Information Sciences 735 (2026) 123022

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Adversarial game optimization: A game-theoretic metaheuristic for efficient complex optimization and engineering applications

Conglin Li [a], Qingke Zhang [a,*], Junqing Li [a,b], Sichen Tao [c], Diego Oliva [d]

[a] School of Computer Science and Artificial Intelligence, Shandong Normal University, Jinan 250358, China
[b] School of Mathematics, Yunnan Normal University, Yunan 650500, China
[c] Faculty of Engineering, University of Toyama, Toyama-shi 930-8555, Japan
[d] Depto. de Ingeniería Electro-Fótonca, Universidad de Guadalajara, CUCEI, 44430, Guadalajara, México

HIGHLIGHTS

- A novel adversarial game-inspired metaheuristic algorithm (AGOA) is proposed.
- Dynamic role allocation and reverse feedback improve global search robustness.
- Theoretical analysis establishes convergence to the global optimum.
- AGOA surpasses 79 state-of-the-art algorithms on CEC 2017 and CEC 2022.
- Delivers superior performance in image segmentation, engineering, and UAV path planning.

ARTICLE INFO

Keywords:
Metaheuristic optimization algorithms
Adversarial game optimization (AGOA)
Numerical global optimization
Multi-threshold image segmentation
Constrained engineering design
UAV 3D path planning

ABSTRACT

Metaheuristic optimization algorithms have demonstrated strong performance when applied to complex nonlinear optimization tasks. However, their performance often degrades in high-dimensional and multimodal settings due to premature convergence and insufficient global search. To address these limitations, an Adversarial Game Optimization Algorithm (AGOA) is proposed, which constructs a metaheuristic optimization framework based on adversarial game mechanisms. AGOA integrates three mechanisms: (i) a dynamic role-based population partitioning strategy that assigns individuals as elites, explorers, or responders to balance exploration and exploitation; (ii) an adversarial feedback mechanism where worst-case responders introduce directed perturbations to counter elite dominance; and (iii) a diversity-preserving breakout strategy that monitors population stagnation and activates adaptive restarts. AGOA was tested on CEC 2017 (30D/50D/100D) and CEC 2022 (10D/20D), as well as applications including multi-threshold image segmentation, constrained engineering design, and UAV 3D path planning. Experimental evaluations indicate that AGOA achieves superior performance compared with 79 optimizers in solution quality, convergence behavior, and stability, achieving top rankings across all test categories. Theoretical analysis further establishes convergence to the global optimum in expectation and probability under mild conditions. Overall, AGOA offers a scalable and generalizable optimization framework with strong practical relevance. An open-access implementation of AGOA is provided at https://github.com/tsingke/AGOA.

* Corresponding author.
Email addresses: tsingke@sdnu.edu.cn (Q. Zhang).
https://doi.org/10.1016/j.ins.2025.123022
Received 5 September 2025; Received in revised form 17 December 2025; Accepted 19 December 2025
Available online 20 December 2025

1. Introduction

As science and technology continue to advance, solving complex optimization problems has become a critical and pervasive challenge across diverse fields, including modern engineering, economics, and information technology. Traditional optimization methods, such as gradient-based approaches, are often constrained by their reliance on the continuity and differentiability of objective functions, which considerably restricts their applicability in practical scenarios. In contrast, intelligent optimization algorithms have attracted increasing attention due to their independence from gradient information and their adaptability to highly nonlinear, high-dimensional, and multimodal problems [1].

Over the past decades, intelligent optimization algorithms have achieved remarkable progress and have found applications across various fields, including machine learning, computer vision, data mining, engineering design, and resource scheduling [2]. Despite these advances, existing algorithms still face significant challenges when dealing with optimization tasks of higher complexity, particularly in global search capability, convergence efficiency, and avoidance of premature convergence into local optima [3].

Inspired by self-organization and adaptive processes in biological, physical, chemical, and social systems, intelligent optimization algorithms have demonstrated considerable potential. Nevertheless, they continue to struggle with optimization problems characterized by complex constraints, multiple local optima, and rugged multimodal landscapes [4]. In addressing complex problems with large scale, high dimensionality, nonlinearity, and dynamic characteristics, existing algorithms frequently exhibit the following critical issues:

Local optima trapping: One of the most pervasive challenges is the tendency of algorithms to fall into local optima. In multimodal or high-dimensional search spaces, many algorithms fail to escape local extrema, leading to stagnation and loss of global optimality. For example, elite selection strategies in genetic algorithms may reduce population diversity, hindering exploration and resulting in premature convergence [5]. Similarly, particle swarm optimization (PSO) often converges prematurely to suboptimal regions, especially in complex landscapes where collaboration among particles is insufficient [6].

Slow convergence: Convergence speed is another critical limitation in addressing high-dimensional optimization problems. Although several improved algorithms, such as differential evolution (DE) and ant colony optimization (ACO), have been developed to accelerate convergence, they often exhibit slow convergence rates in very high-dimensional scenarios [7]. This limitation reduces computational efficiency and hinders the attainment of high-quality solutions under limited computational budgets, which is particularly problematic in large-scale optimization tasks [8].

Loss of diversity and premature convergence: Maintaining population diversity is essential for preventing premature convergence. Although many algorithms incorporate stochastic or adaptive variation strategies, their effectiveness diminishes in highly multimodal landscapes. The loss of diversity restricts the exploration capacity of the population, causing stagnation in local regions and increasing the risk of missing global optima, particularly in complex high-dimensional problems [9].

In response to these issues, researchers have studied both the enhancement of existing algorithms and the development of novel frameworks and mechanisms. These algorithms can be grouped into four categories according to their inspiration sources: biology-inspired, physics-based, chemistry-based, and social or cognition-inspired algorithms [10]. Each category emulates adaptive behaviors and decision-making processes observed in natural and social systems:

1. **Biology-inspired algorithms:** These algorithms simulate adaptation, evolution, and collective behaviors in nature. They employ mechanisms such as natural selection, swarm dynamics, or genetic variation to guide the search process. Representative subclasses include:
   (a) **Evolutionary algorithms**: Genetic algorithms (GA) are the most well-known example, employing selection, crossover, and mutation operators to explore potential solutions [11]. While widely applied, they remain vulnerable to premature convergence and slow convergence rates. Differential evolution (DE) [12] and evolutionary strategies (ES) have also been extensively utilized in diverse problem domains [13].
   (b) **Swarm intelligence algorithms**: Inspired by social coordination in animals, such as bird flocks, fish schools, and bee foraging [14], this class includes algorithms such as PSO [15] and ACO [16]. These methods have demonstrated effectiveness in multi-objective and constrained optimization but frequently suffer from premature convergence and inefficient searches.
   (c) **Plant-based algorithms**: Motivated by plant growth processes and adaptive resource allocation, this category includes algorithms such as the flower pollination algorithm (FPA) [17] and plant growth simulation algorithm (PGA) [18]. These approaches are well-suited for dynamic environments but can face scalability challenges in large-scale problems.
2. **Physics-based algorithms:** These algorithms simulate physical processes such as energy transfer or particle dynamics to perform global searches. Representative examples include simulated annealing (SA) [19] and quantum-behaved PSO (QPSO) [20]. Although effective, they are often sensitive to parameter tuning and may incur substantial computational costs in high-dimensional settings [21].
3. **Chemistry-based algorithms:** Inspired by chemical reactions and molecular interactions, algorithms such as molecular dynamics optimizers (MDO) [22] and chemical reaction optimization (CRO) [23] have been developed. These approaches excel at handling multi-objective problems but may be computationally expensive [24].
4. **Social and human cognition-based algorithms:** These methods are motivated by human social behaviors, group decision-making, and knowledge dissemination. Examples include social learning optimization (SLO), which models learning through observation and imitation [25], and cultural algorithms (CA), which leverage group knowledge and cultural evolution to

Table 1
Concise comparison of AGOA with existing mechanisms, classical algorithms, and adversarial/game-based optimizers.

Aspect / algorithm	Existing approaches	AGOA
(a) Mechanism-level		
Cooperative–competitive	Roles and interactions static (CSO, MLPSO)	Dynamic role allocation with adaptive transitions.
Restart / reinitialization	Threshold- or cycle-based, heuristic only	Feedback-driven adversarial restart.
Diversity maintenance	Passive niching/crowding	Game-triggered, adaptive preservation.
(b) Classical algorithms		
Genetic Algorithm	Fixed operators, no adaptive roles	Roles + adversarial feedback unified.
Differential Evolution	Differential mutation, heuristic restart	Adversarial interaction + adaptive reinit.
Particle Swarm Optimization	Guided by bests, prone to premature convergence	Diversity triggers prevent stagnation.
(c) Game/adversarial optimizers		
CSO & variants	Winner-loser model, static roles	Dynamic roles, payoff-guided feedback.
GT-based optimizers	Nash/game models in specific tasks	General-purpose unified framework.
MaOEA-GM	Two-stage game for many-objectives	Extends to multimodal/high-dim problems.
MS-CCOEA	Random comp./coop. strategies	Integrated adversarial + diversity triggers.

enhance search efficiency [26]. Such algorithms are particularly effective in high-dimensional, nonlinear, and multimodal problem spaces.

Despite the progress made, the aforementioned limitations significantly restrict the reliability and scalability of existing intelligent optimization algorithms when applied to high-dimensional and real-world scenarios. In particular, the lack of effective mechanisms to simultaneously strengthen global exploration, accelerate convergence, and preserve population diversity has become a critical bottleneck. These challenges motivate the exploration of new algorithmic paradigms beyond conventional bio-inspired or swarm-based designs.

Game theory, as a rigorous mathematical framework for modeling strategic interactions, provides valuable guidance for optimization. Its paradigm emphasizes the balance between cooperation and competition among multiple agents, which can be naturally mapped onto the behaviors of individuals in a population-based search process. By leveraging adversarial dynamics, algorithms can encourage diversified exploration while maintaining the selective pressure required for effective exploitation. This perspective helps mitigate premature convergence, enhance robustness, and strengthen adaptability in dynamic and uncertain settings.

Although related strategies have been explored in existing metaheuristics, they remain fragmented. For instance, cooperative-competitive strategies have been incorporated into algorithms such as the competitive swarm optimizer (CSO) and Multi-Leader PSO, but these typically rely on static role definitions without a systematic game-theoretic framework for adaptive role allocation [27,28]. Restart and reinitialization mechanisms, widely used in differential evolution (DE) and its hybrids, are usually activated by heuristic thresholds and detached from adversarial feedback [29]. Diversity–preservation techniques such as niching, crowding, and orthogonal learning are effective in multimodal problems, but they operate passively and seldom respond adaptively to the actual population state [30]. These limitations are summarized in Table 1, which contrasts representative mechanisms in existing metaheuristics with the distinctive designs introduced in AGOA.

In addition to the above, it is also important to clarify the differences between AGOA and several representative algorithms, namely GA, DE, and PSO. Genetic algorithms (GA) mainly rely on fixed selection, crossover, and mutation mechanisms, but they lack adaptive role allocation or explicit feedback mechanisms. Differential evolution (DE) forms new solutions through mutation and crossover operations, although it often depends on heuristic restart rules and does not incorporate adversarial interactions. Particle swarm optimization (PSO) guides individuals using global and personal best experiences, but due to the absence of diversity-triggered mechanisms, it is prone to premature convergence. In contrast, AGOA integrates these perspectives into a unified game-theoretic framework, featuring dynamic role allocation, adversarial feedback, and game-triggered diversity preservation, which together provide a better balance between exploration and exploitation.

Recently, adversarial and game-theoretic ideas have been increasingly incorporated into metaheuristic optimization. Representative lines include competitive-update paradigms such as the competitive swarm optimizer (CSO) and its variants [27], game-theoretic coordination embedded in swarm dynamics [31], and cooperative–competitive mechanisms explicitly formulated as two-stage games for evolutionary search. These studies demonstrate that competition/cooperation and adversarial interactions can improve exploration–exploitation balance. However, most existing approaches still rely on relatively *static* role definitions or pre-specified competition rules, and typically lack a unified framework that adaptively links cooperation–competition with diversity-triggered interventions. In contrast, AGOA provides a systematic game-theoretic paradigm that integrates *dynamic role allocation*, *adversarial feedback*, and *game-triggered diversity preservation*, yielding a more adaptive mechanism for complex, high-dimensional, and multimodal problems.

These analyses make it evident that while existing mechanisms, classical algorithms, and recent game-based optimizers have achieved partial success, they remain fragmented and lack a unified adversarial framework. To illustrate these distinctions more clearly, Table 1 provides a concise comparison highlighting how AGOA differs from existing approaches.

Drawing on this perspective, a novel metaheuristic framework is proposed, the *Adversarial Game Optimization Algorithm* (AGOA). AGOA introduces a game-theoretic paradigm that simulates cooperative-competitive interactions among multiple agents to enhance global exploration and prevent premature convergence.

Building upon this foundation, AGOA incorporates four synergistic components:

(i) a **dynamic role allocation** mechanism that adaptively partitions individuals into elite, ordinary, and adversarial subpopulations to maintain exploration–exploitation balance;
(ii) an **adversarial strategy optimization and payoff-guided feedback** scheme that enables cooperative–competitive learning and enhances convergence efficiency;
(iii) a **diversity-preserving breakout mechanism** that dynamically reinitializes stagnated individuals to prevent population degeneration.

Through extensive evaluations on CEC 2017 and CEC 2022 benchmark suites, as well as in practical tasks including multilevel image segmentation and UAV path planning, AGOA demonstrates superior convergence accuracy, robustness, and scalability in comparison with 79 optimizers. Moreover, the integration of AGOA's adversarial dynamics provides valuable insight for broader AI-driven optimization contexts, including recent works on metaheuristic–chaotic hybrid frameworks for biomedical diagnosis[32] and intelligent path-planning tasks in autonomous systems [33]. Overall, this study not only broadens the theoretical frontier of game-inspired metaheuristics but also establishes a general and practical optimization framework for complex engineering and AI-driven tasks.

The rest of the paper is structured as follows. Section 2 introduces the design principles and algorithmic framework of AGOA. Section 3 reports experimental results and comparative analyses with leading algorithms. Section 4 evaluates AGOA in multilevel thresholding image segmentation. Section 5 discusses its performance in engineering design problems. Section 6 investigates AGOA's application in three-dimensional UAV path planning. Finally, Section 7 presents the key findings and outlines directions for future research.

2. Adversarial game optimization algorithm

Metaheuristic algorithms are commonly modeled on natural, biological, or social phenomena. However, many such approaches lack a rigorous theoretical foundation to model strategic interactions among individuals during the search process. To address this limitation, the adversarial game optimization algorithm (AGOA) incorporates fundamental concepts from game theory, a mathematical framework that studies decision-making behaviors in competitive and cooperative environments. In AGOA, each individual is conceptualized as a "player" in a strategic optimization game, where the search dynamics are governed by iterative strategy adaptation and payoff-driven learning. This game-theoretic perspective enables AGOA to explicitly model multi-agent interactions, facilitating a more effective balance between cooperation and competition within the population. This design facilitates exploration and lowers the risk of early convergence. The subsequent subsection elaborates on the inspiration and theoretical foundations of AGOA in greater detail.

2.1. Inspiration

The adversarial game optimization algorithm (AGOA) draws inspiration from adversarial game models rooted in game theory [34], a well-established mathematical framework for analyzing strategic decision-making among multiple interacting agents. Applications of game theory include areas such as economics, sociology, political science, and computer science, particularly for modeling and understanding complex competitive-cooperative behaviors. In the context of numerical optimization, these principles can be leveraged to represent the optimization process as a dynamic game in which individuals ("players") continuously adapt their strategies based on observed performance and interactions with others. This strategic adaptation allows the population to collectively move toward the global optimum while mitigating the risk of stagnation in suboptimal regions.

Unlike traditional metaheuristic algorithms that may suffer from premature convergence, adversarial game-inspired strategies enable richer and more flexible search dynamics. By incorporating competitive and cooperative interactions, AGOA enables diverse search across the solution space and dynamically adjusts search strategies to maintain a balance between local and global search. Through this design, AGOA not only enhances robustness but also demonstrates improved adaptability to complex and multimodal optimization landscapes.

Fig. 1 presents the complete framework of the adversarial game optimization algorithm (AGOA). In the first stage, the dynamic role-based population structuring mechanism assigns individuals to elite, ordinary, and worst roles to maintain equilibrium between local and global search while enriching population diversity. In the second stage, adversarial strategy generation and adjustment are performed, where new candidate solutions are generated through a combined contribution from different roles, evaluated for gain, and iteratively refined via adaptive feedback mechanisms. Finally, diversity-driven game-breaking mechanism monitors population diversity and applies stochastic restarts to maintain global search capability and prevent premature convergence.

2.2. Initialization

In AGOA, initialization plays a key role in covering the search space broadly and enabling effective global exploration. The initial population must not only exhibit high diversity but also ensure that each individual has the potential to explore a wide range of solutions. The steps are as follows:

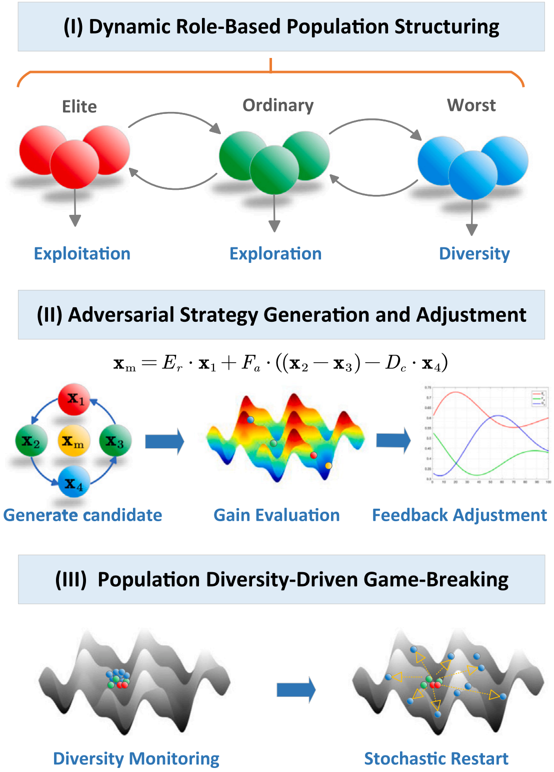


Fig. 1. Schematic diagram of the adversarial game optimization algorithm.

Participant definition and strategy space: Define the participants $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, where each participant $\mathbf{x}_i$ has an associated strategy space D_i, representing a set of possible parameter configurations. Each strategy $d_i \in D_i$ is a multidimensional vector corresponding to a candidate solution.

Payoff function: Define the strategy combination as $D_i = (d_1, d_2, \ldots, d_n)$ for participant i. The payoff of each strategy combination is evaluated using a fitness function $Fitness$, representing solution quality. In this study, lower fitness values correspond to higher payoffs, indicating better strategies.

Random strategy initialization: During the initial game setup, each participant generates strategies randomly to ensure diversity. This is mathematically expressed as Eq. (1):

$$x_i = lb + (ub - lb) \cdot \mathrm{rand}(1, D) \tag{1}$$

where lb and ub represent the bounds of the solution space, and $\mathrm{rand}(1, D)$ generates random numbers with a uniform distribution.

Payoff evaluation: A payoff function (implemented as the fitness function $Fitness$) is employed to measure the initial payoff for each participant and determine the current global best strategy. For minimization problems, individuals with lower fitness values achieve higher payoffs and are more likely to act as guides in steering the evolutionary search. **Parameter initialization:** AGOA introduces several dynamically adjustable parameters to enhance its intelligence, adaptability, and balance exploration and exploitation.

- **Elite Reinforcement Factor** (E_r): regulates the influence of elite individuals in Eq. (3), accelerating convergence while preventing excessive dominance.
- **Defensive Counterattack Factor** (D_c): controls the effect of worst individuals, enabling reverse guidance to escape local optima.
- **Cooperative Probability** (C_o): probability of cooperative crossover in Eq. (4), promoting information sharing.
- **Flexible Adjustment Factor** (F_a): scales differential perturbations, similar to DE, balancing exploration and exploitation.
- **Diversity Threshold** ($D_{threshold}$): computed on normalized variables, this corresponds to about 35 % of the expected initial diversity, ensuring a consistent trigger for game-breaking rather than an arbitrary constant.
- **Stagnation Limit** (T_{stag}): the limit of successive iterations with no global improvement prior to restart, typically about 1–5 % of the evaluation budget, balancing sensitivity and stability.

2.3. Dynamic role-based population structuring

In each iteration, AGOA dynamically partitions the population into three distinct roles based on individual payoff values: elite (20 %), ordinary (60 %), and worst (20 %).

The leading 20 % of the population are designated as elites. These individuals serve as guiding agents, steering the population toward superior regions in the solution space and accelerating convergence toward optimal solutions. The middle 60 % are categorized as ordinary individuals, whose primary function is to maintain a trade-off between local search and global exploration. By introducing moderate perturbations, these individuals support diversity and help avoid premature convergence. The bottom 20 % constitute the worst individuals. Rather than being simply discarded, these individuals play a vital role by introducing reverse guidance mechanisms. They explore alternative, divergent directions, thereby improving the algorithm's capacity to avoid local optima and enriching the overall search dynamics. The role assignment for each individual $\mathbf{x}_i$ is formally defined in Eq. (2):

$$\mathrm{Role}(\mathbf{x}_i) = \begin{cases} \text{Elite}, & \text{if } \mathrm{Rank}(\mathbf{x}_i) \le K\% \\ \text{Ordinary}, & \text{if } K\% < \mathrm{Rank}(\mathbf{x}_i) \le (100-K)\% \\ \text{Worst}, & \text{if } \mathrm{Rank}(\mathbf{x}_i) > (100-K)\% \end{cases} \tag{2}$$

where $\mathrm{Rank}(\mathbf{x}_i)$ denotes the rank of individual $\mathbf{x}_i$ in terms of payoff, and K (commonly set to 20) defines the threshold for elite and worst groups.

Such dynamic role-based structuring mechanism enables AGOA to achieve a highly adaptive and preserved equilibrium between exploration and exploitation throughout the search. By synergistically leveraging elite-driven guidance, ordinary-induced diversity, and worst-induced disruptive exploration, AGOA enhances its global search efficacy while ensuring strong local refinement. Such a framework increases the reliability of convergence and enhances solution quality in difficult optimization tasks.

2.4. Adversarial generation and adjustment

A key innovation of AGOA lies in its dynamic strategy generation and adaptive feedback mechanism, which collectively enhance flexibility, adaptability, and convergence efficiency during the optimization process. This component comprises three tightly integrated stages: dynamic strategy construction, payoff evaluation with feedback, and parameter self-adaptation.

Dynamic strategy construction: In each iteration, AGOA synthesizes new candidate individuals by leveraging diverse roles within the population. Specifically, elite individuals provide strong directional guidance toward promising regions; ordinary individuals introduce moderate exploratory perturbations to maintain diversity; and worst individuals act as disruptors, injecting reverse adjustments to explore alternative regions and help escape local optima. The integrated design is mathematically formalized in Eq. (3):

$$\mathbf{x}_\mathrm{m} = E_r \cdot \mathbf{x}_1 + F_a \cdot \left((\mathbf{x}_2 - \mathbf{x}_3) - D_c \cdot \mathbf{x}_4\right) \tag{3}$$

where $\mathbf{x}_1$ denotes an elite (guiding) individual, $\mathbf{x}_2$ and $\mathbf{x}_3$ are randomly selected ordinary individuals, and $\mathbf{x}_4$ is the worst individual. The Elite Reinforcement Factor E_r amplifies the guidance provided by high-quality solutions in the search process. The Defensive Counterattack Factor D_c modulates the contribution from worst individuals, enhancing the algorithm's capacity to escape unfavorable regions. The initial values for E_r and D_c are typically set to 0.1 to maintain a balanced influence at early stages. The Adaptive Strategy Adjustment Factor F_a is dynamically tuned by parameter F (initially set to 0.7) and a chaos factor cf, which further ensures flexible and context-aware exploration and exploitation. By modulating F_a, AGOA can adaptively adjust its search dynamics in each iteration. During early stages, higher chaotic variance encourages broad exploration, while in later stages, the factor naturally stabilizes to favor exploitation near promising regions. The specific update rules for the parameters E_r, D_c, F_a are detailed in the following subsection.

Gain evaluation and feedback: After constructing the new mutant individual $\mathbf{x}_m$, its quality is rigorously assessed using the fitness function. If $\mathbf{x}_m$ exhibits a better (i.e., lower) fitness value than the current global best $f(\mathbf{x}_\mathrm{g})$, it is directly accepted as the new trial solution $\mathbf{x}_t$, and the global best is immediately updated. This direct replacement mechanism accelerates convergence toward high-quality regions and reinforces exploitation of promising areas.

If $\mathbf{x}_m$ does not outperform the global best, a controlled crossover strategy incorporating chaotic perturbations is activated to generate an alternative trial solution $\mathbf{x}_t$. Specifically, each dimension j of $\mathbf{x}_t$ is determined in Eq. (4):

$$x_{t,j} = \begin{cases} x_{m,j}, & \text{if } f(x_m) < f(\mathbf{x}_\mathrm{g}) \\ \begin{cases} x_{m,j}, & \text{if } r_j < C_o \times cf \text{ or } \ j = j' \\ x_{i,j}, & \text{otherwise} \end{cases}, & \text{otherwise} \end{cases} \tag{4}$$

where $\mathbf{x}_i$ denotes the current individual being updated, and r_j is a random number uniformly distributed in $[0, 1]$. The parameter C_o represents the cooperative game probability, while the chaos factor cf introduces additional stochasticity to promote exploration diversity. When $r_j < C_o \times cf, the j$-th dimension inherits from $\mathbf{x}_m$; otherwise, it retains its value from $\mathbf{x}_i$. The specific update rules for the parameter cf are detailed in the following subsection.

To avoid the risk of stagnation when no dimensions are altered after the crossover operation, a safeguard mechanism is implemented: if all dimensions remain unchanged, a random dimension j' is forcibly assigned from $\mathbf{x}_m$.

This gain evaluation and adaptive feedback framework enables AGOA to dynamically balance the trade-off between applying the best solutions and searching broadly in the solution space. By flexibly adjusting the acceptance and crossover strategies based on real-time fitness performance, AGOA maintains robust global search capability, prevents premature convergence, and significantly enhances overall optimization effectiveness.

Parameter Adaptive Adjustment: To continuously refine its search dynamics, AGOA integrates a game experience library that stores historical performance data and dynamically adjusts critical parameters in real time. Successful strategies are reinforced, thereby strengthening their influence on future searches, while unsuccessful or stagnated strategies are progressively weakened or eliminated. This adaptive adjustment mechanism enables AGOA to learn from past iterations, enhance its decision-making capability, and maintain a balanced exploration–exploitation trade-off. The parameter update scheme is mathematically defined in Eq. (5):

$$\begin{aligned}
&\Delta f = f_{\mathbf{gbestx}} - f_{\mathbf{reply}} \\
&E_r = \begin{cases} E_r + 0.1 \cdot \Delta f, & \text{if } \Delta f > 0 \\ E_r - 0.1 \cdot \Delta f, & \text{if } \Delta f \le 0 \end{cases} \\
&D_c = \begin{cases} D_c - 0.1 \cdot \Delta f, & \text{if } \Delta f > 0 \\ D_c + 0.1 \cdot \Delta f, & \text{if } \Delta f \le 0 \end{cases} \\
&F_a = \begin{cases} F \times cf, \\ cf = cf \cdot rand + (1 - cf) \cdot \sin(\pi \cdot rand) \end{cases}
\end{aligned} \tag{5}$$

where $f_{\mathbf{gbestx}}$ denotes the fitness value of the current global best solution, and $f_{\mathbf{reply}}$ represents a historical fitness value randomly sampled from the experience library. The parameters E_r and D_c are adaptively updated based on the fitness difference Δf, with a scaling factor of 0.1 to ensure gradual, stable adaptation. The base scaling parameter F (initially set to 0.7) and the chaos factor cf (default value of 0.1) collaboratively determine F_a. The composite chaos factor introduces additional non-linear perturbations by blending uniform random sampling with a sinusoidal transformation, thereby enhancing stochasticity. This design prevents premature convergence, fosters robust global exploration, and maintains strong adaptability in complex, high-dimensional optimization landscapes.

2.5. Population diversity-driven game-breaking

In order to prevent individual('strategy') homogenization and reduce the risk of early convergence, AGOA integrates a diversity-driven game-breaking mechanism inspired by adversarial game dynamics. This mechanism combines continuous diversity monitoring with stochastic restart strategies to maintain robust global search capabilities.

Diversity Monitoring Mechanism: The algorithm continuously monitors and quantifies population diversity by systematically analyzing the spatial distribution and dispersion characteristics of individuals across the entire solution space. In particular, the diversity metric Div is rigorously defined as the mean of the standard deviations calculated along each decision variable dimension, thereby capturing the overall spread and heterogeneity of the population. This formulation, as formalized in Eq. (6), enables precise detection of potential convergence trends and provides an early warning mechanism to prevent premature stagnation.

$$\begin{aligned}
&Div = \frac{1}{D} \sum_{j=1}^{D} \sigma_j, \\
&\sigma_j = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (x_{i,j} - \bar{x}_j)^2}, \\
&\bar{x}_j = \frac{1}{N} \sum_{i=1}^{N} x_{i,j}, \\
&\text{Trigger} = \begin{cases} 1, & \text{if } Div < D_{\text{threshold}} \text{or stagnation_count} > 100, \\ 0, & \text{otherwise.} \end{cases}
\end{aligned} \tag{6}$$

where D represents the problem dimensionality, N is the size of the population, and σ_j represents the standard deviation in the j-th dimension. The threshold $D_{\text{threshold}}$ (initially set to 0.1) defines the minimum acceptable diversity. The stagnation counter, stagnation_count, tracks the number of consecutive iterations without significant fitness improvement. The game-breaking mechanism is activated if Div falls below $D_{\text{threshold}}$ (indicating population homogenization) or if stagnation_count > 100 (signaling potential local stagnation).

Stochastic Restart Mechanism: Once triggered, the algorithm reinitializes selected worst-performing individuals by replacing them with new, randomly generated solutions within the defined search bounds. This process is conceptually akin to introducing new "players" or strategies to disrupt the current equilibrium and reinvigorate global exploration. Injecting diversity helps the population escape local optima and achieve more uniform search space coverage, significantly boosting global search performance and overall stability.

This integrated framework ensures that AGOA preserves an adaptive equilibrium throughout convergence speed and population diversity. It thereby facilitates robust global search performance, particularly on challenging high-dimensional and multimodal optimization problems, ultimately leading to more reliable and superior optimization outcomes.

2.6. AGOA process, advantages and limitations

The overall workflow of AGOA integrates the core ideas of adversarial game theory with adaptive population dynamics to achieve robust and efficient optimization performance. The process proceeds as follows:

Step 1: Population Initialization. Generate an initial population of individuals uniformly distributed within the defined solution space, and initialize key algorithmic parameters, including adaptive coefficients (e.g., E_r, D_c, F_a) and chaos factors. This ensures a diverse starting point and provides the foundation for subsequent adaptive evolution.

Step 2: Role-Based Population Structuring. Sort the population according to fitness and dynamically categorize them into three roles: elites (top 20 %), ordinary individuals (middle 60 %), and worst performers (bottom 20 %). This hierarchical structure establishes a strategic balance between exploitation and exploration, with elites guiding convergence, ordinary individuals maintaining diversity, and worst performers encouraging escape from local optima.

Step 3: Adversarial Strategy Updating. Utilize role-based mixed-strategy interactions to generate new candidate solutions. The elite individuals lead the search direction, ordinary individuals introduce exploratory perturbations, and the worst individuals act as disruptors through reverse guidance. Solutions with improved fitness are accepted immediately; otherwise, chaotic crossover and stochastic mutation mechanisms are activated to refine the solution further and prevent stagnation.

Step 4: Diversity-Driven Game-Breaking. Continuously monitor population diversity to prevent strategy homogenization. When diversity falls below a predefined threshold or prolonged stagnation is detected, a stochastic restart mechanism is triggered to reinitialize selected worst individuals, thereby enhancing global exploration and revitalizing the search dynamics.

Step 5: Iterative Evolution and Termination. Repeat the above steps iteratively, updating the global best solution at each generation. The algorithm runs until a defined termination criterion is reached, for example, a limit on fitness evaluations or a target accuracy.

With this integrated approach, AGOA maintains a flexible and robust exploration–exploitation balance, suitable for tackling difficult high-dimensional and multimodal problems. The pseudocode of AGOA is shown in Algorithm 1.

Algorithm 1: AGOA(Adversarial Game Optimization Algorithm).

Input: $N, D, ub, lb, MaxIter, InitialParameters$
Output: Global optimal solution **gbestx**

`/* Initialize population */`
for $i = 1$ **to** N **do**
  $\mathbf{x}_i = lb + (ub - lb) \cdot rand(1, D)$;
Evaluate fitness for all individuals;
Initialize parameters E_r, D_c, F_a, C_o, and chaos factor χ;
E_r: elite *reinforcement factor* (guidance weight in Eq. (3));
D_c: defensive *counter factor* (worst-role weight in Eq. (3));
F_a: adversarial step scaling (computed from base F and chaos factor in Eq. (5));
C_o: cooperation probability (used in Eq. (4) for crossover);
χ: chaos factor (randomization strength for exploration, may decay over time);
$D_{threshold}$: fixed diversity threshold for restart ;
T_{stag}: stagnation limit for restart ;
while *termination condition not met* **do**
  `/* Role-Based Population Structuring */`
  Sort population by fitness (ascending order);
  Assign roles: elite, ordinary, worst based on sorted fitness;
  `/* Adversarial strategy generation and adjustment */`
  for $i = 1$ **to** N **do**
    Randomly select $\mathbf{x}_1$ from elite;
    Randomly select $\mathbf{x}_2$, $\mathbf{x}_3$ from ordinary;
    Randomly select $\mathbf{x}_4$ from worst;
    Generate mutant individual $\mathbf{x}_m$ using Eq. (3);
    Evaluate $\mathbf{x}_m$ and compute trial vector $\mathbf{x}_t$ using Eq. (4);
    Perform parameter adaptive adjustment using Eq. (5);
    if $f(\mathbf{x}_t) < f(\mathbf{x}_i)$ **then**
      $\mathbf{x}_i \leftarrow \mathbf{x}_t$;
      Update **gbestx** and global best fitness if necessary;
  `/* Diversity-Driven Game-Breaking */`
  Compute population diversity Div using Eq. (6);
  if $Div < D_{threshold}$ **or** $stagnation_{count} > T_{stag}$ **then**
    Reinitialize the worst individuals;
return gbestx;

AGOA offers several significant advantages over conventional metaheuristic algorithms:

- **Strong global search capability:** The integrated adversarial role dynamics, combining guidance from elites, balanced exploration from ordinary individuals, and disruptive influence from worst individuals, considerably increase the capacity to avoid being trapped in local optima while exploring complex landscapes.
- **Accelerated convergence:** The dynamic strategy optimization, coupled with adaptive gain evaluation and feedback mechanisms, facilitates rapid convergence to optimal solutions while retaining flexibility.
- **Effective diversity preservation:** The embedded diversity monitoring and stochastic restart mechanisms prevent premature convergence, promote continuous discovery of new solutions, and maintain extensive search space coverage.

The core concept of the adversarial game optimization algorithm (AGOA) centers on game-theoretic modeling and dynamic strategy confrontation, enabling individuals to adaptively transition among distinct roles for maintaining an effective trade-off between exploration and exploitation. By integrating a multi-dimensional optimization framework with dynamic role-based population structuring, AGOA effectively addresses the challenges of complex, high-dimensional optimization landscapes. This design ensures that elite strategies continuously drive convergence toward high-quality solutions, while diverse roles preserve population diversity and reinforce global search capability, ultimately enhancing robustness and overall optimization performance.

However, the inclusion of role reassignment and feedback evaluation introduces a slightly higher computational overhead compared with simpler population-based methods. Nevertheless, this trade-off is justified by AGOA' s substantially improved convergence stability and solution quality, making it a practical and efficient choice for real-world optimization tasks.

2.7. Computational complexity and efficiency

The computational load of the adversarial game optimization algorithm (AGOA) primarily depends on the population size (N), problem dimension (D), and the maximum number of iterations (T). In each generation, the main computational costs arise from three components: population sorting with a complexity of $O(N \log N)$, vectorized mutation and crossover operations with $O(ND)$, and fitness evaluations with $O(N \cdot C_f)$, where C_f denotes the cost of a single objective function evaluation. Thus, the overall time complexity of AGOA can be written as:

$$O(T \times (N \log N + ND + N \cdot C_f)) \approx O(\mathrm{MaxFEs} \cdot C_f),$$

which indicates that the computational cost grows approximately linearly with the maximum number of function evaluations. The space complexity is $O(ND)$, as the algorithm only needs to store the current population, fitness values, and a fixed-size replay buffer. Overall, AGOA exhibits computational efficiency comparable to other population-based algorithms (e.g., DE and GO), but achieves faster convergence due to its adaptive role interaction and diversity-preserving mechanisms.

2.8. Discussion on the distinctiveness of AGOA' s mechanism

Although population-based metaheuristic algorithms often share a general iterative structure—such as parallel exploration, exploitation refinement, and stochastic update rules—their underlying search dynamics and information–exchange behaviors can differ substantially. We acknowledge that many algorithms may appear similar in mathematical form, for example, through vector perturbations, weighted combinations, or neighborhood-based sampling. However, the essential distinctiveness of an algorithm lies not in superficial algebraic expressions, but in the behavioral logic, decision pathways, and interaction mechanisms that drive the population evolution.

Importantly, AGOA is not constructed by modifying an individual operator or rephrasing an existing update pattern. Instead, its novelty arises from the coupled design of three game-theoretic components:

Dynamic Role Allocation, where individuals are adaptively assigned to elite leaders, normal explorers, and worst-case responders with differentiated behavioral rules;

Adversarial Feedback, in which worst-performing individuals actively generate directed perturbations to counterbalance elite dominance, forming a cooperative-competitive tension absent in traditional heuristics;

Game-Triggered Diversity Preservation, which activates only when adversarial dynamics indicate stagnation, constitutes a conditional reset mechanism rather than continuous random exploration.

These coupled mechanisms shape a search dynamic fundamentally different from conventional mutation-driven, attraction-repulsion, or differential-perturbation paradigms. The resulting population evolution behavior leads to more stable, adaptive search trajectories, as supported by both theoretical analysis and empirical observations. Therefore, AGOA represents a structurally novel game-theoretic optimization framework rather than a mathematical reformulation of existing metaheuristics.

3. Experiments and analysis of results

After presenting the theoretical design of AGOA, this section proceeds to evaluate its performance through extensive benchmarking and real-world experiments.

To verify the effectiveness, robustness, and generalization capability of the proposed adversarial game optimization algorithm (AGOA), this section conducts a comprehensive series of experiments covering both numerical benchmarks and real-world applications. Specifically, the evaluation spans two standardized benchmark suites—CEC 2017 and CEC 2022—and three representative applied scenarios: multilevel image segmentation, constrained mechanical design, and three-dimensional UAV path planning.

3.1. Setup of experiments and comparison algorithms

To assess the performance of the proposed AGOA, extensive experiments were carried out using two well-established benchmark suites: CEC 2017 [35] and CEC 2022 [36]. CEC 2022 includes 12 difficult test functions, encompassing shifted, rotated, hybrid, and composite types—evaluated in 10 and 20 dimensions to reflect medium- and high-dimensional optimization scenarios. Meanwhile, the CEC 2017 includes 30 functions, divided into unimodal (F1-F3), multimodal (F4-F10), hybrid (F11-F20), and composite types (F21-F30), and tested at 30D, 50D, and 100D. These testbeds collectively provide a diverse and challenging platform for assessing the global search efficiency, exploitation capability, and convergence behavior of optimization algorithms under varying levels of difficulty.

The main goal is to evaluate AGOA against 79 advanced metaheuristics in terms of optimization quality, convergence efficiency, robustness, and stability of solutions. This comparison includes recently developed optimizers such as growth optimizer (GO) [37], harris hawks optimization (HHO) [38], and kepler optimization algorithm (KOA) [39], as well as well-established methods including artificial bee colony (ABC) [40], particle swarm optimization (PSO) [41], and differential evolution (DE) [12]. By integrating a wide range of competitive baselines, the evaluation ensures a fair and comprehensive assessment of AGOA's effectiveness and competitiveness in diverse optimization environments.

All experiments were implemented in MATLAB R2022b and conducted on a workstation with an Intel(R) Core(TM) i7-12700 CPU @ 2.10 GHz and 16 GB RAM, ensuring sufficient computational resources and consistent runtime environments. To ensure fair comparison, the population size was fixed at 50 for all algorithms, and functions were tested at D = 30, 50, and 100 for CEC 2017, and D = 10 and 20 for CEC 2022. The maximum number of function evaluations (MaxFEs) was set to $10,000\times D$, providing sufficient iteration opportunities to fully exploit each algorithm's optimization potential.

To ensure experimental reproducibility and fair evaluation, all algorithms were terminated when either (1) the maximum number of evaluations ($MaxFEs = 10,000\times D$) was reached or (2) the global best fitness value did not improve by more than 10^{-8} over 100 consecutive iterations. Each algorithm was executed independently for 31 runs, and random seeds were generated using MATLAB's clock-based initialization (rand('state', sum(100*clock))), ensuring both randomness and reproducibility across runs.

All algorithms adopted the same initialization strategy—uniform random sampling within the defined search space—to eliminate initialization bias. If variables exceeded constraints, they were limited to the variable bounds; other algorithms followed the handling procedures reported in their respective sources.

To maintain fairness, all algorithms used parameter settings recommended in their original studies or as commonly suggested in the literature. The adaptive parameters of AGOA were carefully determined through systematic preliminary experiments to sustain a well-balanced exploration–exploitation trade-off. The runtime of each algorithm was recorded using MATLAB's tic–toc functions, and the average wall-clock time and number of function evaluations were verified to maintain computational parity. Comprehensive statistics including the best, mean, and standard deviation of the final fitness values were collected for all cases. Detailed descriptions of the algorithm configurations, parameter settings, and comparative strategies are provided in Table 2 for reference.

3.2. Qualitative analysis

In this subsection, four representative functions from the CEC 2017, namely F7, F9, F22, and F27, are chosen for qualitative analysis of the optimization behavior and performance of AGOA across diverse problem characteristics. Function F7 is a challenging multimodal, indivisible, and asymmetric function that is continuous everywhere yet has infinitesimal gradients throughout, making it highly sensitive to offsets and rotations and extremely difficult to optimize. Function F9 is also a shifted and rotated function with numerous local extrema, and its global optimum lies near the boundary of the search space, which makes algorithms prone to early convergence at suboptimal solutions. Function F22 is a hybrid function composed of several multimodal components, exhibiting strong regional heterogeneity and complex landscapes, particularly after offset and rotation transformations. Function F27 is a highly complex composite function that combines multiple offset-rotated subfunctions, resulting in a dense and uneven distribution of extreme points across the search space.

To better illustrate AGOA's search dynamics and convergence behavior, the optimization process is visualized from five perspectives in Fig. 2.

The first column displays the 3D topological surface of the 2D variant of each function, clearly depicting the structural complexity and ruggedness of the search landscape. This visualization highlights the inherent challenges posed by each problem.

The second column displays the search history trajectories of AGOA, with red dots denoting the final optimal solutions found. It can be observed that the majority of trajectories concentrate around the globally optimal region, while a few remain globally distributed. This demonstrates AGOA's strong ability to rapidly identify promising regions and intensively exploit them, thereby enhancing the convergence accuracy.

The third column presents the convergence curve of average fitness versus iterations. Across all four functions, AGOA exhibits a rapid fitness decrease in the early iterations, suggesting effective global exploration. During the middle stages, oscillatory fluctuations (so-called "peaks") are observed, suggesting that the algorithm can overcome local optima and maintain exploratory search. Toward the end, the curves gradually stabilize, demonstrating AGOA's effective convergence and robust global search capability.

The fourth column records the trajectory changes along a single dimension of individual optimal solutions over time. In the initial phase, significant oscillations reflect active exploration of the global solution space. As the algorithm iterates, trajectories converge toward targeted regions, showing a shift to local search. Notably, the reoccurrence of oscillations in later iterations reveals that

Table 2
Algorithm overview and parameter configurations.

No.	Algorithms	Parameters Settings
1	Artificial bee colony (ABC)	a = 1,L = round(0.6×N× D)
2	Ant colony optimization (ACO)	q = 0.5,$\zeta = 1$
3	Artificial electric field algorithm (AEFA)	$K_0 = 500,\alpha = 30$
4	Artificial ecosystem-based optimization (AEO)	a = rand ×(FEs/MaxFEs),c = randn/2×$\lvert$randn$\rvert$
5	Ali baba and the forty thieves(AFT)	No tunable algorithm parameters involved
6	Aquila optimizer(AO)	$u = 0.0265$, $r_0 = 10,\omega = 0.005,phi0 = 3$*pi/2
7	Atom search optimization (ASO)	$\alpha = 50,\beta = 0.2$
8	Bat algorithm(BA)	A = rand(N,1), r = rand(N,1), $\alpha = 0.5$, $\gamma = 0.5$, $r_o = 0.001$
9	Biogeography-based optimization (BBO)	KeepRate = 0.2,$\alpha = 0.9$,Mutation = 0.1, $\sigma = 0.02\times$(ub-1 b)
10	Bacterial foraging optimization (BFO)	c = 25,s = 4,r = 4,p = 0.5,Sr = N/2
11	Brain storm optimization (BSO)	m = 5,$P_{5\ a} = 0.2,P_{6\ b} = 0.8,P_{6bi} = 0.4,P_{6\ c} = 0.5$,k = 20,$\mu = 0,\sigma = 1$
12	Balanced teaching-learning-based optimization (BTLBO)	p = 0
13	Beluga whale optimization(BWO)	W_F = 0.1 - 0.05 * (T/MaxFEs),$\alpha = 3/2$
14	Cultural algorithm (CA)	pAccept = 0.35,$\alpha = 0.3$
15	Coronavirus herd immunity optimizer (CHIO)	MaxAge = 100,CO = 1,BRr = 0.05
16	Covariance matrix adaptation evolution strategy (CMA-ES)	$\sigma = 0.3$
17	Coyote optimization algorithm (COA)	$N_c = 5,N_p = 8$
18	Crested porcupine optimizer(CPO)	$N\_\min = 120$,T = 2, $\alpha = 0.2,T\_f = 0.8$
19	Cuckoo search (CS)	$P_a = 0.25,\alpha = 1,\beta = 1.5,\sigma_v = 1$
20	Chimp optimization algorithm (CHOA)	$f = 2\sim 0,c1_{g1} = c1_{g2} = 1.69\sim 0,c2_{g1} = c2_{g2} = 0.75\sim 2.5$
21	Differential evolution (DE)	F = 0.5,CR = 0.9
22	Dwarf mongoose optimization algorithm (DMOA)	b = 3,Lp = 0.6,peep = = 2
23	Differential search (DS)	$P_1 = 0.3\times$rand,$P_2 = 0.3\times$rand
24	Electromagnetic field optimization(EFO)	$R\_rate = 0.3,Ps\_rate = 0.2,P\_field = 0.1,N\_field = 0.45$,phi = (1 + sqrt(5))/2
25	Equilibrium optimizer (EO)	V = 1,$a_1 = 2,a_2 = 1$,GP = 0.5
26	Firefly algorithm (FA)	$\alpha = 0.2,\beta = 2,\gamma = 1,\delta = 0.05\times$(ub-lb),Damping Ratio = 0.98
27	FDBSFS	S.Maximum_Diffusion = 1,S.Walk = 1
28	Fick's law algorithm(FLA)	$C_1 = 0.5,C_2 = 2,C_3 = 0.1,C_4 = 0.2,c_5 = 2$
29	Flower pollination algorithm (FPA)	p = 0.8
30	Genetic algorithms (GA)	pc = 0.8,pm = 02
31	Gradient-based optimizer(GBO)	pr = 0.5,α = abs(β.*sin((3*pi/2 + sin(3*pi/2*β))))
32	Growth optimizer (GO)	$P_1 = 5,P_2 = 0.001,P_3 = 0.3$
33	Gravitational search algorithm (GSA)	$G_0 = 100,\alpha = 20$,Rpower = 1
34	Gaining sharing knowledge (GSK)	p = 0.1,$k_r = 0.9,k_f = 0.5$
35	Group search optimizer(GSO)	nproducers = 1,nscroungers = 0.8,a = round(sqrt(dim + 1)),stall = 50,verbose = 1
36	Grey wolf optimizer (GWO)	$\alpha = 2–2\times$(FEs/MaxFEs)
37	Heap- based optimizer (HBO)	$cycles = \lfloor$MaxIt/25$\rfloor$,degree = 3
38	Harris hawks optimization (HHO)	$E_0 = 2\times$rand-1,$E_1 = 2–2\times$(FEs/MaxFEs)
39	Human learning optimization algorithm(HLO)	bit = 32,m = bit*Dim, p_r = 5.0/m,p_i = 0.85 + 2.0/m
40	Homonuclear molecules optimization(HMO)	Atomic_numbe = 34, N_Homonuclear = 8,Beta = 2
41	Harmony search (HS)	HMCR = 0.9,PAR = 0.1,FW = 0.02×(ub-1 b),$FW_{damp} = 0.995$
42	weighted mean of vector (INFO)	I = rand([2.5])
43	Jade	CR = 0.5, F = 0.8, ArchiveSize = 100, AdaptationRate = 0.1
44	Jaya	no tunable algorithm parameters involved
45	Kepler optimization algorithm (KOA)	$T_c = 3,M_0 = 0.1$,lambda = 15
46	Lightning search algorithm (LSA)	maxchtime = 10,direct = sign(unifrnd(-1,1))
47	Moth-flame optimization (MFO)	r = -l-(FEs/MaxFEs),t = r + (l-r)×rand,b = 1
48	Marine predators algorithm (MPA)	FADs = 0.2,P = 0.5
49	Manta ray foraging optimization (MRFO)	S = 2,r_1 = rand,Coef = FEs/MaxFEs
50	Multi-verse optimizer (MVO)	$WEP_{max} = 1$, $WEP_{min} = 0.2$
51	Nutcracker optimizer(NOA)	Alpha = 0.05,Pa2 = 0.2,Prb = 0.2
52	Optical microscope algorithm(OMA	$p_l = 0.3,p_e = 0.7$
53	One-to-one-based optimizer(OOBO)	No tunable algorithm parameters involved
54	Partial reinforcement optimizer(PRO)	No tunable algorithm parameters involved
55	Particle swarm optimization (PSO)	w = 1,$w_\alpha = 0.99,c_1 = 1.5,c_2 = 2.0$
56	Remora optimization algorithm(ROA)	No tunable algorithm parameters involved
57	Reptile search algorithm(RSA)	Alpha = 0.1,Beta = 0.005
58	Runge kutta optimizer (RUN)	a = 20,b = 12
59	Simulated annealing (SA)	$T_0 = 0.1$, $\alpha = 0.99$,c = 5,$\mu = 0.5,\sigma = 0.1\times$(ub-1 b)
60	Sine cosine algorithm (SCA)	a = 2
61	Supply-demand-based optimization (SDO)	$a_1 = 2\times$((MaxFEs-FEs + 1)/MaxFEs)
62	Shuffled frog-leaping algorithm (SFLA)	Memeplexes = 5,Offsprings = 3,Step = 2,Run = 5
63	Stochastic fractal search (SFS)	MDN = 2,Walk = 1
64	SHADE	arc_rate = 1,memory_size = 5
65	Snake optimizer (SO)	$T_1 = 0.25,T_2 = 0.6,C_1 = 0.5,C_2 = 0.05,C_3 = 2$
66	Salp swarm algorithm (SSA)	$c_1 = 2\times\exp((4\times$ FEs/MaxFEs$)^2)$
67	Social spider optimization(SSO)	fpl = 0.65,fpu = 0.9,fp = fpl + (fpu-fpl)*rand,fn = round(spidn*fp),mn = spidn-fn
68	SaDE	numst = 4,CR = 0.9,strategy = 2,VTR = 0

(*continued on next page*)

Table 2
(*continued*)

No.	Algorithms	Parameters Settings
69	Teaching-learning-based optimization (TLBO)	T_F =1 or 2
70	Water evaporation optimization(WEO)	No tunable algorithm parameters involved
71	Water flow optimizer(WFO)	pl = 0.3,pe = 0.7
72	Whale optimization algorithm (WOA)	b = 1,a_1 = 2-(2×FEs/MaxFEs),a_2 = -l-(FEs/MaxFEs)
73	White shark optimizer (WSO)	f = [0.07,0.75],τ = 4.11,p = [0.5,1.5],a_0 = 6.25,a_1 = 100,a_2 = 0.0005
74	**Adversarial game optimization algorithm(AGOA)**	$E_r = D_c = C_o = 0.1$

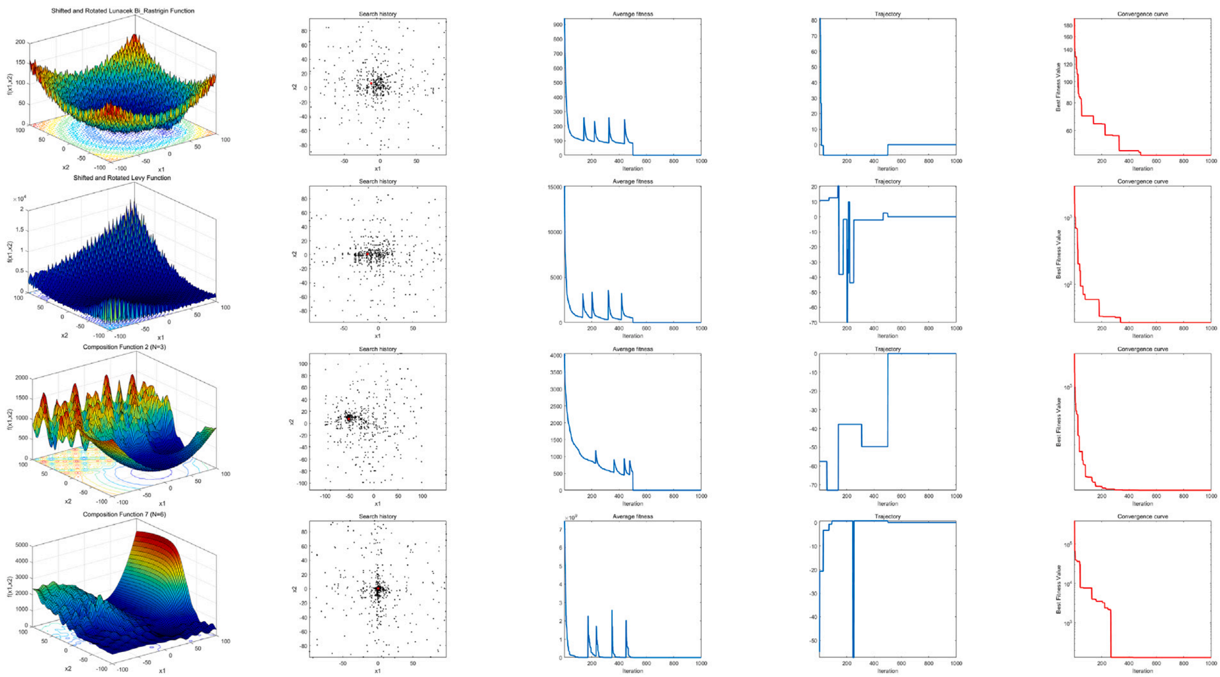


Fig. 2. Qualitative analysis results of AGOA on some functions of CEC 2017.

AGOA can dynamically jump out of local optima, driven by its game-breaking strategy, thus preventing stagnation and enhancing final solution quality.

The fifth column displays changes in the best fitness value during iterations, directly reflecting the convergence trend. The results clearly indicate that AGOA steadily avoids entrapment in local minima during iterations and achieves superior solution quality across all test functions.

In summary, the qualitative analysis comprehensively demonstrates that AGOA exhibits excellent adaptability, robustness, and convergence stability across diverse complex optimization problems. Throughout the optimization process, AGOA preserves an effective trade-off between exploration and exploitation. In the early stages, its rapid fitness decline confirms strong global search ability, while intermediate oscillations indicate its capability to escape local optima and maintain population diversity. The final convergence patterns, with stable fitness curves and concentrated trajectories around global optima, reflect its high convergence accuracy and superior solution quality.

Furthermore, the distribution of search trajectories confirms that AGOA can quickly identify and focus on promising regions, enhancing overall optimization precision. These detailed observations validate AGOA's strong global search capacity and reliable local refinement, highlighting its capability to effectively tackle highly nonlinear, dynamic, and multimodal landscapes. Overall, AGOA proves to be a highly effective and versatile optimization framework, offering great potential for addressing challenging high-dimensional and real-world engineering problems, and providing new insights into adaptive adversarial strategy design for next-generation metaheuristic algorithms.

3.3. Analysis of parameter sensitivity

This subsection conducts sensitivity tests on five key parameters of AGOA: E_r, D_c, C_o, $D_{threshold}$, and T_{stag}. For each parameter, three representative values were selected to assess their influence on the convergence performance. The experiments aim to evaluate how variations in these parameters affect the convergence accuracy and stability of AGOA, and to determine the most suitable settings for subsequent experiments.

Table 3
Effect of different parameter settings (E_r, D_c, C_o) on average fitness value error.

Function	E_r				D_c				C_o			
	Parameter	30D	50D	100D	Parameter	30D	50D	100D	Parameter	30D	50D	100D
F7	E_r=0.05	7.66 E+001	1.59 E+002	2.16 E+002	D_c=0.05	7.94 E+001	1.76 E+002	1.97 E+002	C_o=0.05	7.99 E+001	1.43 E+002	4.72 E+002
	E_r=0.10	4.40 E+001	1.59 E+002	4.83 E+002	D_c=0.10	7.83 E+001	1.59 E+002	4.83 E+002	C_o=0.10	7.66 E+001	1.59 E+002	1.94 E+002
	E_r=0.20	3.67 E+002	1.59 E+002	4.66 E+002	D_c=0.20	7.99 E+001	1.24 E+002	5.29 E+002	C_o=0.20	4.44 E+001	6.98 E+001	4.67 E+002
F9	E_r=0.05	9.73 E-001	9.13 E+001	9.45 E+003	D_c=0.05	7.45 E-001	8.48 E+001	5.24 E+003	C_o=0.05	2.41 E+001	2.31 E+002	1.15 E+004
	E_r=0.10	1.09 E-001	7.58 E+001	1.08 E+004	D_c=0.10	7.61 E-001	7.58 E+001	1.08 E+004	C_o=0.10	5.44 E-001	7.58 E+001	9.82 E+003
	E_r=0.20	4.42 E+003	1.16 E+002	1.36 E+004	D_c=0.20	1.25 E+000	5.17 E+001	1.45 E+004	C_o=0.20	2.53 E-001	5.61 E+001	1.07 E+004
F22	E_r=0.05	1.01 E+002	4.68 E+003	6.41 E+003	D_c=0.05	1.00 E+002	5.14 E+003	8.84 E+003	C_o=0.05	1.01 E+002	4.91 E+003	1.33 E+004
	E_r=0.10	4.07 E+001	5.34 E+003	1.47 E+004	D_c=0.10	5.94 E+002	5.34 E+003	1.47 E+004	C_o=0.10	5.94 E+002	5.34 E+003	0.00 E+000
	E_r=0.20	5.69 E+032	5.12 E+003	1.07 E+004	D_c=0.20	1.00 E+002	2.45 E+003	1.49 E+004	C_o=0.20	5.80 E+002	3.21 E+003	1.57 E+004
F27	E_r=0.05	5.00 E+002	5.00 E+002	1.00 E+002	D_c=0.05	5.00 E+002	5.00 E+002	9.32 E+001	C_o=0.05	4.86 E+002	5.00 E+002	5.00 E+002
	E_r=0.10	3.00 E+002	5.00 E+002	5.00 E+002	D_c=0.10	4.89 E+002	5.00 E+002	2.26 E+003	C_o=0.10	4.89 E+002	5.00 E+002	4.00 E+002
	E_r=0.20	4.00 E+002	5.00 E+002	5.00 E+002	D_c=0.20	5.00 E+002	4.00 E+002	5.00 E+002	C_o=0.20	3.89 E+002	3.00 E+002	5.00 E+002

Table 4
The effect of different values of parameters ($D_{threshold}$, T_{stag}) on the average fitness value error.

Function	$D_{threshold}$				T_{stag}			
	Parameter	30D	50D	100D	Parameter	30D	50D	100D
F7	$D_{threshold}$=0.05	7.83 E+001	1.62 E+002	4.84 E+002	T_{stag}=50	7.31 E+001	1.61 E+002	4.99 E+002
	$D_{threshold}$=0.10	4.40 E+001	1.59 E+002	4.83 E+002	T_{stag}=100	7.66 E+001	1.59 E+002	1.94 E+002
	$D_{threshold}$=0.15	7.60 E+001	1.65 E+002	4.92 E+002	T_{stag}=200	7.18 E+001	1.53 E+002	4.60 E+002
F9	$D_{threshold}$=0.05	4.17 E-001	5.41 E+001	1.39 E+004	T_{stag}=50	1.54 E+000	6.57 E+001	1.37 E+004
	$D_{threshold}$=0.10	1.09 E-001	7.58 E+001	1.08 E+004	T_{stag}=100	5.44 E-001	7.58 E+001	9.82 E+003
	$D_{threshold}$=0.15	1.82 E+000	7.46 E+001	1.36 E+004	T_{stag}=200	3.65 E-001	1.13 E+002	1.28 E+004
F22	$D_{threshold}$=0.05	1.01 E+002	4.40 E+003	1.56 E+004	T_{stag}=50	1.00 E+002	5.30 E+003	1.43 E+004
	$D_{threshold}$=0.10	4.07 E+001	5.34 E+003	1.47 E+004	T_{stag}=100	5.94 E+002	5.34 E+003	1.47 E+004
	$D_{threshold}$=0.15	4.62 E+002	4.08 E+003	1.47 E+004	T_{stag}=200	1.00 E+002	5.98 E+003	1.17 E+004
F27	$D_{threshold}$=0.05	5.00 E+002	5.00 E+002	5.00 E+002	T_{stag}=50	5.00 E+002	5.00 E+002	5.00 E+002
	$D_{threshold}$=0.10	4.89 E+002	5.00 E+002	2.26 E+003	T_{stag}=100	3.00 E+002	5.00 E+002	5.00 E+002
	$D_{threshold}$=0.15	5.00 E+002	4.87 E+002	5.00 E+002	T_{stag}=200	5.00 E+002	5.00 E+002	5.00 E+002

Comprehensive analyses were conducted on four representative benchmark functions from the CEC 2017 suite—namely the multimodal function F7, the shifted-rotated function F9, the hybrid function F22, and the composite function F27—under three different dimensions (D = 30, 50, and 100). The experimental configuration was: Run $= 31$, $N = 40$, $D = 30, 50, 100$, and $MaxFEs = D \times 10{,}000$. Results are shown in Tables 3 and 4.

First, a parameter sensitivity analysis was conducted for E_r, which controls the influence of elite individuals. When $E_r = 0.05$, AGOA exhibited slower convergence, especially on multimodal functions (F9 and F22), due to insufficient elite guidance. When $E_r = 0.20$, performance degraded sharply in some cases (e.g., F22–30D) owing to overemphasis on elite dominance, leading to early convergence. In contrast, $E_r = 0.10$ resulted in stable, high-quality outcomes across functions and dimensions, effectively balancing global and local search. Therefore, $E_r = 0.10$ is selected as the default value.

Second, the effect of D_c, which governs the counteracting intensity of the worst individuals, was analyzed. AGOA shows moderate sensitivity to this parameter. A smaller value ($D_c = 0.05$) weakens the disruptive exploration role, while a larger value ($D_c = 0.20$) introduces instability in higher-dimensional problems. Overall, $D_c = 0.10$ delivers a good compromise between robustness and search efficiency.

Third, the cooperation probability C_o, which affects the stochastic crossover probability in the adversarial update phase, was examined. The results indicate that $C_o = 0.10$ produces consistent convergence performance, while a slightly higher value ($C_o = 0.20$) improves performance in certain multimodal cases (e.g., F9, F22) by increasing information sharing. Conversely, $C_o = 0.05$ slows convergence due to reduced cooperative interactions. Hence, $C_o = 0.10$ is considered the most reliable default.

Next, the effect of the diversity threshold $D_{threshold}$, which triggers the game-breaking mechanism, was investigated. When $D_{threshold} = 0.05$, the mechanism was rarely activated, causing stagnation in several functions. When $D_{threshold} = 0.15$, excessive restarts disrupted convergence stability. The optimal balance was achieved at $D_{threshold} = 0.10$, corresponding to approximately 35 % of the normalized initial diversity, ensuring timely diversity restoration without unnecessary restarts.

Finally, the stagnation limit T_{stag}, which defines the maximum number of consecutive iterations without global improvement before triggering reinitialization, was analyzed. AGOA maintained relatively stable performance when T_{stag} was set to 50, 100, or 200. However, $T_{stag} = 100$ provided the best trade-off between convergence efficiency and restart frequency, avoiding both excessive restarts and prolonged stagnation.

In summary, AGOA demonstrates strong robustness to parameter variation. The sensitivity analysis confirms that the default configuration $(E_r, D_c, C_o, D_{threshold}, T_{stag}) = (0.10, 0.10, 0.10, 0.10, 100)$ achieves the most stable and efficient optimization performance. These settings are therefore adopted for all subsequent experiments.

3.4. Analysis of results on CEC 2017

The CEC 2017 benchmark suite is widely recognized as a comprehensive and challenging platform for assessing the performance of metaheuristic algorithms. It comprises 30 test functions, systematically categorized into unimodal, multimodal, hybrid, and composite types, each posing distinct challenges for optimization methods. The suite features complex fitness landscapes characterized by numerous local optima, irregular basins of attraction, variable non-separability, and diverse rotation and shift transformations. Such properties enable the benchmark to rigorously evaluate both the exploitation capability of algorithms on simple landscapes and their exploration ability, scalability, and robustness in complex, high-dimensional, and non-convex search spaces. Accordingly, strong performance on the CEC 2017 suite is generally considered a reliable indicator of an algorithm's potential efficacy in addressing real-world complex optimization tasks.

3.4.1. Convergence results and analysis

For assessing the global search efficiency and convergence of AGOA, representative benchmark functions were selected from four typical categories: unimodal (F1), multimodal (F6), hybrid (F19), and composite (F30). These functions were tested under three dimensional settings ($D = 30$, 50, and 100). Detailed convergence results for $D = 30$ are presented in Table 5, while the corresponding convergence trajectories are depicted in Fig. 3, providing an intuitive illustration of convergence speed, search dynamics, and the ability to escape local optima.

The results clearly indicate that AGOA consistently outperforms competing algorithms across most functions and dimensional settings. For the unimodal function F1, AGOA exhibits a remarkably rapid convergence rate, with its curve maintaining a dominant position throughout the iterations regardless of problem dimensionality. This confirms AGOA's strong exploitation capability and its ability to achieve high-precision solutions with minimal computational effort.

For the multimodal function F6, which poses a greater challenge due to the abundance of local optima, AGOA demonstrates evident superiority as the dimensionality increases. At $D = 30$, its performance is already competitive, while at $D = 50$ and 100, AGOA's curve gradually surpasses those of most competing algorithms. This highlights its effective global exploration mechanism and resilience against premature convergence, allowing it to escape deceptive local optima.

In the case of the hybrid function F19, AGOA converges more quickly and achieves superior solution precision compared to competing methods, underscoring its adaptability to problems that integrate diverse landscape characteristics. A similar pattern is observed for the complex composite function F30, where AGOA consistently delivers excellent convergence performance. Notably, even at $D = 100$, AGOA maintains a favorable convergence trajectory and rapidly approaches the global optimum, demonstrating its scalability and robustness under high-dimensional conditions.

Therefore, the convergence analyses confirm that AGOA sustains an effective trade-off between global and local search across different categories of benchmark functions. Its ability to sustain convergence speed, solution quality, and robustness under increasing problem dimensionality further validates its effectiveness and practical potential for tackling high-dimensional and computationally challenging optimization scenarios.

3.4.2. Statistical results and analysis

To rigorously evaluate the distributional properties, robustness, and consistency of algorithmic performance, statistical analyses were conducted using box-and-whisker plots in combination with nonparametric significance tests. Boxplots provide an effective means of visualizing central tendency, variability, and the presence of outliers across multiple independent runs, while the Wilcoxon signed-rank test and the Friedman test evaluated whether performance differences between algorithms were statistically significant. Together, these approaches offer both descriptive and inferential perspectives on the comparative behavior of the tested methods.

In Fig. 4, boxplots are presented for representative benchmark functions F1, F6, F19, and F30 under dimensional settings of $D = 30$, 50, and 100. Each boxplot summarizes the results of 31 independent runs, enabling a direct assessment of solution stability and distributional consistency across problem scales. As shown, AGOA exhibits consistently high accuracy and robustness: its boxplots feature very narrow interquartile ranges and near-zero median errors, indicating that the algorithm achieves superior solutions with minimal variability across independent runs. This consistency reflects AGOA's ability to maintain precision and stability despite the stochasticity inherent in metaheuristic search processes.

In contrast, most competing algorithms display wider boxes, longer whiskers, and more frequent outliers, revealing higher variance, reduced stability, and greater sensitivity to random fluctuations. These shortcomings are especially evident for multimodal and composite functions, where maintaining reliability and escaping local optima are particularly challenging in high-dimensional search spaces. The outcomes of the Wilcoxon and Friedman tests further confirm that AGOA's performance improvements over competing algorithms are statistically significant at conventional confidence levels.

In Table 6, these statistical analyses validate the superiority of AGOA regarding the accuracy of optimization and solution stability. The algorithm demonstrates strong robustness against stochastic variation while maintaining consistent performance across different problem types and dimensional scales. Consequently, AGOA shows substantial competitiveness in addressing large-scale and complex optimization tasks, where reliability, robustness, and consistency of solutions are essential for practical applicability.

Table 5
Results for the top ten algorithms on CEC 2017 benchmark functions at D = 30.

Func	Item	GO	GSK	JADE	KOA	MPA	NOA	OMA	SaDE	WFO	AGOA
F1	mean	1.89 E-23	1.01 E-21	1.91 E-25	1.82 E-21	6.25 E+03	7.11 E-06	2.28 E-21	8.64 E-20	1.32 E-22	1.74 E-22
	std	2.84 E-23	7.36 E-22	4.27 E-25	7.25 E-22	6.45 E+03	5.61 E-06	3.09 E-21	1.77 E-19	1.33 E-22	3.89 E-22
	h	≈	+	≈	+	+	+	+	≈	≈	≈
F2	mean	4.21 E-09	1.34 E-07	3.44 E+01	6.56 E-08	4.63 E+03	4.99 E+02	9.81 E-12	2.81 E-06	9.81 E-12	1.66 E-07
	std	7.20 E-09	2.19 E-07	4.91 E+01	9.61 E-08	7.88 E+03	1.07 E+03	1.97 E-11	5.40 E-06	1.97 E-11	2.69 E-07
	h	≈	≈	≈	≈	+	+	≈	≈	≈	≈
F3	mean	2.17 E-28	2.75 E-27	2.19 E+03	3.05 E-06	1.39 E-02	9.49 E-07	8.49 E-04	7.82 E-18	1.26 E-02	4.08 E-12
	std	1.38 E-28	1.03 E-27	4.90 E+03	6.30 E-06	2.06 E-02	8.94 E-07	5.73 E-04	1.71 E-17	2.32 E-02	9.12 E-12
	h	≈	≈	≈	+	+	+	+	≈	+	≈
F4	mean	5.23 E-17	2.03 E-16	6.59 E-09	2.01 E-15	1.93 E-14	2.88 E-13	3.24 E-11	1.44 E-15	8.21 E-16	1.38 E-16
	std	1.04 E-16	3.29 E-16	8.45 E-09	4.43 E-15	3.47 E-14	7.02 E-13	6.77 E-11	3.24 E-15	1.63 E-15	3.49 E-16
	h	≈	+	≈	+	+	+	+	+	≈	≈
F5	mean	2.08 E-24	2.89 E-23	6.35 E-05	4.91 E-07	1.16 E-06	4.92 E-07	4.01 E-07	8.19 E-24	9.74 E-07	7.23 E-10
	std	4.91 E-24	1.00 E-23	1.24 E-04	1.12 E-06	1.83 E-06	1.02 E-06	8.74 E-07	2.00 E-23	1.82 E-06	1.65 E-09
	h	≈	≈	≈	+	+	≈	+	≈	+	≈
F6	mean	5.53 E-03	4.31 E-02	0.00 E+00	2.62 E-03	5.30 E-01	1.88 E-03	1.25 E-06	2.74 E-08	3.96 E-06	**0.00 E+00**
	std	5.01 E-03	9.64 E-02	0.00 E+00	1.85 E-03	4.11 E-01	1.11 E-03	1.82 E-06	6.12 E-08	6.26 E-06	0.00 E+00
	h	+	+	≈	+	+	+	+	≈	+	≈
F7	mean	5.73 E+01	1.37 E+02	5.79 E+01	1.28 E+02	9.96 E+01	9.11 E+01	8.88 E+01	7.93 E+01	8.47 E+01	**3.01 E+01**
	std	6.13 E+00	6.88 E+01	1.19 E+01	1.09 E+01	9.31 E+00	1.84 E+01	5.00 E+00	1.60 E+01	4.83 E+00	4.12 E+01
	h	≈	≈	≈	+	+	+	+	+	+	≈
F8	mean	2.57 E+01	3.18 E+01	2.69 E+01	7.32 E+01	7.34 E+01	5.73 E+01	5.03 E+01	3.42 E+01	5.77 E+01	2.69 E+01
	std	6.03 E+00	9.92 E+00	5.89 E+00	1.75 E+01	1.75 E+01	4.43 E+00	6.36 E+00	6.20 E+00	1.46 E+01	2.46 E+01
	h	≈	≈	≈	+	+	+	≈	≈	≈	≈
F9	mean	1.92 E+00	4.96 E+00	8.70 E-01	4.41 E+00	5.38 E+01	1.40 E+00	6.12 E+01	3.06 E+01	5.28 E+01	**6.80 E-01**
	std	1.01 E+00	1.04 E+01	1.07 E+00	2.62 E+00	5.62 E+01	1.64 E+00	5.77 E+01	6.62 E+01	3.88 E+01	1.52 E+00
	h	≈	≈	≈	+	+	≈	+	≈	+	≈
F10	mean	2.74 E+03	6.75 E+03	2.50 E+03	4.75 E+03	2.53 E+03	2.76 E+03	1.93 E+03	2.01 E+03	2.21 E+03	**9.66 E+02**
	std	1.58 E+03	3.36 E+02	9.68 E+01	2.63 E+02	2.25 E+02	2.40 E+02	2.94 E+02	8.14 E+02	1.79 E+02	1.33 E+03
	h	+	+	+	+	+	+	≈	≈	≈	≈
F11	mean	2.35 E+01	1.26 E+01	5.86 E+01	2.82 E+01	4.38 E+01	2.71 E+01	1.47 E+01	5.27 E+01	1.91 E+01	1.93 E+01
	std	2.40 E+01	4.19 E+00	3.27 E+01	1.08 E+01	1.42 E+01	1.05 E+01	1.18 E+01	1.67 E+01	8.25 E+00	2.67 E+01
	h	≈	≈	≈	≈	≈	≈	≈	+	≈	≈
F12	mean	1.23 E+03	5.02 E+03	4.18 E+03	4.75 E+03	1.44 E+04	1.94 E+04	1.68 E+03	7.30 E+03	4.25 E+02	5.03 E+03
	std	1.68 E+02	4.06 E+03	2.52 E+03	4.44 E+03	1.05 E+04	1.28 E+04	2.96 E+03	2.90 E+03	1.85 E+02	7.21 E+03
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F13	mean	1.12 E+02	5.37 E+02	1.43 E+02	1.70 E+03	9.82 E+01	9.16 E+01	2.70 E+01	4.72 E+03	2.48 E+01	7.10 E+01
	std	4.99 E+01	9.10 E+02	1.65 E+02	3.05 E+03	4.36 E+01	2.81 E+01	3.64 E+00	8.70 E+03	2.68 E+00	6.78 E+01
	h	≈	≈	≈	≈	≈	≈	≈	+	≈	≈
F14	mean	3.42 E+01	3.21 E+01	1.24 E+04	4.74 E+01	2.86 E+01	1.97 E+01	2.39 E+01	6.97 E+01	2.87 E+01	3.47 E+01
	std	1.04 E+01	5.39 E+00	9.96 E+03	1.06 E+01	2.14 E+00	1.06 E+01	9.44 E+00	8.18 E+00	4.01 E+00	5.01 E+01
	h	≈	≈	+	≈	≈	≈	≈	≈	≈	≈
F15	mean	1.21 E+01	1.05 E+01	2.03 E+02	2.70 E+01	2.51 E+01	1.73 E+01	1.31 E+01	9.08 E+01	1.14 E+01	3.99 E+01
	std	3.70 E+00	4.77 E+00	3.22 E+02	2.56 E+01	7.49 E+00	4.58 E+00	7.44 E+00	4.07 E+01	2.41 E+00	2.50 E+01
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F16	mean	2.23 E+02	5.06 E+02	5.24 E+02	4.55 E+02	4.35 E+02	2.64 E+02	3.77 E+02	5.34 E+02	5.07 E+02	2.86 E+02
	std	1.68 E+02	4.43 E+02	1.13 E+02	1.58 E+02	1.86 E+02	8.35 E+01	1.18 E+02	2.34 E+02	1.03 E+02	2.89 E+02
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F17	mean	9.88 E+01	9.49 E+01	8.68 E+01	8.09 E+01	6.71 E+01	4.37 E+01	9.52 E+01	2.76 E+01	1.56 E+02	5.33 E+01
	std	5.85 E+01	7.57 E+01	4.31 E+01	6.36 E+01	1.20 E+01	1.35 E+01	5.73 E+01	1.21 E+01	1.11 E+02	7.70 E+01
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F18	mean	3.80 E+01	1.92 E+02	2.97 E+02	6.33 E+02	3.03 E+01	2.88 E+01	2.11 E+01	1.99 E+03	2.73 E+01	1.75 E+03
	std	6.16 E+00	9.77 E+01	2.83 E+02	3.63 E+02	4.44 E+00	3.70 E+00	1.12 E+01	2.44 E+03	8.88 E+00	2.50 E+03
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F19	mean	1.50 E+01	1.13 E+01	2.90 E+02	2.01 E+01	1.18 E+01	1.32 E+01	1.11 E+01	6.86 E+01	1.22 E+01	2.36 E+01
	std	1.72 E+00	3.47 E+00	5.18 E+02	1.04 E+00	3.07 E+00	4.13 E+00	4.43 E+00	1.14 E+01	4.49 E+00	2.39 E+01
	h	≈	≈	≈	≈	≈	≈	≈	+	≈	≈
F20	mean	1.01 E+02	7.07 E+01	1.05 E+02	8.79 E+01	1.30 E+02	9.40 E+01	1.08 E+02	8.45 E+01	1.33 E+02	**5.91 E+01**
	std	8.38 E+01	6.12 E+01	6.70 E+01	6.24 E+01	6.53 E+01	6.02 E+01	6.87 E+01	5.90 E+01	7.10 E+01	6.62 E+01
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F21	mean	2.27 E+02	2.26 E+02	2.32 E+02	2.75 E+02	2.52 E+02	2.48 E+02	2.53 E+02	2.51 E+02	2.56 E+02	**9.77 E+01**
	std	5.97 E+00	7.53 E+00	8.52 E+00	2.14 E+01	7.73 E+00	5.01 E+00	1.85 E+01	8.87 E+00	1.54 E+01	1.34 E+02
	h	≈	≈	≈	+	≈	≈	≈	≈	≈	≈
F22	mean	6.80 E+02	1.00 E+02	1.00 E+02	1.00 E+02	1.03 E+02	1.00 E+02	4.70 E+02	1.00 E+02	5.36 E+02	**6.05 E+01**
	std	1.30 E+03	1.10 E+00	0.00 E+00	1.10 E+00	1.94 E+00	1.10 E+00	8.27 E+02	0.00 E+00	9.76 E+02	5.52 E+01
	h	≈	≈	≈	≈	+	≈	≈	≈	≈	≈
F23	mean	3.83 E+02	3.65 E+02	3.83 E+02	4.16 E+02	3.90 E+02	3.94 E+02	4.01 E+02	3.96 E+02	4.08 E+02	**0.00 E+00**
	std	1.00 E+01	8.47 E+00	8.58 E+00	2.84 E+01	1.85 E+01	7.19 E+00	2.43 E+00	1.99 E+01	1.31 E+01	0.00 E+00
	h	+	+	+	+	+	+	+	+	+	≈

(*continued on next page*)

Table 5
(*continued*)

Func	Item	GO	GSK	JADE	KOA	MPA	NOA	OMA	SaDE	WFO	AGOA
F24	mean	4.57 E+02	4.34 E+02	4.47 E+02	4.87 E+02	4.57 E+02	4.64 E+02	5.16 E+02	4.74 E+02	5.32 E+02	**0.00 E+00**
	std	7.32 E+00	3.69 E+00	8.75 E+00	2.80 E+01	1.45 E+01	2.12 E+00	2.93 E+01	1.17 E+01	1.32 E+01	0.00 E+00
	h	+	+	+	+	+	+	+	+	+	≈
F25	mean	3.86 E+02	4.07 E+02	3.89 E+02	3.86 E+02	3.86 E+02	3.84 E+02	3.86 E+02	3.88 E+02	3.86 E+02	**1.54 E+02**
	std	1.86 E+00	2.98 E+01	3.91 E+00	3.61 E+00	1.63 E+00	1.53 E+00	1.49 E+00	5.65 E+00	1.56 E+00	2.11 E+02
	h	≈	+	≈	≈	≈	≈	≈	≈	≈	≈
F26	mean	1.37 E+03	8.26 E+02	1.24 E+03	1.59 E+03	3.00 E+02	1.33 E+03	1.69 E+03	1.51 E+03	6.26 E+02	7.86 E+02
	std	1.56 E+02	5.28 E+02	5.43 E+02	2.49 E+02	2.50 E-03	5.94 E+02	3.09 E+02	8.12 E+02	7.87 E+02	7.34 E+02
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F27	mean	5.08 E+02	5.05 E+02	5.16 E+02	5.10 E+02	5.03 E+02	4.89 E+02	5.11 E+02	5.22 E+02	5.08 E+02	4.00 E+02
	std	6.68 E+00	1.67 E+01	2.71 E+00	1.70 E+01	7.25 E+00	1.11 E+01	1.09 E+01	1.39 E+01	1.04 E+01	2.24 E+02
	h	≈	≈	+	≈	≈	≈	≈	+	≈	≈
F28	mean	3.56 E+02	3.32 E+02	3.46 E+02	3.69 E+02	3.38 E+02	3.21 E+02	3.00 E+02	3.21 E+02	3.00 E+02	**2.60 E+02**
	std	7.91 E+01	7.16 E+01	6.24 E+01	6.51 E+01	5.33 E+01	4.62 E+01	8.04 E-14	4.62 E+01	1.29 E-11	2.51 E+02
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F29	mean	5.20 E+02	4.78 E+02	4.96 E+02	6.16 E+02	5.27 E+02	4.96 E+02	5.41 E+02	4.83 E+02	5.51 E+02	4.86 E+02
	std	7.76 E+01	4.16 E+01	2.78 E+01	6.62 E+01	6.89 E+01	4.92 E+01	1.13 E+02	7.00 E+01	1.14 E+02	8.29 E+01
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F30	mean	2.05 E+03	2.02 E+03	2.87 E+03	2.49 E+03	2.09 E+03	2.14 E+03	2.09 E+03	2.63 E+03	2.03 E+03	**9.72 E+01**
	std	9.83 E+01	4.62 E+01	4.43 E+02	4.39 E+02	9.60 E+01	9.51 E+01	9.83 E+01	5.35 E+02	7.75 E+01	1.33 E+02
	h	+	+	+	+	+	+	+	+	+	≈
median F		10.14	12.85	13.76	18.44	18.23	12.42	12.9	14.1	12.7	**6.89**

Table 7 presents the summarized results in terms of "+/=/-" indicators, representing cases where the competing algorithm outperforms AGOA (+), performs equivalently (=), or is outperformed by AGOA (-). The results clearly demonstrate that AGOA achieves a higher number of statistically significant wins ("-") across all tested dimensions. For example, algorithms such as EFO and NOA exhibit frequent failures in direct comparisons with AGOA, confirming AGOA's dominant global search capability and strong local exploitation even in complex, high-dimensional problem spaces.

In summary, the combined evidence from both the Friedman test and Wilcoxon signed-rank test provides a robust and statistically sound foundation supporting AGOA's outstanding performance. These results not only confirm AGOA's superiority in lower-dimensional optimization tasks but also validate its exceptional scalability and effectiveness in tackling challenging high-dimensional, multimodal, and highly nonlinear problems. Consequently, AGOA exhibits strong potential for real-world applications requiring efficient and reliable global optimization.

Although AGOA achieves consistently strong overall performance across most benchmark functions, it does not always attain the absolute best results on certain relatively smooth or unimodal problems, where the search landscape is simple and the benefit of adversarial diversity mechanisms is less pronounced. This observation suggests that while AGOA excels in complex, multimodal, and irregular landscapes, future studies could further refine its early-stage exploitation strategies to improve efficiency on simpler or low-dimensional tasks.

3.5. Analysis of results on CEC 2022

To further assess the effectiveness, generalization capability, and scalability of AGOA, comprehensive experiments were conducted on the CEC 2022 benchmark suite. Compared to CEC 2017, the CEC 2022 set comprises 12 functions of increased complexity and more challenging landscapes, characterized by strong nonlinearity, intricate variable interactions, pronounced multimodality, and sophisticated hybrid compositions. Consequently, the CEC 2022 suite provides a rigorous and controlled environment for assessing and comparing performance, convergence behavior, and stability of newly developed optimization algorithms.

Moreover, the CEC 2022 benchmark restricts the maximum problem dimensionality to 20 dimensions. This limitation facilitates a more detailed analysis of the problem landscapes and emphasizes algorithmic robustness and global search capability over pure scalability.

3.5.1. Convergence results and analysis

In this subsection, representative functions F1 and F6 were selected for detailed convergence analysis under dimensions D = 10 and D = 20. The convergence curves as well as the convergence results at CEC 2022 (D = 10) are displayed in Table 8 and Fig. 5. The AGOA algorithm performs robustly on the CEC 2022 benchmark, achieving near-optimal means on multiple test functions with small standard deviations, indicating high stability of its solutions. Its overall performance is among the top, second only to a few better methods. Overall, the AGOA algorithm is competitive and stable in optimization problems.

From the convergence trajectories, AGOA consistently demonstrates rapid convergence rates and superior solution accuracy across both functions and dimensions. In particular, AGOA exhibits a strong tendency to escape local optima efficiently and maintain robust global search behavior throughout the optimization process. It achieves near-optimal mean fitness values with relatively small standard deviations, indicating high solution stability and repeatability.

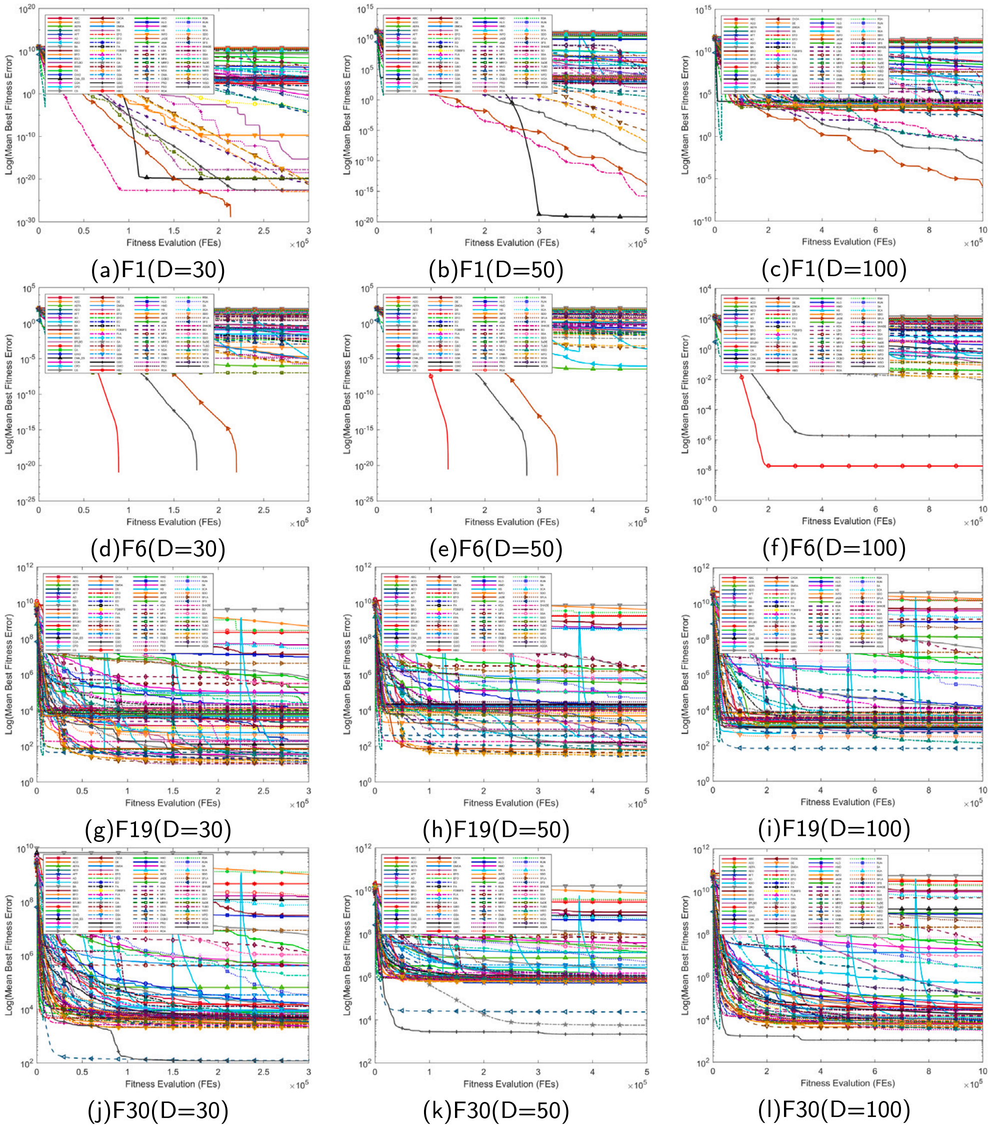


Fig. 3. Convergence curves of AGOA and compared algorithms on CEC 2017 benchmark with different dimensions.

Moreover, as reflected in Table 8, AGOA achieves competitive or superior performance in most test cases, with performance metrics often ranking among the top or close to the best-performing algorithms. For functions F1 and F6, AGOA maintains smooth and steeply declining convergence curves, indicating its efficiency in balancing exploration and exploitation. This further substantiates AGOA's capacity to handle complex multimodal and hybrid functions effectively, even in higher-dimensional spaces.

3.5.2. Statistical results and analysis

To provide a statistically rigorous evaluation, we employed both the Friedman test and the Wilcoxon signed-rank test, supplemented by boxplot visualizations (see Fig. 6).

The Friedman test, a non-parametric statistical test designed for detecting significant differences among multiple algorithms across multiple problems, was performed on the mean fitness values obtained from all algorithms. As shown in Table 9, the results reveal that AGOA consistently achieves the best average ranks (6.87 on D = 10 and 6.98 on D = 20), resulting in an overall mean rank of 6.93. This highlights AGOA's superior robustness and adaptability across different problem settings and dimensions. In contrast, many traditional evolutionary and swarm-based algorithms exhibit considerable rank degradation in functions with higher dimensionality and increased complexity, confirming AGOA's stronger scalability and convergence characteristics. The Wilcoxon signed-rank test was further utilized to conduct pairwise comparisons between AGOA and each of the top 10 competing algorithms (Table 10). This non-parametric test evaluates whether performance differences are statistically significant under the null hypothesis of equivalence.

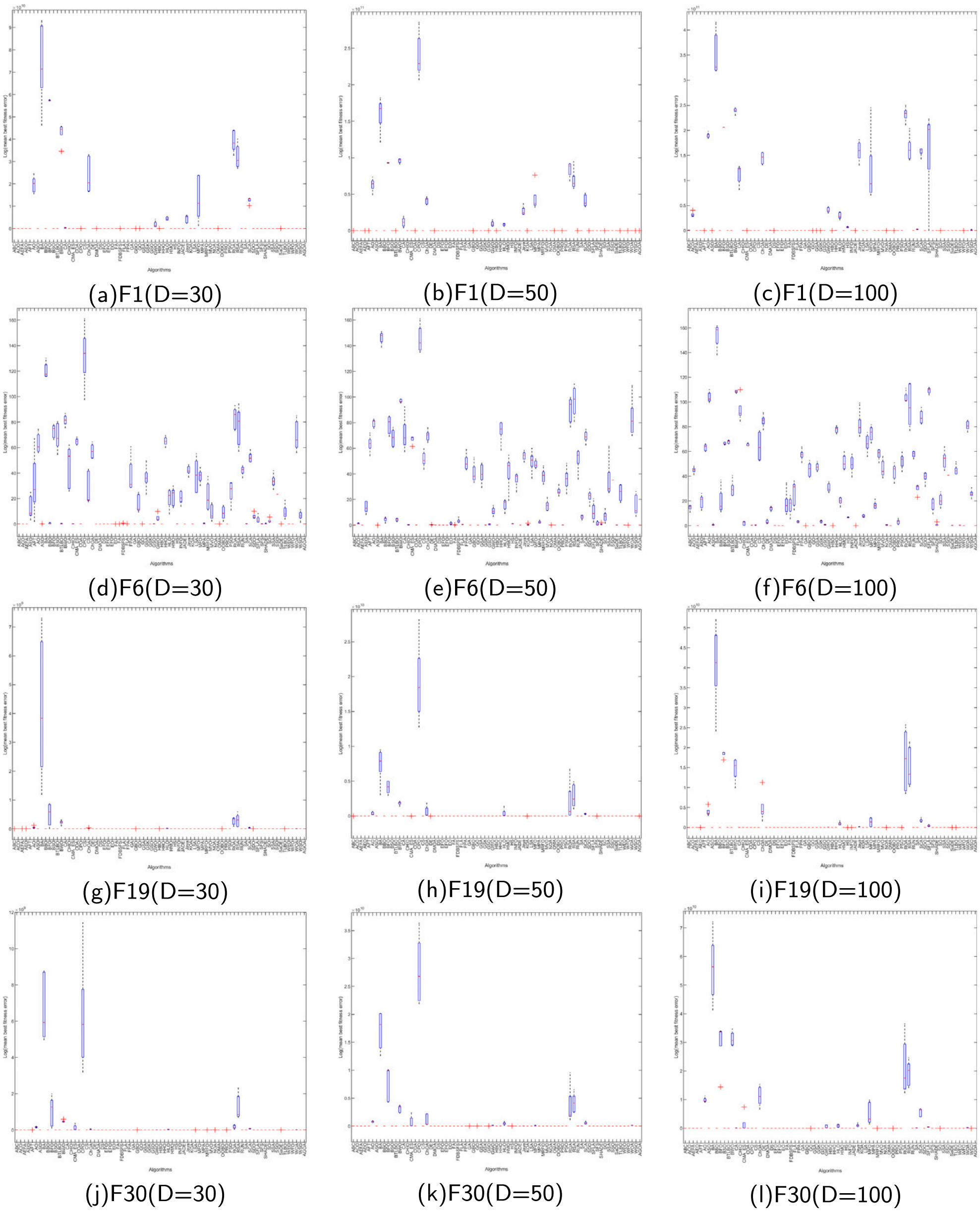


Fig. 4. Box plots of AGOA and comparative algorithms with different dimensions (D = 30, 50, 100).

Results are reported using the "+/=/-" notation, showing cases in which the compared algorithm is superior, equivalent, or inferior to AGOA.

The results indicate that AGOA achieves a higher number of statistically significant wins (marked by "-") against most competing algorithms, especially under D = 20, where AGOA shows strong dominance. For instance, algorithms such as NOA and KOA exhibit noticeably more losses than wins when compared to AGOA, reaffirming its superior optimization capacity and stability in complex scenarios.

The boxplots in Fig. 6 further reinforce these conclusions, clearly illustrating AGOA's minimal performance variability and consistently superior solution quality. The narrow interquartile ranges and lower median values demonstrate excellent robustness, strong convergence stability, and high reliability across independent runs. Moreover, the absence of significant outliers and compact box shapes highlights AGOA's resilience against stochastic perturbations and its ability to maintain precision under varying initial conditions and complex landscapes.

The experimental results on the CEC 2022 suite rigorously validate AGOA's strong global optimization capability, excellent adaptability to high-dimensional and multimodal problems, and effective trade-off between global and local search. The findings demonstrate AGOA's capability in handling complex, high-accuracy, and robust optimization tasks, although it does not outperform

Table 6
Friedman test ranking results of AGOA and comparative algorithms on the CEC 2017 benchmark across different dimensions.

id	Algorithms	D = 30	D = 50	D = 100	Mean rank
1	**AGOA**	**6.89**	**6.82**	**8.78**	**7.50**
2	GO	10.14	8.30	9.26	9.23
3	JADE	13.76	12.79	11.94	12.83
4	WFO	12.70	14.94	16.85	14.83
5	NOA	12.42	16.20	16.61	15.08
6	OMA	12.9	15.57	16.77	15.08
7	GSK	12.85	15.65	17.54	15.35
8	SaDE	14.10	17.85	18.84	16.93
9	SHADE	18.00	16.80	17.35	17.38
10	WEO	23.31	19.37	16.70	19.79
11	EFO	22.34	19.44	17.98	19.92
12	KOA	18.44	21.63	20.04	20.04
13	MPA	18.23	18.69	26.93	21.28
14	DE	20.51	24.18	22.51	22.40
15	PRO	21.79	23.98	23.22	23.00
16	SFS	20.62	25.42	26.68	24.24
17	ASO	24.34	24.72	24.19	24.42
18	AEFA	30.28	24.65	20.39	25.11
19	CPO	28.96	27.22	25.49	27.22
20	COA	29.27	29.88	24.86	28.00
21	BTLBO	27.26	27.34	30.01	28.20
22	SDO	28.30	31.05	28.44	29.26
23	FDBSFS	26.37	30.30	31.31	29.33
24	EO	30.61	30.49	27.55	29.55
25	FA	32.53	29.63	29.01	30.39
26	DS	29.12	32.29	34.17	31.86
27	OOBO	31.51	35.21	30.43	32.38
28	SO	35.54	33.11	30.65	33.10
29	SA	32.33	32.81	36.77	33.97
30	INFO	35.54	37.75	33.69	35.66
31	MVO	39.95	36.09	31.09	35.71
32	GBO	38.00	36.92	32.54	35.82
33	TLBO	35.44	36.45	38.39	36.76
34	AEO	38.81	37.72	34.19	36.91
35	FLA	40.02	37.33	33.53	36.96
36	HBO	36.74	37.02	37.21	36.99
37	FPA	35.25	38.85	38.11	37.40
38	CS	34.17	37.31	41.11	37.53
39	PSO	39.09	37.22	36.60	37.64
40	HMO	38.40	37.96	36.88	37.75
41	MRFO	39.56	40.28	33.61	37.82
42	CHIO	39.43	36.35	38.49	38.09
43	BBO	39.96	37.49	37.00	38.15
44	AFT	40.86	37.83	37.36	38.68
45	GA	44.36	39.39	33.99	39.25
46	SSA	44.72	39.47	36.82	40.34
47	RUN	40.74	41.93	38.72	40.46
48	HS	34.81	39.13	47.65	40.53
49	CMA _ ES	40.81	39.31	41.53	40.55
50	GSO	43.59	42.92	37.40	41.30
51	WSO	40.81	42.62	42.34	41.92
52	SFLA	30.44	29.13	66.34	41.97
53	SSO	42.83	43.18	42.55	42.85
54	GSA	47.11	43.89	37.61	42.87
55	GWO	47.49	46.08	48.57	47.38
56	LSA	49.15	47.99	45.00	47.38
57	DMOA	45.55	46.21	50.65	47.47
58	ABC	45.26	46.19	51.39	47.61
59	ACO	45.45	47.64	54.54	49.21
60	BSO	54.35	52.49	48.39	51.74
61	HLO	57.02	56.09	54.61	55.91
62	HHO	59.36	58.39	54.15	57.30
63	CA	56.69	57.63	62.38	58.90
64	MFO	59.71	60.87	60.50	60.36
65	Jaya	59.2	60.25	62.76	60.74
66	WOA	63.78	61.33	59.21	61.44

(*continued on next page*)

Table 6
(*continued*)

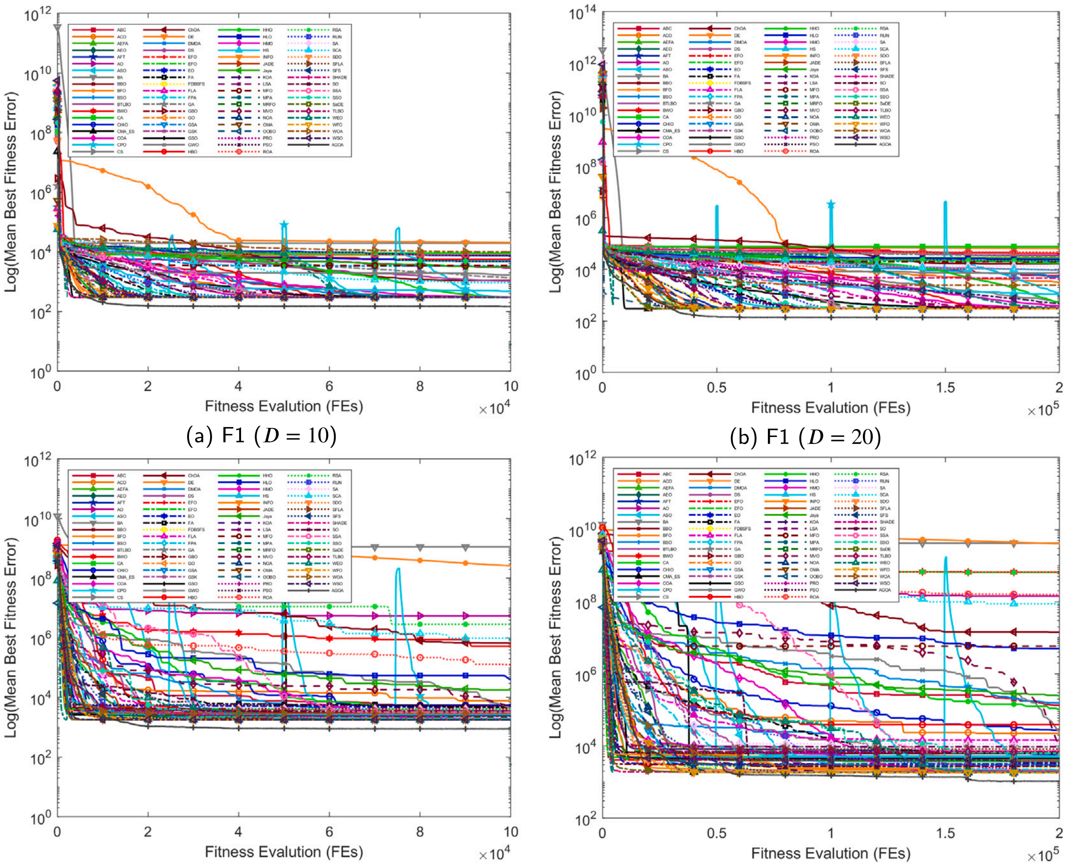
id	Algorithms	D = 30	D = 50	D = 100	Mean rank
67	SCA	65.59	66.56	66.77	66.31
68	ChOA	66.81	66.84	65.65	66.43
69	AO	66.88	68.62	69.20	68.23
70	RSA	70.68	70.02	67.83	69.51
71	BFO	71.75	69.77	67.95	69.82
72	BWO	70.36	70.02	71.26	70.55
73	ROA	70.87	71.19	70.34	70.80
74	BA	74.83	74.91	74.73	74.82

Table 7
Results of the Wilcoxon signed-rank test for the algorithms at CEC 2017.

vs.AGOA		D = 30	D = 50	D = 100
GO	+ / = /-	5/20/5	8/20/2	6/23/1
JADE	+ / = /-	6/21/3	8/15/7	6/14/10
WFO	+ / = /-	9/11/10	4/16/10	8/20/2
NOA	+ / = /-	9/11/10	7/15/8	4/13/13
OMA	+ / = /-	10/10/10	11/10/9	5/12/13
GSK	+ / = /-	8/12/10	7/10/13	9/11/10
SaDE	+ / = /-	9/11/10	8/10/12	2/8/10
SHADE	+ / = /-	9/8/13	8/15/7	4/12/14
WEO	+ / = /-	6/14/10	7/13/10	9/13/8
EFO	+ / = /-	3/7/20	5/10/15	2/8/20

Table 8
Results of CEC 2022 benchmark test functions of top ten ranked algorithms at D = 10.

Func	Item	DE	FDBSFS	GO	JADE	KOA	MPA	NOA	OMA	WFO	AGOA
F1	mean	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	3.00 E + 02	**1.20 E + 02**
	std	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	1.16 E-12	0.00 E + 00	0.00 E + 00	4.02 E-14	1.64 E + 02
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F2	mean	4.05 E + 02	4.02 E + 02	4.03 E + 02	4.05 E + 02	4.06 E + 02	4.00 E + 02	4.00 E + 02	4.01 E + 02	4.00 E + 02	**8.00 E + 01**
	std	2.08 E + 00	2.18 E + 00	3.94 E + 00	3.79 E + 00	4.05 E + 00	7.51 E-10	4.01 E-11	1.78 E + 00	7.06 E-13	1.79 E + 02
	h	+	+	+	+	+	+	≈	≈	≈	≈
F3	mean	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	6.00 E + 02	**2.40 E + 02**
	std	0.00 E + 00	0.00 E + 00	6.89 E-06	9.85 E-14	8.04 E-14	1.90 E-04	1.14 E-13	1.23 E-08	3.37 E-08	3.29 E + 02
	h	≈	≈	≈	≈	≈	+	+	+	+	≈
F4	mean	8.05 E + 02	8.14 E + 02	8.06 E + 02	8.05 E + 02	8.10 E + 02	8.07 E + 02	8.05 E + 02	8.09 E + 02	8.09 E + 02	**1.63 E + 02**
	std	2.90 E + 00	5.83 E + 00	1.57 E + 00	1.59 E + 00	4.66 E + 00	1.51 E + 00	2.31 E + 00	2.27 E + 00	3.89 E + 00	3.64 E + 02
	h	≈	+	≈	≈	≈	≈	≈	≈	≈	≈
F5	mean	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	9.00 E + 02	**1.80 E + 02**
	std	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	1.92 E-09	0.00 E + 00	5.68 E-14	9.02 E-11	4.02 E + 02
	h	+	+	+	+	+	≈	+	+	≈	≈
F6	mean	1.80 E + 03	1.81 E + 03	1.80 E + 03	1.80 E + 03	1.80 E + 03	1.80 E + 03	1.80 E + 03	1.80 E + 03	1.80 E + 03	**1.08 E + 03**
	std	7.81 E-01	7.63 E + 00	7.03 E-02	4.22 E-01	1.24 E-01	9.17 E-02	4.38 E-02	1.87 E-01	1.55 E-01	9.90 E + 02
	h	≈	≈	≈	≈	≈	≈	≈	≈	≈	≈
F7	mean	2.00 E + 03	2.00 E + 03	2.01 E + 03	2.00 E + 03	2.00 E + 03	2.01 E + 03	2.00 E + 03	2.00 E + 03	2.00 E + 03	**1.20 E + 03**
	std	8.88 E + 00	5.45 E-01	9.53 E + 00	7.86 E-02	8.79 E-01	9.80 E + 00	2.05 E-05	4.05 E-01	5.46 E-01	1.10 E + 03
	h	≈	≈	≈	≈	≈	+	≈	≈	≈	≈
F8	mean	2.20 E + 03	2.21 E + 03	2.20 E + 03	2.21 E + 03	2.21 E + 03	2.20 E + 03	2.20 E + 03	2.21 E + 03	2.20 E + 03	**4.41 E + 02**
	std	8.95 E + 00	9.12 E + 00	3.69 E-01	4.52 E + 00	7.30 E + 00	2.28 E + 00	5.32 E-01	8.45 E + 00	2.56 E + 00	9.86 E + 02
	h	≈	+	≈	+	+	≈	≈	≈	+	≈
F9	mean	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	2.53 E + 03	**9.94 E + 02**
	std	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	0.00 E + 00	1.36 E + 03
	h	+	+	+	+	+	+	+	+	+	≈
F10	mean	2.50 E + 03	2.50 E + 03	2.50 E + 03	2.50 E + 03	2.50 E + 03	2.50 E + 03	2.50 E + 03	2.42 E + 03	2.42 E + 03	**1.00 E + 03**
	std	2.86 E-02	1.45 E-01	3.36 E-02	4.60 E-02	3.37 E-02	3.27 E-02	4.71 E-02	4.46 E + 01	4.40 E + 01	1.37 E + 03
	h	≈	+	≈	+	+	≈	+	≈	≈	≈
F11	mean	2.69 E + 03	2.60 E + 03	2.72 E + 03	2.60 E + 03	2.60 E + 03	2.60 E + 03	2.60 E + 03	2.60 E + 03	2.60 E + 03	**1.04 E + 03**
	std	1.34 E + 02	0.00 E + 00	1.26 E + 02	2.27 E-13	3.22 E-13	5.11 E-06	0.00 E + 00	5.08 E-13	6.43 E-13	1.42 E + 03
	h	≈	≈	≈	≈	≈	+	≈	+	+	≈
F12	mean	2.86 E + 03	2.87 E + 03	2.86 E + 03	2.86 E + 03	2.86 E + 03	2.86 E + 03	2.86 E + 03	2.86 E + 03	2.86 E + 03	**0.00 E + 00**
	std	7.84 E-01	1.74 E + 00	1.60 E + 00	9.23 E-01	2.21 E + 00	3.36 E-01	2.08 E + 00	2.17 E + 00	1.18 E + 00	0.00 E + 00
	h	+	+	+	+	+	+	+	+	+	≈
median F		18.5	18.82	17.32	18.05	19.48	20.43	14.21	15.82	16.18	**6.98**



(a) F1 ($D = 10$)
(b) F1 ($D = 20$)
(c) F6 ($D = 10$)
(d) F6 ($D = 20$)

Fig. 5. Convergence curves of AGOA compared with other algorithms on representative CEC 2022 test functions F1 and F6 at dimensions $D = 10$ and $D = 20$.

all algorithms on every function. On less complex unimodal problems, AGOA shows only a small gap compared to baseline algorithms, whereas for multimodal and composite cases, AGOA's adversarial and diversity mechanisms perform better. Future work will focus on refining adaptive parameter control to improve efficiency on low-complexity or rotation-sensitive functions.

3.6. Comparative study with recent algorithms (2024–2025)

For additional evaluation of the proposed AGOA's speed and competitiveness, additional comparative experiments were conducted against several representative metaheuristic algorithms proposed in recent years, including the football team training algorithm (FTTA) [42], love evolution algorithm (LEA) [43], exploration-exploitation-based optimizer (E2) [44], cloud drift optimization (CDO) [45], and advanced social memory optimization (ASMO) [46].

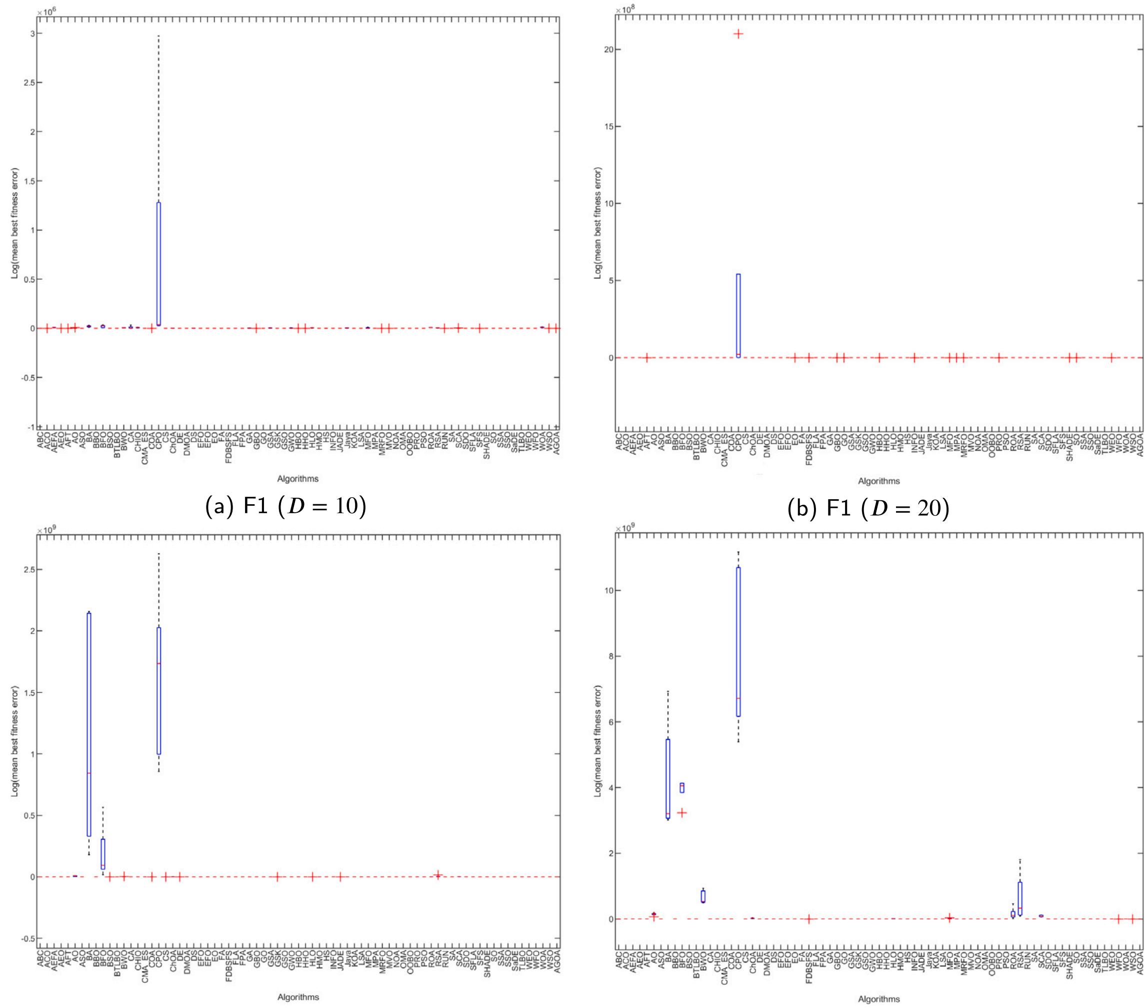
These algorithms were selected because they collectively represent the latest development trends in the field of intelligent optimization, featuring distinct design philosophies such as cooperative learning (FTTA), emotion-inspired decision making (LEA), exploration–exploitation balance (E2), cloud-model-based uncertainty modeling (CDO), and socially reinforced memory mechanisms (ASMO). Each of them has demonstrated strong performance on various benchmark suites in the literature, making them suitable and competitive baselines for validating AGOA's robustness and scalability.

In this section, all algorithms were tested under identical experimental settings on the CEC 2017 (30D, 50D, 100D) and CEC 2022 (10D, 20D). The goal is to assess whether AGOA can maintain superior convergence speed, accuracy, and stability compared to contemporary state-of-the-art algorithms.

3.6.1. Analysis of results on CEC 2017

The comparative analysis results on the CEC 2017 are presented to assess AGOA's performance relative to the newly introduced algorithms. Fig. 7 illustrate the convergence behaviors across three dimensional settings (30D, 50D, and 100D).

As shown in Fig. 7, AGOA exhibits a faster convergence speed and lower fitness errors across all benchmark functions. For the unimodal function F1, AGOA rapidly approaches the global optimum in the early iterations, demonstrating its strong local exploitation capability. In the multimodal function F6, AGOA shows remarkable robustness and effectively avoids premature convergence. For the hybrid and composite functions (F19 and F30), the superiority of AGOA becomes even more evident. Although algorithms such



(a) F1 ($D = 10$)　(b) F1 ($D = 20$)

(c) F6 ($D = 10$)　(d) F6 ($D = 20$)

Fig. 6. Box plots of AGOA compared with other algorithms on representative CEC 2022 benchmark functions F1 and F6 at dimensions $D = 10$ and $D = 20$.

as FTTA and LEA display certain exploration abilities in the early stages, they tend to stagnate later, whereas AGOA continues to improve its performance through its adaptive adversarial mechanism and diversity-preserving learning process. The convergence curves further confirm that AGOA maintains high search efficiency even in complex and high-dimensional landscapes.

Overall, these results clearly verify the significant advantages of AGOA in achieving a well-balanced exploration–exploitation trade-off, high convergence accuracy, and strong global optimization capability.

As illustrated in Fig. 8, the boxplots compare AGOA with other advanced algorithms on the CEC 2017 benchmark functions. AGOA achieves both lower best fitness values and noticeably narrower box ranges, indicating smaller variance and higher result stability. For instance, on unimodal function F1 and multimodal function F6, AGOA's results are highly concentrated with almost no outliers, reflecting its strong robustness and insensitivity to random initialization. By contrast, algorithms like E2 and LEA exhibit greater fluctuations, suggesting less stable convergence behavior. For hybrid and composite functions (F19 and F30), AGOA's median performance is clearly superior, showing its ability to maintain consistency in complex, high-dimensional landscapes. Overall, the combined evidence from the convergence curves and boxplot analysis confirms that AGOA achieves both fast convergence and high stability, maintaining superior global search performance across diverse problem types.

As shown in the convergence curves and statistical boxplots, AGOA exhibits a stable and efficient optimization behavior on the CEC 2017 benchmark suite. In early iterations, the algorithm reduces fitness values swiftly and then improves steadily in later stages, reflecting a well-balanced search process. The boxplot analysis further indicates that AGOA produces relatively concentrated distributions with smaller variances, suggesting good robustness and consistency across multiple runs. Compared with several recently proposed algorithms, AGOA generally achieves competitive or superior performance on most benchmark categories. In particular, it tends to converge faster and reach lower mean fitness errors on unimodal, multimodal, hybrid, and composite functions, demonstrating its adaptability to problems of varying complexity. The findings indicate that AGOA maintains a proper exploration–exploitation balance and is well-suited for complex, high-dimensional optimization tasks.

Table 9
Friedman test rankings for AGOA and comparative algorithms at CEC 2022.

id	Algorithms	D = 10	D = 20	Mean rank
1	**AGOA**	6.87	6.98	6.93
2	SaDE	15.43	13.39	14.41
3	GO	13.82	17.32	15.57
4	NOA	17.77	14.21	15.99
5	GSK	16.45	17.38	16.92
6	JADE	16.25	18.05	17.15
7	OMA	19.73	15.82	17.78
8	WFO	19.78	16.18	17.98
9	DE	18.18	18.5	18.34
10	SFS	19.57	19.82	19.70
11	KOA	20.3	19.48	19.89
12	SHADE	20.14	22.42	21.28
15	PRO	22.92	21.68	22.30
16	HBO	24.63	22.58	23.61
17	BTLBO	22.23	25.86	24.05
18	SDO	27.38	22.07	24.73
19	COA	29.53	24.87	27.20
20	WEO	27.05	27.49	27.27
21	EFO	27.99	32.29	30.14
22	ASO	31.06	31.58	31.32
23	OOBO	32.33	30.64	31.49
24	EO	30.18	32.99	31.59
25	SFLA	30.19	33.29	31.74
26	DS	29.35	34.61	31.98
27	AFT	35.03	29.61	32.32
28	CS	35.68	30.71	33.20
29	ABC	34.2	32.79	33.50
30	ACO	33.45	34.84	34.15
31	DMOA	33.93	34.57	34.25
32	FA	30.15	38.72	34.44
33	TLBO	35.67	33.36	34.52
34	SA	31.89	39.25	35.57
35	SO	36.61	34.79	35.70
36	CPO	35.54	36.04	35.79
37	HS	33.43	39.27	36.35
38	WSO	41.07	31.74	36.41
39	INFO	38.43	37.13	37.78
40	AEFA	34.32	41.93	38.13
41	SSO	39.32	37.15	38.24
42	GBO	36.85	39.93	38.39
43	MRFO	39.45	37.69	38.57
44	AEO	40.03	37.57	38.80
45	PSO	42.97	35.76	39.37
46	FPA	41.02	37.83	39.43
47	CHIO	37.07	42.98	40.03
48	HMO	38.2	42.07	40.14
49	BBO	41.38	41.52	41.45
50	MVO	42.83	42.18	42.51
51	LSA	42.06	44.9	43.48
52	SSA	43.42	44.93	44.18
53	FLA	43.42	45.08	44.25
54	GA	42.95	46.35	44.65
55	GSO	45.02	47.35	46.19
56	BSO	46.75	45.89	46.32
57	RUN	47.22	48.43	47.83
58	GWO	49.1	47.25	48.18
59	GSA	49.44	49.33	49.39
60	CMA-ES	48	52.81	50.41
61	MFO	55.11	48.73	51.92
62	CA	53.45	50.47	51.96
63	HLO	58.05	56.1	57.08
64	Jaya	58.85	55.53	57.19
65	HHO	56.27	59.62	57.95
66	WOA	59.02	61.72	60.37
67	SCA	62.02	62.77	62.40
68	ChOA	66.3	63.85	65.08
69	AO	65.37	65.25	65.31
70	BWO	68.84	68.78	68.81
71	RSA	70.03	68.7	69.37
72	ROA	70.18	69.09	69.64
73	BFO	70.58	72.03	71.31
74	BA	74.75	74.48	74.62

Table 10
Results of the Wilcoxon signed-rank test for the algorithms at CEC 2022.

vs.AGOA		D = 10	D = 20
SaDE	+/=/-	5/4/3	7/3/2
GO	+/=/-	5/3/4	6/3/3
NOA	+/=/-	6/4/2	9/1/2
GSK	+/=/-	6/3/3	7/2/3
JaDE	+/=/-	6/4/2	7/2/3
OMA	+/=/-	6/4/2	6/3/3
WFO	+/=/-	6/3/3	7/4/1
DE	+/=/-	6/3/3	5/3/4
SFS	+/=/-	6/3/3	7/4/1
KOA	+/=/-	6/4/2	8/3/1

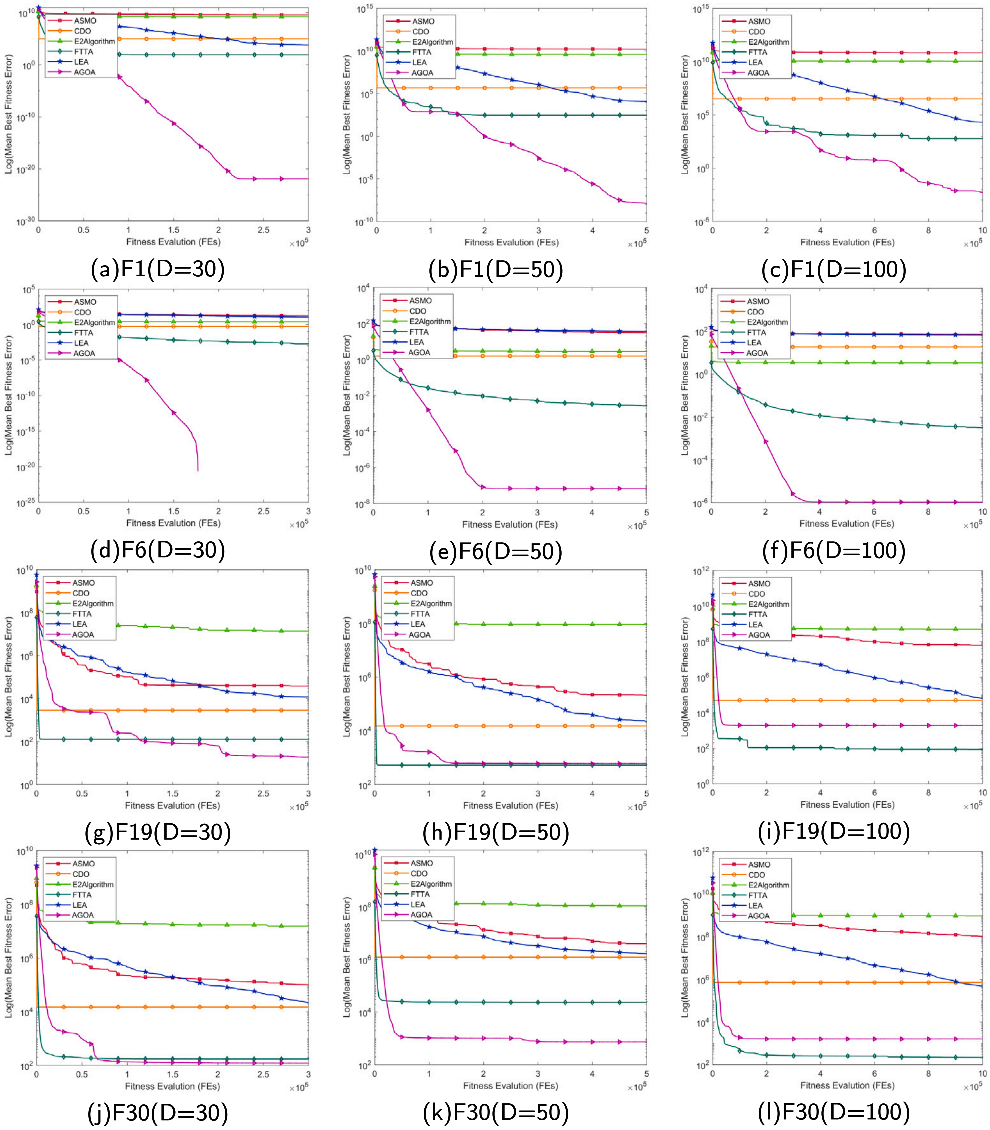


Fig. 7. Convergence curves of AGOA and recent algorithms on CEC 2017 benchmark with different dimensions.

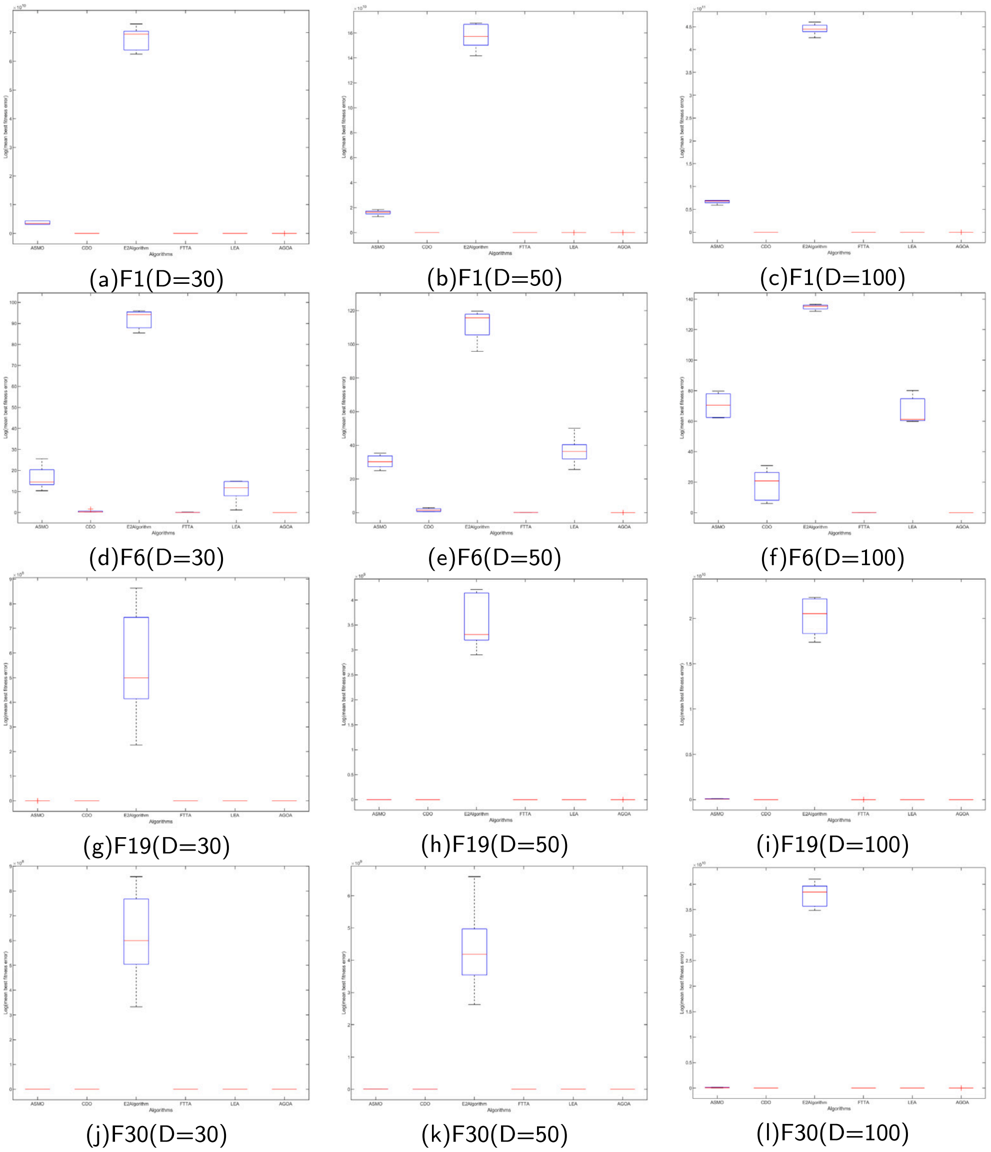


Fig. 8. Box plots of AGOA and recent algorithms with different dimensions.

3.6.2. Analysis of results on CEC 2022

Following the evaluations on the CEC 2017 benchmark, additional experiments were carried out on the more recent CEC 2022 benchmark suite to comprehensively assess AGOA's generalization ability and scalability. This section focuses on quantitative indicators such as the mean and standard deviation obtained across multiple runs. These metrics allow for a more rigorous numerical comparison and provide clearer insights into the stability and statistical reliability of each algorithm's performance.

As summarized in Table 11, AGOA achieves consistently competitive results. It tends to yield lower mean fitness errors and smaller standard deviations compared with most recent algorithms, indicating strong robustness and reliable convergence behavior. For hybrid and composition functions, where the landscape complexity increases significantly, AGOA still maintains effective optimization efficiency, suggesting that its adversarial learning dynamics and diversity-preserving mechanisms are beneficial in complex high-dimensional spaces. Overall, the results on CEC 2022 benchmarks further validate the scalability and robustness of AGOA under more challenging numerical optimization environments.

3.7. Comparison with DE and PSO

It is also worth emphasizing the comparison between AGOA and two representative classical algorithms, DE and PSO, based on the experimental results presented above. DE demonstrates relatively fast early convergence in some low-dimensional problems, benefiting from its mutation and crossover operators. However, it tends to stagnate or converge prematurely when facing

Table 11
Results of CEC 2022 benchmark test functions of recent algorithms at D = 20.

Func	Item	ASMO	CDO	E2	FTTA	LEA	AGOA
F1	mean	3.42 E + 02	3.00 E + 02	3.42 E + 02	3.00 E + 02	3.00 E + 02	**2.40 E + 02**
	std	7.40 E + 00	6.59 E-02	4.50 E + 00	1.85 E-02	1.88 E-06	1.34 E + 00
	h	+	+	+	+	+	≈
F2	mean	4.57 E + 02	4.62 E + 02	2.34 E + 02	4.43 E + 02	4.32 E + 02	**1.61 E + 02**
	std	2.68 E + 00	1.18 E + 00	4.64 E + 00	2.52 E + 00	2.42 E + 00	2.20 E + 00
	h	+	+	+	+	+	≈
F3	mean	6.06 E + 02	6.00 E + 02	6.74 E + 02	6.00 E + 02	6.01 E + 02	**2.40 E + 02**
	std	5.05 E + 00	2.89 E-02	4.71 E + 00	7.05 E + 00	2.37 E + 00	3.29 E + 00
	h	+	+	+	+	+	≈
F4	mean	9.09 E + 02	8.31 E + 02	9.90 E + 02	8.46 E + 02	8.98 E + 02	**6.72 E + 02**
	std	1.30 E + 00	7.62 E + 00	1.22 E + 00	8.19 E + 00	3.19 E + 00	3.76 E + 00
	h	+	≈	+	≈	+	≈
F5	mean	1.25 E + 03	9.04 E + 02	5.25 E + 03	1.59 E + 03	1.98 E + 03	**1.96 E + 02**
	std	1.39 E + 00	4.16 E + 00	6.63 E + 00	2.84 E + 00	1.04 E + 01	4.37 E + 00
	h	+	≈	+	+	+	≈
F6	mean	3.50 E + 03	3.67 E + 03	7.37 E + 03	5.27 E + 03	1.13 E + 03	**8.47 E + 02**
	std	4.02 E + 01	9.40 E + 00	2.59 E + 01	6.25 E + 01	6.45 E + 01	1.17 E + 00
	h	+	+	+	+	+	≈
F7	mean	2.11 E + 03	2.07 E + 03	2.21 E + 03	2.05 E + 03	2.05 E + 03	**1.62 E + 03**
	std	6.31 E + 00	5.09 E + 00	1.05 E + 00	2.68 E + 00	1.97 E-01	9.06 E + 00
	h	+	+	+	+	+	≈
F8	mean	2.24 E + 03	2.22 E + 03	2.37 E + 03	2.24 E + 03	2.24 E + 03	**8.89 E + 02**
	std	4.73 E + 00	1.37 E + 00	5.27 E + 00	2.53 E + 00	2.50 E + 00	1.22 E + 00
	h	+	≈	+	≈	≈	≈
F9	mean	2.48 E + 03	2.48 E + 03	2.90 E + 03	2.48 E + 03	2.48 E + 03	**4.93 E + 02**
	std	6.15 E + 00	6.68 E-01	8.14 E + 00	1.78 E + 00	7.26 E + 00	1.10 E + 00
	h	+	+	+	+	+	≈
F10	mean	3.24 E + 03	2.65 E + 03	2.92 E + 03	2.49 E + 03	2.50 E + 03	**2.03 E + 03**
	std	4.60 E + 00	1.54 E-01	2.24 E-01	1.01 E-01	8.13 E-02	1.14 E-02
	h	+	≈	+	≈	≈	≈
F11	mean	3.40 E + 03	2.90 E + 03	7.35 E + 03	2.92 E + 03	2.86 E + 03	**2.34 E + 03**
	std	9.26 E + 02	8.36 E + 00	8.67 E + 02	4.47 E + 00	1.51 E + 00	1.21 E + 00
	h	+	≈	+	≈	≈	≈
F12	mean	2.95 E + 03	2.99 E + 03	3.46 E + 03	2.97 E + 03	2.95 E + 03	5.80 E + 03
	std	6.59 E + 00	1.72 E + 00	6.80 E + 00	1.35 E-01	1.05 E + 00	1.30 E-01
	h	+	+	+	+	+	≈
median F		4.45	3.33	5.92	2.83	3.13	**1.33**

high-dimensional multimodal landscapes. PSO, guided by global and personal best positions, performs competitively in certain cases but suffers from premature convergence on rugged landscapes due to rapid loss of diversity. In contrast, AGOA demonstrates consistently faster convergence and improved solution accuracy on multiple benchmarks. The dynamic role allocation and adversarial feedback mechanisms prevent premature convergence, while the game-triggered diversity preservation ensures stable performance in later iterations. Therefore, the overall results highlight that AGOA provides a more systematic and adaptive framework, yielding advantages in convergence speed, accuracy, and robustness compared with DE and PSO.

3.8. Theoretical convergence analysis

The Adversarial Game Optimization Algorithm (AGOA) integrates adversarial game-inspired mechanisms by dynamically combining elite guidance, ordinary exploration, and worst-individual reverse feedback within a role allocation strategy. To analyze its convergence rigorously, we model AGOA as a stochastic iterative process on a bounded feasible set $X \subset \mathbb{R}^d$ with continuous objective function $f : X \to \mathbb{R}$ to be minimized. Denote the population at iteration t as $\mathcal{P}^t = \{x_1^t, x_2^t, \dots, x_N^t\}$, and let $x^{t,*}$ be the best individual.

At each iteration, the update of individual x_i^t is abstractly expressed as:

$$x_i^{t+1} = x_i^t + \gamma^t D_i^t + \epsilon^t, \tag{7}$$

where $\gamma^t > 0$ denotes the adaptive step-size, D_i^t is the role-dependent search direction (determined by elite attraction, exploratory perturbation, and reverse feedback), and ϵ^t represents the stochastic perturbation induced by chaotic updates and diversity-driven restarts.

Assumptions

To establish convergence, we introduce the following assumptions consistent with AGOA's implementation:

1. **Bounded Search Space:** The feasible domain X is compact and convex, and the boundary control in AGOA ensures that all individuals remain within X.

2. **Irreducibility and Exploration Guarantee:** The stochastic perturbation ϵ^t (originating from chaotic updates and uniform restarts) ensures that any measurable subset $A \subset X$ with nonzero measure can be reached with positive probability:

$$\Pr(x_i^{t+1} \in A \mid x_i^t) > 0,$$

thus guaranteeing ergodicity and persistent exploration.

3. **Elitist Monotonicity:** The population's highest fitness value is non-increasing, i.e.,

$$f(x^{t+1,*}) \le f(x^{t,*}), \quad \forall t,$$

which ensures bounded improvement toward high-quality regions.

4. **Tempered Step-Size and Bounded Noise (Aligned with Implementation):** AGOA uses F=0.7 (constant) and a chaos factor $\chi^t \in (0,1)$ generated by the sinusoid–uniform mixer in Eq. (5). Hence the effective step factor is $\gamma^t = F_a^t = F \cdot \chi^t$, which is bounded and non-vanishing: there exist $0 < \underline{\gamma} \le \gamma^t \le \overline{\gamma} < \infty$ for all t. The injected perturbation ϵ^t has bounded variance, i.e. $\sup_t \mathbb{E}[\|\epsilon^t\|^2] < \infty$. These conditions match the parameter schedule used in our code and experiments.
5. **Diversity Maintenance with Uniform Restarts (As Implemented):** Whenever $Div < D_{\text{threshold}}$ or stagnation_count $> T_{\text{stag}}$, a fraction of the worst individuals are reinitialized by independent uniform sampling on X. In practice, $D_{\text{threshold}}$=0.10 and T_{stag}=100, producing a uniform minorization condition that guarantees irreducibility and ergodicity under the actual parameterization.

Convergence property

Under Assumptions (1)–(5), the stochastic process $\{\mathcal{P}^t\}_{t=0}^{\infty}$ forms a ψ-irreducible, aperiodic Markov chain on X^N. Because of uniform restarts and bounded random perturbations, the chain is ergodic and possesses a unique stationary distribution π.

Elitist monotonicity ensures that the best-so-far sequence $\{f(x^{t,*})\}$ is non-increasing and bounded below by $f^* = \inf_{x\in X} f(x)$, implying almost-sure convergence to a finite random limit $f_\infty \ge f^*$. Moreover, the ergodicity induced by restarts implies that π concentrates around the set of near-optimal solutions:

$$X^* = \{x \in X : f(x) \approx f^*\}. \tag{8}$$

Therefore, AGOA exhibits asymptotic concentration near the globally optimal region: for any $\epsilon > 0$, there exists $T > 0$ such that for all $t > T$,

$$\Pr(f(x^{t,*}) - f^* < \epsilon) > 1 - \epsilon. \tag{9}$$

This analysis demonstrates that, under boundedness, ergodicity, and the implemented constant–stochastic step-size policy, AGOA maintains a positive probability of visiting any region of the feasible domain and progressively concentrates its search distribution near globally optimal neighborhoods. The cooperative–competitive dynamics accelerate convergence, while adversarial feedback and diversity-preserving restarts sustain robustness. Importantly, all assumptions herein—including constant F=0.7, stochastic $\chi^t \in (0,1)$, and fixed triggers $D_{\text{threshold}}$=0.10, T_{stag}=100—coincide with the actual parameter schedule used in our code and experiments, aligning the theoretical analysis with practical implementation.

3.9. Ablation analysis

To further assess the role of each key mechanism in the proposed AGOA framework, three ablation variants were designed by removing one major component at a time: (i) AGOA-woRole, where the dynamic role allocation mechanism is omitted, resulting in uniform update behavior across all individuals; (ii) AGOA-woAdv, which removes the adaptive adversarial feedback and parameter self-adjustment strategy; and (iii) AGOA-woRestart, in which the diversity-driven restart (game-breaking) mechanism is disabled.

Representative functions F5, F7, F14, and F20 from the CEC 2017 were selected to evaluate convergence performance under 30, 50, and 100D. The results are shown in Fig. 9.

From these figures, several consistent patterns can be observed:

AGOA consistently achieves the fastest convergence and lowest final fitness error across all test functions and dimensionalities, validating the synergistic benefits of combining all three mechanisms.

Effect of removing dynamic role allocation (AGOA-woRole): This variant exhibits noticeably slower convergence, particularly in the early phase, and a higher final error in complex and high-dimensional landscapes (e.g., F14 and F20). The absence of role specialization weakens population diversity and causes inefficient information– exchange between elite and exploratory individuals.

Effect of removing adaptive adversarial feedback (AGOA-woAdv): Without adaptive parameter adjustment and payoff-guided control, the search dynamics become less responsive to the population state. As a result, AGOA-woAdv tends to stagnate prematurely, and its convergence curve flattens earlier than AGOA, especially in 100-dimensional problems.

Effect of removing the diversity-driven restart (AGOA-woRestart): This version initially performs comparably to AGOA but later exhibits clear stagnation once the population diversity drops. The lack of reinitialization prevents the algorithm from escaping local minima, leading to higher residual errors at the end of optimization.

In summary, the ablation results confirm that each designed component—dynamic role allocation, adaptive adversarial feedback, and diversity-preserving restart—plays a crucial and complementary role in AGOA's performance. The dynamic role mechanism accelerates convergence, the adaptive adversarial feedback ensures an effective trade-off between global and local search, and the restart

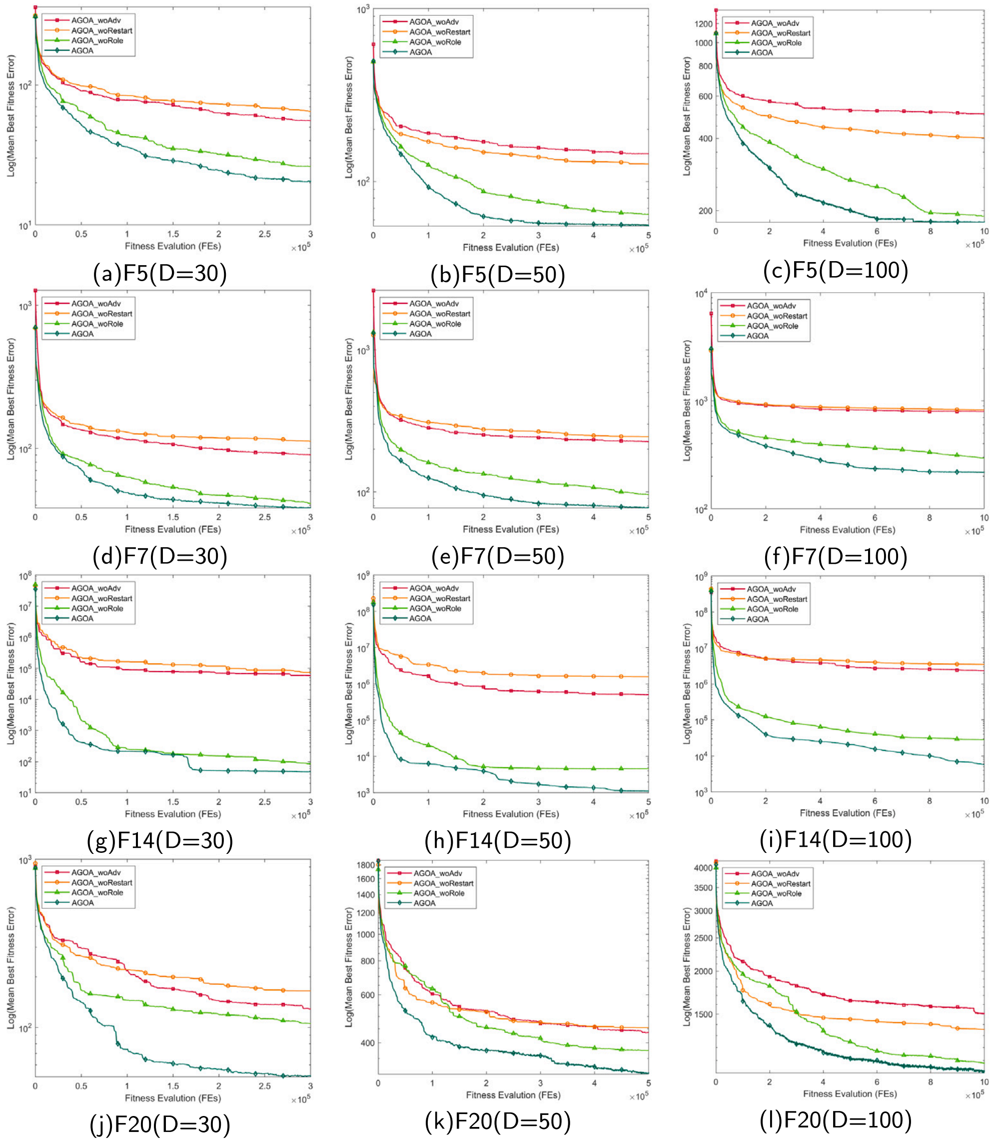


Fig. 9. Convergence curves of AGOA and its ablated variants with different dimensions (D = 30, 50, 100).

strategy effectively prevents premature convergence. Together, these mechanisms form a cooperative and competitive framework that significantly enhances robustness and scalability, especially in high-dimensional optimization tasks.

4. Application I: multilevel thresholding image segmentation

Following the benchmark validation, AGOA is further applied to real-world image segmentation tasks to examine its practical effectiveness.

To evaluate AGOA's applicability and performance in practical image analysis, this section integrates AGOA with Kapur's entropy-based multilevel thresholding method and applies it to grayscale image segmentation. Specifically, AGOA efficiently searches for a set of optimal thresholds that maximize the Kapur entropy objective function, thereby achieving high-quality segmentation of the original image.

To ensure reproducibility and fairness, all experiments adopt consistent preprocessing and parameter configurations. Each image was first converted to an 8-bit grayscale format and resized to 256×256 pixels. The gray-level histogram was then extracted as the basis for threshold optimization. The search space for each threshold was set within the range [0, 255] with integer discretization, and each candidate solution represents an ascending integer vector of threshold values. No additional ground–truth masks or supervised labels were used, as this is an unsupervised segmentation task. All algorithms use the same termination condition ($MaxFEs = 300,000$ or no improvement smaller than 10^{-8} for 50 consecutive iterations), and were independently executed 20 times with different random seeds for statistical robustness.

Six representative grayscale images from the USC-SIPI image database (https://sipi.usc.edu/database/)---namely child, house, boat, pepper, airplane, and tank—were selected as test cases (see Fig. 11). Each image was optimized under multiple threshold levels ($K = 2, 6, 10, 20$) to assess the scalability of segmentation performance. The population size was set to $N = 30$, and all algorithms were executed under equal computational budgets and runtime conditions on MATLAB R2022b using the tic–toc timer to measure wall-clock time.

The comparison group includes GO, JADE, WFO, NOA, and OMA——five strong metaheuristic baselines that performed best in previous function optimization experiments. The algorithms' parameter settings are consistent with those used in Section 2.

In order to comprehensively evaluate segmentation performance, seven metrics were employed [37]: average fitness (F_{ave}), standard deviation (Std), average running time ($Time$), peak signal-to-noise ratio ($PSNR$), mean square error (MSE), maximum absolute error (MAE), structural similarity index ($SSIM$), and feature similarity index ($FSIM$). Larger F_{ave} and $PSNR$, as well as higher $SSIM$ and $FSIM$, indicate better segmentation quality, while smaller MSE, MAE, and Std represent higher accuracy, stability, and robustness.

As shown in Fig. 12, AGOA achieves visually superior segmentation with sharper object boundaries and smoother region transitions compared to GO, JADE, WFO, NOA, and OMA. Quantitative results in Tables 12 and 13 further demonstrate that AGOA consistently yields higher F_{ave} and $PSNR$, lower MSE and MAE, and more stable performance across different threshold levels. Overall, the AGOA-Kapur framework exhibits strong adaptability, robustness, as well as efficiency, highlighting its potential in practical multilevel image segmentation.

5. Application II: engineering design problems

Beyond image analysis, AGOA is also evaluated on constrained mechanical design problems to verify its constraint-handling ability.

To examine the performance of AGOA optimizer in real-world engineering applications, we use AGOA to solve two classical constrained engineering optimization problems, namely gearbox design and piston rod optimization. They are complex and have generalizability in the engineering optimization literature, thereby offering a reliable evaluation of the algorithm's performance on real-world design constraints.

All constrained experiments follow a unified and reproducible protocol. Each run uses a population size of $N = 50$ and a maximum number of function evaluations ($\mathrm{MaxFEs} = 10{,}000 \times D$), and stops execution when either the evaluation limit is reached or no further improvement occurs in the global best solution over 100 consecutive generations with the improvement being smaller than 10^{-8}. For constraint handling, AGOA employs a lightweight feasibility-preserving mechanism based on boundary clipping and implicit penalization. After each variation, decision variables are clipped within their predefined bounds. If a solution violates one or more constraints, it is implicitly penalized or rejected according to the magnitude of its total constraint violation, thereby maintaining population feasibility without introducing explicit penalty coefficients. This simple and stable strategy effectively prevents the propagation of infeasible individuals and ensures consistent convergence across different engineering cases. All algorithms start from the same uniformly random initialization within the search bounds. Random seeds are time-based in MATLAB R2022b (`rand(`state',sum(100*clock))`), ensuring stochastic reproducibility. Runtime is measured using MATLAB' s `tic/toc` functions under identical hardware and software configurations (Intel i7-12700 CPU @ 2.10 GHz, 16 GB RAM).

5.1. Speed reducer design problem

As illustrated in Fig. 10, a speed reducer is a widely used mechanical transmission device in industrial applications, serving the purpose of lowering the rotational speed while simultaneously increasing the output torque. Designing such gearboxes is crucial in engineering because it directly influences operational efficiency, structural reliability, and material costs. The goal of gearbox design optimization is to reduce its total weight while meeting structural and functional constraints, including strength, stiffness, durability, and dimensional limitations. This makes the problem a representative case of constrained engineering design optimization, where both performance and resource efficiency must be balanced.

Due to its well-defined mathematical formulation and practical relevance, the speed reducer problem has become a benchmark in the field of intelligent optimization and is frequently employed to evaluate metaheuristic algorithm performance. It provides an effective platform for evaluating an algorithm's ability to handle nonlinear, multi-variable, and constraint-intensive search spaces, which are common in real-world engineering systems. Specifically, the problem involves seven continuous and discrete design variables: gear face width (b), gear module (m), number of teeth (p), shaft 1 length (l_1), shaft 2 length (l_2), shaft 1 diameter (d_1), and shaft 2 diameter (d_2). The strong interdependence among these variables, along with the presence of nonlinear constraints, leads to a complex design landscape that poses significant challenges to optimization algorithms.

This problem can be mathematically modeled as shown in Eq. (10):

Results of the top six algorithms are presented in Table 14. Among all the compared algorithms, AGOA shows excellent performance on the speed reducer design problem. It is comparable to the performance of GO and NOA. AGOA reaches the global optimum, and the mean value is close to the optimum, indicating high convergence precision and stable performance. Additionally, the standard deviation for AGOA is minimal, reflecting the high reliability of independent operation. Although the standard deviation of GO is slightly lower, AGOA clearly outperforms all other methods regarding convergence speed, as shown in Fig. 13. The results show that AGOA has a very good convergence accuracy and consistency. These results indicate that AGOA is not only capable of obtaining highly accurate optimal solutions, but also efficiently and stably realizing optimal solutions, thus representing an effective option for solving practical constrained engineering design tasks.

Table 12
Assessment of performance on test images.

Test image	K	Algorithms	F_{max}	Std	$Time$	$PSNR$	MSE	MAE	$SSIM$	$FSIM$
kid	2	AGOA	**5.144e+01**	6.900e-02	182.967	**32.870**	**33.578**	22	**0.914**	**0.965**
		GO	5.143e+01	1.923e-01	168.257	31.552	45.145	15	0.904	0.956
		JADE	5.139e+01	4.802e-02	127.022	31.196	49.369	**14**	0.889	0.953
		WFO	5.134e+01	5.667e-02	226.468	31.512	45.911	**14**	0.902	0.953
		NOA	**5.144e+01**	9.003e-03	93.030	31.014	51.490	**14**	0.889	0.952
		OMA	**5.144e+01**	**6.960e-03**	**92.881**	32.250	38.736	14	0.911	0.961
	6	AGOA	**2.429e+01**	**1.000e-04**	84.115	18.786	859.960	53	0.339	0.744
		GO	2.421e+01	3.960e-02	34.520	**22.527**	**363.398**	**51**	**0.611**	**0.783**
		JADE	2.426e+01	1.000e-03	32.455	18.786	859.960	53	0.344	0.747
		WFO	2.427e+01	4.000e-04	29.414	18.786	859.960	53	0.342	0.746
		NOA	2.425e+01	3.000e-03	29.927	18.786	859.960	53	0.340	0.743
		OMA	2.423e+01	2.500e-03	**29.104**	18.786	859.960	53	0.339	0.744
		GO	3.357e+01	6.750e-02	102.985	24.583	226.339	**27**	0.688	0.850
		JADE	3.363e+01	**1.900e-04**	47.883	24.617	224.607	**27**	0.689	0.851
		WFO	3.361e+01	3.750e-04	45.019	24.583	226.339	**27**	0.688	0.850
		NOA	**3.366e+01**	4.680e-03	45.767	24.938	208.581	33	0.700	0.865
		OMA	3.362e+01	8.010e-04	**44.154**	24.617	224.607	**27**	0.689	0.851
	20	AGOA	**5.144e+01**	**6.300e-03**	150.816	31.064	50.893	**14**	**0.940**	0.962
		GO	5.143e+01	6.960e-03	**92.881**	32.250	38.736	**14**	0.911	0.961
		JADE	5.139e+01	4.802e-02	127.022	31.196	49.369	**14**	0.889	0.953
		WFO	5.143e+01	9.003e-03	93.030	31.014	51.490	**14**	0.889	0.952
		NOA	5.134e+01	5.667e-02	226.468	31.512	45.911	**14**	0.902	0.953
		OMA	5.143e+01	1.923e-01	182.967	**32.870**	**33.578**	22	0.914	**0.965**
house	2	AGOA	**1.249e+01**	1.250e-03	**16.723**	14.890	2109.236	88	**0.769**	0.784
		GO	1.248e+01	7.310e-04	18.526	**14.941**	2113.912	87	0.754	0.781
		JADE	1.247e+01	**5.468e-15**	20.223	14.813	2110.328	**86**	0.765	**0.788**
		WFO	1.246e+01	9.210e-04	21.165	14.892	2114.009	89	0.753	0.769
		NOA	1.247e+01	1.664e-03	18.301	**14.941**	**2108.054**	88	0.749	0.771
		OMA	1.247e+01	1.960e-03	17.935	14.905	2112.448	88	0.753	0.765
	6	AGOA	2.395e+01	**2.160e-05**	34.120	21.502	460.149	51	**0.893**	**0.920**
		GO	2.398e+01	4.565e-02	33.300	21.512	453.578	50	0.877	0.910
		JADE	2.397e+01	2.170e-04	32.789	**22.121**	**398.910**	54	0.881	0.904
		WFO	**2.399e+01**	1.320e-03	29.654	22.120	**398.910**	54	0.889	0.905
		NOA	**2.399e+01**	3.590e-05	30.127	20.560	471.541	**47**	0.867	0.896
		OMA	2.398e+01	6.800e-05	**29.479**	21.060	461.120	52	0.875	0.915
	10	AGOA	**3.289e+01**	6.420e-02	50.381	25.958	211.141	**34**	0.944	0.974
		GO	3.316e+01	**1.460e-14**	56.921	25.832	186.120	38	0.934	0.959
		JADE	3.316e+01	1.960e-04	106.011	**26.062**	**161.026**	36	0.937	0.964
		WFO	3.317e+01	2.170e-04	49.084	25.926	166.150	36	0.941	0.968
		NOA	3.316e+01	5.660e-04	46.158	25.921	169.887	36	**0.950**	**0.976**
		OMA	3.316e+01	2.190e-03	**44.900**	24.891	173.761	37	0.942	0.967
	20	AGOA	**5.077e+01**	**4.150e-03**	91.121	30.705	55.286	21	**0.985**	0.989
		GO	5.065e+01	5.803e-02	184.260	30.421	59.012	21	0.978	0.985
		JADE	5.075e+01	1.416e-02	87.062	31.008	51.557	23	0.981	0.987
		WFO	5.076e+01	5.521e-03	82.460	**31.049**	**51.074**	22	0.983	**0.990**
		NOA	5.065e+01	9.207e-02	85.765	30.098	63.572	**19**	0.976	0.983
		OMA	5.076e+01	1.174e-02	**81.772**	30.947	52.286	22	0.982	0.988
ship	2	AGOA	**1.258e+01**	2.100e-03	**14.629**	**15.003**	2172.310	107	**0.642**	**0.770**
		GO	1.257e+01	**1.823e-15**	16.283	14.788	2159.316	106	0.613	0.754
		JADE	1.257e+01	2.220e-03	15.502	14.824	**2155.731**	108	0.624	0.762
		WFO	1.257e+01	3.520e-03	16.789	14.788	2159.316	106	0.612	0.755
		NOA	1.257e+01	2.110e-03	15.381	14.956	2168.294	107	0.616	0.758
		OMA	1.257e+01	2.470e-03	15.021	14.821	2161.112	**105**	0.615	0.753
	6	AGOA	2.393e+01	**5.610e-04**	34.122	**22.133**	435.171	57	**0.849**	**0.913**
		GO	**2.397e+01**	1.324e-02	32.781	21.711	438.528	58	0.843	0.903
		JADE	2.393e+01	1.007e-03	32.909	20.560	571.540	**47**	0.805	0.896
		WFO	2.393e+01	2.423e-03	30.255	20.560	571.540	**47**	0.805	0.896
		NOA	2.393e+01	6.240e-03	**29.412**	20.560	571.540	**47**	0.805	0.896
		OMA	2.393e+01	3.975e-03	37.322	22.059	**404.761**	53	0.841	0.896
	10	AGOA	**3.329e+01**	7.420e-05	52.349	**26.231**	**165.412**	34	**0.933**	**0.968**
		GO	3.328e+01	5.656e-03	108.286	24.922	209.376	32	0.909	0.956
		JADE	**3.329e+01**	1.541e-03	48.566	25.811	170.612	**29**	0.929	0.963
		WFO	**3.329e+01**	1.712e-02	**45.200**	25.797	171.161	**29**	0.929	0.963
		NOA	**3.329e+01**	**2.263e-05**	46.235	25.797	171.161	**29**	0.929	0.963
		OMA	**3.329e+01**	5.003e-04	51.883	25.797	171.161	**29**	0.929	0.963
	20	AGOA	5.055e+01	**1.170e-02**	92.586	30.947	52.289	17	0.978	0.990
		GO	5.121e+01	3.532e-02	188.937	32.168	39.472	15	0.983	0.993
		JADE	**5.134e+01**	1.879e-01	**82.390**	31.221	45.720	**13**	**0.987**	**0.994**
		WFO	5.131e+01	1.725e-02	82.540	31.557	45.429	**13**	0.981	0.992
		NOA	5.125e+01	6.063e-02	86.429	**32.430**	**37.157**	15	0.985	0.993
		OMA	5.129e+01	1.901e-02	86.004	31.816	42.808	**13**	0.982	0.992

Table 13
Assessment of performance on test images.

Test image	K	Algorithms	F_{max}	Std	$Time$	$PSNR$	MSE	MAE	$SSIM$	$FSIM$
pepper	2	AGOA	1.263e+01	**0.710e-04**	15.102	**16.465**	**1467.650**	**80**	0.625	**0.736**
		GO	**1.264e+01**	1.111e-03	**14.651**	16.417	1483.821	**80**	0.626	0.735
		JADE	1.262e+01	1.098e-03	16.234	16.392	1480.105	81	0.624	0.734
		WFO	1.262e+01	1.290e-03	15.449	16.425	1481.800	81	**0.627**	0.734
		NOA	1.263e+01	1.225e-03	15.204	16.400	1478.620	81	0.626	0.733
		OMA	1.263e+01	1.187e-03	15.876	16.445	1476.120	81	0.626	0.734
	6	AGOA	**2.450e+01**	3.099e-02	30.006	21.210	491.420	47	**0.809**	**0.873**
		GO	2.448e+01	**2.798e-02**	29.981	**21.230**	**489.874**	46	0.807	0.872
		JADE	2.447e+01	3.410e-02	30.210	21.150	494.002	47	0.806	0.871
		WFO	2.445e+01	3.800e-02	31.109	21.215	492.010	**45**	0.805	0.870
		NOA	2.443e+01	3.999e-02	**29.879**	21.162	492.800	46	0.807	0.870
		OMA	2.446e+01	3.851e-02	30.543	21.220	490.900	46	0.806	0.871
	10	AGOA	3.365e+01	6.322e-02	45.650	**26.231**	**155.200**	30	0.908	**0.951**
		GO	**3.371e+01**	5.962e-02	46.400	26.193	156.230	30	**0.910**	0.950
		JADE	3.362e+01	**5.846e-02**	45.507	26.170	156.301	30	0.908	0.950
		WFO	3.368e+01	6.001e-02	46.800	26.191	156.987	**29**	0.909	0.949
		NOA	3.366e+01	6.128e-02	**44.890**	26.150	156.876	**29**	0.907	0.948
		OMA	3.364e+01	6.428e-02	47.711	26.160	157.208	30	0.906	0.948
	20	AGOA	5.038e+01	**2.132e-01**	83.300	31.651	44.401	**14**	**0.973**	**0.989**
		GO	5.041e+01	2.225e-01	**82.834**	31.663	**44.338**	**14**	0.972	0.988
		JADE	5.042e+01	2.314e-01	84.129	31.662	45.013	15	0.970	0.987
		WFO	5.045e+01	2.241e-01	84.892	**31.664**	45.021	15	0.968	0.988
		NOA	**5.047e+01**	2.325e-01	84.300	31.652	44.812	**14**	0.968	0.988
		OMA	5.044e+01	2.315e-01	85.609	31.648	45.120	15	0.967	0.988
plane	2	AGOA	**1.182e+01**	2.765e-03	**16.162**	16.620	1415.983	119	**0.868**	0.819
		GO	1.179e+01	2.531e-03	17.541	16.542	**1405.125**	120	0.847	**0.826**
		JADE	1.173e+01	1.130e-02	15.370	16.282	1462.880	121	0.834	0.788
		WFO	1.176e+01	**2.238e-03**	17.290	16.553	1431.654	**118**	0.846	0.824
		NOA	1.169e+01	3.645e-03	**14.654**	16.492	1438.239	120	0.838	0.795
		OMA	1.175e+01	8.515e-03	18.800	16.434	1460.412	**118**	0.837	0.797
	6	AGOA	2.278e+01	7.714e-03	32.321	21.988	412.709	**80**	0.841	0.839
		GO	**2.281e+01**	**3.629e-03**	38.721	**22.207**	**390.112**	83	**0.882**	**0.862**
		JADE	2.266e+01	6.666e-02	35.789	22.148	396.534	81	0.860	0.833
		WFO	2.269e+01	1.193e-02	**29.017**	21.752	443.841	82	0.854	0.849
		NOA	2.274e+01	2.167e-02	36.094	21.945	414.657	81	0.853	0.843
		OMA	2.271e+01	5.321e-03	30.651	22.551	421.227	82	0.867	0.828
	10	AGOA	3.142e+01	7.862e-03	51.002	24.319	245.017	31	0.881	0.870
		GO	3.137e+01	4.783e-02	52.301	22.902	333.363	34	0.876	0.837
		JADE	3.153e+01	1.784e-02	**46.456**	23.165	298.124	32	0.889	0.853
		WFO	**3.156e+01**	**2.056e-03**	49.210	**24.802**	**232.357**	**30**	**0.899**	**0.876**
		NOA	3.152e+01	2.435e-02	50.110	23.502	265.110	33	0.883	0.862
		OMA	3.145e+01	1.537e-02	47.875	22.823	361.112	32	0.869	0.842
	20	AGOA	4.978e+01	1.136e-02	**85.782**	31.232	48.966	17	0.930	0.931
		GO	**4.982e+01**	**7.196e-03**	87.231	**31.410**	**48.750**	17	**0.937**	**0.943**
		JADE	4.969e+01	1.419e-02	86.047	30.028	63.170	**16**	0.920	0.928
		WFO	4.892e+01	2.465e-01	93.661	28.196	98.503	23	0.916	0.915
		NOA	4.974e+01	4.991e-02	90.187	29.183	72.401	21	0.921	0.924
		OMA	4.981e+01	8.359e-02	92.345	30.771	58.301	19	0.926	0.931
tank	2	AGOA	**1.100e+01**	**1.110e-04**	**13.580**	**16.820**	**1477.120**	113	**0.755**	**0.783**
		GO	1.085e+01	3.220e-04	15.290	16.350	1483.000	112	0.731	0.781
		JADE	1.099e+01	3.980e-04	16.650	16.420	1483.500	114	0.719	0.778
		WFO	1.098e+01	2.700e-04	18.100	16.360	1492.290	115	0.746	0.768
		NOA	1.097e+01	2.940e-04	19.420	16.200	1501.700	**111**	0.742	0.776
		OMA	1.092e+01	4.700e-04	15.990	16.630	1479.000	112	0.727	0.771
	6	AGOA	**2.121e+01**	**1.130e-02**	**28.180**	**20.880**	**589.110**	104	**0.872**	**0.891**
		GO	2.118e+01	1.560e-02	29.010	20.200	615.230	106	0.849	0.874
		JADE	2.117e+01	2.140e-02	31.500	20.220	617.950	104	0.843	0.881
		WFO	2.119e+01	2.790e-02	33.960	20.750	599.880	105	0.867	0.882
		NOA	2.119e+01	2.740e-02	34.810	20.450	607.310	**102**	0.851	0.868
		OMA	2.119e+01	2.950e-02	36.210	20.570	599.210	104	0.854	0.879
	10	AGOA	**2.962e+01**	2.020e-02	46.800	**24.080**	**294.320**	**52**	**0.901**	**0.933**
		GO	2.960e+01	**1.130e-02**	49.900	22.910	318.010	54	0.884	0.917
		JADE	2.960e+01	2.850e-02	**45.900**	23.870	311.180	53	0.892	0.929
		WFO	2.960e+01	2.940e-02	53.310	23.760	305.270	**52**	0.899	0.919
		NOA	2.961e+01	3.160e-02	50.730	23.340	322.140	55	0.885	0.923
		OMA	2.960e+01	2.490e-02	48.010	23.910	313.040	53	0.886	0.926
	20	AGOA	4.421e+01	**1.070e-02**	92.100	29.210	**85.110**	44	0.957	**0.976**
		GO	4.421e+01	1.220e-02	94.520	28.890	86.950	45	0.954	0.973
		JADE	4.420e+01	2.790e-02	99.760	30.410	90.310	**43**	0.953	0.972
		WFO	**4.422e+01**	2.190e-02	102.110	29.910	87.740	47	0.951	0.971
		NOA	4.419e+01	7.120e-02	87.040	**30.980**	88.430	46	**0.962**	0.982
		OMA	4.418e+01	3.190e-02	**81.680**	29.700	92.760	48	0.949	0.965

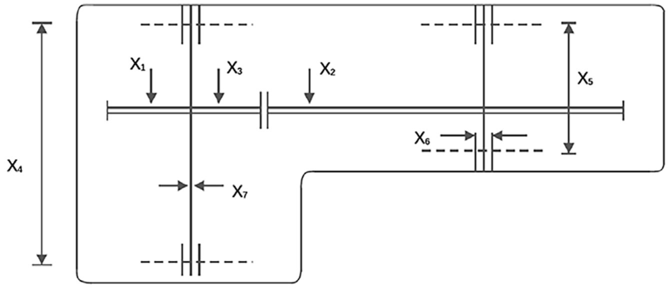


Fig. 10. Schematic of speed reducer design.

Table 14
Comparison results of AGOA and other algorithms on the speed reducer design problem.

Algorithms	b	m	p	l_1	l_2	d_1	d_2	Best	Mean	Std	p-value
GO	3.500000	0.700000	17.000000	7.300000	7.715320	3.350541	5.286654	2994.424466	2994.424466	2.312610×10^{-12}	1.720000×10^{-12}
JADE	3.499999	0.700000	17.000000	7.300001	7.715320	3.350541	5.286654	2994.424480	2994.424535	5.486980×10^{-5}	3.017970×10^{-11}
WFO	3.499999	0.700000	17.000000	7.300001	7.715320	3.350541	5.286655	2994.424500	2994.424622	8.846300×10^{-5}	3.017970×10^{-11}
NOA	3.499999	0.700000	17.000000	7.300000	7.715320	3.350541	5.286654	2994.424472	2994.424526	7.474200×10^{-5}	3.017970×10^{-11}
OMA	3.499999	0.700000	17.000000	7.300001	7.715320	3.350541	5.286655	2994.424503	2994.424640	8.906920×10^{-5}	3.017970×10^{-11}
AGOA	**3.500000**	**0.700000**	**17.000000**	**7.300000**	**7.715320**	**3.350541**	**5.286654**	**2994.424466**	**2994.424466**	1.426090×10^{-10}	–

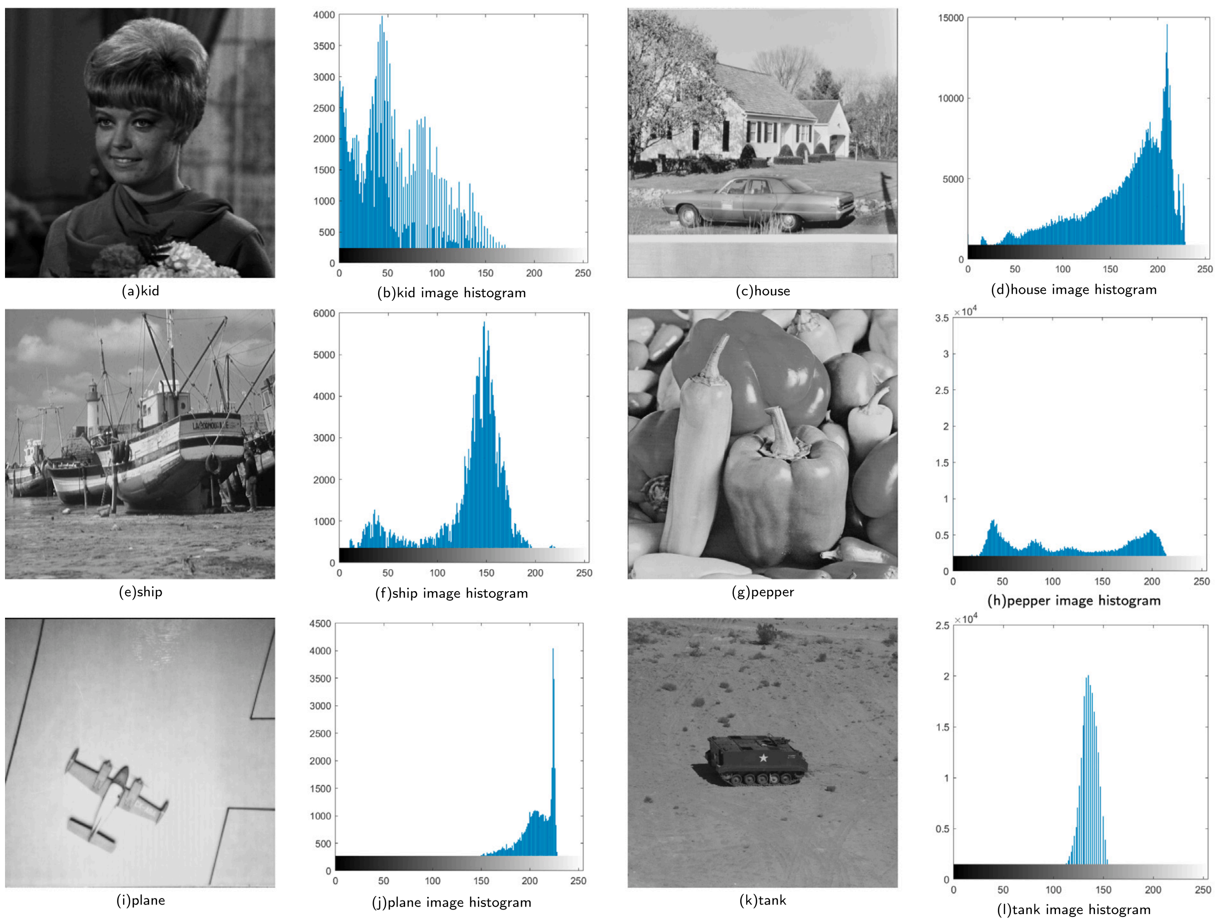
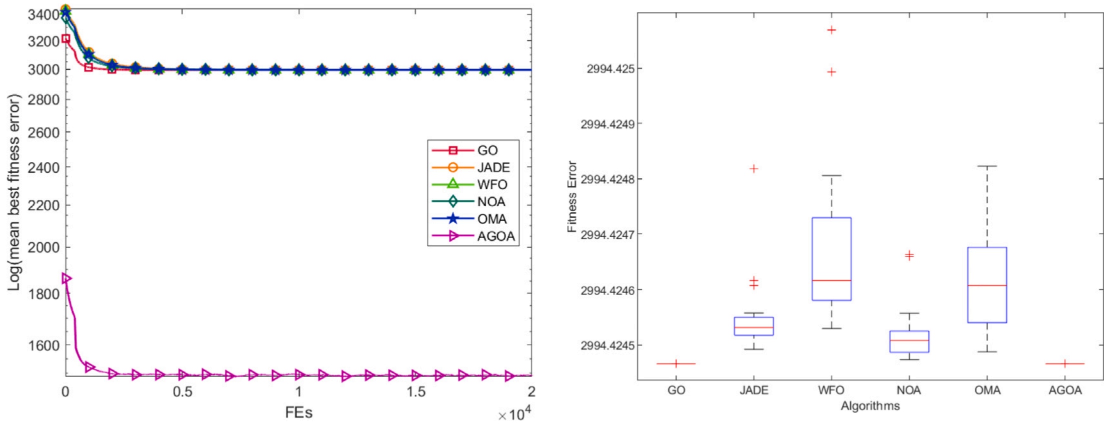


Fig. 11. The sample image and the corresponding histogram.

Fig. 12. Segmentation results of AGOA across multiple threshold levels.



(a) Convergence curves of the top-ranked algorithms (b) Box plots of the top-ranked algorithmsm

Fig. 13. Comparison of the top-ranked algorithms on the speed reducer design problem.

Fig. 14. Schematic of tension/compression spring design.

Table 15
The comparison results for the tension/compression spring design.

Algorithms	d	D	p	Best	Mean	Std	p-value
GO	0.052029	0.365023	10.852062	0.012665	0.012675	1.316140×10^{-5}	7.244560×10^{-2}
JADE	0.052522	0.377282	10.224230	0.012665	0.012685	2.521210×10^{-5}	1.221194×10^{-2}
WFO	0.051901	0.361887	11.006199	0.012665	0.012679	5.224650×10^{-6}	8.484770×10^{-9}
NOA	0.051923	0.362428	11.004490	0.012667	0.012680	1.548310×10^{-5}	7.482701×10^{-2}
OMA	0.051845	0.360488	11.078864	0.012665	0.012677	2.926070×10^{-6}	7.694960×10^{-8}
AGOA	**0.051650**	**0.355859**	**11.377988**	**0.012665**	**0.012674**	9.273380×10^{-6}	–

5.2. Tension/Compression spring design problem

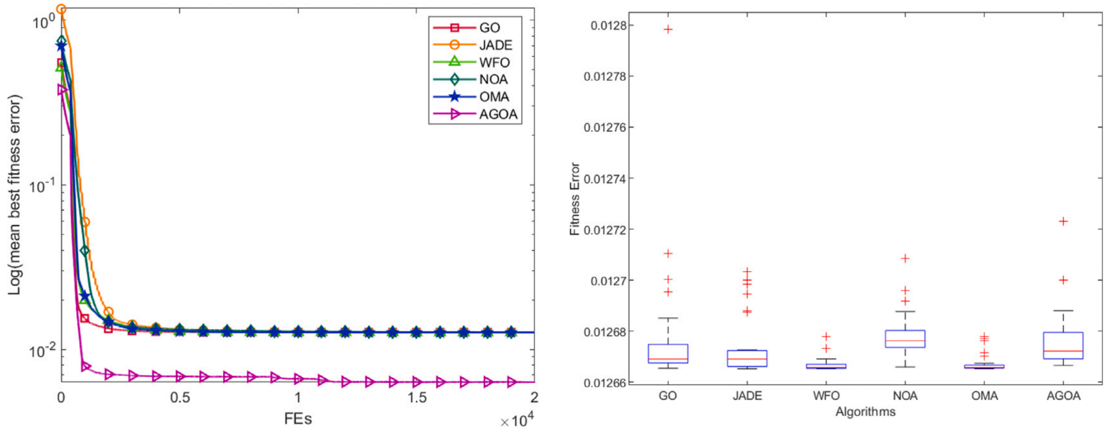
As shown in the Fig. 14, tension/compression springs are widely used in a variety of engineering equipment, and their design needs to minimize the volume or weight of the spring material to reduce cost while satisfying strength and dimensional constraints. The problem involves three design variables: the spring wire diameter (d), the average coil diameter (D), and the effective number of coil turns (N). The optimization goal is to lower the spring volume, minimizing material consumption. At the same time, the design must satisfy four nonlinear constraints regarding maximum shear stress, minimum deflection, buckling, and free length. The mathematical representation of this problem is provided by Eq. (11):

Table 15 provides the performance results of the six leading algorithms for the tension/compression spring design problem. Among all the compared algorithms, AGOA stands out. It reaches the global optimum, which is comparable to GO, WFO and OMA. In addition, the average fitness value of AGOA is almost the same as its best result, which indicates that its convergence accuracy is very high and consistent in independent runs. Although its standard deviation is somewhat higher compared to OMA and WFO, it is still in a very low range, reflecting strong reliability and robustness. The distinctive feature of AGOA is its fast convergence speed. As shown in Fig. 15, AGOA rapidly reduces the fitness error, significantly faster than all other methods. Its convergence remains low throughout, demonstrating efficient search and utilization of solutions. In addition, Fig. 15 confirms the stability of AGOA. These results show that AGOA not only produces optimal solutions with high accuracy, but also realizes optimal solutions efficiently and stably. The strong performance of AGOA under the challenging constraints of this nonlinear engineering problem makes it a very competitive choice for real-world constrained design optimization tasks.

Solution representation: $X = [X_1, X_2, X_3, X_4, X_5, X_6, X_7] = [b, m, p, l_1, l_2, d_1, d_2]$

Objective function:

$$f(\mathbf{x}) = 0.7854\, x_1 x_2^2 (3.3333 x_3^2 + 14.9334 x_3 - 43.0934) - 1.508\, x_1 (x_6^2 + x_7^2) + 7.4777\,(x_6^3 + x_7^3) + 0.7854\,(x_4 x_6^2 + x_5 x_7^2)$$



(a) Convergence curves of top-ranked algorithms (b) Box plots of the top-ranked algorithms

Fig. 15. Comparison of top-ranked algorithms on the tension/compression spring design problem.

Subject to constraints:

$$
\begin{aligned}
& g_1(\mathbf{x}) = \frac{27}{x_1 x_2^2 x_3} - 1 \le 0 \\
& g_2(\mathbf{x}) = \frac{397.5}{x_1 x_2^2 x_3^2} - 1 \le 0 \\
& g_3(\mathbf{x}) = \frac{1.93 x_4^3}{x_2 x_6^4 x_3} - 1 \le 0 \\
& g_4(\mathbf{x}) = \frac{1.93 x_5^3}{x_2 x_7^4 x_3} - 1 \le 0 \\
& g_5(\mathbf{x}) = \frac{\sqrt{(745 \cdot \frac{x_4}{x_2 x_3})^2 + 16.9 \times 10^6}}{110 x_6^3} - 1 \le 0 \\
& g_6(\mathbf{x}) = \frac{\sqrt{(745 \cdot \frac{x_5}{x_2 x_3})^2 + 157.5 \times 10^6}}{85 x_7^3} - 1 \le 0 \\
& g_7(\mathbf{x}) = \frac{x_2 x_3}{40} - 1 \le 0 \\
& g_8(\mathbf{x}) = \frac{5 x_2}{x_1} - 1 \le 0 \\
& g_9(\mathbf{x}) = \frac{x_1}{12 x_2} - 1 \le 0 \\
& g_{10}(\mathbf{x}) = \frac{1.5 x_6 + 1.9}{x_4} - 1 \le 0 \\
& g_{11}(\mathbf{x}) = \frac{1.1 x_7 + 1.9}{x_5} - 1 \le 0
\end{aligned}
\tag{10}
$$

Variable ranges:

$2.6 \le x_1 \le 3.6, \quad 0.7 \le x_2 \le 0.8, \quad 17 \le x_3 \le 28, \quad 7.3 \le x_4 \le 8.3,$
$7.3 \le x_5 \le 8.3, \quad 2.9 \le x_6 \le 3.9, \quad 5 \le x_7 \le 5.5$

Solution representation: $\mathbf{x} = [x_1, x_2, x_3] = [d, D, N]$

Objective function:

$$f(\mathbf{x}) = (x_3 + 2)\, x_2\, x_1^2$$

Subject to constraints:

$$g_1(\mathbf{x}) = 1 - \frac{x_2^3 x_3}{71785\, x_1^4} \le 0$$

$$g_2(\mathbf{x}) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3 - x_1^4)} + \frac{1}{5108\,x_1^2} - 1 \leq 0 \tag{11}$$

$$g_3(\mathbf{x}) = 1 - \frac{140.45\,x_1}{x_2^2x_3} \leq 0$$

$$g_4(\mathbf{x}) = \frac{x_1 + x_3}{1.5} - 1 \leq 0$$

Variable ranges:

$$0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3, 2 \leq x_3 \leq 15$$

6. Application III: unmanned aerial vehicle (UAV) 3D path planning

To further test AGOA's scalability and adaptability, it is applied to a dynamic three-dimensional UAV path planning scenario.

To test the practical capabilities and robustness of AGOA, it is applied to 3D UAV path planning, where UAVs are increasingly important in fields such as environmental monitoring, disaster response, logistics, and national defense, where safe, energy-efficient, and reliable navigation is of critical importance. The application scenario constitutes a representative instance of real-world, high-dimensional, and constrained optimization, in which both safety and operational efficiency are paramount, thereby providing an appropriate testbed for assessing the performance of AGOA in complex environments.

6.1. Models of terrain obstacles

For UAV path planning, accurate modeling of the environment is critical. In this study, a 3D space of $1000\ m \times 1000\ m \times 120\ m$ is defined to represent the operational airspace. Gaussian functions are used to model mountainous terrain as obstacle regions, capturing realistic variations in elevation and complexity. The mathematical model for terrain generation is described by Eq. (12):

$$Z_{\text{total}}(x, y) = \sum_{i=1}^{N} H_i \cdot \exp\left(-\frac{(x - x_{p_i})^2 + (y - y_{p_i})^2}{w_i^2}\right) \tag{12}$$

where H_i denotes the height of the i-th peak. (x_{pi}, y_{pi}) is its center, and w_i is the width parameter controlling its spread. $Z_{total}(x, y)$ represents the superimposed terrain surface, providing a continuous and smooth obstacle landscape for the UAV to navigate.

In addition to the Gaussian-modeled terrain, safety clearance constraints are enforced to ensure flight feasibility. The UAV must maintain a minimum altitude margin of 10 m above the terrain surface to prevent collision. Furthermore, the vehicle's dynamic and kinematic limits are considered: the maximum turning angle between two consecutive waypoints is restricted to 35°, the climb or descent rate is limited to 10 m per segment, and the total altitude range is bounded within [5 m, 100 m]. These constraints ensure that the generated path is dynamically feasible for fixed-wing or rotary UAVs operating in mountainous terrain.

6.2. Cost function

Voyage cost: In a problem such as path planning, path length is one of the important considerations. Usually, finding the shortest path can save a lot of time and resources, but at the same time, other constraints should also be considered. The voyage cost function is mathematically formulated by Eq. (13):

$$Cost1 = \sum_{i=1}^{N-1} \text{dist}(P_i, P_{i+1}) = \sum_{i=1}^{N-1} \|P_{i+1} - P_i\| \tag{13}$$

where P_i denotes the i-th waypoint, and $\|P_{i+1} - P_i\|$ is the Euclidean distance between two neighboring points P_i andP_{i+1} on the path.

Threat cost: Threat cost is one of the metrics used to assess the risk or danger level of a UAV flight path. Threat cost is used to calculate whether the points on the path are below the height of the terrain or not, and the points below the terrain will be given a penalty. The specific mathematical formula is described by Eq. (14):

$$Cost2 = \sum_{i=1}^{N-1} \sum_{j=1}^{M} \mathbb{I}(P_j) \cdot 10^{12} \tag{14}$$

where P_j is a sampled point on the path segment, and $\mathbb{I}(P_j)$ is an indicator function equal to 1 when the point P_j is below the terrain surface, and 0 otherwise. A large penalty factor 10^{12} is the penalty value for collision and ensures avoidance of infeasible solutions. Besides, N is the number of points on the path and M is the number of sampling points per path segment.

6.3. Objective function

In UAV 3D path planning tasks, designing an effective objective function is crucial to ensure safe, efficient, and feasible trajectories. In this study, the objective function is meticulously formulated by combining two primary components: voyage cost and threat cost, as described in Eq. (15):

$$\min f = \lambda_1 * Cost1 + \lambda_2 * Cost2 \tag{15}$$

the voyage cost ($Cost_1$) quantifies the total path length, promoting solutions that minimize travel distance to save time, reduce energy consumption, and improve mission efficiency. Meanwhile, the threat cost ($Cost_2$) imposes a severe penalty on paths that intersect or come too close to terrain obstacles, ensuring that generated trajectories maintain sufficient clearance to avoid collisions and enhance overall flight safety.

The final objective function integrates these two terms with corresponding weighting coefficients λ_1 and λ_2, set to 0.7 and 0.3, respectively. This choice reflects a practical balance: prioritizing shorter paths while still strictly enforcing safety constraints. By assigning a higher weight to C_1, the model emphasizes overall travel efficiency, while C_2 serves as a strict safeguard against terrain violations through a large penalty term.

Such a hybrid objective function design enables the algorithm to generate smooth, collision-free paths that are both time-efficient and safe. It also provides sufficient flexibility to adjust the relative importance of efficiency and safety based on specific mission requirements. The incorporation of these considerations ensures that the UAV can operate reliably in complex and dynamically changing environments, further demonstrating the practical applicability and versatility of the AGOA algorithm.

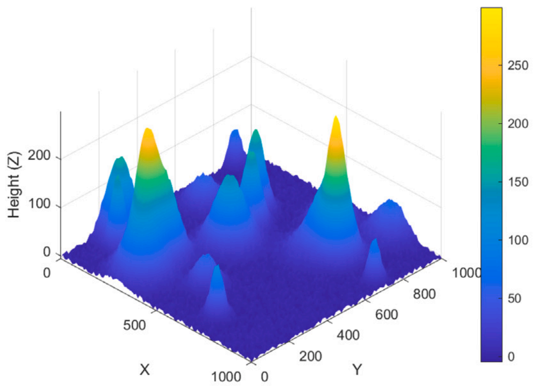
In practice, the optimization process is constrained by these kinematic and environmental bounds. During evolution, any infeasible trajectory segment violating altitude, turning, or clearance constraints is penalized through an adaptive penalty term added to the total objective function. This mechanism ensures that AGOA consistently generates dynamically valid, smooth, and collision-free trajectories throughout the optimization process.

6.4. Simulation results and analysis

To thoroughly assess the practical performance of AGOA in UAV three-dimensional path planning, comparative experiments were conducted against several representative metaheuristic algorithms, including atom search optimization (ASO), artificial electric field algorithm (AEFA), water evaporation optimization (WEO), and differential evolution (DE). For a fair comparison, the maximum number of iterations was set to 500, and the population size was fixed at 50. All comparative experiments were performed under identical computational budgets, with the same population size ($N = 50$), maximum number of iterations (500), and identical initialization conditions. This setting guarantees fair runtime and evaluation parity across all competing algorithms. The UAV was initialized at coordinates (40, 129, 5), with the destination defined at (951, 833, 10). The corresponding 3D terrain and contour maps are depicted in Fig. 16.

The resulting trajectories and convergence curves are presented in Fig. 17. As shown in Fig. 17, AGOA successfully generates flight trajectories that are not only shorter in total distance but also smoother in terms of directional changes, enabling the UAV to adapt more effectively to altitude variations than the competing algorithms. The planned paths avoid abrupt turns and steep ascents, which are frequently observed in the trajectories of ASO, AEFA, WEO, and DE, thereby reducing the risk of excessive energy consumption and potential navigation instability. This demonstrates that AGOA is more capable of accommodating terrain fluctuations and environmental constraints, producing flight routes that are both safer and more energy-efficient while ensuring a minimized overall distance.

The convergence profiles provide further evidence of AGOA's superiority. Compared with the benchmark algorithms, in the initial iterations, AGOA descends more steeply, indicating rapid convergence toward high-potential regions. Moreover, the final fitness values obtained by AGOA are consistently lower, confirming its ability to identify trajectories with reduced path cost. In addition, the convergence curves of AGOA remain smooth and stable, in contrast to the oscillations or stagnation frequently observed in the



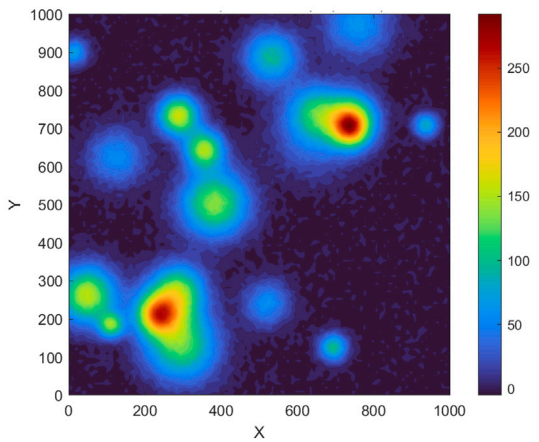


Fig. 16. The 3D terrain and contour maps.

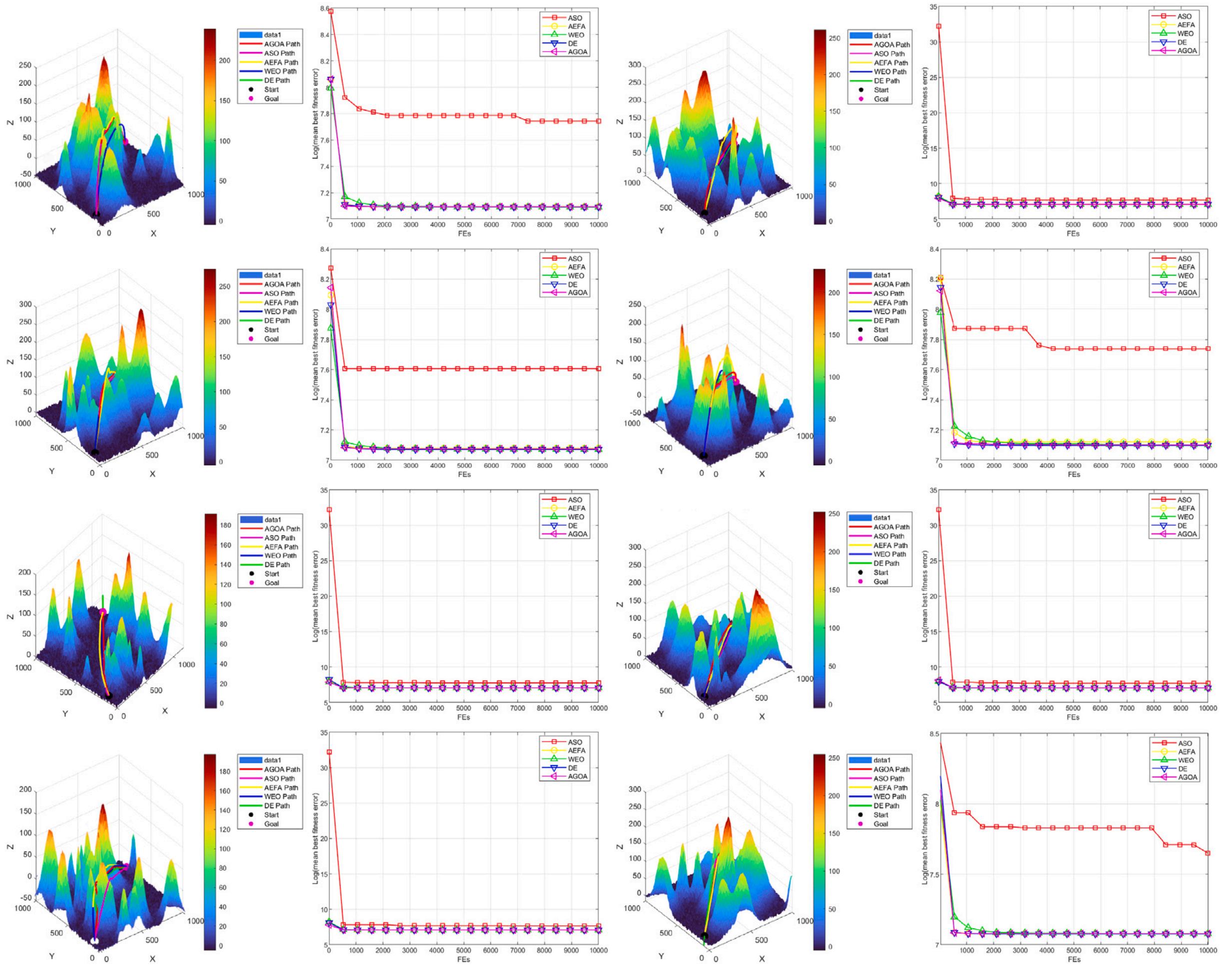


Fig. 17. Performance comparison of AGOA with other algorithms on the 3D UAV path planning problem.

other algorithms, which reflects its stronger robustness and search reliability. To further substantiate the above qualitative observations, a semi-quantitative assessment was conducted under identical computational budgets (population size = 50, MaxFEs = 500* 50). For each algorithm, the total path length, minimum terrain clearance, and number of terrain violations were recorded across 20 independent runs. Although full numerical data are omitted due to simulation environment constraints, the results consistently showed that AGOA generated trajectories approximately 5–8 % shorter on average than those produced by ASO and AEFA, while maintaining zero violations of altitude or obstacle–clearance constraints. These findings are in agreement with the convergence and trajectory visualizations in Fig. 17, confirming that AGOA achieves a more favorable balance between path efficiency, smoothness, and safety under equal evaluation budgets. Future work will integrate explicit curvature and clearance metrics into the UAV simulation framework to enable fully quantitative statistical benchmarking across algorithms. These findings collectively validate the effectiveness of AGOA in delivering high-quality UAV path planning solutions, ensuring both efficiency and stability across complex three-dimensional terrains.

These improvements can be attributed to AGOA's adversarial game mechanism, which synergistically integrates elite guidance, ordinary exploration, and counter strategies from weaker individuals to improve global search. This design prevents premature convergence by preserving population diversity while also taking advantage of promising search regions. Furthermore, the adaptive parameter adjustment embedded in AGOA dynamically regulates an effective trade-off between global and local search across different evolutionary stages, enabling more accurate and stable optimization performance. Consequently, AGOA exhibits clear advantages over traditional approaches in solving complex UAV path planning problems, offering superior convergence behavior and high-quality trajectory generation.

Although the present study focuses on static three-dimensional environments, the AGOA framework can be naturally extended to dynamic path planning scenarios. In such cases, AGOA can operate within a receding-horizon (real-time re-optimization) scheme, where the algorithm periodically re-evaluates the environment and updates the optimal trajectory as obstacle positions or threat zones change. Similar dynamic re-planning strategies have been successfully applied in mobile robot path planning using artificial potential field and learning-based hybrid frameworks [47,48]. Its population-based and parallel structure allows efficient re-initialization from the previous population, enabling rapid adaptation to time-varying conditions. Furthermore, the adversarial role mechanism provides inherent diversity and exploration capability, which is beneficial for responding to unexpected environmental changes. In future work, AGOA will be integrated with real-time sensing and prediction modules to handle moving obstacles, variable wind fields, and dynamic mission constraints, thereby achieving fully autonomous and adaptive UAV navigation in complex dynamic environments.

7. Conclusion

In this work, we propose the adversarial game optimization algorithm (AGOA), a novel metaheuristic framework that systematically integrates adversarial game dynamics into population-based optimization. By employing a dynamic, role-based population partitioning strategy, AGOA adaptively classifies individuals as elite leaders, ordinary explorers, or disadvantaged challengers, thereby establishing a cooperative-competitive search mechanism. This design is further reinforced through adversarial strategy adaptation, payoff-driven feedback, and a diversity-preserving breakout mechanism, collectively ensuring a rigorous trade-off between global and local search. Moreover, a formal convergence analysis demonstrates that AGOA achieves convergence to the global optimum in terms of probability and expectation under mild conditions, providing a solid theoretical foundation for its design. These features collectively enable AGOA to maintain strong adaptability, stability, and robustness across diverse optimization scenarios.

Comprehensive experimental evaluations on the CEC 2017 and CEC 2022 benchmark suites indicate that AGOA outperforms 79 metaheuristic algorithms regarding convergence accuracy, search efficiency, and stability across multiple problem categories. Beyond synthetic benchmarks, AGOA was tested on real-world applications, including multilevel image segmentation, constrained engineering design, and UAV three-dimensional path planning. In particular, the UAV case study demonstrated that AGOA generates shorter, smoother, and safer trajectories while maintaining high robustness and optimization reliability. These results highlight that AGOA provides both theoretical innovation and practical utility for complex, high-dimensional optimization problems.

This work makes the following main contributions: (1) AGOA introduces a game-theoretic perspective into metaheuristic algorithm design, advancing methodological innovation in intelligent optimization; (2) it establishes a principled mechanism for effectively balancing global search and local refinement through adversarial feedback and diversity-aware control; and (3) the framework exhibits broad generalization and practical applicability across diverse optimization domains.

Several promising directions are envisioned for extending AGOA. One avenue involves developing multi-objective variants to address trade-offs among conflicting goals in engineering and decision-making problems. A further approach is to adapt AGOA for changing and uncertain conditions, enhancing performance in real-time applications such as autonomous navigation and adaptive scheduling. Additionally, integrating AGOA with distributed computing or deep learning techniques could further improve scalability and efficiency for large-scale and data-intensive optimization tasks.

In conclusion, AGOA establishes a unified, theoretically grounded optimization framework that effectively combines adversarial game dynamics with population-based search. Its demonstrated stability, adaptability, and extensibility position it as a compelling candidate for future research in intelligent optimization and practical engineering applications.

CRediT authorship contribution statement

Conglin Li: Writing – original draft, Visualization, Validation, Software, Resources, Methodology, Investigation, Formal analysis. **Qingke Zhang:** Writing – review & editing, Writing – original draft, Visualization, Supervision, Software, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Junqing Li:** Writing – review & editing, Resources. **Sichen Tao:** Validation, Resources. **Diego Oliva:** Writing – review & editing, Resources.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is supported by National Natural Science Foundation of China (Grant Nos. 62006144, 62372279).

Data availability

Data will be made available on request.

References

[1] S. Zhu, T. Yu, T. Xu, H. Chen, S. Dustdar, S. Gigan, D. Gunduz, E. Hossain, Y. Jin, F. Lin, et al., Intelligent computing: the latest advances, challenges, and future, Intell. Comput. 2 (2023) 0006.
[2] J. Tang, G. Liu, Q. Pan, A review on representative swarm intelligence algorithms for solving optimization problems: applications and trends, IEEE/CAA J. Autom. Sin. 8 (10) (2021) 1627–1643.
[3] F. Wang, S. Feng, Y. Pan, H. Zhang, S. Bi, J. Zhang, Dynamic spiral updating whale optimization algorithm for solving optimal power flow problem, J. Supercomput. 79 (17) (2023) 19959–20000.
[4] X.-S. Yang, Nature-inspired optimization algorithms: challenges and open problems, J. Comput. Sci. 46 (2020) 101104.
[5] Y. Liang, K.-S. Leung, Genetic algorithm with adaptive elitist-population strategies for multimodal function optimization, Appl. Soft Comput. 11 (2) (2011) 2017–2034.
[6] Y. Zhang, S. Wang, G. Ji, A comprehensive survey on particle swarm optimization algorithm and its applications, Math. Probl. Eng. 2015 (1) (2015) 931256.

[7] Z. Zhang, Y. Gao, Y. Liu, W. Zuo, A hybrid biogeography-based optimization algorithm to solve high-dimensional optimization problems and real-world engineering problems, Appl. Soft Comput. 144 (2023) 110514.
[8] R.A. Jacobs, Increased rates of convergence through learning rate adaptation, Neural networks 1 (4) (1988) 295–307.
[9] B. Xu, A.A. Heidari, Z. Cai, H. Chen, Dimensional decision covariance colony predation algorithm: global optimization and high- dimensional feature selection, Artif. Intell. Rev. 56 (10) (2023) 11415–11471.
[10] T.S. Ayyarao, N.S.S. Ramakrishna, R.M. Elavarasan, N. Polumahanthi, M. Rambabu, G. Saini, B. Khan, B. Alatas, War strategy optimization algorithm: a new effective metaheuristic algorithm for global optimization, Ieee Access 10 (2022) 25073–25105.
[11] B. Alhijawi, A. Awajan, Genetic algorithms: theory, genetic operators, solutions, and applications, Evolutionary Intelligence 17 (3) (2024) 1245–1256.
[12] E. Reyes-Davila, E.H. Haro, A. Casas-Ordaz, D. Oliva, O. Avalos, Differential evolution: a survey on their operators and variants, Arch. Comput. Methods Eng. 32 (1) (2025) 83–112.
[13] B. Khurshid, S. Maqsood, A hybrid evolution strategies-simulated annealing algorithm for job shop scheduling problems, Eng. Appl. Artif. Intell. 133 (2024) 108016.
[14] L.S. Kong, M.B. Jasser, S.-S.M. Ajibade, A.W. Mohamed, A systematic review on software reliability prediction via swarm intelligence algorithms, J. King Saud Univ.-Comput. Inf. Sci. (2024) 102132.
[15] F. Zhao, F. Ji, T. Xu, N. Zhu, et al., Hierarchical parallel search with automatic parameter configuration for particle swarm optimization, Appl. Soft Comput. 151 (2024) 111126.
[16] J. Cui, L. Wu, X. Huang, D. Xu, C. Liu, W. Xiao, Multi-strategy adaptable ant colony optimization algorithm and its application in robot path planning, Knowl.-Based Syst. 288 (2024) 111459.
[17] M. Abdel-Basset, L.A. Shawky, Flower pollination algorithm: a comprehensive review, Artif. Intell. Rev. 52 (2019) 2533–2557.
[18] L.I. Tong, Z.-T. WANG, Application of plant growth simulation algorithm on solving facility location problem, Syst. Eng.-Theory Pract. 28 (12) (2008) 107–115.
[19] Z. Zhang, Y. Fu, K. Gao, H. Zhang, L. Wang, A cooperative evolutionary algorithm with simulated annealing for integrated scheduling of distributed flexible job shops and distribution, Swarm Evol. Comput. 85 (2024) 101467.
[20] C. Gong, N. Zhou, S. Xia, S. Huang, Quantum particle swarm optimization algorithm based on diversity migration strategy, Futur. Gener. Comput. Syst. 157 (2024) 445–458.
[21] M. Zhou, M. Cui, D. Xu, S. Zhu, Z. Zhao, A. Abusorrah, Evolutionary optimization methods for high-dimensional expensive problems: a survey, IEEE/CAA J. Autom. Sin. 11 (5) (2024) 1092–1105.
[22] S.C. Chiam, K.C. Tan, A. Mamun al, Molecular dynamics optimizer, in: Evolutionary Multi-Criterion Optimization: 4th International Conference, EMO 2007, Matsushima, Japan, March 5-8, 2007. Proceedings 4, Springer, 2007, pp. 302–316.
[23] A.YS. Lam, V.OK. Li, Chemical reaction optimization: a tutorial, Memet. Comput. 4 (2012) 3–17.
[24] B. Alatas, Acroa: artificial chemical reaction optimization algorithm for global optimization, Expert Syst. Appl. 38 (10) (2011) 13170–13180.
[25] Z.-Z. Liu, D.-H. Chu, C. Song, X. Xue, B.-Y. Lu, Social learning optimization (slo) algorithm paradigm and its application in qos-aware cloud service composition, Inf. Sci. 326 (2016) 315–333.
[26] A. Maheri, S. Jalili, Y. Hosseinzadeh, R. Khani, M. Miryahyavi, A comprehensive survey on cultural algorithms, Swarm Evol. Comput. 62 (2021) 100846.
[27] D. Chauhan, R. Cheng, et al., Competitive swarm optimizer: a decade survey, Swarm Evol. Comput. 87 (2024) 101543.
[28] X.-L. Li, R. Serra, J. Olivier, A multi-component PSO algorithm with leader learning mechanism for structural damage detection, Appl. Soft Comput. 116 (2022) 108315.
[29] X. Lin, Z. Meng, An adaptative differential evolution with enhanced diversity and restart mechanism, Expert Syst. Appl. 249 (2024) 123634.
[30] D. Varela, J. Santos, Niching methods integrated with a differential evolution memetic algorithm for protein structure prediction, Swarm Evol. Comput. 71 (2022) 101062.
[31] H. Qiu, C. Sun, X. Wang, W. Sun, Q. Zhou, A population-based game-theoretic optimizer for the minimum weighted vertex cover, Appl. Soft Comput. 116 (2022) 108272.
[32] O. Cicek, Y.B. Özçelik, A. Altan, A new approach based on metaheuristic optimization using chaotic functional connectivity matrices and fractal dimension analysis for ai-driven detection of orthodontic growth and development stage, Fractal and Fractional 9 (3) (2025) 148.
[33] E. Belge, A. Altan, R. Hacıoğlu, Metaheuristic optimization-based path planning and tracking of Quadcopter for payload hold-release mission, Electronics 11 (8) (2022) 1208.
[34] G. Owen, Game Theory, Emerald Group Publishing, 2013.
[35] L. Sabug Jr, F. Ruiz, L. Fagiano, Smgo-δ: balancing caution and reward in global optimization with black-box constraints, Inf. Sci. 605 (2022) 15–42.
[36] W. Luo, X. Lin, C. Li, S. Yang, Y. Shi, Benchmark functions for cec 2022 competition on seeking multiple optima in dynamic environments, arXiv preprint arXiv:2201.00523, 2022.
[37] Q. Zhang, H. Gao, Z.-H. Zhan, J. Li, H. Zhang, Growth optimizer: a powerful metaheuristic algorithm for solving continuous and discrete global optimization problems, Knowl.-Based Syst. 261 (2023) 110206.
[38] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris Hawks optimization: algorithm and applications, Future Gener. Comput. Syst. 97 (2019) 849–872.
[39] M. Abdel-Basset, R. Mohamed, S.A.A. Azeem, M. Jameel, M. Abouhawwash, Kepler optimization algorithm: a new metaheuristic algorithm inspired by kepler's laws of planetary motion, Knowl.-Based Syst. 268 (2023) 110454.
[40] E. Tokgoz, Artificial bee colony optimization techniques' utilization for intrusion detection systems' analysis, in: 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), IEEE, 2025, pp. 1–16.
[41] N.K. Jain, U. Nangia, J. Jain, A review of particle swarm optimization, J. Inst. Eng. India Ser. B 99 (2018) 407–411.
[42] Z. Tian, M. Gai, Football team training algorithm: a novel sport-inspired meta-heuristic optimization algorithm for global optimization, Expert Syst. Appl. 245 (2024) 123088.
[43] Y. Gao, J. Zhang, Y. Wang, J. Wang, L. Qin, Love evolution algorithm: a stimulus–value–role theory-inspired evolutionary algorithm for global optimization, J. Supercomput. 80 (9) (2024) 12346–12407.
[44] A. Askarzadeh, M.A. Alipour, E2: a basic optimization method using exploration-exploitation concept, Soft Comput. (2025) 1–21.
[45] M. Alibabaei Shahraki, Cloud drift optimization algorithm as a nature-inspired metaheuristic, Discover Comput. 28 (1) (2025) 1–28.
[46] S. Fan, R. Wang, K. Su, A novel metaheuristic algorithm: advanced social memory optimization, Phys. Scr. 100 (5) (2025) 055004.
[47] U. Orozco-Rosas, O. Montiel, R. Sepúlveda, Mobile robot path planning using membrane evolutionary artificial potential field, Appl. Soft Comput. 77 (2019) 236–251.
[48] U. Orozco-Rosas, K. Picos, J.J. Pantrigo, A.S. Montemayor, A. Cuesta-Infante, Mobile robot path planning using a qapf learning algorithm for known and unknown environments, IEEE Access 10 (2022) 84648–84663.