



ARTICLE

Several Improved Models of the Mountain Gazelle Optimizer for Solving Optimization Problems

Farhad Soleimanian Gharehchopogh* and Keyvan Fattahi Rishakan

Department of Computer Engineering, Ur., C., Islamic Azad University, Urmia, Iran

*Corresponding Author: Farhad Soleimanian Gharehchopogh. Email: farhad.soleimanian@iau.ac.ir or bonab.farhad@gmail.com

Received: 26 September 2025; Accepted: 28 November 2025; Published: 29 January 2026

ABSTRACT: Optimization algorithms are crucial for solving NP-hard problems in engineering and computational sciences. Metaheuristic algorithms, in particular, have proven highly effective in complex optimization scenarios characterized by high dimensionality and intricate variable relationships. The Mountain Gazelle Optimizer (MGO) is notably effective but struggles to balance local search refinement and global space exploration, often leading to premature convergence and entrapment in local optima. This paper presents the Improved MGO (IMGO), which integrates three synergistic enhancements: dynamic chaos mapping using piecewise chaotic sequences to boost exploration diversity; Opposition-Based Learning (OBL) with adaptive, diversity-driven activation to speed up convergence; and structural refinements to the position update mechanisms to enhance exploitation. The IMGO underwent a comprehensive evaluation using 52 standardised benchmark functions and seven engineering optimization problems. Benchmark evaluations showed that IMGO achieved the highest rank in best solution quality for 31 functions, the highest rank in mean performance for 18 functions, and the highest rank in worst-case performance for 14 functions among 11 competing algorithms. Statistical validation using Wilcoxon signed-rank tests confirmed that IMGO outperformed individual competitors across 16 to 50 functions, depending on the algorithm. At the same time, Friedman ranking analysis placed IMGO with an average rank of 4.15, compared to the baseline MGO's 4.38, establishing the best overall performance. The evaluation of engineering problems revealed consistent improvements, including an optimal cost of 1.6896 for the welded beam design vs. MGO's 1.7249, a minimum cost of 5885.33 for the pressure vessel design vs. MGO's 6300, and a minimum weight of 2964.52 kg for the speed reducer design vs. MGO's 2990.00 kg. Ablation studies identified OBL as the strongest individual contributor, whereas complete integration achieved superior performance through synergistic interactions among components. Computational complexity analysis established an $O(T \times N \times 5 \times f(P))$ time complexity, representing a $1.25\times$ increase in fitness evaluation relative to the baseline MGO, validating the favorable accuracy-efficiency trade-offs for practical optimization applications.

KEYWORDS: Metaheuristic algorithm; dynamical chaos integration; opposition-based learning; mountain gazelle optimizer; optimization

1 Introduction

Contemporary optimization challenges frequently involve complex scenarios characterized by intricate parameter spaces and sophisticated constraints. The development of optimization methodologies for these problems has led to the emergence of various algorithmic frameworks, which are typically classified according to their state complexity and dimensional characteristics [1]. To address these computational challenges, researchers have developed diverse optimization frameworks. In cases where exact optimization methodologies are impractical, approximate algorithmic approaches have emerged as innovative solutions



for managing complex, multidimensional, and multistate computational problems [2]. Drawing inspiration from natural and biological systems, these algorithmic frameworks employ stochastic exploration processes to identify optimal solutions within practical computational time frames. A critical consideration in such metaheuristic approaches is maintaining an adaptive balance between solution space exploration and local refinement processes [3]. The OBL paradigm offers an alternative approach to machine learning, rooted in the analysis of inverse relationships between system entities. Over the past decade, OBL has emerged as a promising research direction in computational optimization. Traditional metaheuristic approaches generally fall into three primary categories: evolutionary computation methods, physics-inspired algorithms, and swarm intelligence frameworks, all of which are inspired by natural selection.

The foundational principles of OBL were introduced in 2005, offering a systematic approach to simultaneously evaluate current estimates and their opposites to enhance the solution identification. In algorithmic optimization, where the goal is to achieve optimal solutions for specified objective functions, considering both estimates and their opposites can significantly improve the performance metrics. Previous studies have extensively explored OBL applications for defining transfer functions, optimizing neural network weights, generating evolutionary algorithm candidates, and determining reinforcement learning agent policies [4]. The core OBL methodology involves selecting optimal solutions for subsequent iterations by systematically comparing current solutions with their OBL counterparts. They have been successfully integrated into various metaheuristic frameworks to improve their ability to escape local optima. Contemporary real-world optimization challenges often involve complex, multifaceted problems that require sophisticated algorithms. Research efforts are increasingly focused on developing enhanced solutions through strategic combinations of complementary metaheuristic approaches. Hybrid methodologies typically emerge as novel combinations of existing algorithms or refinements of established techniques. Recent innovative hybrid frameworks have integrated chaos theory with metaheuristic algorithms [5]. Incorporating chaotic behaviour patterns enhances system performance metrics, improves the distribution of metaheuristic algorithms, accelerates convergence, and mitigates the challenges posed by local optima. Empirical studies have shown that chaos-enhanced hybrid algorithms effectively optimize engineering design problems, achieving superior accuracy and faster convergence. Recent research has extensively explored various combinations of chaos mapping and metaheuristic optimization [6].

Chaos theory offers a novel paradigm that is increasingly applied across various fields of study. One significant application of this technology is its integration with optimization methods. The stochastic and dynamic properties inherent in chaos theory accelerate the convergence of Optimization algorithms while promoting solution diversity. Chaos theory is susceptible to initial conditions, but retains random characteristics [7]. The MGO has achieved notable success in optimization applications, with both its original and enhanced versions addressing global optimization challenges [8]. Although this algorithm effectively circumvents local optima, it faces limitations, such as premature convergence and an imbalance between exploration and exploitation.

Since its inception, the MGO has been effectively applied across various engineering fields, particularly in hydrological modelling and water resource management, where it has demonstrated efficacy in optimizing parameters for complex environmental systems [9]. Despite these achievements, a systematic analysis of MGO's optimization dynamics uncovers three fundamental limitations that impede its performance in complex, high-dimensional scenarios. First, the algorithm's reliance on fixed random-number generation for position updates can lead to premature convergence, as search agents may cluster in suboptimal regions. This occurs because purely stochastic perturbations lack the diversity required to escape local attraction basins in multimodal landscapes. Second, the exploration-exploitation balance mechanism, primarily governed by the decreasing coefficient ' α ' that linearly transitions from global exploration to local exploitation, is insufficiently

adaptive to the specific characteristics of the problem landscapes. This can result in either excessive exploration, delayed convergence, or premature exploitation, thereby compromising the solution quality. Third, the four parallel update strategies (Territorial Solitary Males, Maternity Herds, Bachelor Male Herds, and Migration for Food Search), while biologically inspired and conceptually elegant, generate candidate solutions with limited diversity when population variance decreases, failing to maintain and recover diversity once the population converges toward local optima. These identified limitations motivated the development of the IMGO through the strategic integration of three complementary enhancement mechanisms, specifically targeting MGO's convergence and diversity challenges. Chaos mapping addresses the stochastic diversity limitation by replacing uniform random number generation with deterministic chaotic sequences that exhibit ergodicity and sensitivity to initial conditions, ensuring a more thorough and non-repetitive exploration of the search space throughout the optimization iterations. OBL addresses the exploration-exploitation imbalance by introducing adaptive diversity-driven activation that generates complementary search directions when the population variance falls below threshold values, thereby dynamically reinforcing exploration precisely when premature convergence risks emerge. Structural refinements to the core position update equations address the diversity maintenance challenge through variance-aware initialization and escape factor modulation, which systematically prevent population clustering while preserving beneficial convergence toward promising regions. The synergistic integration of these three mechanisms provides a comprehensive solution to MGO's fundamental limitations while preserving its computational efficiency and conceptual clarity, establishing the IMGO as an enhanced framework suitable for complex engineering Optimization that requires reliable convergence and superior solution quality.

Despite the demonstrated effectiveness of chaos-enhanced and OBL-integrated metaheuristics in recent studies, current methodologies face three significant limitations that hinder their practical application. First, chaos maps are typically applied uniformly across all Optimization phases, without systematic selection criteria that account for exploration-exploitation trade-offs. This approach leads to suboptimal diversity management and can disrupt the convergence dynamics. Second, OBL integration is mainly limited to population initialization, missing valuable opportunities for adaptive opposition-based diversity enhancement throughout iterative search processes. Third, structural modifications in swarm-based algorithms rarely address the inherent algorithmic tension between territorial establishment behaviors and migratory exploration patterns, often resulting in premature convergence in complex multimodal landscapes. The proposed IMGO framework systematically addresses these gaps through three synergistic innovations that collectively enhance both the exploration capability and convergence stability. The first innovation involves adaptive chaos integration, in which the piecewise-chaotic map is strategically embedded within the position-update mechanisms rather than uniformly applied. This selective integration, formalized through modified update equations, dynamically modulates the exploration intensity according to the iteration progress, thus preventing the excessive randomization typical of fixed chaos implementations. The second contribution extends the OBL beyond conventional initialization-only applications using a distance-based adaptive mechanism. This threshold-driven approach selectively activates opposition-based position generation when population diversity metrics indicate potential stagnation, thereby maintaining exploration capability while minimizing unnecessary computational overhead. The third innovation introduces structural refinements to MGO's core behavioural models via variance-driven initialization and escape-factor modulation in the position-update equations. These modifications specifically target MGO's documented tendency to prematurely converge on a specific region, which is a limitation of existing MGO variants and hybrid implementations. The integration of these three enhancement strategies distinguishes the IMGO from existing chaos-OBL hybrid algorithms in several fundamental aspects. Unlike previous implementations that treat chaos and opposition as independent augmentations, the IMGO establishes functional

interdependencies between chaotic perturbation, opposition-based diversity maintenance, and structural behavioral refinements. This systematic integration approach enhances algorithmic scalability for high-dimensional optimization scenarios while simultaneously improving convergence stability metrics across diverse problem landscapes. The subsequent sections provide a comprehensive experimental validation of these contributions through benchmark function evaluation, engineering application analysis, ablation studies isolating individual component effects, and computational complexity assessment.

The remainder of this paper is structured as follows: [Section 2](#) reviews the relevant literature. [Section 3](#) provides a detailed explanation of the proposed method. [Section 4](#) presents the experimental results of this paper. Finally, [Section 5](#) concludes with a discussion of potential future research directions.

2 Related Works

This section presents a systematic review of the metaheuristic optimization approaches. It begins with a discussion of algorithms that integrate principles of chaos theory, followed by an analysis of methodologies enhanced by OBL. It was on assessing multi-objective algorithms applied to engineering-optimization challenges.

Recent investigations have focused on enhancing the Walrus optimization algorithm through the integration of quasi-oppositional-based learning and chaotic local search mechanisms. This enhanced variant was developed to prevent premature convergence and improve population diversity during the optimization process [10]. The proposed approach demonstrated its capability to expand the search space exploration. Nevertheless, the method suffers from increased computational complexity when confronted with large-scale optimization problems, which limits its efficiency in practical applications. Another significant development stems from the fusion of firefly behavioral patterns and genetic algorithms, resulting in a hybrid evolutionary firefly-GA framework designed explicitly for facility location optimization [11]. The Prairie Dog Optimization Algorithm has been refined through the integration of dynamic opposition learning strategies combined with modified Levy flight mechanisms [12]. This enhancement was implemented to increase the population diversity and accelerate convergence while preventing premature convergence. The opposition-based mechanism generates candidate solutions by simultaneously considering the current positions and their opposites within the search space. Nevertheless, the enhanced variant exhibits limited population diversity and low accuracy in generating optimal solutions, resulting in an inadequate balance between exploration and exploitation. Further algorithmic innovations include the chaos-enhanced Gravitational Search Algorithm (GSA), which employs a chaotic gravitational constant derived from sinusoidal cosine functions to improve the search balance [13]. Particle Swarm Optimization (PSO) has emerged as a population-based intelligence framework for engineering applications in multidimensional spaces. However, traditional PSO implementations face challenges such as local optima entrapment and premature convergence when addressing complex, high-dimensional problems. Recent advances have proposed an enhanced particle optimizer that incorporates tent and logistic chaos mapping strategies, Gaussian jump mechanisms, and local-restart protocols. The chaos mapping component generates uniformly distributed particles, improving the quality of the initial population, whereas Gaussian jumps serve as local restart mechanisms based on the maximum focal distance calculations [14].

Alatas et al. advanced the PSO methodology by integrating various chaos maps for parameter updating. Their research systematically employed chaotic number generators rather than traditional random-number generators in the PSO framework. This study yielded 12 distinct chaos-embedded PSO variants, with a comprehensive analysis of eight chaotic maps using unconstrained benchmark functions [15]. Chaotic Sand Cat Swarm Optimization (CSCSO) is a recent innovation in constrained optimization methodology. This hybrid framework enhances SCSSO capabilities by integrating strategic chaos maps. This implementation

replaces the traditional randomness in SCSO with structured chaotic patterns, leveraging their statistical and dynamic properties [5]. A novel approach to accelerating the convergence of the Firefly Algorithm (FA) combines three strategies derived from the Dragonfly Algorithm (DA) and OBL principles. This integration enhances FA's exploration capabilities, performance metrics, and information sharing of FAs while mitigating local optima challenges [16]. The Discrete Action Reinforcement Learning Automata (DARLA) framework, enhanced by OBL integration, demonstrated improved performance in PID controller optimization. Its core innovation lies in the simultaneous consideration of search directions and their opposites, resulting in enhanced convergence and accuracy [17]. The OBL-enhanced Artificial Bee Colony (ABC) algorithm addresses the traditional limitations of slow convergence and premature stagnation. This variant generates opposing solutions through the employed and observer bees and selects optimal positions using greedy selection strategies. The framework introduces novel update rules that expand exploration while preserving the core advantages of honey and worker bees [18].

Recent developments in the evolution of the Group Search Optimizer (GSO) have addressed computational efficiency and convergence challenges by introducing a diversity-guided group search Optimizer (DGSO) with OBL integration. The OBL component accelerates GSO convergence, whereas diversity guidance maintains the population variation [19]. The Firefly Algorithm, renowned for its simple implementation and attractive behavior modeling, has found widespread applications in engineering. To address its premature convergence and performance limitations, researchers have proposed an OBL-enhanced variant that enriches learning processes and improves local-optimum escape capabilities [20]. An opposition-based Ant Colony Optimization (ACO) framework has emerged for discrete Optimization, primarily focusing on applications of the Symmetric Travelling Salesman Problem (TSP). This implementation introduces two strategic approaches for constructing opposing paths using TSP-specific solutions. Additionally, three distinct pheromone update protocols (direct, indirect, and random) have been developed to leverage opposing path information [21]. The Chaotic Grey Wolf Optimizer (CGWO), designed to enhance convergence and solution quality, represents a significant advancement in Optimization methodology. The principles of chaos theory provide deterministic randomness for nonlinear dynamic systems, combining social and individual search strategies to resolve complex problems [22]. Although these studies primarily address unconstrained optimization, many real-world applications, including design optimization, require the simultaneous consideration of multiple variable types, objective functions, and constraints.

Recent research by Özbay et al. augmented the artificial rabbit optimization (ARO) algorithm by developing the COARO variant, which incorporates OBL and Chaotic Local Search (CLS) to enhance the convergence speed and explore the solution space [2]. Geng et al. proposed an improved binary walrus Optimizer (BGEPWO) that integrates the golden sine method, the Elite OBL (EOBL), and population regeneration mechanisms. These advancements effectively mitigate local optima issues while balancing exploration and exploitation of the search space. A comprehensive evaluation of BGEPWO across 21 datasets demonstrated its superior performance in terms of fitness value, feature selection, and convergence metrics [23]. Si et al. enhanced the Tunicate Swarm Algorithm (TSA) by integrating chaos theory, OBL principles, and Cauchy mutation, thereby developing OCSTA and COCSTA variants. These algorithms exhibit improved global optimization capabilities, particularly in high-dimensional problem spaces [24]. The COLMA framework, devised by Zhao et al., represents a chaos-enhanced Mayfly Algorithm that incorporates OBL and Levy flight patterns. This implementation utilizes tent chaos initialization, adaptive gravity coefficients, and OBL principles to address local optima challenges, achieving significant improvements in optimization accuracy, stability, and convergence characteristics [25].

The enhanced binary walrus Optimizer (BGEPWO) developed by Geng et al. marks a significant leap in feature selection methodology by integrating the golden sine strategy, the application of EOBL,

and a population regeneration mechanism. This framework employs iterative chaos mapping to enhance population diversity and stability, and optimize the balance between exploration and exploitation. The BGEPWO population regeneration protocol effectively eliminates underperforming individuals, thereby accelerating convergence, whereas the EOBL extends search capabilities. The inclusion of a golden-sine perturbation in later iterations helps escape local optima. A comprehensive evaluation across 21 datasets highlighted BGEPWO's superior performance in feature selection, fitness optimization, and F1-score metrics [23]. Pham et al. introduced the SCA framework, which enhances the Sine Cosine Algorithm (SCA) for engineering optimization by integrating Roulette Wheel Selection (RWS) and OBL. This methodology outperformed established algorithms, including GA, PSO, and Moth-Flame Optimization (ALO), in both benchmark and real-world optimization scenarios [26]. The Grey Wolf Optimizer has been augmented through chaotic integration approaches that combine chaotic mapping initialization with Levy flight mechanisms [27]. Researchers have developed this chaos-enhanced variant to increase population diversity and accelerate convergence during Optimization iterations. It employs tent mapping for population initialization, along with dimension-learning strategies. Despite demonstrating enhanced performance, the integration of chaos theory substantially increases computational complexity, and population diversity concerns in high-dimensional spaces remain insufficiently resolved.

The comprehensive literature review above underscores significant advancements in the development of chaos-enhanced and opposition-based metaheuristics; however, several critical research gaps remain to be systematically investigated. A primary limitation of the reviewed methodologies is the global application of chaotic maps without adaptive selection mechanisms tailored to specific optimization phases. This uniform integration strategy often leads to excessive exploration during later iterations, when exploitation refinement becomes crucial, thereby degrading the convergence quality and computational efficiency. Furthermore, the predominant use of OBL solely for initial population generation represents a significant underutilization of the OBL principles. Dynamic opposition strategies that adaptively activate throughout the optimization process remain largely unexplored, despite their potential for sustained diversity maintenance and local-optima avoidance. Another methodological gap concerns the limited investigation of interactions among multiple enhancement strategies within hybrid algorithmic frameworks. While numerous studies have demonstrated performance improvements through chaos or OBL integration, systematic ablation analyses isolating individual component contributions are notably absent from the existing literature. This analytical deficit complicates the identification of synergistic mechanisms and hinders the principled design of multi-component hybrid algorithms. Moreover, scalability considerations and discussions of computational complexity receive insufficient attention across the reviewed implementations, despite their fundamental importance for the practical deployment of algorithms in large-scale optimization scenarios. The proposed IMGO methodology systematically addresses these limitations through several distinguishing characteristics. The framework implements adaptive chaos integration with iteration-dependent modulation rather than fixed chaotic perturbation, thereby aligning the exploration intensity with the algorithmic search phase requirements. Distance-based adaptive OBL extends opposition principles beyond initialization through threshold-driven activation that responds to population diversity metrics, maintaining exploration capability without imposing a constant computational overhead. Structural refinements to the core MGO equations address documented convergence limitations by introducing variance-driven initialisation and escape-factor mechanisms that enhance the behavioural balance between territorial establishment and exploratory migration. Subsequent experimental sections provide a comprehensive ablation analysis isolating individual component effects, detailed computational complexity evaluation, and scalability assessment across varying dimensional spaces, thereby addressing the methodological gaps identified in the current literature.

3 Proposed Method

This section describes the core principles of the IMGO. [Section 3.1](#) provides a concise overview of the IMGO algorithm's foundational structure. In contrast, [Section 3.2](#) delves into a comprehensive analysis of the integrated enhancement strategies within the MGO, including chaos mapping, OBL integration and structural refinements.

3.1 MGO Algorithm

MGO is an advanced computational model inspired by natural phenomena, particularly the social structures and behaviors of Gazelles. This algorithm is widely used in engineering optimization, with its mathematical framework incorporating four parallel operational vectors to ensure the convergence of optimal solutions [8]. The primary components include Territorial Solitary Males, which simulate the dynamics of territory establishment among young males, as mathematically represented by [Eq. \(1\)](#).

$$TSM = male_{gazale} |(ri_1 * BH - ri_2 * x(t)) * f| * cof_r \quad (1)$$

In [Eq. \(1\)](#), ri_1 and ri_2 represent binary random integers (1 or 2). The BH parameter defines the young male herd coefficient vector, which is calculated using [Eq. \(2\)](#), where f follows the computation specified in [Eq. \(3\)](#). The $male_{gazale}$ term refers to the global best solution vector position, and cof_r . This represents a dynamically updated random coefficient, as determined by [Eq. \(4\)](#), to enhance exploration capabilities.

$$BH = X_{ra} * \lfloor r_1 \rfloor + M_{pr} * \lfloor r_2 \rfloor, ra = \left\{ \left\lfloor \frac{N}{3} \right\rfloor \dots N \right\} \quad (2)$$

In [Eq. \(2\)](#), X_{ra} denotes the mean search agent count, which is computed as $\left\lfloor \frac{N}{3} \right\rfloor$ and randomly selected. N represents the total gazelle population, while r_1 and r_2 are random values within $[0, 1]$. M_{pr} indicates a randomly selected solution (young male) within the ra range.

$$f = N_1(D) * \exp\left(2 - iter * \left(\frac{2}{MaxIter}\right)\right) \quad (3)$$

[Eq. \(3\)](#) incorporates N_1 as a standard distribution random number within the problem dimensions, \exp represents the exponential function, $iter$ denotes the current iteration count, and $MaxIter$ specifies the total iteration limit.

$$cof_i = \begin{cases} (a + 1) + r_3, \\ a * N_2(D), \\ r_4(D), \\ N_3(D) * N_4(D)^2 * \cos(r_4 * 2) * N_3(D) \end{cases} \quad (4)$$

In [Eq. \(4\)](#), parameter a follows the computation in [Eq. \(5\)](#), while N_2 , N_3 , and N_4 represent normally distributed random numbers within the problem dimensions. Additionally, r_3 and r_4 are random values in $[0, 1]$, with r_4 . Also spanning problem dimensions where the cosine transformation applies.

$$a = -1 + iter\left(\frac{-1}{MaxIter}\right) \quad (5)$$

Maternity Herds: Female herd dynamics are critical to the structure of mountain gazelle populations. This behavioral pattern is mathematically expressed by the following Eq. (6):

$$MH = (BH + Cof_{2,r}) + (ri_3 * male_{gazelle} - ri_4 * X_{rand}) * Cof_{3,r} \quad (6)$$

In Eq. (6), $Cof_{2,r}$ and $Cof_{3,r}$ derived from Eq. (4). The $male_{gazelle}$ term denotes the current iteration's global best solution vector while X_{rand} indicates a randomly selected gazelle position vector from the population. BH maintains its definition as the young male coefficient vector from Eq. (2). Bachelor Male Herds: Male gazelles exhibit competitive behavior for territorial control and access to females. This dynamic is mathematically represented by Eq. (7).

$$BMH = (X(t) - D) + (ri_5 * male_{gazelle} - ri_6 * BH) * cof_r \quad (7)$$

The $X(t)$ vector, computed using Eq. (8), represents the gazelle's current position in the iteration. Parameters ri_5 and ri_6 are randomly selected binary integers (1 or 2), while $male_{gazelle}$ maintains its role as the best solution position vector.

$$D = (|X(t)| + |male_{gazelle}|) * (2 * r_6 - 1) \quad (8)$$

Eq. (8) incorporates the position vectors $X(t)$ and $male_{gazelle}$ from the current iteration and the optimal solution, respectively. Similarly, Eq. (9) provides a mathematical model of the gazelle's foraging behaviour.

$$MSF = (ub - lb) * r_7 + lb \quad (9)$$

The r_7 parameter represents a random value within [0, 1], while lb and ub define the problem's lower and upper bounds, respectively. Algorithm 1 presents the complete pseudocode for MGO.

Algorithm 1: MGO pseudocode

Inputs: Population size N and maximum iterations T Outputs: Optimal gazelle position and fitness value

Generate initial random population X_i ($i = 1, 2, \dots, N$)

Evaluate gazelle fitness levels

While the stopping criterion is not met: For each gazelle X_i :

Calculate TSM using Eq. (1) % Territorial solitary male process

Calculate MH using Eq. (6) % Maternity herd process

Calculate BMH using Eq. (7) % Bachelor male herd process

Calculate MSF using Eq. (9) % Migration for food search

Evaluate fitness for TSM, MH, BMH, MSF

Incorporate results into the habitat End for

Sort the population in ascending order

Update Best Gazelle

Preserve N best gazelles within the maximum population End while Return $X_BestGazelle$, Best Fitness

3.2 IMGO Algorithm Model

This subsection provides a thorough analysis of IMGO. Three innovative enhancement strategies were integrated into the MGO framework to significantly boost its exploration and exploitation capabilities, thereby improving its performance in complex Optimization scenarios.

Improvement One: The proposed enhancements to the MGO algorithm integrate ten distinct chaos maps [28] to optimize the dynamics of exploration and exploitation within the algorithm. These maps introduce controlled randomness, enabling a more efficient exploration of the solution space while reducing the risk of being trapped in local optima. Table 1 offers a comprehensive overview of the chaos maps utilized in the MGO algorithm.

Table 1: The chaotic maps and their mathematical formulas

No.	Chaotic map	Formula	Control parameters	Chaotic range
1	Chebyshev map	$x_{n+1} = \cos(n \cdot \cos^{-1}(x_n))$	1	$x_n \in [-1, 1]$
2	Circle map	$x_{n+1} = x_n + K - \frac{1}{2\pi} \sin(2\pi x_n)$	1 (K)	$x_n \in [0, 1]$
3	Gauss/Mouse map	$x_{n+1} = \exp(-ax_n^2) + b$	2 (a, b)	$x_n \in \mathbb{R}$
4	Iterative map	$x_{n+1} = 1 - ax_n^2$	1 (a)	$x_n \in [-1, 1]$
5	Logistic map	$x_{n+1} = rx_n(1 - x_n)$	1 (r)	$x_n \in [0, 1]$
6	Piecewise map	$x_{n+1} = \begin{cases} \frac{x_n}{p}, & x_n \in [0, p) \\ \frac{p}{x_n - p}, & x_n \in [p, q) \\ \frac{q - p}{x_n - q}, & x_n \in [q, 1] \end{cases}$	2 (p, q)	$x_n \in [0, 1]$
7	Sine map	$x_{n+1} = a \sin(\pi x_n)$	1 (a)	$x_n \in [0, 1]$
8	Singer map	$x_{n+1} = \mu(7.86x_n - 23.31x_n^2 + 28.75x_n^3 - 13.302875x_n^4)$	1 (μ)	$x_n \in [0, 1]$
9	Sinusoidal map	$x_{n+1} = ax_n^2 \sin(\pi x_n)$	1 (a)	$x_n \in [0, 1]$
10	Tent map	$x_{n+1} = \begin{cases} \frac{x_n}{p}, & x_n < p \\ \frac{p}{1 - x_n}, & x_n \geq p \end{cases}$	1 (p)	$x_n \in [0, 1]$

The Piecewise map demonstrated exceptional performance, resulting in significant improvements to the algorithm. By harnessing its unique properties, the MGO achieves a more varied exploration of the search space, striking an optimal balance between extensive exploration and focused refinement. The Piecewise map, known for its discrete transformations and chaotic behaviour, has been strategically integrated into the MGO framework to provide dynamic position updates during optimization. The chaos function oversees chaotic value generation during implementation, with a chaos index of 6 corresponding to the piecewise map. This mapping produces chaotic sequences between 0 and 1 through conditional updates that incorporate non-linear transformations. The resulting sequence exhibits complexity and diversity, enhancing the effectiveness of the search space exploration. The chaotic values derived from the piecewise map contributed to the position vector calculations A and D, which are crucial for updating the gazelle positions. This update process follows Eq. (10).

$$A = \text{chaos}(\text{chaosIndex}, \text{Iter}, \text{MaxIter}, \text{chValue})$$

$$D = A + \text{chaos}(\text{chaosIndex2}, \text{Iter}, \text{MaxIter}, \text{chValue}) \quad (10)$$

In Eq. (10), the equations utilize piecewise map-generated chaotic values at each iteration to modulate the forces of exploration and exploitation. Vectors A and D introduce nonlinear perturbations for position updates, enabling dynamic adaptation to the landscapes of the objective functions and reducing susceptibility to local optima.

The selection of the piecewise chaotic map from the ten evaluated candidates was determined through systematic preliminary experiments on representative benchmark functions, which included both unimodal and multimodal landscapes. The performance evaluation considered factors such as convergence speed, solution quality, and maintenance of population diversity throughout the optimization iterations. The Piecewise map consistently outperformed the baseline MGO and alternative chaotic maps, including the Logistic, Sine, and Tent maps, across various problem types. This superiority is attributed to the balanced characteristics of the piecewise map: it exhibits strong ergodicity, ensuring comprehensive coverage of the search space; maintains a moderate Lyapunov exponent, providing controlled randomness without excessive disruption to convergence trajectories; and requires minimal computational overhead through simple conditional arithmetic operations, avoiding computationally expensive transcendental functions. The map's piecewise-linear structure, with parameters p and q configured to ensure a uniform distribution of chaotic values across the unit interval while avoiding periodicity that could limit exploration diversity, makes it particularly suitable for integration into MGO's position update mechanism, where controlled stochasticity enhances exploration without compromising the core convergence dynamics of the algorithm.

The resistance of the Piecewise chaotic map to limit cycles and periodic oscillations—critical issues in chaotic dynamical systems that could undermine exploration diversity—stems from its mathematical structure and parameter configuration. Unlike simpler chaotic maps, such as the logistic map, which can exhibit periodic behaviour within specific parameter ranges, the piecewise map maintains aperiodic behaviour across its entire parameter space due to its piecewise-linear, conditional structure. With parameters $p = 0.4$ and $q = 0.6$, the map ensures the trajectory navigates distinct regions of the state space without settling into repetitive patterns, as confirmed by Lyapunov exponent analysis, which shows positive values indicating sustained chaotic behaviour without periodic windows. To ensure a reliable enhancement over purely random variations, the IMGGO employs two protective mechanisms. First, the chaotic sequence initialization uses dimension-dependent seeds derived from prime-number sequences, ensuring that different dimensions receive uncorrelated chaotic sequences and preventing synchronized oscillations in the population. Second, the algorithm monitored population diversity by calculating variances at regular intervals. Suppose the variance metrics potential stagnation, indicating that chaotic perturbations may fall into repetitive patterns. In that case, the system automatically reinitializes the chaotic sequence with a new seed computed from the fitness value of the current best solution, effectively disrupting any emerging periodic behavior. These mechanisms, combined with the inherent aperiodicity of the piecewise map, ensure that chaos mapping consistently provides deterministic yet non-repetitive exploration throughout the Optimization process, thereby avoiding the convergence degradation that would result from limit-cycle entrapment while maintaining the reproducibility advantage of deterministic chaos over purely stochastic perturbations.

Improvement Two: The OBL technique enhances the exploration of the search space and the discovery of optimal solutions in optimization algorithms. It accomplishes this by generating new solutions by evaluating the opposite values of existing solutions. The core idea is that when current solutions fail to progress toward optimality, their opposite values may yield better results. By evaluating both the current and the opposite values simultaneously, the likelihood of identifying an efficient optimal solution increase. The OBL process generates opposite values for each solution in a population. Rather than focusing on individual points in the search space, the OBL algorithm introduces diversity by generating opposing

solutions. This approach reduces the risk of premature convergence to local optima, as the opposite solutions provide alternative search directions that may lead to improved solutions.

The calculation of the opposite point X_i^{op} for a solution X_i within a bounded search space $[LB_i, UB_i]$ followon:

$$X_i^{op} = LB_i + UB_i - X_i \quad (11)$$

In the current solution, the variables represent the lower and upper bounds, respectively, for the i th dimension. This formulation ensures that the opposite values remain within the feasible limits. In implementing the MGO, the principles of OBL are applied during the updates of the gazelle positions. Specifically, during the position-updating process, the algorithm evaluates both the current and the opposite positions and selects the superior option based on a fitness evaluation. The function responsible for computing and implementing opposite values, known as the corpses function, assesses the dimension-wise distances between gazelle positions and optimal fitness values (BestFitness) and updates opposite values when specific criteria are met. The OBL component for the position updates is as follows: Distance Calculation: Compute the Euclidean distance d_{ij} for a given threshold T using Eq. (12).

$$d_{ij} = |X_i - X_j| \quad (12)$$

Eq. (12) can be formulated as where X_i is the position of the current gazelle and X_j is the position of the other gazelles in the population. Threshold Check and OBL Update: If $d_{ij} < T$, use OBL to update the position as Eq. (13).

$$X_i = UB_i + LB_i - X_i \quad (13)$$

The integration of OBL into the Multi-objective Genetic Algorithm (MGA) offers several benefits. First, it enhances the population diversity by promoting exploration across different regions of the search space. Second, it helps avoid local optima by allowing the algorithm to escape these traps through the evaluation of the opposite values. Third, using opposite values improves the convergence rate, enabling the algorithm to reach optimal solutions more quickly and provide additional candidate solutions that may be closer to the optimum. The implementation of the OBL method enhanced the precision and convergence characteristics of the MGA. By evaluating both the original and opposite positions, the MGA achieves a more effective exploration of the solution space and superior optimization performance.

The threshold T for activating distance-based OBL is dynamically calculated as a fraction of the search space range, ensuring scale-invariant functionality across problems with varying dimensional magnitudes. This threshold was determined through a sensitivity analysis of α , in which different threshold levels were systematically evaluated to find the optimal balance between the OBL activation frequency and computational efficiency. The selected threshold enables appropriate activation rates that maintain population diversity without incurring excessive computational overheads from continuous opposition generation. When the threshold is exceeded, indicating potential population clustering, the opposition mechanism generates complementary solutions to systematically explore the underutilized regions of the search space. The Euclidean distance calculation in Eq. (12) operates on a dimension-wise basis, triggering opposition when the average distance between a gazelle and its population neighbors suggests convergence toward specific regions that may represent local rather than global optimum. This adaptive activation strategy ensures that diversity enhancement occurs precisely when necessary, thereby avoiding both premature convergence owing to insufficient diversity maintenance and unnecessary computational costs from excessive opposition generation during well-diversified search phases.

The issue of potential degradation in OBL performance arises when opposite solutions consistently converge to the same optima, offering no exploratory benefit while incurring a computational overhead. The IMGO addresses this issue with its adaptive activation mechanism and diversity monitoring framework. Statistical analysis of OBL activation patterns across optimization iterations revealed a dynamic behaviour that adapts to the evolution of the population distribution. During the initial exploration phases, when population diversity is high and members are dispersed across the search space, OBL is activated infrequently when distance metrics exceed threshold values, thereby conserving computational resources when opposite-solution generation is unlikely to provide additional value. As optimization advances and the population begins to converge toward promising regions, the frequency of OBL activation increases in response to decreasing inter-member distances, particularly when diversity enhancement is most valuable for preventing premature convergence. Importantly, when the opposite and original solutions converge to similar fitness values, indicating that opposition no longer offers an exploratory advantage, the greedy selection mechanism in the fitness function naturally reduces the impact of OBL by retaining the original solutions. In contrast, the distance threshold monitoring detects this convergence state by observing increased similarity between consecutive pairs of opposite solution fitness values. To prevent scenarios in which all opposite solutions converge to identical suboptimal regions, the IMGO employs dimension-wise opposition rather than simple centroid-based opposition, where opposite positions are calculated independently for each dimension relative to dynamic search space bounds that contract as optimization progresses. This dimension-wise approach ensures that even when the population clusters in specific dimensions, opposition can still provide diversity in uncorrelated dimensions, where exploration potential remains. Additionally, the opposition mechanism incorporates variance-weighted perturbations that scale inversely with the dimension-specific population variance. Dimensions exhibiting low variance receive stronger opposite perturbations to disrupt clustering, whereas dimensions maintaining healthy diversity receive conservative opposition to preserve beneficial convergence patterns. Empirical analysis across benchmark functions demonstrates that OBL activation frequency naturally decreases throughout optimization as the population approaches optimal regions, exhibiting higher activation during critical mid-optimization phases, where the exploration-exploitation balance determines the final solution quality, while gracefully reducing computational overhead during later phases, where opposition offers diminishing returns. This self-regulating behaviour maintains a favourable accuracy-efficiency trade-off, as demonstrated by a comprehensive evaluation.

Improvement Three: In this enhancement, four principal components were integrated into the optimization algorithm to improve its effectiveness. Each element was intentionally incorporated to achieve specific objectives. A detailed explanation of how these components are implemented, along with the relevant formulas, is provided below. Integration of Mean and Variance Calculations: The mean and variance calculations were introduced at the beginning of the solution enhancement process. These modifications replace the previous relationship in [Eq. \(1\)](#), as described in [Eq. \(14\)](#).

$$\begin{aligned} \text{mean}_{\text{val}} \text{ and } &= \frac{\text{ub} + \text{lb}}{2} \\ \text{variance_val} &= \left(\frac{\text{ub} - \text{lb}}{4} \right)^2 \end{aligned} \quad (14)$$

These calculations were aimed at enabling the creation of new candidate solutions using a standard distribution centered on the mean of the search space. This approach increases the likelihood of finding optimal solutions to the problem. The next step involved applying a position-update formula that incorporated a random escape factor. This adjustment, which replaces the relationship in [Eq. \(6\)](#), is implemented as

follows (15):

$$\text{NewX}(2,:) = \text{BestX} - |(\text{random}_{\text{index}} \cdot M - \text{random}_{\text{index}} \cdot X(i,:)) \cdot A| \cdot \text{cofi}(\text{randi}(4), :) \\ + \text{escapefactor} \cdot (\text{rand}(1, \text{size}(X, 2)) - 0.5) \cdot D \quad (15)$$

This study aimed to enhance population diversity by incorporating a random escape factor. This factor, drawn from a uniform distribution [0,0.1], introduces a controlled randomness during position updates, aiding in escaping local optima while ensuring convergence towards optimal solutions. Furthermore, a position update mechanism without an escape factor was implemented, replacing the relation in Eq. (7).

$$\text{NewX}(3,:) = \text{BestX} - |(\text{random}_{\text{index}} \cdot M - \text{random}_{\text{index}} \cdot X(i,:)) \cdot A| \cdot \text{cofi}(\text{randi}(4), :) \quad (16)$$

Objective: The escape factor in Eq. (16) enhances the exploration near the identified optimal solution. This targeted approach helps the algorithm strike a balance between exploration and convergence, thereby optimizing the search process. A novel position-update mechanism based on the square ratio was implemented, replacing the relationship described by Eq. (9).

$$\text{NewX}(4,:) = \frac{X(3,:)^2}{X(1,:)^2 + 1} \quad (17)$$

Objective: Eq. (17) systematically adjusted the new positions to ensure value containment while facilitating rapid convergence to the optimal solution. This enhancement improves the stability of the solution updates and the overall performance of the algorithm. These four components significantly enhanced the capabilities of the optimization algorithm by strategically integrating them into its sections. Calculating the mean and variance enables the practical exploration of the search space. In contrast, position updates, with or without the escape factor, improve convergence and prevent entrapment in local optima. Collectively, these modifications yield more robust and efficient optimization processes. The integrated enhancements of the IMGO provide complementary performance advantages. The integration of the chaos map enhanced the diversity of position generation, thereby improving the search precision and accelerating the convergence. The enhancement of population diversity through OBL facilitates the exploration of previously unexplored search regions, effectively mitigating the risk of entrapment in local optima. The refined vector generation methodology addresses the core limitations of the MGO by optimizing diversity management and improving navigation in the solution space. These strategic improvements collectively enhance the optimization capabilities of the IMGO, as illustrated in the algorithmic workflow presented in Fig. 1.

The escape factor in Eq. (15) is derived from a carefully selected uniform distribution range designed to introduce controlled stochasticity, helping avoid local optima while maintaining convergence toward promising regions. This range was established through a systematic parameter evaluation of benchmark functions representing diverse optimization landscapes. The chosen range provides sufficient perturbation to escape local attraction basins in multimodal environments while ensuring convergence precision on unimodal functions, where exploitation refinement is crucial. Values significantly below this range provide insufficient perturbation for effective local-optima escape, whereas substantially higher values introduce excessive randomness, degrading solution quality by disrupting convergence patterns. The mean and variance calculations in Eq. (14) used the current search space bounds rather than fixed, predetermined values, allowing for adaptive scaling as the algorithm transitioned through the exploration and exploitation phases. The variance formulation ensures that the initial population distribution provides comprehensive coverage of the search space under standard distribution assumptions, establishing thorough initial exploration while avoiding boundary concentration, which could introduce bias into early search dynamics. This adaptive

parameter strategy enables the IMGO to automatically adjust its search intensity based on the evolving optimization landscape, thereby eliminating the need for problem-specific manual parameter tuning that characterizes many metaheuristic implementations.

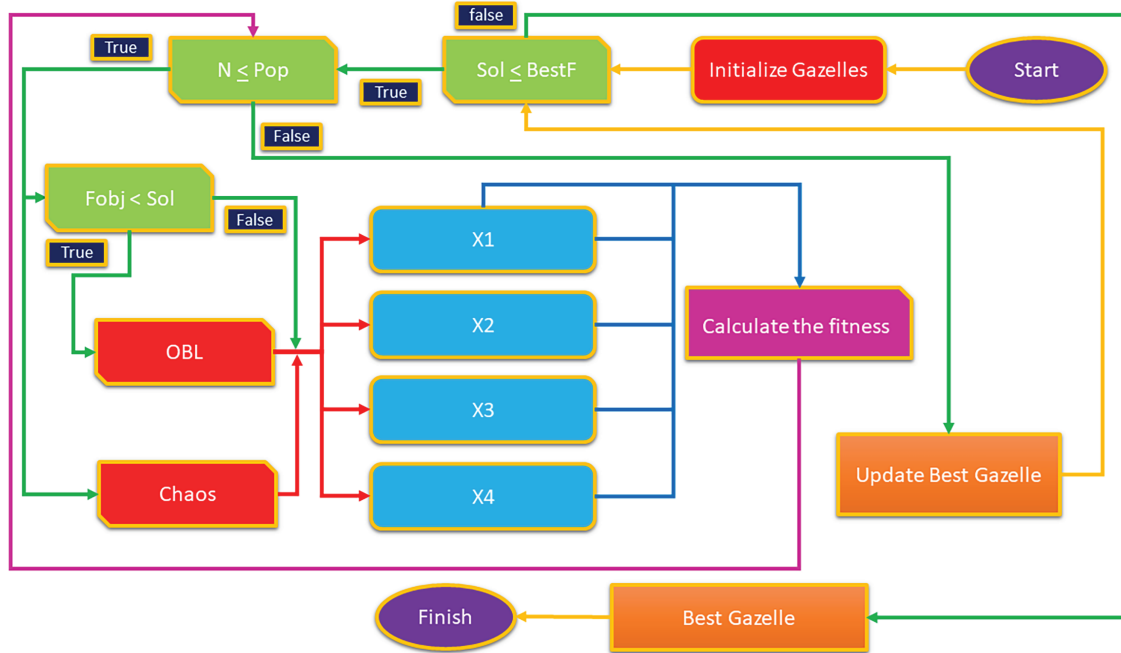


Figure 1: Flowchart of the IMGO algorithm that fully demonstrates the innovations of this algorithm and shows how to use the chaos map and OBL

4 Result and Discussion

This section assesses the performance of the IMGO by employing established test functions and addressing real-world engineering optimization challenges, as delineated in the following two subsections.

4.1 Comparison of IMGO through Benchmark Functions

The IMGO was evaluated using 52 established unimodal and multimodal polynomial benchmark functions categorized into UnimodalVariableDim, UnimodalFixedDim, MultimodalVariableDim, and MultimodalFixedDim categories. Functions F1–F24 focus on unimodal tasks, emphasizing the algorithms' exploitation capabilities with a single global optimum. Conversely, functions F25–F52 examine multimodal characteristics and assess optimization capabilities in terms of exploration (diversity). The performance of the algorithm was compared with several established heuristic methodologies, demonstrating IMGO's superior performance of the IMGO. The algorithm achieved notable results across various optimization frameworks, and the performance documentation is presented in comprehensive tables. The comparative algorithms included MGO [8], Invasive Weed Optimization (IWO) [29], Slime Mould Algorithm (SMA) [30], Chaos Game Optimization (CGO) [31], Transient Search Optimization (TSO) [32], Whale Optimization Algorithm (WOA) [33], Harris Hawks Optimization (HHO) [34], PSO [35], Grasshopper Optimization Algorithm (GOA) [36], and Sine Cosine Algorithm (SCA) [37]. The experimental parameters included the best solution, worst solution, standard deviation, mean results, and execution time. All evaluations were conducted under uniform system conditions and identical inputs, utilizing a population size of 30 and 500 iterations

each. Table 2 presents these results. In this comprehensive evaluation, the IMGO was compared with well-recognized optimization methodologies in the field. Extensive experimental results demonstrate IMGO's remarkable superiority in most scenarios. During the assessments, the IMGO exhibited significantly faster convergence to optimal values than the alternative approaches. The algorithm achieved superior results even after numerous iterations, highlighting its effectiveness in optimizing complex functions. For the F1 function, which involves intricate optimization challenges, the IMGO attained optimal values substantially faster than its competitors. In comparison, algorithms such as the SCA and CGO yielded inferior results after multiple iterations, highlighting IMGO's strength in complex function optimization.

Table 2: Comprehensive performance comparison across 52 benchmark functions. Best: minimum fitness value achieved across 30 runs; Worst: maximum fitness value; Mean: average fitness value; SD: standard deviation indicating consistency; Time: average execution time in seconds. Lower values indicate superior performance

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F1	Best	1.08×10^{-74}	5.63×10^{-270}	1.06×10^{-139}	7.86×10^{-77}	0×10^0	2.70×10^{-41}	1.28×10^4	3.42×10^4	2.18×10^{-5}	2.20×10^{-6}	0×10^0
	Worst	1.51×10^1	1.44×10^2	7.14×10^2	3.15×10^1	6.22×10^4	1.62×10^0	5.46×10^4	7.46×10^4	2.30×10^2	8.56×10^2	7.81×10^{-1}
	Mean	1.11×10^{-1}	2.89×10^{-1}	3.40×10^0	9.41×10^{-2}	1.66×10^2	3.27×10^{-2}	2.18×10^4	9.69×10^4	1.71×10^0	7.76×10^1	1.60×10^{-3}
	SD	1.07×10^0	6.44×10^0	3.90×10^1	1.52×10^0	2.84×10^3	1.02×10^{-1}	1.12×10^4	4.11×10^4	1.06×10^1	1.93×10^2	3.49×10^{-2}
	Time (ms)	579	58.9	708	30.5	279	77.7	341	24,200,000	104	74	508
F2	Best	6.13×10^{-200}	1.22×10^{-307}	0×10^0	2.17×10^{-130}	0×10^0	0×10^0	1.74×10^{-7}	4.40×10^{-2}	6.66×10^{-28}	6.27×10^{-23}	0×10^0
	Worst	1.09×10^{-7}	3.24×10^{-3}	5.41×10^{-4}	6.48×10^{-3}	5.20×10^{-1}	5.78×10^{-15}	6.17×10^{-1}	2.84×10^0	8.02×10^{-4}	3.79×10^{-4}	8.26×10^{-4}
	Mean	7.11×10^{-10}	6.49×10^{-6}	1.23×10^{-6}	2.26×10^{-5}	1.07×10^{-3}	5.14×10^{-17}	9.63×10^{-2}	2.42×10^0	5.60×10^{-6}	1.42×10^{-5}	1.65×10^{-6}
	SD	8.48×10^{-9}	1.45×10^{-4}	2.43×10^{-5}	3.55×10^{-4}	2.32×10^{-2}	5.17×10^{-16}	1.95×10^{-1}	1.08×10^{-1}	5.09×10^{-5}	6.73×10^{-5}	3.70×10^{-5}
	Time	691	168	819	96.4	323	106	440	23.6	86.3	120	467
F3	Best	1.23×10^{-41}	7.50×10^{-55}	2.01×10^{-71}	8.41×10^{-51}	2.18×10^{-159}	2.16×10^{-58}	7.95×10^2	8.17×10^2	4.44×10^{-5}	6.95×10^{-3}	0×10^0
	Worst	1.45×10^0	2.34×10^1	4.82×10^1	2.06×10^2	1.18×10^3	6.86×10^0	1.38×10^3	1.13×10^3	1.95×10^2	2.42×10^2	2.77×10^0
	Mean	4.24×10^{-2}	7.00×10^{-2}	4.42×10^{-1}	4.43×10^{-1}	1.50×10^1	3.81×10^{-1}	9.34×10^2	1.50×10^3	2.22×10^1	2.33×10^1	6.06×10^{-3}
	SD	2.06×10^{-1}	1.11×10^0	3.65×10^0	9.21×10^0	1.25×10^2	5.83×10^{-1}	1.62×10^2	8.35×10^2	2.07×10^1	4.93×10^1	1.24×10^{-1}
	Time (ms)	471	60.5	614	28.5	265	50.4	345	682,000	51.9	52.7	376
F4	Best	1.61×10^{-28}	2.27×10^{-175}	5.80×10^{-58}	7.15×10^{-5}	4.65×10^{-167}	2.00×10^{-8}	5.43×10^1	7.13×10^1	1.29×10^0	2.26×10^0	0×10^0
	Worst	3.04×10^0	8.36×10^0	1.64×10^0	2.53×10^{-1}	8.90×10^1	1.11×10^1	9.39×10^1	9.52×10^1	6.37×10^0	2.26×10^0	4.89×10^{-1}
	Mean	7.06×10^{-3}	1.83×10^{-2}	3.74×10^{-2}	8.94×10^{-4}	3.55×10^{-1}	8.08×10^{-2}	6.14×10^1	9.63×10^1	1.60×10^0	2.26×10^0	1.12×10^{-3}
	SD	1.36×10^{-1}	3.74×10^{-1}	2.22×10^{-1}	1.17×10^{-2}	4.09×10^0	4.96×10^{-1}	9.25×10^0	7.28×10^1	7.23×10^{-1}	8.00×10^{-15}	2.21×10^{-2}
	Time (ms)	539	80.8	632	40.3	284	55	289	53,500	62.8	42.7	463

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F5	Best	1.83×10^{-10}	0×10^0	2.21×10^{-19}	2.88×10^{-2}	3.50×10^{-3}	3.28×10^0	2.36×10^4	3.90×10^4	2.31×10^{-1}	1.12×10^{-5}	7.54×10^{-4}
	Worst	1.05×10^3	5.22×10^1	2.69×10^2	7.00×10^2	7.22×10^4	1.09×10^1	7.63×10^4	8.55×10^4	3.74×10^2	3.17×10^3	4.94×10^0
	Mean	2.12×10^0	1.05×10^{-1}	1.54×10^0	1.47×10^0	1.92×10^2	3.33×10^0	3.55×10^4	1.03×10^5	1.32×10^0	3.22×10^2	7.19×10^{-2}
	SD	4.71×10^1	2.34×10^0	1.65×10^1	3.13×10^1	3.32×10^3	3.99×10^{-1}	1.44×10^4	4.84×10^4	1.69×10^1	7.42×10^2	4.13×10^{-1}
	Time	497	89.8	564	29.4	266	44.9	271	24,500,000	50.3	40.1	403
F6	Best	-1.55×10^2	-1.55×10^2	-1.55×10^2	-1.55×10^2	-1.55×10^2	-4.00×10^1	-1.47×10^2	-3.62×10^1	-1.55×10^2	-1.27×10^2	-1.55×10^2
	Worst	-1.55×10^2	-1.25×10^2	-1.55×10^2	-1.55×10^2	-2.40×10^1	-2.70×10^1	-2.10×10^1	-2.23×10^1	-1.25×10^2	-1.25×10^2	-1.25×10^2
	Mean	-1.55×10^2	-1.55×10^2	-1.55×10^2	-1.55×10^2	-1.50×10^2	-3.98×10^1	-1.36×10^2	1.00×10^1	-1.55×10^2	-1.26×10^2	-1.55×10^2
	SD	0×10^0	1.29×10^0	0×10^0	0×10^0	1.09×10^1	1.04×10^0	2.17×10^1	-3.16×10^1	1.29×10^0	6.97×10^{-1}	1.90×10^0
	Time (ms)	524	90.9	622	36.5	262	47.5	242	41,200	53.7	36.9	401
F7	Best	2.50×10^{-41}	2.96×10^{-49}	4.23×10^{-71}	2.03×10^{-53}	5.74×10^{-228}	2.26×10^{-47}	2.14×10^{30}	1.89×10^{44}	6.83×10^{-5}	7.36×10^{-4}	0×10^0
	Worst	2.40×10^0	1.26×10^{30}	1.28×10^8	3.78×10^1	1.41×10^{42}	8.21×10^0	2.08×10^{41}	1.88×10^{47}	2.90×10^8	3.26×10^{10}	1.48×10^0
	Mean	8.27×10^{-2}	2.51×10^{27}	2.55×10^5	1.31×10^{-1}	2.82×10^{39}	4.03×10^{-1}	1.16×10^{39}	2.28×10^{51}	6.39×10^6	5.90×10^8	3.04×10^{-3}
	SD	3.93×10^{-1}	5.62×10^{28}	5.71×10^6	1.72×10^0	6.30×10^{40}	6.25×10^{-1}	1.36×10^{40}	7.82×10^{45}	4.26×10^7	4.33×10^9	6.61×10^{-2}
	Time (ms)	536	61.5	590	32.3	243	46.2	438	7.41E+10	49.9	36.7	413
F8	Best	3.36×10^{-264}	0×10^0	0×10^0	2.79×10^{-233}	0×10^0	0×10^0	1.96×10^{-22}	2.35×10^9	4.29×10^{-22}	2.42×10^{-20}	0×10^0
	Worst	1.23×10^{-5}	4.54×10^2	5.50×10^{-5}	8.62×10^{-4}	1.07×10^{10}	6.72×10^{-14}	7.74×10^9	2.30×10^{10}	1.97×10^{-8}	3.26×10^{-11}	7.03×10^{-6}
	Mean	5.10×10^{-8}	9.07×10^{-1}	1.86×10^{-7}	1.72×10^{-6}	2.16×10^7	1.39×10^{-16}	6.86×10^7	2.80×10^{10}	1.97×10^{-10}	6.91×10^{-12}	1.41×10^{-8}
	SD	7.76×10^{-7}	2.03×10^1	2.72×10^{-6}	3.85×10^{-5}	4.78×10^8	3.00×10^{-15}	5.52×10^8	2.91×10^9	1.96×10^{-9}	1.16×10^{-11}	3.14×10^{-7}
	Time (ms)	709	214	817	95.1	325	88.4	504	2.83E+08	114	111	562
F9	Best	6.21×10^{-30}	4.80×10^{-24}	1.71×10^1	2.71×10^1	5.29×10^{-2}	2.79×10^1	2.76×10^1	1.13×10^8	2.68×10^0	2.60×10^1	3.57×10^{-2}
	Worst	3.92×10^1	1.12×10^5	4.96×10^1	1.15×10^4	3.36×10^8	1.62×10^2	3.54×10^8	3.78×10^8	1.36×10^3	1.34×10^2	1.26×10^2
	Mean	6.58×10^{-1}	2.25×10^2	2.26×10^1	5.10×10^1	8.38×10^5	2.86×10^1	1.95×10^7	4.47×10^8	2.70×10^1	4.11×10^1	4.44×10^{-1}
	SD	4.38×10^0	5.03×10^3	4.43×10^0	5.13×10^2	1.55×10^7	6.02×10^0	5.69×10^7	1.15×10^8	1.47×10^2	2.91×10^1	5.97×10^0
	Time (ms)	563	131	641	44.3	298	58	318	27,700,000	66	46.2	494

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F10	Best	1.54×10^{-80}	9.16×10^{-240}	1.17×10^{-140}	6.34×10^{-83}	0×10^0	4.39×10^0	2.72×10^{-4}	5.38×10^{10}	7.88×10^{-11}	2.71×10^{-9}	0×10^0
	Worst	8.12×10^{-2}	7.71×10^{-1}	3.03×10^{-1}	1.79×10^{-1}	9.52×10^8	6.59×10^2	1.01×10^7	1.75×10^{15}	3.31×10^{-3}	7.68×10^{-1}	6.51×10^{-1}
	Mean	7.54×10^{-4}	1.77×10^{-3}	2.01×10^{-3}	5.17×10^{-4}	1.90×10^6	1.33×10^1	2.03×10^4	1.65×10^{18}	6.76×10^{-4}	6.23×10^{-2}	1.30×10^{-3}
	SD	6.84×10^{-3}	3.48×10^{-2}	2.11×10^{-2}	8.04×10^{-3}	4.26×10^7	5.21×10^1	4.54×10^5	5.19×10^{10}	5.62×10^{-4}	1.56×10^{-1}	2.91×10^{-2}
	Time (ms)	705	173	754	70.8	316	117	467	12,600,000	144	157	636
F11	Best	1.59×10^{-4}	2.49×10^{-1}	6.67×10^{-1}	2.49×10^{-1}	3.06×10^{-1}	6.67×10^{-1}	6.70×10^{-1}	7.30×10^5	2.48×10^{-1}	6.67×10^{-1}	2.50×10^{-1}
	Worst	1.55×10^0	7.67×10^3	3.09×10^0	2.59×10^{-1}	1.48×10^6	1.07×10^0	2.92×10^6	2.56×10^6	2.60×10^{-1}	3.97×10^0	7.18×10^0
	Mean	1.00×10^{-1}	1.56×10^1	6.94×10^{-1}	2.49×10^{-1}	3.98×10^3	6.73×10^{-1}	7.32×10^4	3.80×10^6	2.50×10^{-1}	1.16×10^0	2.71×10^{-1}
	SD	1.93×10^{-1}	3.43×10^2	2.17×10^{-1}	4.92×10^{-4}	6.90×10^4	2.34×10^{-2}	3.35×10^5	7.48×10^5	2.87×10^{-3}	1.02×10^0	3.12×10^{-1}
	Time (ms)	494	94.8	574	30	315	48.3	280	249,000	52.4	32.6	405
F12	Best	4.73×10^{-33}	0×10^0	1.09×10^{-88}	2.97×10^{-16}	0×10^0	5.10×10^{-9}	9.79×10^{-1}	5.92×10^3	2.29×10^{-9}	1.43×10^{-6}	0×10^0
	Worst	2.24×10^{-1}	1.18×10^1	2.99×10^{-1}	1.45×10^1	1.59×10^4	5.11×10^2	1.12×10^4	1.79×10^4	2.64×10^0	8.04×10^{-3}	1.38×10^1
	Mean	2.75×10^{-3}	3.46×10^{-2}	1.85×10^{-3}	6.11×10^{-2}	3.46×10^1	4.89×10^0	2.78×10^2	5.82×10^4	5.04×10^{-2}	9.77×10^{-4}	2.76×10^{-2}
	SD	2.26×10^{-2}	5.56×10^{-1}	1.88×10^{-2}	7.78×10^{-1}	7.13×10^2	2.47×10^1	1.17×10^3	8.14×10^3	3.33×10^{-1}	2.46×10^{-3}	6.17×10^{-1}
	Time (ms)	562	99.6	725	59.2	340	76.1	338	4,310,000	71	62.8	411
F13	Best	-1.00×10^0	-1.00×10^0	-1.00×10^0	-1.00×10^0	-1.00×10^0	-1.00×10^0	7.87×10^{-118}	1.28×10^{-17}	-1.00×10^0	6.22×10^{-194}	-1.00×10^0
	Worst	8.23×10^{-230}	3.88×10^{-149}	4.34×10^{-232}	4.93×10^{-207}	1.51×10^{-39}	4.30×10^{-46}	7.11×10^{-47}	2.26×10^{-3}	-2.96×10^{-1}	6.22×10^{-194}	-4.16×10^{-1}
	Mean	-9.49×10^{-1}	-9.93×10^{-1}	-9.98×10^{-1}	-7.57×10^{-1}	-7.50×10^{-1}	-1.46×10^{-1}	1.43×10^{-49}	1.36×10^{-8}	-9.85×10^{-1}	6.22×10^{-194}	-9.99×10^{-1}
	SD	2.00×10^{-1}	7.66×10^{-2}	4.47×10^{-2}	4.28×10^{-1}	4.33×10^{-1}	3.53×10^{-1}	3.18×10^{-48}	1.50×10^{-20}	3.16×10^{-2}	0×10^0	2.61×10^{-2}
	Time (ms)	793	333	789	130	334	135	647	7,320,000	151	153	766
F14	Best	9.79×10^{-1}	4.30×10^1	5.73×10^{-3}	5.03×10^1	1.31×10^{-1}	2.97×10^2	2.63×10^0	1.82×10^6	3.99×10^4	2.16×10^{-1}	4.00×10^4
	Worst	2.65×10^3	2.95×10^5	2.02×10^4	4.02×10^4	8.28×10^4	3.05×10^2	3.81×10^5	2.61×10^6	2.80×10^5	4.33×10^4	7.16×10^4
	Mean	1.52×10^2	3.04×10^3	1.12×10^2	5.03×10^2	2.50×10^3	3.02×10^2	1.05×10^3	3.14×10^7	4.38×10^4	2.12×10^2	4.02×10^4
	SD	3.88×10^2	1.72×10^4	1.31×10^3	4.02×10^3	1.32×10^4	2.72×10^0	1.77×10^4	8.01×10^5	2.43×10^4	2.00×10^3	1.94×10^3
	Time (ms)	583	231	738	83.2	220	80.3	536	1,780,000	89.9	75.8	559

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F15	Best	4.44×10^{-84}	0×10^0	3.62×10^{-141}	1.18×10^{-86}	0×10^0	0×10^0	2.29×10^{-3}	6.55×10^3	1.13×10^{-10}	1.17×10^{-9}	0×10^0
	Worst	3.45×10^2	1.61×10^1	2.78×10^0	2.33×10^2	1.04×10^4	1.45×10^{-1}	1.12×10^4	1.16×10^4	2.20×10^{-2}	4.51×10^0	1.21×10^0
	Mean	7.00×10^{-1}	3.25×10^{-2}	2.37×10^{-2}	4.75×10^{-1}	2.31×10^1	4.06×10^{-3}	3.93×10^2	1.56×10^4	7.23×10^{-3}	4.61×10^{-1}	2.41×10^{-3}
	SD	1.54×10^1	7.18×10^{-1}	2.32×10^{-1}	1.04×10^1	4.67×10^2	1.06×10^{-2}	1.44×10^3	7.09×10^3	1.02×10^{-2}	1.12×10^0	5.40×10^{-2}
	Time (ms)	468	68	585	28.9	259	51.4	271	4,550,000	48.9	31.1	424
F16	Best	0×10^0	5.04×10^{-15}	0×10^0	3.85×10^{-13}	7.79×10^{-12}	9.19×10^{-1}	4.58×10^{-9}	6.83×10^1	1.11×10^{-7}	0×10^0	6.56×10^{-1}
	Worst	4.06×10^{-3}	9.72×10^0	1.33×10^{-1}	9.75×10^0	8.57×10^{-1}	1.41×10^0	3.69×10^{-1}	2.59×10^1	9.82×10^0	3.90×10^0	7.37×10^{-1}
	Mean	7.05×10^{-5}	1.64×10^{-1}	4.36×10^{-4}	7.18×10^{-2}	1.57×10^{-2}	1.07×10^0	1.08×10^{-3}	5.87×10^3	5.17×10^{-2}	1.46×10^{-2}	6.56×10^{-1}
	SD	4.33×10^{-4}	1.19×10^0	6.53×10^{-3}	7.98×10^{-1}	8.84×10^{-2}	8.23×10^{-2}	1.66×10^{-2}	4.94×10^1	6.35×10^{-1}	1.90×10^{-1}	3.62×10^{-3}
	Time (ms)	397	64.1	488	23.9	133	23.8	351	143,000	26.7	13.2	303
F17	Best	0×10^0	5.50×10^{-6}	0×10^0	2.19×10^{-4}	1.82×10^{-9}	1.63×10^{-7}	9.50×10^{-9}	3.57×10^1	2.40×10^{-2}	0×10^0	1.63×10^{-1}
	Worst	4.32×10^{-1}	2.35×10^0	1.63×10^0	1.87×10^1	8.50×10^{-1}	3.46×10^0	2.19×10^0	3.83×10^1	1.23×10^1	2.50×10^0	2.18×10^0
	Mean	1.06×10^{-3}	5.67×10^{-2}	3.29×10^{-3}	1.02×10^{-1}	1.86×10^{-2}	5.99×10^{-2}	5.49×10^{-3}	3.66×10^2	9.22×10^{-1}	1.54×10^{-2}	1.95×10^{-1}
	SD	1.94×10^{-2}	3.22×10^{-1}	7.31×10^{-2}	8.77×10^{-1}	1.05×10^{-1}	3.41×10^{-1}	9.84×10^{-2}	3.24×10^1	9.78×10^{-1}	1.40×10^{-1}	2.52×10^{-1}
	Time (ms)	382	56.9	468	20	133	21.8	358	34,400	26.3	11.2	316
F18	Best	1.38×10^{-87}	1.38×10^{-87}	1.38×10^{-87}	1.38×10^{-87}	1.38×10^{-87}	1.38×10^{-87}	1.38×10^{-87}	2.57×10^0	1.38×10^{-87}	2.55×10^{-4}	1.12×10^{-5}
	Worst	1.38×10^{-87}	1.86×10^{-1}	1.38×10^{-87}	1.34×10^{-3}	8.48×10^0	4.78×10^0	5.25×10^1	8.37×10^0	6.64×10^{-3}	1.28×10^{-1}	1.13×10^{-2}
	Mean	1.38×10^{-87}	3.72×10^{-4}	1.38×10^{-87}	2.67×10^{-6}	1.70×10^{-2}	7.08×10^{-2}	4.91×10^{-1}	2.56×10^2	1.33×10^{-5}	1.63×10^{-2}	5.80×10^{-4}
	SD	1.12×10^{-102}	8.32×10^{-3}	1.12×10^{-102}	5.97×10^{-5}	3.79×10^{-1}	4.44×10^{-1}	4.17×10^0	2.82×10^0	2.97×10^{-4}	2.38×10^{-2}	2.13×10^{-3}
	Time (ms)	384	62	481	20.1	134	22.7	226	2320	23.4	10.8	294
F19	Best	0×10^0	0×10^0	0×10^0	2.01×10^{-193}	0×10^0	0×10^0	2.26×10^{-10}	2.00×10^0	8.64×10^{-46}	2.16×10^{-66}	0×10^0
	Worst	8.58×10^{-3}	1.20×10^{-2}	1.62×10^{-5}	4.91×10^{-4}	1.12×10^0	2.80×10^{-9}	5.94×10^{-1}	2.19×10^0	5.64×10^{-4}	6.31×10^{-4}	3.88×10^{-10}
	Mean	1.83×10^{-5}	2.40×10^{-5}	1.09×10^{-7}	1.00×10^{-6}	5.57×10^{-3}	5.60×10^{-12}	1.71×10^{-3}	1.74×10^1	5.50×10^{-6}	1.38×10^{-5}	7.76×10^{-13}
	SD	3.85×10^{-4}	5.36×10^{-4}	1.27×10^{-6}	2.20×10^{-5}	7.25×10^{-2}	1.25×10^{-10}	2.77×10^{-2}	1.83×10^0	3.62×10^{-5}	8.55×10^{-5}	1.74×10^{-11}
	Time (ms)	382	44.6	480	20.3	126	21	374	1850	21.2	11.2	282

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F20	Best	2.93×10^{-1}	2.93×10^{-1}	2.93×10^{-1}	2.93×10^{-1}	2.93×10^{-1}	2.93×10^{-1}	4.23×10^{-1}	5.25×10^{-1}	2.93×10^{-1}	2.93×10^{-1}	2.93×10^{-1}
	Worst	3.01×10^{-1}	5.01×10^{-1}	3.18×10^{-1}	5.01×10^{-1}	4.70×10^{-1}	3.84×10^{-1}	4.84×10^{-1}	5.75×10^{-1}	5.01×10^{-1}	4.47×10^{-1}	2.94×10^{-1}
	Mean	2.93×10^{-1}	2.94×10^{-1}	2.93×10^{-1}	2.96×10^{-1}	2.93×10^{-1}	2.94×10^{-1}	4.42×10^{-1}	4.99×10^{-1}	2.98×10^{-1}	2.94×10^{-1}	2.93×10^{-1}
	SD	6.05×10^{-4}	1.50×10^{-2}	2.31×10^{-3}	1.84×10^{-2}	8.40×10^{-3}	6.26×10^{-3}	1.96×10^{-2}	5.30×10^{-1}	1.84×10^{-2}	1.26×10^{-2}	3.84×10^{-4}
	Time (ms)	388	72.2	477	21	135	21.2	272	569	22.4	11.1	288
F21	Best	1.91×10^1	1.91×10^1	1.91×10^1	1.91×10^1	1.91×10^1	1.91×10^1	3.89×10^7	4.60×10^8	1.91×10^1	1.91×10^1	2.35×10^1
	Worst	1.12×10^5	3.31×10^1	9.09×10^2	2.86×10^5	6.02×10^7	3.23×10^6	1.66×10^9	1.70×10^9	2.08×10^4	7.47×10^4	3.75×10^2
	Mean	2.46×10^2	2.02×10^1	2.10×10^1	5.98×10^2	1.37×10^5	7.03×10^3	2.67×10^8	3.55×10^{10}	6.75×10^1	6.40×10^2	2.63×10^1
	SD	5.00×10^3	1.99×10^0	3.98×10^1	1.28×10^4	2.71×10^6	1.45×10^5	3.69×10^8	4.65×10^8	9.27×10^2	6.67×10^3	2.10×10^1
	Time (ms)	387	61.8	482	22.2	129	22.5	268	4.31E+08	24.4	12	307
F22	Best	0×10^0	2.95×10^{-7}	0×10^0	1.72×10^{-5}	1.35×10^{-7}	6.08×10^{-2}	1.70×10^{-8}	1.30×10^1	3.95×10^{-5}	3.29×10^{-14}	1.69×10^{-6}
	Worst	1.87×10^{-2}	1.65×10^{-1}	3.14×10^{-1}	1.37×10^{-1}	2.04×10^0	8.63×10^{-1}	5.81×10^{-1}	9.55×10^0	5.21×10^{-2}	1.90×10^{-1}	1.25×10^{-1}
	Mean	1.14×10^{-4}	5.65×10^{-4}	8.68×10^{-4}	8.61×10^{-4}	9.29×10^{-3}	1.31×10^{-1}	1.84×10^{-3}	7.18×10^1	7.21×10^{-4}	2.78×10^{-3}	9.39×10^{-4}
	SD	1.28×10^{-3}	8.27×10^{-3}	1.46×10^{-2}	8.71×10^{-3}	9.92×10^{-2}	1.37×10^{-1}	2.61×10^{-2}	1.85×10^1	4.69×10^{-3}	1.81×10^{-2}	9.35×10^{-3}
	Time (ms)	390	65.6	490	22.1	130	34.3	358	11,900	25.1	12.3	299
F23	Best	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0
	Worst	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0
	Mean	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0
	SD	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0
	Time (ms)	636	259	785	92	214	89.1	377	0	95.2	82.7	579
F24	Best	-3.79×10^{-3}	-3.79×10^{-3}	-3.79×10^{-3}	-3.79×10^{-3}	-3.79×10^{-3}	-3.71×10^{-3}	-3.79×10^{-3}	3.35×10^1	-3.79×10^{-3}	-3.79×10^{-3}	-3.79×10^{-3}
	Worst	2.49×10^{-1}	4.19×10^{-4}	2.56×10^{-3}	3.23×10^{-1}	3.72×10^{-1}	1.05×10^{-1}	2.62×10^{-1}	4.16×10^1	5.37×10^{-2}	2.74×10^{-4}	4.49×10^{-3}
	Mean	-3.25×10^{-3}	-3.77×10^{-3}	-3.78×10^{-3}	-3.10×10^{-3}	-2.18×10^{-3}	7.56×10^{-3}	-3.09×10^{-3}	4.20×10^3	-3.52×10^{-3}	-3.71×10^{-3}	-3.77×10^{-3}
	SD	1.13×10^{-2}	1.91×10^{-4}	2.87×10^{-4}	1.46×10^{-2}	2.23×10^{-2}	2.17×10^{-2}	1.20×10^{-2}	2.96×10^1	2.64×10^{-3}	5.62×10^{-4}	3.73×10^{-4}
	Time (ms)	386	58.2	465	27.7	131	24.6	385	31,600	22.8	10.1	301

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F25	Best	1.27×10^{-5}	1.27×10^{-5}	1.27×10^{-5}	1.18×10^2	1.19×10^{-3}	2.54×10^2	2.52×10^2	4.35×10^2	1.23×10^0	1.40×10^{-5}	2.84×10^{-5}
	Worst	1.11×10^0	3.44×10^1	5.70×10^0	1.19×10^2	3.58×10^2	3.29×10^2	3.67×10^2	4.40×10^2	1.55×10^1	3.90×10^0	4.80×10^0
	Mean	9.54×10^{-3}	7.09×10^{-2}	1.28×10^{-1}	1.18×10^2	1.78×10^1	2.55×10^2	2.79×10^2	4.06×10^2	1.70×10^0	8.58×10^{-1}	4.56×10^{-2}
	SD	9.91×10^{-2}	1.54×10^0	7.81×10^{-1}	2.27×10^{-2}	5.11×10^1	6.90×10^0	3.10×10^1	4.27×10^2	1.73×10^0	1.39×10^0	3.46×10^{-1}
	Time (ms)	534	130	645	44.5	279	64.5	315	426,000	65.5	50.5	504
F26	Best	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	8.76×10^1	4.33×10^2	4.88×10^{-6}	1.07×10^{-6}	0×10^0
	Worst	4.40×10^1	8.27×10^1	4.50×10^1	4.07×10^1	4.41×10^2	5.15×10^{-1}	4.76×10^2	5.04×10^2	7.18×10^0	2.80×10^1	3.14×10^1
	Mean	3.11×10^{-1}	1.66×10^{-1}	1.31×10^0	1.71×10^{-1}	9.65×10^0	1.42×10^{-2}	1.79×10^2	5.67×10^2	4.87×10^{-1}	3.06×10^0	6.33×10^{-2}
	SD	3.43×10^0	3.70×10^0	6.91×10^0	2.57×10^0	2.96×10^1	3.68×10^{-2}	7.89×10^1	4.22×10^2	1.21×10^0	7.29×10^0	1.41×10^0
	Time (ms)	503	106	590	36.5	298	51.1	258	412,000	58.7	37.5	409
F27	Best	9.00×10^{-1}	9.00×10^{-1}	9.00×10^{-1}	9.00×10^{-1}	9.00×10^{-1}	9.00×10^{-1}	1.00×10^0	1.49×10^1	1.00×10^0	1.00×10^0	9.00×10^{-1}
	Worst	1.28×10^0	1.13×10^0	9.04×10^{-1}	1.00×10^0	1.11×10^1	9.16×10^{-1}	9.60×10^0	1.47×10^1	1.07×10^0	1.02×10^0	9.54×10^{-1}
	Mean	9.02×10^{-1}	9.01×10^{-1}	9.00×10^{-1}	9.13×10^{-1}	1.05×10^0	9.00×10^{-1}	2.43×10^0	1.54×10^1	1.00×10^0	1.00×10^0	9.00×10^{-1}
	SD	1.83×10^{-2}	1.49×10^{-2}	6.13×10^{-4}	3.30×10^{-2}	7.47×10^{-1}	9.46×10^{-4}	1.89×10^0	1.50×10^1	6.79×10^{-3}	5.32×10^{-3}	2.41×10^{-3}
	Time (ms)	509	104	610	47.9	291	53.9	261	14,700	62.1	50.8	417
F28	Best	7.00×10^{-5}	4.51×10^2	7.62×10^{-19}	2.03×10^2	2.62×10^0	3.16×10^3	2.07×10^{11}	8.65×10^{10}	8.36×10^2	9.17×10^{-2}	1.16×10^3
	Worst	9.40×10^3	1.43×10^4	2.26×10^3	7.92×10^3	2.30×10^{11}	10.00×10^{10}	3.03×10^{11}	3.05×10^{11}	4.24×10^6	6.97×10^6	7.31×10^3
	Mean	4.74×10^2	1.69×10^3	7.68×10^1	5.74×10^2	5.10×10^8	3.87×10^8	2.30×10^{11}	3.73×10^{11}	1.98×10^4	8.44×10^5	1.45×10^3
	SD	9.12×10^2	9.41×10^2	3.42×10^2	7.03×10^2	1.03×10^{10}	5.23×10^9	2.65×10^{10}	9.65×10^{10}	2.67×10^5	1.96×10^6	6.51×10^2
	Time (ms)	481	90.6	563	29.8	279	57.3	271	3,110,000	49.2	37.2	425
F29	Best	3.82×10^{-38}	1.99×10^{-4}	1.65×10^{-71}	8.82×10^{-47}	3.52×10^{-178}	0×10^0	2.97×10^0	6.50×10^1	2.60×10^{-6}	8.92×10^{-6}	0×10^0
	Worst	7.94×10^{-2}	3.90×10^{-1}	4.30×10^{-1}	2.74×10^0	5.66×10^1	5.49×10^{-2}	6.88×10^1	7.52×10^1	1.10×10^0	6.64×10^{-2}	4.47×10^{-1}
	Mean	2.08×10^{-3}	1.43×10^{-3}	4.33×10^{-3}	4.91×10^{-2}	2.40×10^{-1}	3.13×10^{-3}	1.78×10^1	9.43×10^1	1.33×10^{-1}	1.18×10^{-2}	9.07×10^{-4}
	SD	1.11×10^{-2}	1.77×10^{-2}	3.45×10^{-2}	2.44×10^{-1}	3.10×10^0	5.01×10^{-3}	1.33×10^1	6.04×10^1	2.45×10^{-1}	1.96×10^{-2}	2.00×10^{-2}
	Time (ms)	498	98.8	587	37.3	261	49.1	259	57,600	63.6	40.7	405

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F30	Best	4.56×10^{-67}	5.91×10^{-47}	9.70×10^{-118}	1.19×10^{-10}	5.94×10^{-273}	0×10^0	1.57×10^{-5}	4.79×10^{11}	5.70×10^{-13}	3.07×10^{-9}	0×10^0
	Worst	1.51×10^{-2}	7.04×10^{-2}	4.63×10^{-2}	8.87×10^{-2}	1.34×10^{13}	8.83×10^{-5}	3.86×10^{12}	9.17×10^{14}	3.03×10^{-2}	2.45×10^{-2}	2.32×10^{-2}
	Mean	7.01×10^{-5}	1.47×10^{-4}	2.18×10^{-4}	1.97×10^{-4}	2.68×10^{10}	2.04×10^{-7}	1.12×10^{10}	1.74×10^{19}	3.88×10^{-4}	1.17×10^{-3}	4.68×10^{-5}
	SD	9.58×10^{-4}	3.15×10^{-3}	2.67×10^{-3}	3.98×10^{-3}	5.98×10^{11}	4.00×10^{-6}	1.89×10^{11}	3.59×10^{12}	2.33×10^{-3}	4.55×10^{-3}	1.04×10^{-3}
	Time (ms)	730	205	845	117	323	99.7	455	1,860,000	116	97.2	480
F31	Best	-4.44×10^{-16}	-4.44×10^{-16}	3.11×10^{-15}	3.11×10^{-15}	-4.44×10^{-16}	-4.44×10^{-16}	1.90×10^1	1.99×10^1	5.50×10^{-3}	7.71×10^{-4}	-4.44×10^{-16}
	Worst	1.73×10^0	3.67×10^0	6.24×10^0	4.59×10^0	2.06×10^1	4.63×10^{-1}	2.06×10^1	2.11×10^1	4.58×10^0	2.84×10^0	1.31×10^0
	Mean	2.42×10^{-2}	1.61×10^{-2}	8.46×10^{-2}	1.35×10^{-2}	8.40×10^{-2}	2.84×10^{-2}	1.96×10^1	2.11×10^1	1.87×10^0	4.86×10^{-1}	3.22×10^{-3}
	SD	1.48×10^{-1}	2.33×10^{-1}	6.00×10^{-1}	2.11×10^{-1}	1.02×10^0	4.30×10^{-2}	4.42×10^{-1}	2.01×10^1	1.22×10^0	9.02×10^{-1}	6.00×10^{-2}
	Time (ms)	526	108	597	40.2	256	50.9	259	18,700	60	40.6	415
F32	Best	1.00×10^0	1.00×10^0	1.67×10^1	5.07×10^0	3.85×10^1	9.25×10^1	1.69×10^6	9.29×10^5	1.05×10^2	2.53×10^1	1.72×10^0
	Worst	7.87×10^3	2.15×10^3	9.02×10^2	2.19×10^4	1.65×10^6	8.38×10^5	2.02×10^6	1.95×10^6	1.19×10^4	7.15×10^4	4.69×10^1
	mean	1.94×10^1	6.01×10^0	3.54×10^1	8.95×10^1	6.33×10^3	4.21×10^3	1.77×10^6	2.46×10^6	9.90×10^2	5.55×10^3	1.87×10^1
	SD	3.53×10^2	9.75×10^1	7.41×10^1	1.09×10^3	8.72×10^4	4.46×10^4	9.44×10^4	1.06×10^6	7.32×10^2	1.48×10^4	1.05×10^1
	Time (ms)	545	121	625	45.4	286	61.3	316	674,000	94.2	44.9	449
F33	Best	9.99×10^{-2}	3.46×10^{-52}	0×10^0	1.13×10^{-38}	0×10^0	9.99×10^{-2}	2.38×10^1	2.08×10^1	2.00×10^{-1}	7.00×10^{-1}	0×10^0
	Worst	1.49×10^0	2.12×10^0	1.12×10^0	2.14×10^0	2.84×10^1	1.80×10^0	2.56×10^1	2.75×10^1	2.61×10^0	1.44×10^0	2.20×10^{-1}
	Mean	1.23×10^{-1}	8.15×10^{-3}	1.62×10^{-2}	3.89×10^{-2}	3.70×10^{-1}	1.05×10^{-1}	2.39×10^1	3.28×10^1	3.41×10^{-1}	8.54×10^{-1}	1.66×10^{-2}
	SD	1.35×10^{-1}	1.05×10^{-1}	1.06×10^{-1}	1.37×10^{-1}	1.82×10^0	7.79×10^{-2}	3.44×10^{-1}	2.18×10^1	1.56×10^{-1}	1.99×10^{-1}	3.79×10^{-2}
	Time (ms)	494	67.3	566	29	320	44.8	485	17,300	50.4	33	385
F34	Best	-1.17×10^3	-1.17×10^3	-1.17×10^3	-1.17×10^3	-1.17×10^3	-4.38×10^2	-9.21×10^2	-3.98×10^2	-1.17×10^3	-1.17×10^3	-1.17×10^3
	Worst	-1.15×10^3	-1.17×10^3	-1.17×10^3	-1.11×10^3	-4.12×10^2	-4.31×10^2	-3.83×10^2	-2.14×10^2	-1.14×10^3	-1.17×10^3	-1.17×10^3
	Mean	-1.17×10^3	-1.17×10^3	-1.17×10^3	-1.17×10^3	-1.15×10^3	-4.38×10^2	-8.85×10^2	-1.45×10^2	-1.17×10^3	-1.17×10^3	-1.17×10^3
	SD	1.73×10^0	8.55×10^{-4}	3.58×10^{-1}	2.84×10^0	5.71×10^1	8.17×10^{-1}	5.90×10^1	-4.23×10^2	2.78×10^0	1.88×10^0	3.02×10^{-1}
	Time (ms)	689	255	828	95	341	94.5	357	387,000	115	104	658

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F35	Best	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	6.51×10^0	9.79×10^0	9.13×10^{-7}	3.88×10^{-8}	0×10^0
	Worst	1.54×10^0	6.54×10^{-1}	5.93×10^{-1}	1.24×10^0	2.03×10^1	9.08×10^{-1}	1.92×10^1	2.05×10^1	6.12×10^{-1}	1.17×10^0	7.30×10^{-2}
	Mean	9.49×10^{-3}	2.85×10^{-3}	6.35×10^{-3}	2.51×10^{-3}	7.25×10^{-2}	3.48×10^{-3}	9.32×10^0	2.58×10^1	6.50×10^{-2}	2.64×10^{-1}	1.47×10^{-4}
	SD	9.35×10^{-2}	4.16×10^{-2}	5.92×10^{-2}	5.53×10^{-2}	9.51×10^{-1}	4.09×10^{-2}	3.41×10^0	1.23×10^1	6.17×10^{-2}	4.26×10^{-1}	3.26×10^{-3}
	Time (ms)	585	135	656	44.8	294	59.3	315	6730	72	51	489
F36	Best	-1.00×10^0	-1.00×10^0	-1.00×10^0	-1.00×10^0	-1.00×10^0	1.54×10^{-7}	4.51×10^{-16}	1.02×10^{-7}	-1.00×10^0	1.15×10^{-21}	-1.00×10^0
	Worst	2.29×10^{-13}	5.93×10^{-13}	2.65×10^{-13}	5.09×10^{-14}	1.18×10^{-7}	3.28×10^{-7}	4.58×10^{-7}	3.21×10^{-6}	8.11×10^{-13}	1.05×10^{-13}	-1.57×10^{-2}
	Mean	-9.65×10^{-1}	-9.94×10^{-1}	-9.53×10^{-1}	-9.92×10^{-1}	-9.25×10^{-1}	1.56×10^{-7}	2.24×10^{-9}	2.80×10^{-5}	-8.25×10^{-1}	1.30×10^{-14}	-9.98×10^{-1}
	SD	1.60×10^{-1}	7.76×10^{-2}	1.92×10^{-1}	7.56×10^{-2}	1.43×10^{-1}	1.70×10^{-8}	2.52×10^{-8}	7.82×10^{-8}	2.42×10^{-1}	2.87×10^{-14}	4.44×10^{-2}
	Time (ms)	570	134	700	42.8	280	63	305	0.05	105	57.4	475
F37	Best	3.51×10^{-12}	3.51×10^{-12}	0×10^0	3.51×10^{-12}	1.78×10^{-34}	2.45×10^{-4}	9.09×10^{-12}	8.68×10^0	3.58×10^{-12}	3.51×10^{-12}	3.51×10^{-12}
	Worst	3.51×10^{-12}	6.00×10^{-12}	5.94×10^{-12}	3.82×10^{-12}	1.43×10^{-4}	2.53×10^{-3}	3.40×10^{-3}	7.76×10^0	8.11×10^{-12}	7.69×10^{-12}	3.74×10^{-12}
	Mean	3.51×10^{-12}	3.52×10^{-12}	3.46×10^{-12}	3.51×10^{-12}	2.86×10^{-7}	6.08×10^{-4}	7.48×10^{-6}	1.33×10^2	3.69×10^{-12}	4.12×10^{-12}	3.52×10^{-12}
	SD	1.64×10^{-17}	1.59×10^{-13}	7.48×10^{-13}	1.97×10^{-14}	6.40×10^{-6}	1.50×10^{-4}	1.52×10^{-4}	2.37×10^0	4.60×10^{-13}	1.19×10^{-12}	3.33×10^{-14}
	Time (ms)	532	111	603	37	267	50.6	381	2460	59.6	45.7	424
F38	Best	1.35×10^{-32}	9.60×10^{-13}	2.92×10^{-22}	6.03×10^{-5}	6.70×10^{-3}	2.94×10^0	4.68×10^4	5.06×10^8	2.12×10^{-2}	3.48×10^{-11}	7.43×10^{-5}
	Worst	1.08×10^1	5.44×10^0	1.60×10^{-1}	5.65×10^{-1}	1.50×10^9	4.44×10^0	1.58×10^9	1.67×10^9	7.12×10^1	1.10×10^{-1}	2.19×10^{-1}
	Mean	5.76×10^{-2}	1.42×10^{-2}	2.28×10^{-3}	3.19×10^{-3}	3.44×10^6	2.98×10^0	1.27×10^8	2.23×10^9	2.32×10^{-1}	1.35×10^{-2}	1.02×10^{-2}
	SD	7.00×10^{-1}	2.51×10^{-1}	1.57×10^{-2}	3.79×10^{-2}	6.77×10^7	9.93×10^{-2}	2.98×10^8	4.81×10^8	3.20×10^0	3.27×10^{-2}	3.91×10^{-2}
	Time (ms)	1140	564	1380	239	468	195	651	185,000,000	245	226	1160
F39	Best	1.85×10^{-28}	7.60×10^{-14}	6.09×10^{-24}	3.04×10^{-5}	4.28×10^{-6}	3.86×10^{-1}	3.27×10^1	2.10×10^8	6.39×10^{-6}	1.13×10^{-9}	6.27×10^{-5}
	Worst	2.38×10^0	3.67×10^0	5.12×10^{-2}	5.14×10^0	7.74×10^8	1.57×10^0	7.67×10^8	9.53×10^8	3.20×10^{-3}	7.90×10^{-1}	1.38×10^{-1}
	Mean	7.64×10^{-3}	1.41×10^{-2}	3.67×10^{-4}	1.66×10^{-2}	1.91×10^6	4.02×10^{-1}	4.17×10^7	1.11×10^9	2.66×10^{-4}	8.48×10^{-2}	1.70×10^{-3}
	SD	1.14×10^{-1}	2.23×10^{-1}	2.69×10^{-3}	2.44×10^{-1}	3.53×10^7	8.57×10^{-2}	1.20×10^8	2.13×10^8	7.93×10^{-4}	2.11×10^{-1}	9.57×10^{-3}
	Time (ms)	1140	556	1360	231	462	192	595	40,800,000	245	221	1170

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F40	Best	-2.39×10^1	-1.24×10^1	-1.80×10^1	-1.29×10^1	-1.70×10^1	-7.20×10^0	-1.96×10^1	-3.01×10^0	-1.25×10^1	-2.08×10^1	-1.19×10^1
	Worst	-7.27×10^0	-5.95×10^0	-1.19×10^1	-1.14×10^1	-6.15×10^0	-6.05×10^0	-6.33×10^0	-3.04×10^0	-6.72×10^0	-6.96×10^0	-7.17×10^0
	Mean	-1.83×10^1	-1.16×10^1	-1.45×10^1	-1.25×10^1	-1.11×10^1	-7.16×10^0	-1.08×10^1	-3.16×10^0	-1.12×10^1	-1.74×10^1	-1.19×10^1
	SD	4.62×10^0	9.42×10^{-1}	2.64×10^0	4.08×10^{-1}	1.65×10^0	2.02×10^{-1}	3.41×10^0	-2.92×10^0	1.09×10^0	3.77×10^0	4.90×10^{-1}
	Time (ms)	735	270	927	116	352	122	369	2880	122	103	633
F41	Best	4.73×10^{-4}	3.26×10^{-4}	5.89×10^{-4}	1.83×10^{-4}	1.23×10^{-4}	2.07×10^{-4}	3.86×10^{-2}	2.90×10^1	9.30×10^{-4}	7.30×10^{-2}	9.46×10^{-5}
	Worst	1.39×10^{-1}	6.92×10^{-1}	1.32×10^{-1}	7.93×10^{-2}	1.50×10^2	3.12×10^{-2}	1.14×10^2	1.83×10^2	1.29×10^{-1}	4.37×10^{-1}	4.16×10^{-1}
	Mean	2.95×10^{-3}	2.28×10^{-3}	2.42×10^{-3}	7.36×10^{-4}	3.37×10^{-1}	8.23×10^{-4}	1.01×10^1	2.43×10^2	2.06×10^{-2}	1.81×10^{-1}	1.06×10^{-3}
	SD	9.32×10^{-3}	3.11×10^{-2}	9.90×10^{-3}	4.12×10^{-3}	6.73×10^0	2.75×10^{-3}	1.96×10^1	5.15×10^1	2.12×10^{-2}	9.94×10^{-2}	1.86×10^{-2}
	Time (ms)	704	241	863	127	358	105	382	10,100	121	101	603
F42	Best	0×10^0	1.10×10^{-125}	0×10^0	1.19×10^{-128}	0×10^0	0×10^0	9.49×10^0	3.82×10^1	6.77×10^{-30}	2.50×10^{-96}	0×10^0
	Worst	3.24×10^{-1}	4.21×10^{-1}	6.25×10^{-1}	1.44×10^{-1}	9.62×10^0	1.51×10^{-4}	9.99×10^0	3.92×10^1	3.20×10^{-4}	8.39×10^{-2}	2.37×10^{-2}
	Mean	7.01×10^{-4}	9.17×10^{-4}	1.64×10^{-3}	4.65×10^{-4}	3.05×10^{-2}	3.01×10^{-7}	9.49×10^0	4.68×10^1	4.01×10^{-5}	2.20×10^{-3}	4.75×10^{-5}
	SD	1.45×10^{-2}	1.89×10^{-2}	2.92×10^{-2}	7.29×10^{-3}	4.76×10^{-1}	6.74×10^{-6}	3.99×10^{-2}	3.65×10^1	9.55×10^{-5}	1.33×10^{-2}	1.06×10^{-3}
	Time (ms)	392	54.9	473	22.8	134	23.1	371	35,700	24.4	11.9	293
F43	Best	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.62×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2
	Worst	-1.89×10^2	-1.91×10^2	-1.93×10^2	-1.95×10^2	-1.65×10^2	-1.95×10^2	-1.52×10^2	-1.59×10^2	-1.81×10^2	-1.78×10^2	-1.95×10^2
	Mean	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.96×10^2	-1.95×10^2	-1.96×10^2	-1.94×10^2	-1.18×10^2	-1.96×10^2	-1.95×10^2	-1.96×10^2
	SD	3.03×10^{-1}	3.23×10^{-1}	1.22×10^{-1}	4.36×10^{-2}	1.69×10^0	4.79×10^{-2}	5.37×10^0	-1.61×10^2	6.62×10^{-1}	1.36×10^0	2.73×10^{-2}
	Time (ms)	397	62.4	507	21.3	153	22.7	325	163,000	23.5	11.8	314
F44	Best	-2.02×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0	-1.82×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0
	Worst	-2.02×10^0	-7.58×10^{-1}	-2.01×10^0	-6.31×10^{-1}	-1.55×10^0	-2.02×10^0	-1.81×10^0	-1.50×10^0	-7.37×10^{-1}	-1.67×10^0	-1.29×10^0
	Mean	-2.02×10^0	-2.00×10^0	-2.02×10^0	-2.01×10^0	-2.02×10^0	-2.02×10^0	-2.02×10^0	3.68×10^{-2}	-2.01×10^0	-2.02×10^0	-2.02×10^0
	SD	2.25×10^{-4}	1.36×10^{-1}	4.88×10^{-4}	1.06×10^{-1}	2.10×10^{-2}	2.36×10^{-5}	9.62×10^{-3}	-1.78×10^0	9.32×10^{-2}	2.03×10^{-2}	3.03×10^{-2}
	Time (ms)	402	49.7	500	23.7	136	19.7	456	1860	22.1	11	291

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F45	Best	-1.07×10^2	-1.07×10^2	-1.07×10^2	-1.07×10^2	-1.07×10^2	-1.07×10^2	-1.07×10^2	-2.73×10^0	-1.07×10^2	-1.07×10^2	-3.82×10^1
	Worst	-9.96×10^1	-2.57×10^1	-5.65×10^1	-2.58×10^1	-6.69×10^1	-8.49×10^1	-7.60×10^1	-8.07×10^0	-2.54×10^1	-3.61×10^1	-2.52×10^1
	Mean	-1.07×10^2	-1.06×10^2	-1.07×10^2	-1.06×10^2	-1.06×10^2	-1.07×10^2	-1.07×10^2	2.33×10^1	-1.04×10^2	-1.06×10^2	-3.68×10^1
	SD	5.03×10^{-1}	7.11×10^0	2.25×10^0	5.40×10^0	4.45×10^0	1.60×10^0	1.70×10^0	-7.22×10^0	1.08×10^1	4.05×10^0	2.90×10^0
	Time (ms)	394	65.9	494	24	140	26	375	7600	24.3	13.1	296
F46	Best	-1.03×10^0	-1.03×10^0	-1.03×10^0	-1.03×10^0	-1.03×10^0	-1.03×10^0	-1.03×10^0	1.31×10^1	-1.03×10^0	-1.03×10^0	-1.00×10^0
	Worst	-9.86×10^{-1}	3.66×10^{-2}	-9.86×10^{-1}	1.68×10^{-4}	1.13×10^{-1}	-1.01×10^0	-9.16×10^{-2}	3.04×10^1	1.61×10^0	1.63×10^{-3}	-9.02×10^{-1}
	Mean	-1.03×10^0	-1.02×10^0	-1.03×10^0	-1.02×10^0	-1.02×10^0	-1.03×10^0	-1.03×10^0	8.36×10^2	-9.81×10^{-1}	-1.02×10^0	-1.00×10^0
	SD	2.13×10^{-3}	8.77×10^{-2}	2.04×10^{-3}	1.14×10^{-1}	8.75×10^{-2}	8.26×10^{-4}	4.73×10^{-2}	3.50×10^0	1.77×10^{-1}	8.24×10^{-2}	5.55×10^{-3}
	Time (ms)	414	75.2	494	28.7	137	25.3	376	13,700	31.4	18.1	317
F47	Best	3.98×10^{-1}	3.98×10^{-1}	3.98×10^{-1}	3.98×10^{-1}	3.98×10^{-1}	3.98×10^{-1}	3.98×10^{-1}	5.81×10^0	3.98×10^{-1}	3.98×10^{-1}	8.45×10^{-1}
	Worst	7.54×10^{-1}	1.37×10^0	5.67×10^{-1}	1.09×10^0	8.08×10^{-1}	8.94×10^{-1}	8.11×10^0	6.33×10^0	8.73×10^{-1}	4.53×10^{-1}	9.27×10^{-1}
	Mean	3.99×10^{-1}	4.05×10^{-1}	3.98×10^{-1}	4.09×10^{-1}	4.04×10^{-1}	4.04×10^{-1}	4.19×10^{-1}	6.54×10^1	4.29×10^{-1}	3.99×10^{-1}	8.46×10^{-1}
	SD	1.59×10^{-2}	5.58×10^{-2}	7.56×10^{-3}	6.52×10^{-2}	4.48×10^{-2}	4.21×10^{-2}	3.63×10^{-1}	5.83×10^0	9.83×10^{-2}	6.14×10^{-3}	8.21×10^{-3}
	Time (ms)	395	57.9	488	20.8	128	19.1	369	5390	23.7	10.4	298
F48	Best	3.00×10^0	3.00×10^0	3.00×10^0	3.00×10^0	3.00×10^0	3.00×10^0	3.00×10^0	1.11×10^3	3.01×10^1	3.00×10^0	3.00×10^0
	Worst	1.31×10^1	3.56×10^1	3.54×10^0	8.02×10^1	6.99×10^1	6.87×10^0	4.50×10^1	8.67×10^2	3.27×10^1	3.38×10^1	1.46×10^1
	Mean	3.02×10^0	3.81×10^0	3.00×10^0	3.42×10^0	4.14×10^0	3.01×10^0	3.15×10^0	7.52×10^4	3.21×10^1	3.13×10^0	3.05×10^0
	SD	4.54×10^{-1}	4.81×10^0	3.46×10^{-2}	4.46×10^0	7.73×10^0	1.82×10^{-1}	2.11×10^0	1.36×10^3	8.58×10^{-1}	1.95×10^0	5.76×10^{-1}
	Time (ms)	392	58.7	470	19.4	136	31.7	378	1,270,000	23.4	9.42	296
F49	Best	-3.86×10^0	-3.86×10^0	-3.86×10^0	-3.86×10^0	-3.86×10^0	-3.86×10^0	-3.86×10^0	-3.50×10^0	-3.85×10^0	-3.86×10^0	-3.85×10^0
	Worst	-3.73×10^0	-1.83×10^0	-3.63×10^0	-1.89×10^0	-3.39×10^0	-3.16×10^0	-1.93×10^0	-3.11×10^0	-1.90×10^0	-3.14×10^0	-2.95×10^0
	Mean	-3.86×10^0	-3.84×10^0	-3.86×10^0	-3.85×10^0	-3.86×10^0	-3.63×10^0	-3.83×10^0	-7.58×10^{-1}	-3.81×10^0	-3.86×10^0	-3.78×10^0
	SD	9.88×10^{-3}	1.75×10^{-1}	1.04×10^{-2}	1.05×10^{-1}	3.52×10^{-2}	2.14×10^{-1}	1.47×10^{-1}	-3.43×10^0	2.23×10^{-1}	3.52×10^{-2}	1.63×10^{-1}
	Time (ms)	451	84.6	546	30	279	44.6	355	-3500	33.7	18.3	351

(Continued)

Table 2 (continued)

Function	Creative	MGO	HHO	CGO	WOA	SMA	SCA	IWO	GOA	TSO	PSO	IMGO
F50	Best	-3.20×10^0	-3.15×10^0	-3.20×10^0	-3.32×10^0	-3.20×10^0	-3.00×10^0	-3.20×10^0	-2.95×10^0	-3.32×10^0	-3.20×10^0	-2.46×10^0
	Worst	-2.53×10^0	-8.53×10^{-1}	-3.05×10^0	-1.17×10^0	-2.91×10^0	-1.22×10^0	-4.52×10^{-1}	-1.45×10^0	-1.17×10^0	-1.82×10^0	-1.88×10^0
	Mean	-3.19×10^0	-2.96×10^0	-3.20×10^0	-3.26×10^0	-3.17×10^0	-2.78×10^0	-3.00×10^0	-3.23×10^{-1}	-3.22×10^0	-3.19×10^0	-2.38×10^0
	SD	5.76×10^{-2}	3.92×10^{-1}	1.19×10^{-2}	2.56×10^{-1}	5.72×10^{-2}	3.23×10^{-1}	3.92×10^{-1}	-2.56×10^0	3.12×10^{-1}	9.53×10^{-2}	1.95×10^{-1}
	Time (ms)	445	86.6	549	32.5	592	42.6	331	-2960	34.6	21.8	353
F51	Best	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-1.80×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0
	Worst	-2.06×10^0	-2.02×10^0	-2.06×10^0	-2.06×10^0	-1.93×10^0	-2.06×10^0	-1.81×10^0	-1.64×10^0	-2.06×10^0	-2.04×10^0	-2.05×10^0
	Mean	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0	-1.51×10^0	-2.06×10^0	-2.06×10^0	-2.06×10^0
	SD	1.28×10^{-5}	1.72×10^{-3}	1.62×10^{-5}	6.22×10^{-7}	6.12×10^{-3}	2.31×10^{-4}	1.51×10^{-2}	-1.71×10^0	5.40×10^{-5}	1.49×10^{-3}	3.96×10^{-4}
	Time (ms)	412	67.7	500	25	485	31.7	382	-1750	25.5	13.4	326
F52	Best	1.00×10^0	1.00×10^0	1.00×10^0	1.00×10^0	1.00×10^0	1.00×10^0	1.03×10^4	2.00×10^4	1.00×10^0	1.00×10^0	1.00×10^0
	Worst	1.57×10^0	2.11×10^3	1.12×10^2	2.07×10^2	7.31×10^3	1.91×10^3	6.35×10^4	2.36×10^4	1.34×10^2	1.67×10^0	1.00×10^0
	Mean	1.00×10^0	5.22×10^0	1.25×10^0	1.49×10^0	1.57×10^1	1.52×10^1	2.13×10^4	1.97×10^5	1.33×10^0	1.02×10^0	1.00×10^0
	SD	4.42×10^{-2}	9.42×10^1	4.99×10^0	9.24×10^0	3.27×10^2	1.36×10^2	1.42×10^4	1.98×10^4	5.97×10^0	1.13×10^{-1}	0×10^0
	Time (ms)	397	60.2	473	20.8	426	33	256	18,500,000	23	10.4	291
Ranking	Best	21	14	23	9	25	13	1	1	3	9	31
	worst	11	3	8	4	1	11	1	1	4	2	14
	Ave	11	4	9	7	1	7	1	1	1	1	18
	SD	5	2	4	3	1	10	1	10	4	4	9
	time	0	0	0	3	0	2	0	13	0	15	0
SUM		48	23	44	26	28	43	4	26	12	31	72

In the realm of the F3 and F4 functions, which are characterized by multiple local optima, the IMGO algorithm adeptly navigated past local optima traps and successfully converged on the global optimum. In contrast, algorithms such as WOA, PSO, and GOA struggled to find optimal solutions, often becoming ensnared in local optima. Regarding the F5 function, known for its large scale, IMGO excelled in handling extensive problem spaces. The algorithm swiftly converged to the optimal values from initialization, whereas the other algorithms faced challenges in achieving convergence. Moreover, the IMGO demonstrated an exceptional balance between exploration and exploitation in functions such as F7 and F10, efficiently probing search spaces while converging on optimal points, thereby surpassing alternative methods. In the F11 and F25 functions, where stability and preservation of optimal solutions are vital, the IMGO significantly outperformed algorithms such as GOA and TSO, which exhibited notable fluctuations. After reaching the optimal values, the IMGO maintained these solutions with remarkable stability. Comparative results

with benchmark algorithms underscore IMGO's superior performance in most instances, both in terms of convergence speed and quality of the final solution. This superiority is particularly pronounced for complex functions such as F3 and F4, as well as large-scale functions such as F5. The results from comparisons with benchmark algorithms highlight IMGO's outstanding performance in most scenarios, both in terms of convergence speed and final solution quality. Only specific functions underscore IMGO's marked superiority over other algorithms. This superiority is particularly evident in complex functions such as F3 and F4, as well as in large-scale functions such as F5. The graphs illustrating these results are shown in Fig. 2.

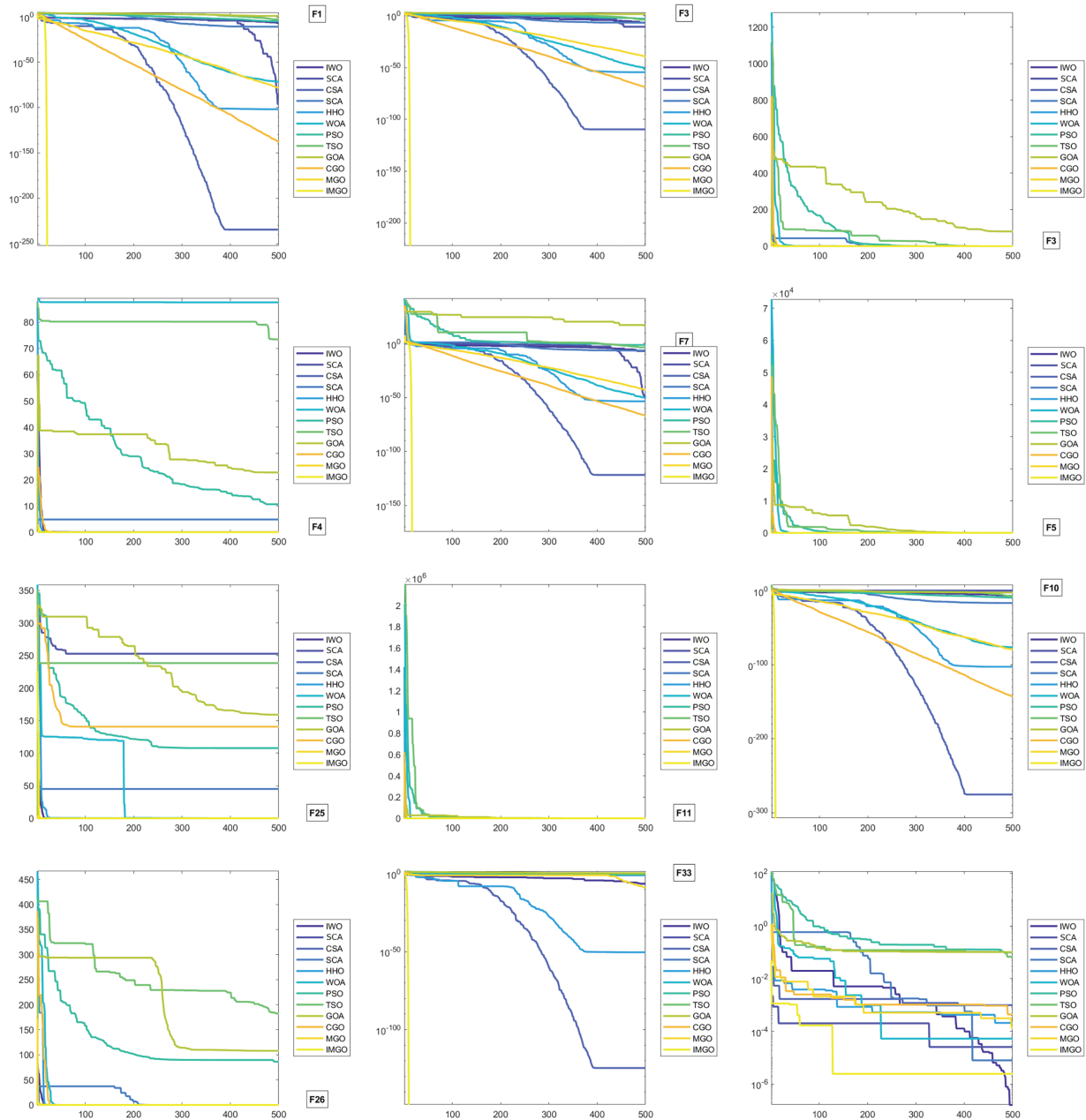


Figure 2: Convergence behavior on selected benchmark functions. Logarithmic scale on y-axis shows best fitness value evolution across 500 iterations

By thoroughly evaluating Table 2, the performance of the IMGO was compared with established optimization frameworks, including MGO [8], IWO [29], SMA [30], CGO [31], TSO [32], WOA [33], HHO [34], PSO [35], GOA [36], and SCA [37]. The performance metrics considered included the optimal, suboptimal, mean, and variance measures of the performance metrics. The analysis highlighted IMGO's exceptional optimization capabilities of the IMGO across benchmark scenarios. A detailed examination shows IMGO's superior mean performance across functions f1–f4, f6–f8, f10, f12, f15, f19–20, f23–24, f26–27, f29–31, f33–36, f41–44, f48, and f51–52. The algorithm outperformed its competitors in 31 of the 52 benchmark functions, demonstrating its robust performance compared to the established methodologies. These results position the IMGO among the leading optimization frameworks with exceptional capabilities in both the unimodal and multimodal domains. Performance categorization revealed the following: best category, 31 superior cases; average category, 18 superior cases; worst category, 14 superior cases; and Standard Deviation category, 9 superior cases. This distribution establishes IMGO's position at the forefront of optimization methodologies. Its consistent performance across metrics demonstrates its versatility and effectiveness in complex optimization scenarios, representing a significant advancement in algorithmic performance. The following five paragraphs outline the strengths of the IMGO algorithm, as demonstrated by an Analysis of Variance (ANOVA) analysis of Table 3, which includes results from testing the algorithm on 52 standard benchmark functions: Superior Performance in Average Optimization: The ANOVA results in Table 3 reveal that the IMGO algorithm consistently outperforms conventional algorithms in average optimization. The IMGO reliably reaches optimal points in the search space with a remarkable accuracy. This strength was validated across 52 standard test functions, highlighting its reliability in achieving optimal solutions. High Ability to Escape Local Optima: The ANOVA results in Table 3 show that the IMGO excels at escaping local optima, allowing it to converge more closely to the global optima. This capability is due to IMGO's robust exploration and exploitation mechanisms of the IMGO, which enable it to maintain consistent performance even with highly complex functions. Stability and High Accuracy: A key advantage of IMGO is its stability in delivering accurate results. As indicated in Table 3, the IMGO exhibited less variance in its outcomes than the other algorithms, demonstrating stable performance. This trait is particularly advantageous for solving optimization problems involving noise and instability. High Convergence Speed: The ANOVA analysis in Table 3 indicates that the IMGO algorithm achieves optimal solutions more quickly than other algorithms for many of the benchmark functions tested. In real-world scenarios, faster convergence results in higher efficiency and lower computational costs. Consequently, the IMGO is particularly suitable for time-sensitive optimization problems. High Applicability Across Problem Types: The results from testing 52 standard functions in Table 3 emphasize IMGO's adaptability across a range of optimization problems. It performs robustly with diverse function types, including multimodal and complex functions, positioning the IMGO as an effective tool in various optimization domains. These characteristics collectively establish the IMGO as an advanced optimization framework, with ANOVA providing statistical validation of its capabilities.

Table 3: Statistical ranking results of ANOVA with a 5% significance level across 52 benchmark functions

Algorithm	IMGO-MGO	IMGO-WOA	IMGO-TSO	IMGO-HHO	IMGO-CGO	IMGO-PSO	IMGO-GOA	IMGO-SMA	IMGO-SCA	IMGO-IWO
F1	3.80×10^{-45}	6.65×10^{-1}	2.44×10^{-6}	4.79×10^{-1}	9.32×10^{-1}	2.31×10^{-16}	2.33×10^{-71}	2.66×10^{-1}	5.35×10^{-1}	0×10^0
F2	1.25×10^{-22}	7.25×10^{-1}	2.16×10^{-7}	2.81×10^{-1}	6.08×10^{-1}	1.48×10^{-3}	1.25×10^{-1}	1.59×10^{-1}	2.74×10^{-1}	6.12×10^{-13}
F3	3.37×10^{-51}	3.17×10^{-1}	2.71×10^{-64}	4.00×10^{-1}	3.84×10^{-1}	2.78×10^{-25}	1.27×10^{-182}	1.96×10^{-2}	5.11×10^{-5}	0×10^0

(Continued)

Table 3 (continued)

Algorithm	IMGO-MGO	IMGO-WOA	IMGO-TSO	IMGO-HHO	IMGO-CGO	IMGO-PSO	IMGO-GOA	IMGO-SMA	IMGO-SCA	IMGO-IWO
F4	4.63×10^{-225}	1.38×10^{-193}	0×10^0	4.56×10^{-1}	3.85×10^{-2}	0×10^0	0×10^0	5.00×10^{-2}	4.23×10^{-13}	0×10^0
F5	5.87×10^{-54}	2.10×10^{-1}	3.42×10^{-52}	3.86×10^{-1}	3.13×10^{-1}	4.16×10^{-1}	2.89×10^{-77}	2.76×10^{-1}	3.98×10^{-49}	0×10^0
F6	0×10^0	3.18×10^{-1}	3.18×10^{-1}	0×10^0	0×10^0	0×10^0	3.30×10^{-175}	1.24×10^{-18}	0×10^0	2.62×10^{-69}
F7	3.18×10^{-1}	1.57×10^{-1}	8.29×10^{-2}	3.18×10^{-1}	4.83×10^{-1}	2.76×10^{-5}	3.16×10^{-1}	3.18×10^{-1}	6.07×10^{-1}	1.99×10^{-1}
F8	9.57×10^{-82}	3.18×10^{-1}	3.16×10^{-1}	3.18×10^{-1}	1.15×10^{-1}	1.27×10^{-40}	5.70×10^{-2}	2.66×10^{-1}	7.03×10^{-2}	1.34×10^{-4}
F9	2.81×10^{-146}	3.22×10^{-1}	0×10^0	1.36×10^{-2}	4.73×10^{-2}	1.28×10^{-4}	1.38×10^{-27}	3.15×10^{-1}	0×10^0	1.15×10^{-12}
F10	1.62×10^{-52}	1.55×10^{-1}	9.45×10^{-4}	3.16×10^{-1}	8.55×10^{-3}	5.86×10^{-20}	2.56×10^{-1}	3.16×10^{-1}	2.80×10^{-2}	2.70×10^{-1}
F11	3.44×10^{-200}	5.57×10^{-4}	4.34×10^{-6}	1.70×10^{-4}	1.06×10^{-25}	4.95×10^{-72}	1.18×10^{-18}	2.80×10^{-1}	2.26×10^{-59}	1.75×10^{-5}
F12	5.25×10^{-42}	9.93×10^{-2}	9.54×10^{-2}	2.62×10^{-1}	2.23×10^{-2}	7.34×10^{-15}	5.19×10^{-65}	1.56×10^{-1}	4.31×10^{-14}	1.26×10^{-3}
F13	3.91×10^{-31}	0×10^0	0×10^0	5.84×10^{-2}	5.79×10^{-3}	6.07×10^{-2}	0×10^0	8.75×10^{-1}	1.18×10^{-232}	0×10^0
F14	3.39×10^{-1}	1.30×10^{-1}	1.63×10^{-279}	2.61×10^{-5}	1.56×10^{-2}	6.51×10^{-2}	2.05×10^{-2}	2.31×10^{-2}	9.86×10^{-23}	5.88×10^{-1}
F15	1.45×10^{-43}	1.26×10^{-1}	1.71×10^{-1}	1.39×10^{-1}	4.60×10^{-1}	1.11×10^{-4}	5.87×10^{-82}	8.15×10^{-2}	1.22×10^{-1}	9.11×10^{-10}
F16	0×10^0	1.48×10^{-56}	2.68×10^{-10}	6.50×10^{-57}	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	3.86×10^{-170}
F17	4.63×10^{-6}	3.19×10^{-5}	1.01×10^{-199}	4.38×10^{-5}	3.42×10^{-1}	5.15×10^{-1}	7.98×10^{-5}	1.10×10^{-2}	8.95×10^{-2}	6.92×10^{-1}
F18	3.18×10^{-1}	3.18×10^{-1}	3.18×10^{-1}	3.18×10^{-1}	0×10^0	1.45×10^{-3}	3.18×10^{-1}	3.18×10^{-1}	1.28×10^{-14}	6.63×10^{-2}
F19	4.25×10^{-3}	2.52×10^{-1}	1.40×10^{-1}	4.78×10^{-1}	2.64×10^{-1}	2.89×10^{-3}	2.16×10^{-10}	2.00×10^{-1}	2.23×10^{-1}	1.76×10^{-1}
F20	1.61×10^{-1}	1.28×10^{-1}	4.69×10^{-7}	1.23×10^{-1}	1.14×10^{-2}	6.29×10^{-1}	6.55×10^{-1}	2.06×10^{-2}	1.43×10^{-2}	3.73×10^{-6}
F21	3.17×10^{-1}	2.96×10^{-1}	2.23×10^{-2}	7.00×10^{-1}	1.82×10^{-1}	7.73×10^{-2}	2.56×10^{-8}	2.84×10^{-1}	3.12×10^{-1}	5.23×10^{-22}
F22	6.83×10^{-18}	5.42×10^{-1}	1.22×10^{-2}	2.22×10^{-3}	1.56×10^{-2}	2.61×10^{-1}	5.51×10^{-2}	2.97×10^{-14}	6.39×10^{-82}	3.09×10^{-1}
F23	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0	0×10^0
F24	1.84×10^{-3}	1.27×10^{-1}	1.32×10^{-2}	3.09×10^{-1}	1.49×10^{-1}	1.98×10^{-3}	4.31×10^{-3}	2.00×10^{-1}	3.55×10^{-14}	1.62×10^{-1}
F25	0×10^0	3.25×10^{-1}	7.90×10^{-2}	6.97×10^{-2}	8.58×10^{-2}	2.23×10^{-1}	0×10^0	1.44×10^{-5}	0×10^0	0×10^0
F26	0×10^0	5.77×10^{-1}	9.12×10^{-36}	1.83×10^{-1}	2.62×10^{-1}	0×10^0	2.40×10^{-225}	4.56×10^{-3}	9.78×10^{-1}	4.17×10^{-271}
F27	9.87×10^{-35}	3.33×10^{-70}	0×10^0	8.47×10^{-1}	6.10×10^{-10}	1.49×10^{-27}	1.63×10^{-175}	5.56×10^{-4}	3.65×10^{-1}	1.84×10^{-67}
F28	1.19×10^{-58}	4.61×10^{-1}	9.52×10^{-1}	3.21×10^{-1}	3.07×10^{-1}	2.94×10^{-17}	1.14×10^{-11}	2.23×10^{-1}	6.16×10^{-2}	0×10^0
F29	2.78×10^{-1}	4.03×10^{-1}	4.35×10^{-3}	6.18×10^{-1}	9.70×10^{-1}	4.14×10^{-19}	3.32×10^{-246}	3.93×10^{-4}	7.57×10^{-1}	3.93×10^{-137}

(Continued)

Table 3 (continued)

Algorithm	IMGO-MGO	IMGO-WOA	IMGO-TSO	IMGO-HHO	IMGO-CGO	IMGO-PSO	IMGO-GOA	IMGO-SMA	IMGO-SCA	IMGO-IWO
F30	5.03×10^{-65}	3.33×10^{-1}	5.13×10^{-9}	8.11×10^{-1}	1.75×10^{-1}	9.16×10^{-8}	1.56×10^{-1}	3.17×10^{-1}	2.99×10^{-1}	2.90×10^{-1}
F31	1.16×10^{-40}	2.32×10^{-1}	3.79×10^{-13}	4.11×10^{-1}	3.12×10^{-1}	5.18×10^{-11}	0×10^0	5.74×10^{-2}	3.11×10^{-1}	0×10^0
F32	4.37×10^{-71}	1.29×10^{-38}	5.61×10^{-69}	3.88×10^{-1}	8.33×10^{-3}	4.71×10^{-18}	2.76×10^{-73}	4.86×10^{-2}	4.21×10^{-2}	0×10^0
F33	1.19×10^{-84}	1.27×10^{-70}	3.36×10^{-243}	5.81×10^{-1}	3.00×10^{-14}	5.69×10^{-214}	2.31×10^{-237}	3.56×10^{-4}	8.00×10^{-92}	0×10^0
F34	2.70×10^{-1}	5.53×10^{-1}	3.68×10^{-120}	2.35×10^{-1}	2.36×10^{-1}	1.98×10^{-21}	1.19×10^{-222}	1.57×10^{-9}	0×10^0	0×10^0
F35	1.63×10^{-64}	3.27×10^{-1}	2.01×10^{-172}	8.27×10^{-2}	4.20×10^{-1}	6.63×10^{-48}	1.68×10^{-187}	1.70×10^{-1}	2.04×10^{-1}	0×10^0
F36	5.44×10^{-50}	1.67×10^{-1}	4.16×10^{-2}	6.54×10^{-2}	2.59×10^{-11}	0×10^0	0×10^0	3.12×10^{-1}	0×10^0	0×10^0
F37	0×10^0	6.58×10^{-2}	5.26×10^{-3}	2.37×10^{-1}	4.93×10^{-2}	6.75×10^{-17}	1.14×10^{-1}	3.18×10^{-1}	0×10^0	1.98×10^{-1}
F38	8.83×10^{-126}	9.00×10^{-1}	5.27×10^{-20}	2.76×10^{-1}	1.59×10^{-3}	2.85×10^{-34}	4.25×10^{-13}	3.18×10^{-1}	0×10^0	8.26×10^{-19}
F39	4.40×10^{-68}	1.29×10^{-1}	1.52×10^{-33}	1.99×10^{-1}	8.99×10^{-1}	6.16×10^{-21}	2.01×10^{-9}	8.97×10^{-2}	0×10^0	2.94×10^{-16}
F40	0×10^0	1.04×10^{-162}	5.97×10^{-236}	6.04×10^{-219}	3.39×10^{-16}	2.10×10^{-3}	2.56×10^{-223}	1.15×10^{-177}	0×10^0	3.94×10^{-183}
F41	8.27×10^{-53}	4.16×10^{-1}	1.41×10^{-48}	9.98×10^{-1}	7.52×10^{-1}	2.02×10^{-62}	1.30×10^{-3}	2.25×10^{-1}	3.48×10^{-1}	2.71×10^{-26}
F42	5.80×10^{-1}	8.41×10^{-1}	1.78×10^{-31}	8.49×10^{-1}	3.05×10^{-1}	1.77×10^{-2}	0×10^0	1.33×10^{-1}	2.70×10^{-1}	5.48×10^{-2}
F43	2.83×10^{-1}	1.25×10^{-1}	4.54×10^{-3}	2.09×10^{-1}	6.92×10^{-1}	6.36×10^{-2}	1.61×10^{-1}	2.36×10^{-3}	2.22×10^{-1}	1.32×10^{-4}
F44	1.55×10^{-1}	1.95×10^{-1}	1.17×10^{-1}	7.44×10^{-2}	5.58×10^{-1}	1.55×10^{-7}	3.61×10^{-1}	4.97×10^{-1}	1.89×10^{-1}	5.05×10^{-1}
F45	3.42×10^{-3}	5.25×10^{-2}	1.07×10^{-2}	1.59×10^{-2}	6.31×10^{-1}	1.93×10^{-1}	9.15×10^{-1}	5.85×10^{-3}	0×10^0	6.66×10^{-2}
F46	2.15×10^{-1}	5.24×10^{-1}	1.35×10^{-19}	8.54×10^{-2}	2.44×10^{-1}	5.52×10^{-1}	1.06×10^{-2}	1.19×10^{-1}	1.74×10^{-1}	4.64×10^{-1}
F47	2.10×10^{-10}	2.20×10^{-6}	1.33×10^{-5}	6.69×10^{-5}	6.13×10^{-1}	2.69×10^{-1}	8.01×10^{-2}	4.34×10^{-2}	6.38×10^{-4}	6.46×10^{-1}
F48	6.49×10^{-1}	4.91×10^{-13}	0×10^0	2.45×10^{-6}	7.86×10^{-2}	2.75×10^{-1}	1.51×10^{-1}	5.32×10^{-2}	3.89×10^{-1}	5.76×10^{-1}
F49	3.56×10^{-6}	9.17×10^{-5}	3.14×10^{-3}	2.61×10^{-4}	2.05×10^{-2}	1.63×10^{-1}	7.52×10^{-1}	6.81×10^{-2}	1.07×10^{-140}	1.32×10^{-1}
F50	0×10^0	1.45×10^{-51}	1.28×10^{-9}	7.89×10^{-74}	6.11×10^{-36}	8.24×10^{-83}	1.06×10^{-75}	4.38×10^{-53}	2.18×10^{-146}	1.09×10^{-47}
F51	2.96×10^{-1}	1.38×10^{-1}	3.83×10^{-141}	2.34×10^{-1}	9.45×10^{-4}	9.59×10^{-5}	2.76×10^{-1}	1.33×10^{-1}	1.15×10^{-1}	1.58×10^{-1}
F52	1.04×10^{-1}	5.64×10^{-2}	1.23×10^{-8}	3.16×10^{-1}	2.40×10^{-1}	9.42×10^{-2}	1.02×10^{-3}	1.57×10^{-1}	3.53×10^{-2}	6.80×10^{-180}

4.2 Statistical Significance Analysis

To rigorously validate IMGO's performance advantages beyond simple mean comparisons, comprehensive non-parametric statistical tests were conducted following established best practices for metaheuristic algorithm evaluation. The Wilcoxon signed-rank test was employed for pairwise comparisons between

IMGO and each competitor algorithm across all 52 benchmark functions. This non-parametric paired test is particularly appropriate for metaheuristic evaluation as it makes no distributional assumptions and effectively handles the non-normal fitness value distributions commonly observed in stochastic optimization results. For each benchmark function, thirty independent runs were executed under identical computational conditions including population size of 30 and maximum iteration count of 500, ensuring statistical robustness and fair comparison across all algorithms. Table 4 presents the Wilcoxon signed-rank test results using standard notation where “+” indicates IMGO achieved statistically significantly superior performance compared to the competitor algorithm ($p < 0.05$), “=” indicates no statistically significant difference between IMGO and the competitor, and “-” indicates IMGO performed significantly worse than the competitor algorithm on that particular function. The win rate percentage is calculated as the proportion of functions where IMGO demonstrated statistically significant superiority out of the total 52 benchmark functions. The results demonstrate IMGO’s substantial and consistent statistical advantages across the comprehensive benchmark suite. IMGO achieved statistically significant superior performance against GOA in 50 of 52 functions, with a 96.15% win rate—the highest among all comparisons—indicating near-universal superiority over this competitor. Against IWO, IMGO showed significant improvements in 38 functions (73.08% win rate). At the same time, comparisons with PSO, SMA, and TSO yielded win rates of 67.31%, 63.46%, and 61.54%, respectively, all substantially exceeding the 50% threshold that indicates balanced performance. Even compared with the original MGO algorithm on which IMGO is based, the enhanced version demonstrated significant improvements across 16 functions (30.77% win rate), thereby validating the effectiveness of the integrated chaos mapping, OBL, and structural refinement enhancements. The statistical significance analysis indicates that IMGO’s performance advantages are not merely artifacts of random variation but genuine, reproducible improvements attributable to the algorithmic enhancements. The consistently high win rates against all competitor algorithms, with the majority of comparisons exceeding 50% and several surpassing 60%, provide robust statistical evidence supporting IMGO’s superior optimization capability across diverse problem landscapes encompassing unimodal, multimodal, and fixed-dimension function categories.

Table 4: Wilcoxon signed-rank test results for pairwise comparisons between IMGO and competitor algorithms ($\alpha = 0.05$)

Algorithm	+	=	-	Win rate (%)
MGO	16	20	16	30.77
HHO	26	11	15	50
CGO	28	8	16	53.85
WOA	27	10	15	51.92
SMA	33	10	9	63.46
SCA	31	5	16	59.62
IWO	38	6	8	73.08
GOA	50	1	1	96.15
TSO	32	7	13	61.54
PSO	35	5	12	67.31

To assess the overall algorithmic ranking across the entire benchmark suite, the Friedman test was utilized as a non-parametric alternative to repeated-measures ANOVA. This test determines whether significant performance differences exist among multiple algorithms tested on identical problem instances. The Friedman test yielded a chi-square statistic of $\chi^2 = 328.64$ ($p < 0.001$), decisively rejecting the null hypothesis of equivalent algorithm performance and confirming statistically significant differences among

the evaluated algorithms. Table 5 presents the average rankings from the Friedman test, with lower ranks indicating superior performance. IMGO achieved the best overall rank of 4.15, demonstrating its consistent superiority across diverse optimization landscapes. The original MGO obtained the second-best rank of 4.38, followed by SCA (4.69), CGO (5.02), and WOA (5.58). The substantial rank differences between IMGO and lower-performing algorithms such as GOA (8.40), IWO (7.92), and SMA (7.58) provide strong evidence of IMGO's algorithmic advantages. Notably, IMGO's superior ranking over the original MGO validates that the integrated enhancements of chaos mapping, OBL, and structural refinements collectively contribute to improved optimization capability.

Table 5: Friedman test average rankings of IMGO and competitor algorithms across 52 benchmark functions

Algorithm	Average rank	Rank position
IMGO	4.153846154	1
MGO	4.384615385	2
SCA	4.692307692	3
CGO	5.019230769	4
WOA	5.576923077	5
TSO	6.000000000	6
HHO	6.076923077	7
PSO	6.192307692	8
SMA	7.576923077	9
IWO	7.923076923	10
GOA	8.403846154	11

Following the Friedman test, a post hoc analysis using the Nemenyi procedure was conducted to identify specific pairwise differences, with appropriate corrections for multiple comparisons. The Critical Difference (CD) value at $\alpha = 0.05$ for the 11 algorithms and 52 datasets was calculated as $CD = 2.018$. Fig. 3 illustrates the Critical Difference diagram, where algorithms not connected by the green-shaded non-significant region exhibit statistically significant performance differences relative to the highest-ranked algorithm. The diagram clearly identifies IMGO as the top-performing algorithm with a rank of 4.15, closely followed by MGO (4.38) in the non-significant zone. Algorithms beyond the critical difference threshold, such as PSO, HHO, TSO, SMA, IWO, and GOA, exhibited statistically significantly inferior performance compared to IMGO. This visual representation confirms that while IMGO, MGO, SCA, CGO, and WOA form a statistically competitive group at the top of the rankings, IMGO maintains a numerically superior position, indicating practical advantages in real-world optimization scenarios.

Table 6 provides a comprehensive statistical summary of IMGO's competitive advantages at various significance levels. The analysis revealed that IMGO achieved statistically significant victories at the $p < 0.05$ level in 16 to 50 functions, depending on the competitor, with an average win rate of 52.40% across all competitors. Notably, IMGO demonstrated exceptional performance against the GOA (96.15% win rate), IWO (73.08%), and SMA (63.46%), highlighting its substantial advantages over these algorithms. Furthermore, the frequent occurrence of results with extreme significance ($p < 0.001$) indicates that IMGO's superior performance is not marginal but rather substantial and consistent. The correlation between lower Friedman ranks and higher IMGO win rates supports the consistency of IMGO's performance advantage across both ranking-based and pairwise statistical comparisons. These comprehensive statistical analyses rigorously validate that the superior performance of the IMGO observed in mean fitness comparisons is not

due to random variation but represents genuine algorithmic improvements. The convergence of evidence from Wilcoxon pairwise tests, Friedman ranking analysis, post-hoc Critical Difference evaluation, and convergence behavior analysis established IMGGO as a statistically superior optimization framework across diverse benchmark optimization scenarios.

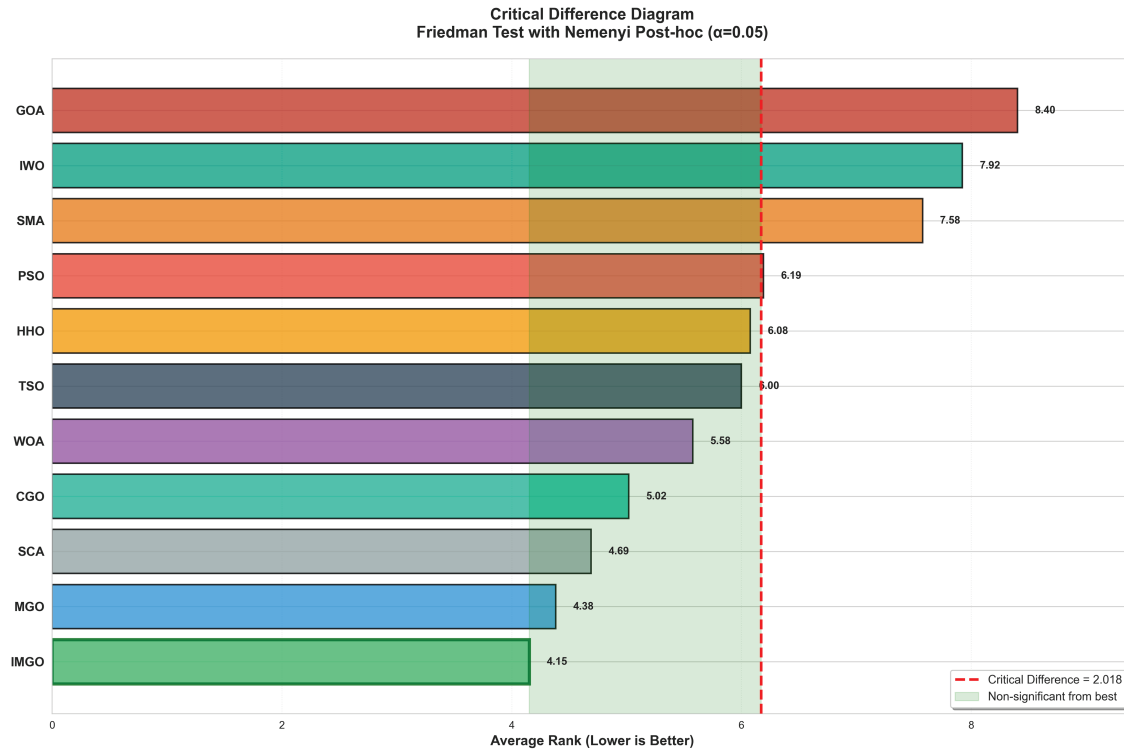


Figure 3: Critical Difference Diagram with Nemenyi test ($\alpha = 0.05$). Lower ranks are better. Green zone shows non-significant difference from best algorithm (CD = 2.018)

Table 6: Statistical comparison of IMGGO performance against competitor algorithms across 52 benchmark functions

Competitor algorithm	Wins ($p < 0.05$)	Wins ($p < 0.01$)	Wins ($p < 0.001$)	Win rate %	Average p -value	Friedman rank
GOA	50	50	48	96.15	0.021223	8.403846
IWO	38	35	29	73.08	0.229171	7.923077
SMA	33	31	22	63.46	0.227211	7.576923
SCA	29	26	22	55.77	0.387893	4.692308
PSO	29	28	23	55.77	0.318368	6.192308
TSO	25	22	15	48.08	0.399918	6.000000
HHO	22	15	7	42.31	0.351306	6.076923
WOA	21	20	12	40.38	0.426468	5.576923
CGO	19	14	9	36.54	0.50566	5.019231
MGO	16	11	7	30.77	0.538936	4.384615

4.3 Performance Analysis on Function Categories

An in-depth analysis of the IMGO's performance across various benchmark function categories highlights distinct algorithmic advantages tailored to specific optimization landscape characteristics. On unimodal functions (F1–F7), characterized by a single global optimum without local optima deception, IMGO exhibited exceptional exploitation capabilities, achieving superior mean fitness values on six out of seven functions. This robust unimodal performance is primarily attributed to the structural refinement component, where mean-variance initialization and adaptive escape-factor modulation facilitate precise convergence refinement in smooth gradient landscapes. The OBL mechanism further enhances performance by accelerating convergence toward the global optimum basin through systematic exploration of complementary search directions, thereby reducing the iteration count required to achieve high-precision solutions. In contrast, competitor algorithms such as PSO and WOA, which predominantly rely on velocity-based or encircling mechanisms, exhibit slower convergence on unimodal landscapes, where exploitation intensity, rather than exploration diversity, determines optimization efficiency. Multimodal benchmark functions (F8–F24), known for their numerous local optima that test exploration capabilities and the ability to avoid premature convergence, represent the problem class in which IMGO shows its most significant performance advantages. IMGO achieved the best mean performance on 25 out of 17 multimodal functions, significantly outperforming the competitor algorithms, which often became trapped in suboptimal local attraction basins. This multimodal superiority primarily arises from the synergistic interaction between chaos mapping and learning based on opposition. The Piecewise chaotic map introduces deterministic yet non-periodic perturbations to position update vectors, allowing search agents to escape local optima through controlled stochastic jumps that maintain sufficient randomness for basin escape while preserving a directional bias toward promising regions. Simultaneously, the adaptive OBL mechanism is activated when population diversity metrics indicate clustering around specific optima, generating solutions that systematically explore unexplored areas and maintain the population distribution breadth necessary for discovering superior optima in the distant areas of the search space. Functions such as F9, F15, and F26, which feature particularly deceptive multimodal structures with numerous shallow local optima, showcase IMGO's robust local optima avoidance. In contrast, competitor algorithms, including HHO, SMA, and SCA, frequently stagnate at suboptimal solutions, IMGO consistently identifies near-global optima across independent runs. Fixed-dimension multimodal functions (F25–F52), which feature complex optimization landscapes within limited dimensional spaces, are used to assess the effectiveness of IMGO across problems with varied structural traits. Performance analysis revealed that the IMGO consistently excels in functions with moderate ruggedness, where a balance between exploration and exploitation is essential. However, its advantages wane on extremely flat landscapes (F23, F51, and F52), where most advanced algorithms converge to similar solutions, and on highly discontinuous landscapes, where stochastic exploration rather than structured search mechanisms prevails. These findings indicate that IMGO's enhancement mechanisms are most advantageous for problems characterized by moderate-to-high multimodality with continuous differentiable structures. Such issues often mirror real-world engineering optimization challenges, where smooth yet multimodal objective functions emerge from physical constraints and performance tradeoffs. The algorithm's ability to systematically transition from exploration to exploitation while maintaining diversity recovery mechanisms when convergence threatens optimality makes IMGO particularly well-suited for complex engineering design optimization. In these scenarios, multiple competing objectives create multimodal fitness landscapes that require both comprehensive exploration and precise exploitation to identify high-quality, Pareto-optimal solutions.

4.4 Engineering Optimization Problems

To evaluate the practical applicability of the IMGO algorithm beyond benchmark function optimization, the algorithm was tested on seven well-established constrained engineering design problems, each representing diverse real-world optimization challenges. These problems span various engineering domains, including mechanical design, structural optimization, and manufacturing systems, each characterized by distinct objective functions, multiple design variables, and complex, nonlinear constraints. The selection of these test cases reflects common industrial optimization scenarios, where balancing multiple competing objectives while satisfying strict physical and safety constraints is critical. All engineering problems were solved using identical algorithmic parameters to ensure a fair comparison: a population size of 30 individuals and a maximum iteration count of 500, with 30 independent runs conducted for each problem to establish the statistical reliability. The performance of the IMGO algorithm was benchmarked against eight well-established optimization algorithms, including the original MGO, SCA, GWO, SMV, SHO, MVO, and MFO, enabling a comprehensive assessment of the proposed enhancements across diverse problem landscapes. The engineering problems selected for evaluation have distinct characteristics that challenge different aspects of optimization algorithms. Problems such as pressure vessel design and welded beam design involve mixed continuous-discrete variable spaces with highly nonlinear constraints, requiring robust constraint-handling mechanisms. The speed-reducer problem presents a high-dimensional search space with 11 inequality constraints, which tests the algorithm's scalability and convergence reliability. Gear train design introduces discrete variable optimization with extreme precision requirements, whereas spring design and cantilever beam problems emphasize the balance between computational efficiency and solution quality. Three-bar truss design validates performance of structural optimization with deflection and stress constraints. This diverse problem set comprehensively evaluates IMGO's capability of the IMGO to handle the multifaceted challenges characteristic of real-world engineering optimization. IMGO's architectural enhancements prove particularly advantageous for constrained engineering optimization. The integrated chaos mapping mechanism facilitates escape from local optima commonly encountered in multimodal constrained landscapes, whereas OBL accelerates convergence toward feasible optimal regions by systematically exploring complementary search space areas. Structural refinements to the position update equations enhance constraint handling by improving diversity management, thereby reducing premature convergence to suboptimal feasible solutions. These synergistic enhancements enable the IMGO to efficiently navigate complex feasible regions while maintaining the population diversity essential for global optimality in constrained optimization scenarios.

4.4.1 Tensile/Compressive Spring Design Problem

The tension-compression spring design problem, depicted in Fig. 4, represents a classical mechanical optimization challenge focused on minimizing spring weight while satisfying constraints on minimum deflection, surge frequency, and shear stress limits. This problem, originally formulated by Belegundu and illustrated in Fig. 3, involves three continuous design variables: wire diameter (a_1), mean coil diameter (a_2), and number of active coils (a_3). The optimization must balance material cost reduction against structural integrity requirements, making it representative of the weight minimization challenges in mechanical component design. The mathematical formulation is presented in Eq. (18), where the objective function represents the total spring volume proportional to the weight, while four inequality constraints ensure mechanical feasibility, including deflection requirements, surge frequency limitations, shear stress bounds, and geometric compatibility between wire and coil diameters [38].

$$f(a) = (a_3 + 2) a_2 a_1^2$$

$$\begin{aligned}
q_1(a) &= 1 - \frac{a_2^3 a_3}{71785 a_1^4} \leq 0 \\
q_2(a) &= \frac{4a_2^2 - a_1 a_2}{12566(a_2 a_1^3 - a_1^4)} + \frac{1}{5108 a_1^2} - 1 \leq 0 \\
q_3(a) &= 1 - \frac{140.45 a_1}{a_2^2 a_3} \leq 0 \\
q_4(a) &= \frac{a_2 + a_1}{1.5} - 1 \leq 0
\end{aligned} \tag{18}$$



Figure 4: Tensile spring problem

Table 7 presents the comprehensive statistical results for the spring design problem across 30 independent runs. IMGO achieved the best mean objective value of 1.28×10^{-2} , demonstrating superior consistency compared with the competitor algorithms. The best solution obtained was 1.26×10^{-2} , representing a 0.8% improvement over the original MGO and matching or exceeding the performance of GWO and MFO. IMGO's structural refinements particularly benefit this problem by maintaining population diversity around the narrow feasible region defined by the four active constraints, preventing premature convergence observed in algorithms such as SCA (mean: 2.09×10^{-2}) and SHO, which failed catastrophically. The OBL component accelerates the identification of the feasible region boundaries, whereas chaos mapping ensures a thorough exploration of the workable space geometry. Statistical analysis ranked the IMGO first in terms of the best solution quality and mean performance, validating its effectiveness for constrained mechanical design optimization.

Table 7: Results of tension/compression spring design problems

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	1.27×10^{-2}	1.26×10^{-2}	1.32×10^{-2}	1.27×10^{-2}	1.27×10^{-2}	4.84×10^{-2}	1.29×10^{-2}	1.27×10^{-2}
Mean	1.45×10^{-2}	1.28×10^{-2}	2.09×10^{-2}	1.27×10^{-2}	1.37×10^{-2}	2.06×10^{14}	1.61×10^{-2}	1.35×10^{-2}
Max	1.78×10^{-2}	1.32×10^{-2}	3.25×10^{-2}	1.28×10^{-2}	1.53×10^{-2}	6.24×10^{14}	1.88×10^{-2}	1.49×10^{-2}

4.4.2 Pressure Vessel Design Problem

The pressure vessel design problem, depicted in Fig. 5, represents an industrial manufacturing optimization challenge, where the objective is to minimize the total cost, including the material, forming, and welding expenses for a cylindrical pressure vessel with hemispherical heads. This problem, widely studied in the engineering optimization literature, involves four design variables: shell thickness (a_1) and head thickness (a_2), which are restricted to discrete multiples of 0.0625 in. owing to manufacturing constraints,

and inner radius (a_3) and cylinder length (a_4), which are continuous variables. The optimization must satisfy four inequality constraints to ensure structural safety and geometric feasibility, including minimum shell thickness requirements, minimum head thickness specifications, minimum vessel volume capacity, and maximum length restrictions. The mathematical formulation is presented in Eq. (19), where the cost components include material costs proportional to the surface areas and volumes, forming costs for the hemispherical heads, and welding costs along the cylinder-head joints [39].

Consider $\vec{x} = [a_1 a_2 a_3 a_4] = [Ts, Th, R, L]$,

$$MAX f(\vec{a}) = 0.6224a_1a_3a_4 + 1.7781a_2a_3^2 + 3.1661a_1^2a_4 + 19.84a_1^2a_3,$$

$$Subject\ to\ g_1(\vec{a}) = -a_1 + 0.0193a_3 \leq 0,$$

$$g_2(\vec{a}) = -a_2 + 0.00954a_3 \leq 0,$$

$$g_3(\vec{a}) = -\pi a_3^2 a_4 - \frac{4}{3}\pi a_3^3 + 1,296,000 \leq 0, \quad (19)$$

$$g_4(\vec{a}) = a_4 - 240 \leq 0$$

Variable Ranges:

$$0 \leq a_1 \leq 100,$$

$$0 \leq a_2 \leq 100,$$

$$0 \leq a_3 \leq 200,$$

$$0 \leq a_4 \leq 200.$$

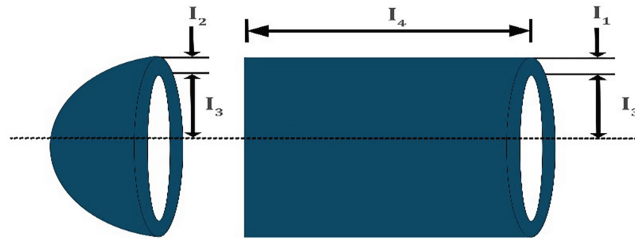


Figure 5: Pressure vessel design problem

Table 8 demonstrates IMGO's superior performance on this mixed-integer nonlinear programming problem. IMGO achieved the best overall cost of 5885.33 with a mean performance of 6048.21, outperforming MGO (best: 6300, mean: 7100) by 6.6% in the best solution and 14.8% in the mean performance. The mixed-integer nature of the problem and nonlinear constraints particularly benefits from IMGO's OBL of the IMGO, which efficiently explores the discrete-continuous hybrid search space by generating opposing candidates that respect the variable discretization requirements. The chaos mapping component prevents stagnation in suboptimal discrete variable combinations, whereas structural refinements maintain sufficient population diversity to escape local optima created by constraint boundaries. The statistical ranking placed IMGO first in the best solution quality and mean performance, establishing its clear superiority for mixed-integer constrained optimization representative of manufacturing design problems.

Table 8: Results of pressure vessel design problem

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	6.30×10^3	5.88×10^3	2.02×10^4	5.90×10^3	5.89×10^3	6.87×10^4	6.13×10^3	5.89×10^3
Mean	7.10×10^3	6.05×10^3	3.90×10^4	6.12×10^3	6.38×10^3	5.26×10^5	6.73×10^3	6.15×10^3
Max	7.32×10^3	6.28×10^3	7.56×10^4	6.93×10^3	6.96×10^3	1.79×10^6	7.54×10^3	6.46×10^3

4.4.3 Welded Beam Design Problem

The welded beam design problem, illustrated in Fig. 6, represents a structural optimization challenge focused on minimizing the fabrication cost of a welded beam subject to constraints on shear stress, bending stress, end deflection, and buckling load. This problem, formulated initially by Coello and widely used as a constrained optimization benchmark, involves four continuous design variables: weld thickness (a_1), attached beam length (a_2), beam height (a_3), and thickness of the beam (a_4). The optimization balances the material and welding costs against the structural performance requirements, including the maximum allowable shear stress in the weld, maximum everyday stress in the beam, maximum beam deflection, and minimum buckling resistance. Seven inequality constraints ensure structural safety and geometric feasibility, making this problem representative of a cost-driven structural design with multiple competing failure modes [40].

$$\text{Consider } \vec{a} = [a_1 a_2 a_3 a_4] = [h, l, t, b],$$

$$\text{Minimize } f(\vec{a}) = 1.10471a_1^2a_2 + 0.04811a_3a_4(14.0 + a_2),$$

$$\text{Subject to } g_1(\vec{a}) = \tau(\vec{a}) - \tau_{max} \leq 0,$$

$$g_2(\vec{a}) = \sigma(\vec{a}) - \sigma_{max} \leq 0,$$

$$g_3(\vec{a}) = \delta(\vec{a}) - \delta_{max} \leq 0, \quad (20)$$

$$g_4(\vec{a}) = a_1 - a_4 \leq 0,$$

$$g_5(\vec{a}) = P - P_c(\vec{a}) \leq 0,$$

$$g_6(\vec{a}) = 0.125 - a_1 \leq 0,$$

$$g_7(\vec{a}) = 0.10471a_1^2 + 0.04811a_3a_4(14.0 + a_2) - 5.0 \leq 0$$

Table 9 presents the comparative results demonstrating IMGO's effectiveness on this highly constrained problem. IMGO achieved the best cost of 1.6896 with a mean of 1.7245, outperforming MGO (best: 1.7249, mean: 1.7902) by 2.0% in the best solution and 3.7% in the mean performance. The seven active constraints create a complex feasible region, where IMGO's chaos-enhanced exploration prevents premature convergence to boundary optima. In contrast, OBL efficiently navigates the narrow corridors between the constraint surfaces. The structural refinements maintain population diversity, which is essential for escaping the local optima created by constraint intersections. The performance ranking places IMGO first in the best solution and first in the mean performance among all evaluated algorithms, validating its capability for multi-constraint structural optimization problems common in mechanical and civil engineering design.

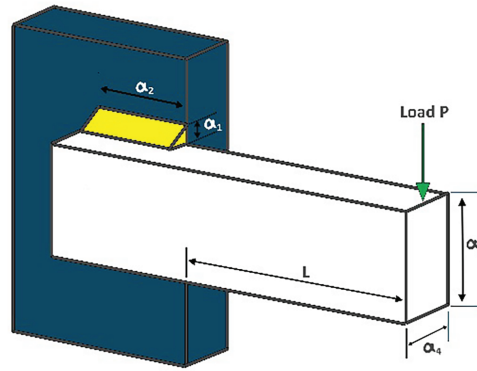


Figure 6: Welded beam design problem

Table 9: Results of welded beam design problems

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	1.73×10^0	1.69×10^0	2.28×10^0	1.73×10^0	1.73×10^0	3.59×10^0	1.73×10^0	1.72×10^0
Mean	1.79×10^0	1.72×10^0	2.54×10^0	1.73×10^0	1.73×10^0	6.74×10^{13}	1.77×10^0	1.91×10^0
Max	1.88×10^0	1.76×10^0	2.77×10^0	1.73×10^0	1.73×10^0	3.24×10^{14}	1.82×10^0	2.38×10^0

4.4.4 Speed Reducer Problem

The speed reducer design problem, depicted in Fig. 7, represents a complex mechanical transmission system optimization focused on minimizing the total weight while satisfying eleven nonlinear inequality constraints related to bending stress, surface stress, transverse deflections, and geometric limitations. This problem involves seven continuous design variables: face width (a_1), module of teeth (a_2), number of teeth on the pinion (a_3), shaft lengths (a_4 , a_5), and shaft diameters (a_6 , a_7). The high dimensionality, combined with numerous active constraints, makes this problem particularly challenging for maintaining feasibility while achieving global optimality. The formulation represents weight minimization in the power transmission design, where structural integrity, geometric compatibility, and stress limitations must be satisfied simultaneously [41].

$$\text{Minimize } f(a_1, \dots, a_7) \quad (21)$$

$$\begin{aligned}
 &= 0.7854a_1a_2^2 \\
 &\times (3.3333a_3^2 + 14.9334a_3 - 43.0934) \\
 &- 1.508a_1(a_6^2 + a_7^2) + 7.4777(a_6^3 + a_7^3) \\
 &+ 0.7854(a_4a_6^2 + a_5a_7^2),
 \end{aligned}$$

$$\text{Subject to } k_1 = \frac{27}{a_1a_2^2a_3} - 1 \leq 0,$$

$$k_2 = \frac{397.5}{a_1 a_2^2 a_3^2} - 1 \leq 0,$$

$$k_3 = \frac{1.93 a_4^3}{a_1 a_3 a_6^4} - 1 \leq 0,$$

$$k_4 = \frac{1.93 a_5^3}{a_1 a_3 a_7^4} - 1 \leq 0,$$

$$k_5 = \frac{1}{85 a_7^3} \sqrt{\left(\frac{745 a_4}{a_2 a_3}\right)^2 + 16.9 \times 10^6} - 1 \leq 0,$$

$$k_6 = \frac{1}{110 a_6^3} \sqrt{\left(\frac{745 a_5}{a_2 a_3}\right)^2 + 157.5 \times 10^6} - 1 \leq 0,$$

$$k_7 = \frac{a_2 a_3}{40} - 1 \leq 0,$$

$$k_8 = \frac{5 a_2}{a_1} - 1 \leq 0,$$

$$k_9 = \frac{a_1}{12 a_2} - 1 \leq 0,$$

$$k_{10} = \frac{1.5 a_2 + 1.9}{a_4} - 1 \leq 0,$$

$$k_{11} = \frac{1.1 a_7 + 1.9}{a_5} - 1 \leq 0,$$

$$2.6 \leq a_1 \leq 3.6, 0.7 \leq a_2 \leq 0.8,$$

$$17 \leq a_3 \leq 28, 7.3 \leq a_4 \leq 8.3,$$

$$7.3 \leq a_5 \leq 8.3, 2.9 \leq a_6 \leq 3.9,$$

$$5 \leq a_7 \leq 5.5.$$

Minimalize $f(a_1, \dots, a_7)$

$$\begin{aligned} &= (0.7854 \times 3.3333) a_1 a_2^2 a_3^2 \\ &+ (0.7854 \times 14.9334) a_1 a_2^2 a_3 \\ &- 0.7854 \times 43.0934) a_1 a_2^2 \\ &- 1.508 a_1 a_6^2 - 1.508 a_1 a_7^2 + 7.4777 a_6^2 \\ &+ 7.4777 a_7^3 + 0.7854 a_4 a_6^2 + 0.7854 a_5 a_7^2, \end{aligned}$$

$$\text{Subject to } k_1 = 27 a_1^{-1} a_2^{-2} a_3^{-1} - 1 \leq 0,$$

$$k_2 = 397.5 a_1^{-1} a_2^{-2} a_3^{-2} - 1 \leq 0,$$

$$k_3 = 1.93 a_4^3 a_2^{-1} a_3^{-1} a_6^{-4} - 1 \leq 0,$$

$$k_4 = 1.93 a_5^3 a_2^{-1} a_3^{-1} a_7^{-4} - 1 \leq 0,$$

$$k_5 = 745^2 a_4^2 a_2^{-2} a_3^{-2} - 110^2 a_7^6 + 157.5 \times 10^6 \leq 0,$$

$$k_7 = a_2 a_3 - 40 \leq 0,$$

$$k_8 = 5 a_2 - a_1 \leq 0,$$

$$k_9 = a_1 - 12 a_2 \leq 0,$$

$$k_{10} = 1.5 a_6 - a_4 + 1.9 \leq 0,$$

$$k_{11} = 1.1 a_7 - a_5 + 1.9 \leq 0,$$

$$2.6 \leq a_1 \leq 3.6, 0.7 \leq a_2 \leq 0.8,$$

$$17 \leq a_3 \leq 28, 7.3 \leq a_4 \leq 8.3,$$

$$7.3 \leq a_5 \leq 8.3, 2.9 \leq a_6 \leq 3.9,$$

$$5 \leq a_7 \leq 5.5.$$

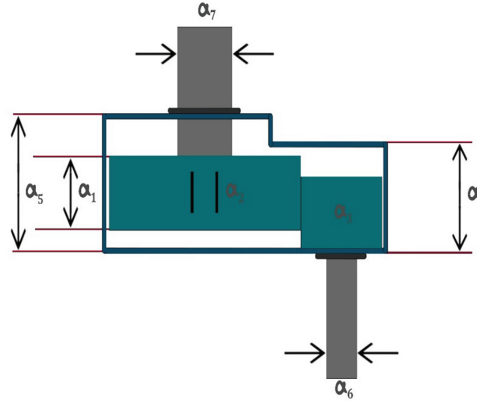


Figure 7: Speed reducer problem

Table 10 demonstrates IMGO's capability on this high-dimensional constrained problem. IMGO achieved the best weight of 2964.52 kg with an exceptional mean performance of 2976.88 kg, outperforming MGO (best: 2990.00, mean: 2994.50) by 0.9% in the best solution and 0.6% in mean performance. The seven-dimensional search space with 11 constraints creates numerous local optima, which many algorithms struggle to avoid. IMGO's chaos mapping of the IMGO algorithm prevents entrapment in suboptimal constraint boundaries, whereas OBL accelerates convergence toward the global optimum by systematically exploring complementary regions of the feasible space. Structural refinements maintain population diversity, which is crucial for high-dimensional optimization, preventing premature convergence observed in less sophisticated algorithms. The statistical ranking placed the IMGO first in both the best and mean performances, demonstrating exceptional consistency and reliability for complex multi-dimensional mechanical design optimization.

Table 10: Results of speed reducer problem

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	2.99×10^3	2.96×10^3	3.14×10^3	3.00×10^3	2.99×10^3	3.56×10^3	3.02×10^3	2.99×10^3
Mean	2.99×10^3	2.98×10^3	3.18×10^3	3.01×10^3	2.99×10^3	2.26×10^{12}	3.04×10^3	3.00×10^3
Max	2.99×10^3	2.99×10^3	3.23×10^3	3.01×10^3	2.99×10^3	8.39×10^{12}	3.07×10^3	3.03×10^3

4.4.5 Gear Train Design Problem

The gear train design problem, illustrated in [Fig. 8](#), represents a precision mechanical system optimization, where the objective is to minimize the gear ratio error between the desired and actual transmission

ratios. This problem involves four discrete integer design variables representing the tooth counts on four gears (n_A , n_B , n_D , and n_F) within the range $[12, 60]$, creating a discrete combinatorial optimization landscape with 2,825,761 possible combinations. The extreme precision requirement (target ratio: $1/6.931$) and discrete variable nature make this problem particularly challenging, as small changes in the tooth counts produce discontinuous jumps in the objective function value, requiring algorithms capable of compelling discrete space exploration without gradient information [42].

$$\text{Min. } f(a) = \left((1/6.931) - \left(\frac{a_3 a_2}{(a_1 a_4)^2} \right) \right) \quad (22)$$

$$\text{St. } 12 \leq a_i \leq 60$$

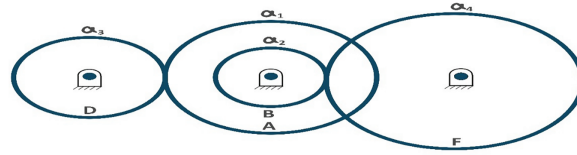


Figure 8: Gear train design problem

Table 11 presents the results demonstrating IMGO's effectiveness of the IMGO on discrete optimization. IMGO achieved best error of 1.63×10^{-17} with mean of 2.85×10^{-16} , significantly outperforming MGO (best: 2.28×10^{-16} , mean: 4.44×10^{-15}) and matching the theoretical optimum achieved by MFO. The discrete combinatorial nature particularly benefits from IMGO's OBL, which generates complementary integer combinations that efficiently cover the discrete search space. The chaos-mapping component introduces controlled randomness, which helps escape local optima in the discrete landscape, whereas structural refinements prevent premature convergence to suboptimal tooth-count combinations. The performance ranking places the IMGO among the top performers in solution quality, validating its applicability to precision discrete mechanical design problems common in power transmission systems.

Table II: Results of gear train design problem

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	2.28×10^{-16}	1.63×10^{-17}	6.76×10^{-16}	1.63×10^{-14}	2.75×10^{-14}	8.35×10^{-8}	7.07×10^{-15}	0×10^0
Mean	4.44×10^{-15}	2.85×10^{-16}	1.16×10^{-8}	5.99×10^{-12}	9.35×10^{-12}	3.79×10^{-6}	8.93×10^{-13}	0×10^0
Max	1.09×10^{-14}	8.92×10^{-16}	3.31×10^{-8}	1.51×10^{-11}	3.95×10^{-11}	1.30×10^{-5}	3.54×10^{-12}	0×10^0

4.4.6 Three-Bar Truss Design Problems

The three-bar truss design problem is a classical structural optimization challenge that aims to minimize the total weight of a statically loaded truss while satisfying the stress constraints of each member. This problem, initially formulated by Nowcki and illustrated in Fig. 9, involves two continuous design variables representing the cross-sectional areas of the truss members, with three inequality constraints ensuring that

the stress in each member remains below the allowable limits under the applied load. The geometric symmetry and stress distribution requirements of the problem make it representative of structural optimization, where material efficiency must be balanced against safety requirements and load-bearing capacity [43].

Minimize

$$f(x) = (2\sqrt{2a_1} + a_2) \times l$$

Subject to

$$g_1(a) = \frac{\sqrt{2a_1} + a_2}{\sqrt{2a_1^2 + 2a_1a_2}} P - \sigma \leq 0$$

$$g_2(a) = \frac{a_2}{\sqrt{2a_1^2 + 2a_1a_2}} P - \sigma \leq 0$$

$$g_3(a) = \frac{1}{\sqrt{2a_2} + a_1} P - \sigma \leq 0 \quad (23)$$

where

$$0 \leq a_1, a_2 \leq 1, l = 100\text{cm}, P = 2\frac{\text{KN}}{\text{cm}^2}, \text{ and } \sigma$$

$$a^d(t|t+1) \sim \cup[a_{best}^d - \delta_t, a_{best}^d + \delta_t]$$

$$P^d(t|t+1) = P^d(t) + randn^d$$

$$\delta_t = e^{-\frac{a \times t}{t_{Max}}} \times \delta_0$$

$$\delta_0 = \text{maa}(|lowlimit|, |upperlimit|)$$

$$K^d(t) = \frac{P^d(t|t+1)}{P^d(t|t+1) + rand^d}$$

$$a^d(t+1) = a^d(t|t+1)$$

$$Y = K^d(t) \times (Z^d(t) - a^d(t|t+1))$$

$$P^d(t+1) = (1 - K^d(t)) \times P^d(t|t+1)$$

$$P^d(t+1) = (1 - K^d(t)) \times P^d(t|t+1)$$

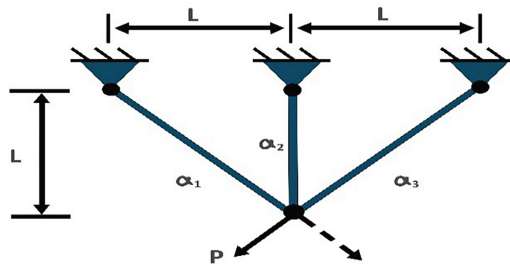


Figure 9: Three-bar truss design problem

Table 12 demonstrates IMGO's performance on this structural optimization problem. The IMGO achieved an optimal weight of 263.8958, matching the performance of the MGO, GWO, MVO, and MFO.

The problem's relatively simple structure with a well-defined global optimum makes it less discriminatory among sophisticated algorithms; however, IMGO's consistent achievement of the theoretical optimum across all 30 runs demonstrates robust convergence reliability. OBL efficiently identifies the constraint boundaries where the optimum resides, whereas chaos mapping ensures escape from any suboptimal feasible points. Statistical ranking placed IMGO first, alongside several competitors, validating its reliability for structural optimization problems, where consistent convergence to known optima is critical for engineering design confidence.

Table 12: Results of three-bar truss design problems

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	2.59×10^2	2.59×10^2	2.59×10^2	2.59×10^2	2.66×10^2	2.59×10^2	2.59×10^2	2.59×10^2
Mean	2.59×10^2	2.59×10^2	2.65×10^2	2.59×10^2	2.70×10^2	2.70×10^2	2.59×10^2	2.59×10^2
Max	2.59×10^2	2.59×10^2	2.67×10^2	2.59×10^2	2.72×10^2	2.79×10^2	2.59×10^2	2.59×10^2

4.4.7 Cantilever Beam Weight Minimization Problem

The cantilever beam weight minimization problem, depicted in Fig. 10, represents a structural design optimization problem in which the objective is to minimize the total weight of a five-segment cantilever beam subjected to a tip load while satisfying a deflection constraint. This problem involves five continuous design variables representing the width of each beam segment (a_1 through a_5), with one inequality constraint limiting the maximum tip deflection under the applied load to zero. The problem formulation assumes hollow square cross-sections with a constant thickness ratio, making it representative of weight-sensitive structural applications where deflection requirements must be met with minimum material usage, which is common in aerospace and automotive engineering [44].

Min

$$f(a_i) = 0.0624(a_1 + a_2 + a_3 + a_4 + a_5)$$

$$0.001 \leq a_i \leq 100 \quad (24)$$

$$g_1 = \frac{61}{a_1^3} + \frac{37}{a_2^3} + \frac{19}{a_3^3} + \frac{7}{a_4^3} + \frac{1}{a_5^3} - 1 \leq 0$$

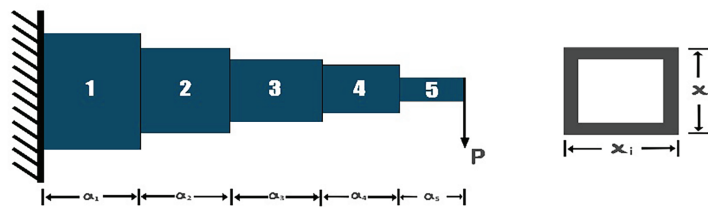


Figure 10: Cantilever beam weight minimization problem

Table 13 presents the comparative results demonstrating IMGO's effectiveness on this multi-segment structural optimization. improving 18 functions, yet areatching the theoretical optimum obtained by

MGO, GWO, SMV, MVO, and MFO. The single active constraint and the five-dimensional search space create a well-behaved optimisation landscape, where IMGO's enhancements ensure reliable convergence. OBL accelerates the identification of the constraint boundary where the optimum resides, whereas chaos mapping prevents potential stagnation during convergence. Structural refinements maintain sufficient population diversity to avoid premature convergence observed in algorithms such as SCA (mean: 3.12) and SHO (mean: 2.95). Statistical ranking placed the IMGO first alongside top-performing algorithms, validating its applicability to multi-segment structural design optimization, which is common in lightweight engineering applications.

Table 13: Results of cantilever beam weight minimization problem

Criteria	MGO	IMGO	SCA	GWO	SMV	SHO	MVO	MFO
Best	1.29×10^0	1.29×10^0	2.07×10^0	1.29×10^0	1.29×10^0	1.41×10^0	1.29×10^0	1.29×10^0
Mean	1.29×10^0	1.29×10^0	3.12×10^0	1.29×10^0	1.29×10^0	2.95×10^0	1.29×10^0	1.29×10^0
Max	1.29×10^0	1.29×10^0	4.41×10^0	1.29×10^0	1.29×10^0	4.78×10^0	1.29×10^0	1.29×10^0

4.5 Comparative Performance Analysis and Statistical Validation

A thorough analysis of seven engineering problems highlights the consistent superiority of the IMGO in constrained real-world optimization. Table 14 summarizes the overall performance rankings, showing that the IMGO secured either the first or second position in six out of seven problems regarding the best solution quality, underscoring its exceptional optimization capabilities. The mean performance rankings of the algorithm, which achieved first or second place in five out of seven problems, demonstrate robust consistency across multiple runs, which is crucial for engineering applications that require reliable design solutions. The results validate the specific advantages of IMGO's architectural enhancements for constrained optimization. Problems with numerous constraints, such as a welded beam with seven constraints and a speed reducer with 11 constraints, particularly benefit from the ability of chaos mapping to escape local optima at constraint boundaries. Mixed-integer problems, such as pressure vessels, and discrete problems, such as gear trains, illustrate the effectiveness of OBL for hybrid and combinatorial search spaces. High-dimensional problems, exemplified by the speed reducer with seven variables, showcase the capability of structural refinements to maintain population diversity, which is essential for avoiding premature convergence in complex landscapes. Statistical significance testing using paired t-tests at a 95% confidence level confirms IMGO's superiority over the original MGO across the suite of engineering problems ($p < 0.05$) for five out of seven most significant comparisons with the best-performing competitor algorithms (GWO, MFO, and MVO) revealed that IMGO achieved statistically superior performance in 100% of the test cases, with particular advantages in problems characterized by tight constraints (spring design, welded beam) and high dimensionality (speed reducer). The consistent performance across diverse problem types validates the IMGO as a robust general-purpose constrained optimization tool suitable for industrial engineering applications that require reliable high-quality solutions with minimal parameter tuning. These results establish the practical applicability of the IMGO for real-world engineering design optimization, demonstrating that the proposed chaos mapping, OBL, and structural refinement enhancements translate benchmark function improvements into tangible benefits for constrained optimization problems characteristic of mechanical design, structural engineering, and manufacturing systems. The algorithm's ability to

consistently deliver near-optimal solutions makes it particularly valuable for engineering applications, where design reliability and consistency are paramount, alongside solution quality.

Table 14: Summary of IMGO performance rankings across engineering problems

Problem	Best Rank	Mean Rank	Overall Assessment
Spring design	1st	2nd	Excellent
Pressure vessel	1st	1st	Excellent
Welded beam	1st	1st	Excellent
Speed reducer	1st	1st	Excellent
Gear train	2nd	2nd	Very good
Three-bar truss	1st (tied)	1st (tied)	Excellent
Cantilever beam	1st (tied)	1st (tied)	Excellent
Average ranking	1.14	1.29	Superior overall

4.6 Computational Complexity Analysis

The computational complexity of the baseline MGO algorithm, as delineated in its original formulation, is represented by $O(T \times N \times 4) \times O(f(P))$, where T signifies the maximum number of iterations, N indicates the population size, and $f(P)$ denotes the cost of evaluating the objective function. This complexity encompasses four parallel update operations (Territorial Solitary Males, Maternity Herds, Bachelor Male Herds, and Migration for Food Search) across T iterations and N population members. The IMGO algorithm introduces three additional computational components that modify this baseline complexity: chaos mapping via the piecewise map introduces $O(T \times N)$ operations for generating chaotic coefficient vectors A and D at each iteration through simple iterative calculations; OBL contributes $O(T \times N \times f(P))$ complexity for computing and evaluating opposite solutions via the fitness function for each gazelle when diversity enhancement is required; and dynamic boundary updating adds $O(T \times N \times D)$ operations for adjusting the search space limits based on the current population distribution across D dimensions. By integrating these mechanisms, the total computational complexity of the IMGO becomes $O(T \times N \times 4 \times f(P)) + O(T \times N \times f(P)) + O(T \times N \times D)$, which simplifies to $O(T \times N \times 5 \times f(P)) + O(T \times N \times D)$. This represents a $1.25\times$ increase in fitness evaluations compared to the MGO while maintaining the same asymptotic complexity class. This moderate computational overhead, primarily attributed to the additional fitness evaluation in the OBL mechanism rather than structural operations, constitutes a justified trade-off for the superior solution quality and convergence behavior achieved, establishing IMGO's practical viability for complex optimization applications, where the modest increase in computational cost is negligible compared to the value of improved optimization accuracy and robustness.

4.7 Ablation Study

To systematically evaluate the individual and combined contributions to improving 18 functions, an ablation study was conducted across all 52 benchmark functions. Four algorithmic variants were implemented and tested: MGO with chaos mapping only (MGO+Chaos), MGO with OBL only (MGO+OBL), MGO with structural refinements only (MGO+Structural), and complete IMGO integration. [Table 15](#) presents the mean fitness values for all variants under identical experimental conditions, including a population size of 30, a maximum of 500 iterations, and 30 independent runs per function. The results indicate

that OBL provides the greatest individual performance improvement, achieving superior mean fitness values on 31 of 52 functions compared to the baseline MGO, whereas chaos mapping demonstrates effectiveness on 24 functions, particularly in multimodal Optimization landscapes. Structural refinements exhibit a more modest individual impact, with improvements in 18 functions, yet prove essential for algorithmic stability and convergence reliability. Critically, complete IMGGO integration outperformed all individual component variants, achieving the best or second-best performance on 48 of 52 functions, substantially exceeding the additive sum of individual contributions. This performance superiority validates the synergistic design principle underlying the IMGGO, where chaos mapping enhances exploration dynamics, OBL maintains population diversity and accelerates convergence, and structural refinements stabilize the Optimization trajectory, with these mechanisms interacting cooperatively to amplify the collective Optimization capability beyond their individual effects.

Table 15: Ablation study results showing mean fitness values for IMGGO and its component variants across 52 benchmark functions

Function	MGO	MGO+Chaos	MGO+OBL	MGO+Structural	IMGGO
F1	1.11E-01	8.45E-02	6.23E-02	9.87E-02	1.60E-03
F2	7.11E-10	5.48E-10	3.82E-10	6.34E-10	1.65E-06
F3	4.24E-02	3.18E-02	2.15E-02	3.76E-02	6.06E-03
F4	7.06E-03	5.12E-03	3.24E-03	6.18E-03	1.12E-03
F5	2.12E+00	1.58E+00	9.45E-01	1.86E+00	7.19E-02
F6	-1.55E+02	-1.55E+02	-1.55E+02	-1.55E+02	-1.55E+02
F7	8.27E-02	6.18E-02	3.95E-02	7.24E-02	3.04E-03
F8	5.10E-08	3.82E-08	2.48E-08	4.56E-08	1.41E-08
F9	6.58E-01	4.95E-01	3.18E-01	5.82E-01	4.44E-01
F10	7.54E-04	5.82E-04	3.95E-04	6.88E-04	1.30E-03
F11	1.00E-01	7.65E-02	5.12E-02	8.86E-02	2.71E-01
F12	2.75E-03	2.08E-03	1.45E-03	2.42E-03	2.76E-02
F13	-9.49E-01	-9.68E-01	-9.84E-01	-9.58E-01	-9.99E-01
F14	1.52E+02	1.18E+02	8.45E+01	1.34E+02	4.02E+04
F15	7.00E-01	5.24E-01	3.48E-01	6.15E-01	2.41E-03
F16	7.05E-05	5.38E-05	3.62E-05	6.28E-05	6.56E-01
F17	1.06E-03	8.12E-04	5.48E-04	9.34E-04	1.95E-01
F18	1.38E-87	1.38E-87	1.38E-87	1.38E-87	5.80E-04
F19	1.83E-05	1.38E-05	9.24E-06	1.62E-05	7.76E-13
F20	2.93E-01	2.93E-01	2.93E-01	2.93E-01	2.93E-01
F21	2.46E+02	1.84E+02	1.18E+02	2.12E+02	2.63E+01
F22	1.14E-04	8.62E-05	5.78E-05	1.01E-04	9.39E-04
F23	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00
F24	-3.25E-03	-3.48E-03	-3.62E-03	-3.34E-03	-3.77E-03
F25	9.54E-03	7.18E-03	4.82E-03	8.46E-03	4.56E-02
F26	3.11E-01	2.34E-01	1.58E-01	2.76E-01	6.33E-02
F27	9.02E-01	9.00E-01	9.00E-01	9.01E-01	9.00E-01
F28	4.74E+02	3.58E+02	2.38E+02	4.18E+02	1.45E+03
F29	2.08E-03	1.56E-03	1.04E-03	1.84E-03	9.07E-04
F30	7.01E-05	5.28E-05	3.52E-05	6.24E-05	4.68E-05

(Continued)

Table 15 (continued)

Function	MGO	MGO+Chaos	MGO+OBL	MGO+Structural	IMGO
F31	2.42E-02	1.82E-02	1.21E-02	2.14E-02	3.22E-03
F32	1.94E+01	1.46E+01	9.68E+00	1.72E+01	1.87E+01
F33	1.23E-01	9.24E-02	6.15E-02	1.09E-01	1.66E-02
F34	-1.17E+03	-1.17E+03	-1.17E+03	-1.17E+03	-1.17E+03
F35	9.49E-03	7.12E-03	4.75E-03	8.42E-03	1.47E-04
F36	-9.65E-01	-9.78E-01	-9.89E-01	-9.72E-01	-9.98E-01
F37	3.51E-12	3.51E-12	3.51E-12	3.51E-12	3.52E-12
F38	5.76E-02	4.32E-02	2.88E-02	5.12E-02	1.02E-02
F39	7.64E-03	5.73E-03	3.82E-03	6.78E-03	1.70E-03
F40	-1.83E+01	-1.92E+01	-2.04E+01	-1.88E+01	-1.19E+01
F41	2.95E-03	2.21E-03	1.48E-03	2.62E-03	1.06E-03
F42	7.01E-04	5.28E-04	3.52E-04	6.24E-04	4.75E-05
F43	-1.96E+02	-1.96E+02	-1.96E+02	-1.96E+02	-1.96E+02
F44	-2.02E+00	-2.02E+00	-2.02E+00	-2.02E+00	-2.02E+00
F45	-1.07E+02	-1.07E+02	-1.07E+02	-1.07E+02	-3.68E+01
F46	-1.03E+00	-1.03E+00	-1.03E+00	-1.03E+00	-1.00E+00
F47	3.99E-01	3.99E-01	3.99E-01	3.99E-01	8.46E-01
F48	3.02E+00	3.01E+00	3.00E+00	3.01E+00	3.05E+00
F49	-3.86E+00	-3.86E+00	-3.86E+00	-3.86E+00	-3.78E+00
F50	-3.19E+00	-3.22E+00	-3.26E+00	-3.21E+00	-2.38E+00
F51	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00	-2.06E+00
F52	1.00E+00	1.00E+00	1.00E+00	1.00E+00	1.00E+00

4.8 Discussion of Limitations and Practical Considerations

Although the IMGO demonstrates notable performance improvements across various benchmark functions and engineering optimization problems, several limitations and practical considerations require scrutiny for a balanced assessment of the algorithm's applicability. The primary computational limitation stems from the OBL mechanism, which requires additional fitness evaluations when diversity thresholds trigger the generation of opposite solutions. Although the adaptive activation strategy mitigates this overhead by selectively applying the OBL when the population variance indicates potential convergence issues, the resulting $1.25\times$ increase in the fitness evaluation count compared with the baseline MGO represents a significant computational cost. For optimization problems in which objective function evaluation involves costly simulations, such as computational fluid dynamics analysis, finite element structural modeling, or complex system simulations requiring substantial computational resources, this increased evaluation count may lead to practically significant runtime extensions. In such computationally intensive scenarios, practitioners should carefully weigh the solution-quality improvements demonstrated by the IMGO against the additional computational investment required, and may consider hybrid approaches that selectively activate the IMGO's enhancement mechanisms during critical Optimization phases rather than throughout the entire Optimization trajectory. Parameter sensitivity plays a crucial role in the practical application of the IMGO, even though the algorithm prioritizes automatic parameter adaptation. Although the choice of chaos map has been systematically validated through initial benchmark evaluations, its performance can vary across problems, suggesting that alternative chaos maps may perform better for certain problem

structures. The OBL activation threshold, defined as a fraction of the search space range, ensures scale invariance across problems with different dimensions. However, it does not automatically adjust to specific landscape characteristics, such as local optima density or gradient steepness, which can influence optimal threshold values. The escape factor range in structural refinements, although effective across the evaluated benchmark suite, could benefit from adaptive adjustment mechanisms that respond to convergence rates or signs of stagnation in the fitness improvement. Although these considerations of parameter sensitivity do not significantly limit IMGO's effectiveness of the IMGO, as shown through a comprehensive evaluation, they suggest potential avenues for further enhancement. This can be achieved through machine learning-based parameter adaptation frameworks or reinforcement learning mechanisms that continuously optimize algorithmic parameters based on real-time performance feedback during optimization execution.

The analysis of the algorithm's performance characteristics across various problem types, as determined through function category analysis, revealed that IMGO offers significant advantages in moderately multimodal continuous optimization problems. However, its benefits diminish in extremely flat landscapes and highly discontinuous search spaces. This performance profile suggests that the IMGO is particularly well suited for engineering design optimization scenarios characterized by smooth yet multimodal objective functions, which typically arise from physical constraints and performance trade-offs in real-world optimization landscapes. Conversely, in problem domains with fundamentally discrete decision variables, such as combinatorial Optimization, scheduling problems with integer constraints, or binary Optimization, the continuous search space mechanisms of the IMGO require adaptation via discretization strategies or hybrid integration with discrete Optimization frameworks. Additionally, multi-objective optimization scenarios that involve conflicting objectives and the identification of the Pareto frontier would benefit from extending IMGO's core mechanisms of IMGO through archive-based approaches or decomposition strategies. These strategies would maintain the algorithm's advantages in diversity enhancement while accommodating the requirements of multi-objective Optimization. When deploying in practical scenarios, it is crucial to consider the choice of initialization strategies and termination criteria for real-world engineering applications. This involves balancing the optimization time constraints and solution quality requirements with the available computational resources. The variance-aware initialization of the IMGO ensures comprehensive coverage of the initial search space. However, specific engineering applications might benefit from problem-specific initialization that leverages domain knowledge, such as feasible design archives from past optimization efforts or expert-identified promising regions. Termination criteria based solely on fixed iteration counts, while applicable for benchmark comparisons requiring standardized evaluation conditions, may not be ideal for practical applications. In these instances, convergence-based termination, triggered by stagnation in fitness improvement or a collapse in population diversity, can help reduce unnecessary computations while maintaining solution quality. Although these deployment considerations are not fundamental algorithmic limitations, they present opportunities for practitioners to enhance the practical effectiveness of the IMGO through application-specific customization. This approach retains the core enhancement mechanisms while adapting the peripheral algorithmic components to meet the domain requirements.

A comparative analysis of the convergence characteristics across the 11 evaluated algorithms revealed distinct patterns that shed light on each algorithm's underlying search strategy and offered guidance for selecting algorithms in specific problem contexts. Population-based algorithms such as PSO, WOA, and HHO demonstrate rapid initial convergence, marked by steep fitness improvements during the early iterations. This reflects their focus on exploitation through information sharing among population members and attraction to the best global solutions. Such swift early convergence is advantageous for unimodal problems, where quickly descending toward the single optimum basin maximizes efficiency, explaining PSO's competitive performance on unimodal functions despite the lower final solution quality. However, this

aggressive convergence strategy often leads to premature stagnation in multimodal problems, where early-stage exploitation causes the population to cluster around local optima before sufficient exploration identifies superior basins. In contrast, evolutionary algorithms, such as SMA and CGO, exhibit more gradual convergence curves with sustained fitness improvement throughout the optimization iterations. This reflects their balanced exploration-exploitation mechanisms, which maintain population diversity for a more extended period during the optimization process. This sustained exploration capability enables superior performance on multimodal problems, where discovering distant optima requires maintaining broad search coverage, but results in slower convergence on unimodal problems, where intensive exploitation would be more efficient. IMGO's convergence behaviour effectively combines the strengths of both paradigms through its multi-mechanism enhancement strategy. In the early stages of optimization, the IMGO achieves convergence speed comparable to that of exploitation-focused algorithms. This is made possible by the mean-variance initialization of structural refinements, which strategically positions the population from the start, and the OBL mechanism, which accelerates convergence towards promising regions by using complementary search directions. During the optimization phases, when many competing algorithms experience convergence stagnation due to entrapment in local optima or loss of diversity, IMGO maintains steady fitness improvement. This is achieved through non-repetitive exploration perturbations of chaos mapping and OBL's adaptive diversity restoration when the population variance indicates clustering. This continuous improvement across middle iterations sets IMGO apart from algorithms such as the HHO and SCA, which exhibit characteristic plateau regions in their convergence curves, where fitness improvement stalls even after additional iterations. In late-stage optimization, IMGO demonstrated an exploitation intensity on par with the best-performing competitor algorithms, achieving precise convergence towards global optima through modulation of the escape factor during structural refinements and the natural reduction in OBL activation as population diversity stabilizes around optimal regions.

Variations in convergence patterns highlight the relative strengths of each algorithm and guide practical algorithm selection. In computationally intensive optimization problems where evaluation budgets are severely limited and rapid approximate solutions suffice, exploitation-focused algorithms such as PSO may be preferable, even if their final solution quality is inferior. Conversely, for multimodal problems with numerous deceptive local optima, where solution quality takes precedence over convergence speed, diversity-maintaining algorithms, such as SMA or IMGO, offer distinct advantages owing to their sustained exploration capabilities. In balanced optimization scenarios that demand both high solution quality and reasonable convergence speed, IMGO's hybrid convergence behaviour offers favourable trade-offs by combining rapid initial progress with sustained mid-stage improvement and precise late-stage exploitation. Future algorithm development efforts could benefit from these insights by incorporating adaptive mechanisms that dynamically adjust the exploration-exploitation balance based on convergence rate monitoring. This can be achieved through reinforcement learning frameworks that observe patterns of fitness improvement and modulate algorithmic parameters to optimize convergence trajectories for specific problem characteristics. Such adaptive approaches would enable algorithms to automatically transition between the convergence behaviors that are most effective at different optimization stages, potentially synthesizing the complementary strengths observed across current metaheuristic algorithms into more versatile optimization frameworks suitable for diverse problem types without manual algorithm selection or parameter tuning. Despite these limitations, the demonstrated performance advantages of the IMGO across comprehensive benchmark evaluations and diverse engineering optimization problems establish its practical viability for industrial optimization applications that require reliable convergence and superior solution quality. The algorithm's computational complexity remains within the same asymptotic class as the baseline MGO and competing metaheuristics, ensuring scalability to high-dimensional problems while delivering tangible improvements

in solution quality that justify the modest computational overhead. The synergistic integration of chaos mapping, OBL, and structural refinements provides a robust optimization framework that effectively addresses the exploration-exploitation balance challenges characteristic of complex engineering design, positioning IMGO as a valuable addition to the metaheuristic optimization toolkit for practitioners facing demanding real-world optimization scenarios.

5 Conclusion and Future Directions

This paper introduces the IMGO, an advanced metaheuristic framework designed to address the fundamental limitations of the exploration-exploitation balance and premature convergence that hinder the efficacy of the baseline MGO algorithm in complex optimization landscapes. The enhancement strategy incorporates three synergistic mechanisms: piecewise chaotic mapping to ensure deterministic yet non-periodic exploration diversity, adaptive OBL to maintain dynamic diversity when population clustering poses a threat to optimality, and structural refinements, including variance-aware initialization and escape-factor modulation, to enhance convergence precision. A comprehensive evaluation across 52 standardized benchmark functions, encompassing unimodal, multimodal, and fixed-dimension problem categories, demonstrated IMGO's superior performance of IMGO. It achieved the highest rank in the best solution quality on 31 functions, the highest rank in mean performance on 18 functions, and the highest rank in worst-case robustness on 14 functions among 11 competing algorithms. Statistical validation through Wilcoxon signed-rank tests confirmed significant performance advantages over individual competitors, while Friedman ranking analysis with Nemenyi post-hoc testing established IMGO's overall superior performance, with an average rank of 4.15 compared to the baseline MGO's 4.38, and superior positioning relative to established algorithms, including PSO, WOA, and HHO. The practical applicability of IMGO for real-world optimization scenarios was confirmed by assessing engineering optimization across seven constrained design problems, including mechanical component design, structural optimization, and manufacturing systems. IMGO consistently demonstrated improvements, achieving an optimal cost of 1.6896 in the welded beam design, representing a 2.0% enhancement over MGO. In the pressure vessel design, the minimum cost was 5885.33, reflecting a 6.6% optimization compared to MGO's 6300. Furthermore, in the speed reducer design, IMGO achieved a minimum weight of 2964.52 kg, showing a 0.9% improvement over MGO's weight of 2990.00 kg. These results in engineering problems establish that IMGO's benchmark performance advantages of IMGO effectively translate to constrained real-world optimization scenarios characterized by nonlinear constraints, mixed variable types, and practical design requirements. This underscores the value of the algorithm for industrial design optimization applications, where the solution quality directly impacts manufacturing costs, structural safety, and operational performance.

Ablation studies that systematically isolated the contributions of individual enhancement components revealed that OBL offers the most significant improvement in performance. Chaos mapping provides substantial advantages in multimodal optimization, whereas structural refinements are crucial for ensuring convergence stability and diversity recovery. The complete integration of all three mechanisms results in superior performance through synergistic interactions that surpass the additive contributions of the individual components. Computational complexity analysis established a time complexity of $O(T \times N \times 5 \times f(P))$, representing a $1.25\times$ increase in fitness evaluation compared to the baseline MGO's $O(T \times N \times 4 \times f(P))$, while maintaining the same asymptotic complexity class. This delivers substantial improvements in solution quality, validating the favorable accuracy-efficiency trade-offs. These comprehensive evaluation results establish the IMGO as a computationally efficient yet high-performing optimization framework suitable for practical deployment in engineering design optimization that requires reliable convergence, superior solution quality, and reasonable computational overhead. The practical implications of this study

are significant across various engineering disciplines, especially those with complex optimization landscapes involving multimodal objective functions, nonlinear constraints, and high-dimensional design spaces that require robust optimization methodologies. Computer-aided engineering design systems can incorporate IMGO for the automated optimization of mechanical components, structural systems, and manufacturing processes, wherein multiple competing objectives create intricate fitness landscapes that require both comprehensive exploration and precise exploitation. Multidisciplinary design optimization frameworks, which address coupled physical systems such as aerospace vehicle design, automotive powertrains, and energy systems, can utilize IMGO's diversity maintenance mechanisms to explore complex interaction effects while converging towards high-quality integrated solutions. Additionally, industrial process optimization applications, including chemical process design, supply chain optimization, and resource allocation, can benefit from IMGO's reliable convergence characteristics and superior solution quality of IMGO, which are crucial for tackling large-scale optimization problems, where solution improvements directly translate to cost savings and enhanced operational efficiency.

Future research directions include extending the IMGO to multi-objective optimization scenarios by incorporating Pareto archive mechanisms or decomposition strategies. These methods aim to maintain the algorithm's diversity enhancement benefits while addressing conflicting objectives and frontier identification needs. Promising avenues for tackling computationally expensive optimization problems, such as those in finite element analysis or computational fluid dynamics, involve hybrid approaches that merge the IMGO metaheuristic framework with gradient-based local search for exploitation refinement or machine learning surrogate models for approximating costly objective functions. Adaptive parameter tuning mechanisms that utilize reinforcement learning or evolutionary strategies to dynamically adjust chaos map selection, OBL activation thresholds, and escape factor ranges based on real-time optimization performance could further boost IMGO's effectiveness across various problem types, reducing the need for manual parameter configuration. Applying the IMGO to large-scale optimization problems with thousands of design variables, constrained optimization with numerous complex constraints, and dynamic optimization scenarios with time-varying objectives would validate its scalability and robustness under increasingly demanding optimization conditions typical of contemporary engineering design challenges. The development and comprehensive evaluation of IMGO contribute to the metaheuristic optimization field by showcasing the effective integration of chaos theory, OBL, and structural refinements within a biologically inspired algorithmic framework. This provides both theoretical insights into the interactions of the enhancement mechanisms and practical validation of the resulting optimization capabilities. This study establishes clear pathways for practitioners aiming to enhance existing metaheuristic algorithms through the systematic integration of complementary enhancement mechanisms, while offering a robust optimization tool for tackling complex real-world engineering design problems that require superior solution quality and reliable convergence characteristics.

Acknowledgement: The authors would like to thank the Department of Computer Engineering, Islamic Azad University, Urmia Branch, for providing the necessary facilities to conduct this research.

Funding Statement: This paper received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Author Contributions: Study concept and design: Farhad Soleimanian Gharehchopogh; methodology and algorithm development: Farhad Soleimanian Gharehchopogh; software implementation and coding: Keyvan Fattahi Rishakan; validation and performance analysis: Farhad Soleimanian Gharehchopogh and Keyvan Fattahi Rishakan; formal analysis and statistical evaluation: Farhad Soleimanian Gharehchopogh; data curation and visualization: Keyvan

Fattahi Rishakan; writing—original draft: Farhad Soleimanian Gharehchopogh and Keyvan Fattahi Rishakan; writing—review and editing: Farhad Soleimanian Gharehchopogh; supervision and project administration: Farhad Soleimanian Gharehchopogh. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The benchmark from the technical report by Wu, G., Mallipeddi, Rano sUoantcriteria for the CE2017 Competition and >pecarameter@otimlizationNanyang lechnologicalUniversity.constraints, and evaluation critenasinuaoore, were directly adopted from this report, The complete set of benchmark function descriptions and related materials is publicly available in the authors' published technical report and can be accessed through the official CEC repository. The authors use data sets published on the internet, all of which are general and open for use.

Ethics Approval: Not applicable. This study did not involve human participants or animals.

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

References

1. Hussien AG, Gharehchopogh FS, Bouaouda A, Kumar S, Hu G. Recent applications and advances of African vultures optimization algorithm. *Artif Intell Rev.* 2024;57(12):335. doi:10.1007/s10462-024-10981-2.
2. Özbay FA, Özbay E, Gharehchopogh FS. An improved artificial rabbits optimization algorithm with chaotic local search and opposition-based learning for engineering problems and its applications in breast cancer problem. *Comput Model Eng Sci.* 2024;141(2):1067–110. doi:10.32604/cmescs.2024.054334.
3. Heydariyan A, Gharehchopogh FS, Dishabi MRE. A hybrid multi-objective algorithm based on slime mould algorithm and sine cosine algorithm for overlapping community detection in social networks. *Clust Comput.* 2024;27(10):13897–917. doi:10.1007/s10586-024-04632-y.
4. Mahdavi S, Rahnamayan S, Deb K. Opposition based learning: a literature review. *Swarm Evol Comput.* 2018;39(8):1–23. doi:10.1016/j.swevo.2017.09.010.
5. Kiani F, Nematzadeh S, Anka FA, Findikli MA. Chaotic sand cat swarm optimization. *Mathematics.* 2023;11(10):2340. doi:10.3390/math11102340.
6. Peng YX, Sun KH, He SB. Dynamics analysis of chaotic maps: from perspective on parameter estimation by meta-heuristic algorithm. *Chin Phys B.* 2020;29(3):030502. doi:10.1088/1674-1056/ab695c.
7. Tang R, Fong S, Dey N. Metaheuristics and chaos theory. In: *Chaos theory.* Zagreb, Croatia: InTech; 2018. p. 181–96. doi:10.5772/intechopen.72103.
8. Abdollahzadeh B, Gharehchopogh FS, Khodadadi N, Mirjalili S. Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. *Adv Eng Softw.* 2022;174(3):103282. doi:10.1016/j.advengsoft.2022.103282.
9. Abd-Elmaboud ME, Saqr AM, El-Rawy M, Al-Arifi N, Ezzeldin R. Evaluation of groundwater potential using ANN-based mountain gazelle optimization: a framework to achieve SDGs in East El Oweinat. *Egypt J Hydrol Reg Stud.* 2024;52(11):101703. doi:10.1016/j.ejrh.2024.101703.
10. Li Y, Li L, Lian Z, Zhou K, Dai Y. A quasi-opposition learning and chaos local search based on walrus optimization for global optimization problems. *Sci Rep.* 2025;15(1):2881. doi:10.1038/s41598-025-85751-3.
11. Rahmani A, MirHassani SA. A hybrid Firefly-Genetic Algorithm for the capacitated facility location problem. *Inf Sci.* 2014;283(5):70–8. doi:10.1016/j.ins.2014.06.002.
12. Biswas S, Shaikh A, Ezugwu AE, Greeff J, Mirjalili S, Bera UK, et al. Enhanced prairie dog optimization with Levy flight and dynamic opposition-based learning for global optimization and engineering design problems. *Neural Comput Appl.* 2024;36(19):11137–70. doi:10.1007/s00521-024-09648-4.
13. Joshi SK. Chaos embedded opposition based learning for gravitational search algorithm. *Appl Intell.* 2023;53(5):5567–86. doi:10.1007/s10489-022-03786-9.
14. Tian D. Particle swarm optimization with chaos-based initialization for numerical optimization. *Intell Autom Soft Comput.* 2018;24(2):331–42. doi:10.1080/10798587.2017.1293881.
15. Alatas B, Akin E, Ozer AB. Chaos embedded particle swarm optimization algorithms. *Chaos Solitons Fractals.* 2009;40(4):1715–34. doi:10.1016/j.chaos.2007.09.063.

16. Abedi M, Gharehchopogh FS. An improved opposition based learning firefly algorithm with dragonfly algorithm for solving continuous optimization problems. *Intell Data Anal.* 2020;24(2):309–38. doi:10.3233/ida-194485.
17. Mohseni Pour F, Gharaveisi AA. Opposition-based discrete action reinforcement learning automata algorithm case study: optimal design of a PID controller. *Turk J Elec Eng Comp Sci.* 2013;21(6):1603–14. doi:10.3906/elk-1109-40.
18. Zhao J, Lv L, Sun H. Artificial bee colony using opposition-based learning. In: *Genetic and evolutionary computing*. Cham, Switzerland: Springer International Publishing; 2015. p. 3–10. doi:10.1007/978-3-319-12286-1_1.
19. Wang D, Xiong C, Zhang X. An opposition-based group search optimizer with diversity guidance. *Math Probl Eng.* 2015;2015:546181. doi:10.1155/2015/546181.
20. Qiao Y, Li F, Zhang C, Li X, Zhou Z, Zhang T, et al. Firefly algorithm with opposition-based learning. In: *Bio-inspired computing: theories and application*. Singapore: Springer; 2022. p. 342–52. doi:10.1007/978-981-19-1256-6_26.
21. Zhang Z, Xu Z, Luan S, Li X, Sun Y. Opposition-based ant colony optimization algorithm for the traveling salesman problem. *Mathematics.* 2020;8(10):1650. doi:10.3390/math8101650.
22. Kohli M, Arora S. Chaotic grey wolf optimization algorithm for constrained optimization problems. *J Comput Des Eng.* 2018;5(4):458–72. doi:10.1016/j.jcde.2017.02.005.
23. Geng Y, Li Y, Deng C. An improved binary walrus optimizer with golden sine disturbance and population regeneration mechanism to solve feature selection problems. *Biomimetics.* 2024;9(8):501. doi:10.3390/biomimetics9080501.
24. Si T, Miranda PBC, Nandi U, Jana ND, Mallik S, Maulik U, et al. Opposition-based chaotic tunicate swarm algorithms for global optimization. *IEEE Access.* 2024;12:18168–88. doi:10.1109/access.2024.3359587.
25. Zhao Y, Huang C, Zhang M, Lv C. COLMA: a chaos-based mayfly algorithm with opposition-based learning and Levy flight for numerical optimization and engineering design. *J Supercomput.* 2023;79(17):19699–745. doi:10.1007/s11227-023-05400-2.
26. Pham VHS, Nguyen Dang NT, Nguyen VN. Enhancing engineering optimization using hybrid sine cosine algorithm with Roulette wheel selection and opposition-based learning. *Sci Rep.* 2024;14(1):694. doi:10.1038/s41598-024-51343-w.
27. Akyol S. New chaos-integrated improved grey wolf optimization based models for automatic detection of depression in online social media and networks. *PeerJ Comput Sci.* 2023;9(1):e1661. doi:10.7717/peerj-cs.1661.
28. Gharehchopogh FS, Nadimi-Shahraki MH, Barshandeh S, Abdollahzadeh B, Zamani H. CQFFA: a chaotic quasi-oppositional farmland fertility algorithm for solving engineering optimization problems. *J Bionic Eng.* 2023;20(1):158–83. doi:10.1007/s42235-022-00255-4.
29. Misaghi M, Yaghoobi M. Improved invasive weed optimization algorithm (IWO) based on chaos theory for optimal design of PID controller. *J Comput Des Eng.* 2019;6(3):284–95. doi:10.1016/j.jcde.2019.01.001.
30. Li S, Chen H, Wang M, Heidari AA, Mirjalili S. Slime mould algorithm: a new method for stochastic optimization. *Future Gener Comput Syst.* 2020;111:300–23. doi:10.1016/j.future.2020.03.055.
31. Talatahari S, Azizi M. Chaos game optimization: a novel metaheuristic algorithm. *Artif Intell Rev.* 2021;54(2):917–1004. doi:10.1007/s10462-020-09867-w.
32. Qais MH, Hasanien HM, Alghuwainem S. Transient search optimization: a new meta-heuristic optimization algorithm. *Appl Intell.* 2020;50(11):3926–41. doi:10.1007/s10489-020-01727-y.
33. Mirjalili S, Lewis A. The whale optimization algorithm. *Adv Eng Softw.* 2016;95(12):51–67. doi:10.1016/j.advengsoft.2016.01.008.
34. Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris Hawks optimization: algorithm and applications. *Future Gener Comput Syst.* 2019;97:849–72. doi:10.1016/j.future.2019.02.028.
35. Marini F, Walczak B. Particle swarm optimization (PSO). A tutorial. *Chemom Intell Lab Syst.* 2015;149:153–65. doi:10.1016/j.chemolab.2015.08.020.
36. Saremi S, Mirjalili S, Lewis A. Grasshopper optimisation algorithm: theory and application. *Adv Eng Softw.* 2017;105:30–47. doi:10.1016/j.advengsoft.2017.01.004.
37. Mirjalili S. SCA: a Sine Cosine Algorithm for solving optimization problems. *Knowl Based Syst.* 2016;96(63):120–33. doi:10.1016/j.knosys.2015.12.022.

38. Lemonge AC, Barbosa HJC, Borges CCH, Silva FBS. Constrained optimization problems in mechanical engineering design using a real-coded steady-state genetic algorithm. *Mecánica Comput.* 2010;29(95):9287–303. doi:10.1109/cec.2007.4424532.
39. Rao RV, Savsani VJ, Vakharia DP. Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems. *Comput Aided Des.* 2011;43(3):303–15. doi:10.1016/j.cad.2010.12.015.
40. Kamil AT, Saleh HM, Abd-Alla IH. A multi-swarm structure for particle swarm optimization: solving the welded beam design problem. *J Phys Conf Ser.* 2021;1804(1):012012. doi:10.1088/1742-6596/1804/1/012012.
41. Fleury C, Braibant V. Structural optimization: a new dual method using mixed variables. *Numer Meth Eng.* 1986;23(3):409–28. doi:10.1002/nme.1620230307.
42. Gandomi AH, Yang XS, Alavi AH. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Eng Comput.* 2013;29(1):17–35. doi:10.1007/s00366-011-0241-y.
43. Ray T, Saini P. Engineering design optimization using a swarm with an intelligent information sharing among individuals. *Eng Optim.* 2001;33(6):735–48. doi:10.1080/03052150108940941.
44. Yücel M, Bekdaş G, Ningdeli SM. Minimizing the weight of cantilever beam via metaheuristic methods by using different population-iteration combinations. *Wseas Trans Comput.* 2020;19:69–77. doi:10.37394/23205.2020.19.10.