustering by Novel Fitness Function Based on Voronoi Diagram in Wireless Sensor Networks. *Sensors* **2025**, *25*, 4339.
2. Cai, X.; Wang, W.; Wang, Y. Multi-strategy enterprise development optimizer for numerical optimization and constrained problems. *Sci. Rep.* **2025**, *15*, 10538.
3. Chen, B.; Hai, M.; Di, G.; Zhou, B.; Zhang, Q.; Wang, M.; Guo, Y. Enhanced Dung Beetle Optimizer-Optimized KELM for Pile Bearing Capacity Prediction. *Buildings* **2025**, *15*, 2654.
4. Yu, M.; Yang, H.; Zhang, J.; Ouyang, K.; Fu, S.; Tan, P.; Jiang, F.; Xu, J. Bounty hunter optimizer: A novel metaheuristic with an application to multi-UAV mobile edge computing and path planning. *Knowl.-Based Syst.* **2026**, *341*, 115836.
5. Zhang, C.; Yang, Y.; Chen, W. Multi-strategy ensemble wind driven optimization algorithm for robot path planning. *Math. Comput. Simul.* **2025**, *231*, 144–159.
6. Feng, L.; Liu, H.; He, Z.; Ye, H.; Yang, S. Multi-strategy secretary bird optimization algorithm for UAV path planning in complex environment. *Sci. Rep.* **2025**, *16*, 101.
7. Yu, M.; Zhang, J.; Yu, L.; Yang, H.; Lu, H.; Wei, X.; Xu, J. Alzheimer's disease brain image segmentation using multi-feature fusion in 3D Rényi entropy model and quantum hybrid optimization. *Artif. Intell. Rev.* **2026**, *59*, 95.
8. Yu, M.; Yang, H.; Wei, X.; Fu, S.; Xu, J.; Zhang, J. Carbon Footprint Efficiency Ratio: A Unified Indicator for Green Evaluation of Evolutionary Algorithms. In *IEEE Transactions on Emerging Topics in Computational Intelligence*; IEEE: New York, NY, USA, 2026; pp. 1–14.
9. Holland, J.H. Genetic algorithms. *Sci. Am.* **1992**, *267*, 66–73.
10. Kennedy, J.; Eberhart, R. Particle swarm optimization. In *Proceedings of ICNN'95-International Conference on Neural Networks*; IEEE: New York, NY, USA, 1995; Volume 4, pp. 1942–1948.
11. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer Advances. *Adv. Eng. Softw.* **2014**, *69*, 46–61.
12. Storn, R.; Price, K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359.
13. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67.
14. Guan, Z.; Ren, C.; Niu, J.; Wang, P.; Shang, Y. Great Wall Construction Algorithm: A novel meta-heuristic algorithm for engineer problems. *Expert Syst. Appl.* **2023**, *233*, 120905.
15. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H. Marine Predators Algorithm: A nature-inspired metaheuristic. *Expert Syst. Appl.* **2020**, *152*, 113377.
16. Liu, X.; Li, S.; Wu, Y.; Fu, Z. Graduate Student Evolutionary Algorithm: A Novel Metaheuristic Algorithm for 3D UAV and Robot Path Planning. *Biomimetics* **2025**, *10*, 616.
17. Fu, Y.; Liu, D.; Chen, J.; He, L. Secretary bird optimization algorithm: A new metaheuristic for solving global optimization problems. *Artif. Intell. Rev.* **2024**, *57*, 123.
18. Wang, T.-L.; Gu, S.-W.; Liu, R.-J.; Chen, L.-Q.; Wang, Z.; Zeng, Z.-Q. Cuckoo catfish optimizer: A new meta-heuristic optimization algorithm. *Artif. Intell. Rev.* **2025**, *58*, 326.
19. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. In *IEEE Transactions on Evolutionary Computation*; IEEE: New York, NY, USA, 1997; Volume 1, pp. 67–82.
20. Zhao, Y.; Zheng, Z.; Liu, Y. Survey on computational-intelligence-based UAV path planning. *Knowl.-Based Syst.* **2018**, *158*, 54–64.
21. Bashir, N.; Boudjit, S.; Dauphin, G.; Zeadally, S. An obstacle avoidance approach for UAV path planning. *Simul. Model. Pract. Theory* **2023**, *129*, 102815.
22. Haam, S.; Kim, H.; Yoo, M.; Song, W.S. Optimal additional evacuation route analysis in deep underground station structures using Dijkstra's algorithm. *Dev. Built Environ.* **2025**, *24*, 100784.
23. Milin, V.; Stanivuk, T.; Skoko, I.; Bulić, T. Dijkstra and A* Algorithms for Algorithmic Optimization of Maritime Routes and Logistics of Offshore Wind Farms. *J. Mar. Sci. Eng.* **2025**, *13*, 1863.
24. Xing, J.; Wang, C.; Gao, Q.; Wang, T.; Zheng, H. MSRIME: A multi-strategy RIME optimization algorithm for three-dimensional UAV path planning. *Meas. Sci. Technol.* **2025**, *36*, 096214.
25. Wang, Z.; Zhang, X.; Li, W.; Feng, Z. Real-Time UUV Obstacle Avoidance Through Flexible Steering Technology Based on Improved Soft Actor-Critic Framework. In *IEEE Transactions on Instrumentation and Measurement*; IEEE: New York, NY, USA, 2025; Volume 74, pp. 1–14.

26. Meng, F.; Xu, H.; Gao, Z.; Li, Q. Real-time trajectory planning of unmanned surface vehicles: A constraint-embedded model predictive control approach. *Ocean Eng.* **2026**, *363*, 126812.
27. Li, B.; Zhang, H.; Rong, Y.; Han, Z. A Control-Based Design of Beamforming and Trajectory for UAV-Enabled ISAC System. In *IEEE Transactions on Wireless Communications*; IEEE: New York, NY, USA, 2026; Volume 25, pp. 3469–3484.
28. Tran, N.H.; Dao, Q.T.; Nguyen, T.V.-A.; Bui, N.T. Trajectory and Parameter Optimization in Robust Tracking Control of a Quadrotor. *IEEE Access* **2025**, *13*, 155215–155232.
29. Gu, T.; Zhang, Y.; Wang, L.; Zhang, Y.; Deveci, M.; Wen, X. A comprehensive analysis of multi-strategic RIME algorithm for UAV path planning in varied terrains. *J. Ind. Inf. Integr.* **2025**, *43*, 100742.
30. Fu, S.; Li, K.; Huang, H.; Ma, C.; Fan, Q.; Zhu, Y. Red-billed blue magpie optimizer: A novel metaheuristic algorithm for 2D/3D UAV path planning and engineering design problems. *Artif. Intell. Rev.* **2024**, *57*, 134.
31. Ye, M.; Han, H.; Liu, M.; Wan, D.; Abdel-Maksoud, M. A novel hybrid physics-data driven encoder-decoder model for wind turbine wake predictions. *Ocean Eng.* **2026**, *361*, 126169.
32. Wang, R.; Hu, R.; Geng, F.; Xu, L.; Chu, S.; Pan, J.; Meng, Z.; Mirjalili, S. The Animated Oat Optimization Algorithm: A nature-inspired metaheuristic for engineering optimization and a case study on Wireless Sensor Networks. *Knowl.-Based Syst.* **2025**, *318*, 113589.
33. Zhou, J.; Liu, K.; Jiang, Z.; Liao, P. Lithium-ion battery state of charge prediction under multiple aging stages: An animated oat optimization algorithm-optimized temporal convolutional network-attention model study. *J. Energy Storage* **2025**, *140*, 119117.
34. Yuan, X.; Zhu, Z.; Yang, Z.; Zhang, Y. MSCSO: A Modified Sand Cat Swarm Optimization for Global Optimization and Multi-level Thresholding Image Segmentation. *Symmetry* **2025**, *17*, 2012.
35. Ahmed, R.; Rangaiah, G.P.; Mahadzir, S.; Mirjalili, S.; Hassan, M.H.; Kamel, S. Memory, evolutionary operator, and local search based improved Grey Wolf Optimizer with linear population size reduction technique. *Knowl.-Based Syst.* **2023**, *264*, 110297.
36. Wu, G.; Mallipeddi, R.; Suganthan, P. *Problem Definitions and Evaluation Criteria for the CEC 2017 Competition and Special Session on Constrained Single Objective Real-Parameter Optimization*; Nanyang Technological University: Singapore, 2016.
37. Mohamed, A.W.; Hadi, A.A.; Mohamed, A.K.; Awad, N.H. Evaluating the performance of adaptive gainsharing knowledge based algorithm on CEC 2020 benchmark problems. In *2020 IEEE Congress on Evolutionary Computation (CEC)*; IEEE: New York, NY, USA, 2020; pp. 1–8.
38. Luo, W.; Lin, X.; Li, C.; Yang, S.; Shi, Y. Benchmark functions for CEC 2022 competition on seeking multiple optima in dynamic environments. *arXiv* **2022**, arXiv: 2201.00523.
39. Shami, T.M.; Mirjalili, S.; Al-Eryani, Y.; Daoudi, K.; Izadi, S.; Abualigah, L. Velocity pausing particle swarm optimization: a novel variant for global optimization. *Neural Comput. Appl.* **2023**, *35*, 9193–9223.
40. Abbasi, B.; Majidnezhad, V.; Mirjalili, S. ADE: Advanced differential evolution. *Neural Comput. Appl.* **2024**, *36*, 15407–15438.
41. Yu, M.; Xu, J.; Liang, W.; Qiu, Y.; Bao, S.; Tang, L. Improved multi-strategy adaptive Grey Wolf Optimization for practical engineering applications and high-dimensional problem solving. *Artif. Intell. Rev.* **2024**, *57*, 277.
42. Ahmadianfar, I.; Heidari, A.A.; Gandomi, A.H.; Chu, X.; Chen, H. RUN beyond the metaphor: An efficient optimization algorithm based on Runge Kutta method. *Expert Syst. Appl.* **2021**, *181*, 115079.
43. Xue, J.; Shen, B. A novel swarm intelligence optimization approach: Sparrow search algorithm. *Syst. Sci. Control Eng.* **2020**, *8*, 22–34.
44. Xiao, Y.; Cui, H.; Khurma, R.A.; Castillo, P.A. Artificial lemming algorithm: A novel bionic meta-heuristic technique for solving real-world engineering optimization problems. *Artif. Intell. Rev.* **2025**, *58*, 84.
45. Xue, J.; Shen, B. Dung beetle optimizer: A new meta-heuristic algorithm for global optimization. *J. Supercomput.* **2023**, *79*, 7305–7336.
46. Shen, Y.; Zhang, C.; Gharehchopogh, S.; Mirjalili, S. An improved whale optimization algorithm based on multi-population evolution for global optimization and engineering design problems. *Expert Syst. Appl.* **2023**, *215*, 119269.
47. Cao, L.; Wei, Q. SZOA: An Improved Synergistic Zebra Optimization Algorithm for Microgrid Scheduling and Management. *Biomimetics* **2025**, *10*, 664.
48. Yu, M.; Yang, H.; Fu, S.; Kong, D.; Xu, X.; Zhang, J.; Xu, J. Improved Coverage and Redundancy Management in WSN Using ENMDBO: An Enhanced Metaheuristic Solution. In *IEEE Transactions on Network and Service Management*; IEEE: New York, NY, USA, 2025.
49. Zhang, Y.; Wang, J.; Zhang, X.; Wang, B. ACPOA: An Adaptive Cooperative Pelican Optimization Algorithm for Global Optimization and Multilevel Thresholding Image Segmentation. *Biomimetics* **2025**, *10*, 596.

50. Yu, M.; Zhang, J.; Kong, D.; Zhan, K.; Fu, S.; Jiang, F.; Xu, J. Fractional-quantum reinforcement learning differential evolution for large-scale edge computing offloading. *Knowl.-Based Syst.* **2026**, *346*, 116188.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.