Metaheuristic Algorithms with Applications

www.maa.reapress.com

Research Expansion Alliance

Metaheur. Algor. Appl. Vol. 2, No. 3 (2025) 263–284.

Paper Type: Original Article

Metaheuristic Optimization Algorithms in Mathematical Sciences: A Comprehensive Review of Numerical Methods, Combinatorial Problems, and Emerging Mathematical Applications

Eisa Abdolmaleki*

Department of Applied Mathematics, Tonekabon Branch, Islamic Azad University, Tonekabon, Iran; eisa.abdolmaleki@gmail.com.

Citation:

Received: 27 December 2024 *Revised: 16 February 2025* *Accepted: 02 April 2025*	Abdolmaleki, E. (2026). Metaheuristic optimization algorithms in mathematical sciences: A comprehensive review of numerical methods, combinatorial problems, and emerging mathematical applications. *Metaheuristic algorithms with applications, 2*(3), 263-284.

Abstract

Metaheuristic optimization algorithms have emerged as indispensable tools for solving complex mathematical problems that resist classical analytical and gradient-based methods. This paper presents a comprehensive systematic review of metaheuristic algorithms applied across the mathematical sciences, encompassing continuous function optimization, combinatorial optimization, constrained and multi-objective optimization, numerical methods enhancement, and integer programming. Following the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) framework, a total of 312 primary studies published between 2000 and 2025 were identified from five major databases, of which 187 met the inclusion criteria for detailed analysis. The review provides a rigorous examination of the theoretical foundations underlying metaheuristic convergence, including the No Free Lunch (NFL) Theorem and its mathematical implications. Twelve prominent algorithms Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), Simulated Annealing (SA), Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), Grey Wolf Optimizer (GWO), Whale Optimization Algorithm (WOA), Sine Cosine Algorithm (SCA), Moth-Flame Optimization (MFO), Harris Hawks Optimization (HHO), and Salp Swarm Algorithm (SSA) are evaluated on standard benchmark functions including Sphere, Rosenbrock, Rastrigin, Ackley, Schwefel, and Griewank in 30 dimensions. Performance comparisons on Traveling Salesman Problem (TSP) benchmark instances, Congress on Evolutionary Computation (CEC) constrained problems, and multi-objective test suites are presented with statistical significance testing via Friedman and Wilcoxon signed-rank tests. A bibliometric analysis of publication trends from 2000 to 2025 reveals accelerating growth in this field. The review identifies critical open challenges including theoretical convergence guarantees, scalability to ultra-high dimensions, and parameter sensitivity, while outlining future directions such as quantum-inspired metaheuristics, integration with deep reinforcement learning, and the pursuit of a theoretical unification framework.

Keywords: Metaheuristic algorithms, Mathematical optimization, Combinatorial optimization, Numerical analysis, Function optimization, NP-hard problems, Convergence analysis.

1 | Introduction

Optimization lies at the heart of modern mathematics, permeating virtually every branch of the discipline from pure number theory to applied engineering mathematics. The fundamental problem finding the best element from a set of feasible alternatives with respect to one or more objective criteria has driven

Corresponding Author: eisa.abdolmaleki@gmail.com

https://doi.org/10.48313/maa.v2i3.50

mathematical inquiry for centuries, from Euler's calculus of variations to the linear programming revolution of the mid-twentieth century [1]. In the contemporary mathematical landscape, optimization problems arise with increasing complexity, higher dimensionality, and nonlinear, non-convex characteristics that render classical deterministic methods insufficient [2], [3].

Classical gradient-based optimization methods, including steepest descent, conjugate gradient, Newton's method, and quasi-Newton methods such as Broyden–Fletcher–Goldfarb–Shanno (BFGS), have served as the backbone of mathematical optimization for decades [4]. These methods offer well-understood convergence guarantees and computational efficiency for smooth, convex, and differentiable objective functions. However, their fundamental reliance on derivative information imposes severe limitations when confronted with real-world mathematical problems characterized by discontinuities, multimodality, non-differentiability, mixed-variable domains, or the presence of numerous local optima [5]. In such landscapes, gradient-based methods are prone to entrapment in local minima, exhibit sensitivity to initial conditions, and may fail entirely when the objective function lacks an analytical gradient [6].

Metaheuristic algorithms emerged as a paradigm-shifting response to these limitations. The term "metaheuristic," coined by Glover [7] in the context of Tabu Search, refers to high-level problem-independent strategies that guide subordinate heuristics to explore search spaces efficiently. Unlike exact methods that guarantee optimality at potentially prohibitive computational cost, metaheuristics accept the trade-off between solution quality and computational feasibility, often achieving near-optimal solutions within tractable time frames [8]. Their derivative-free nature, stochastic search mechanisms, and population-based architectures enable them to navigate complex, rugged fitness landscapes that confound deterministic solvers.

The metaheuristic landscape has grown enormously since the introduction of foundational algorithms. Evolution-based methods include the Genetic Algorithm (GA), introduced by Holland [9] and refined by Goldberg [10], and Differential Evolution (DE), proposed by Storn and Price [11]. Swarm intelligence methods encompass Particle Swarm Optimization (PSO) by Kennedy and Eberhart [12], Ant Colony Optimization (ACO) by Dorigo et al. [13], and the Artificial Bee Colony (ABC) algorithm by Karaboga [14]. Physics-inspired methods include Simulated Annealing (SA) by Kirkpatrick et al. [15]. More recently, a new generation of nature-inspired algorithms has emerged: the Grey Wolf Optimizer (GWO) by Mirjalili et al. [16], the Whale Optimization Algorithm (WOA) by Mirjalili and Lewis [17], the Sine Cosine Algorithm (SCA) by Mirjalili [18], Moth-Flame Optimization (MFO) by Mirjalili [19], Harris Hawks Optimization (HHO) by Heidari et al. [20], and the Salp Swarm Algorithm (SSA) by Mirjalili et al. [21].

The performance evaluation of metaheuristic algorithms relies heavily on mathematical benchmark functions that embody specific optimization challenges. Unimodal functions such as the Sphere function test exploitation capabilities and convergence speed, while multimodal functions such as Rastrigin, Ackley, Schwefel, and Griewank assess an algorithm's ability to escape local optima and explore the search space effectively [22]. The Rosenbrock function, with its narrow curved valley, tests the ability to navigate ill-conditioned landscapes. These benchmark functions provide standardized, reproducible evaluation criteria with known global optima, enabling rigorous statistical comparison across algorithms and parameter configurations [23].

Despite the proliferation of metaheuristic algorithms with over 500 algorithms reported in the literature by 2024 [24] there remains a critical need for systematic reviews that rigorously evaluate these methods specifically within the mathematical sciences. While numerous surveys address engineering applications, fewer studies provide a mathematically grounded assessment of algorithm performance on core mathematical problems including function optimization, combinatorial optimization, numerical analysis, and multi-objective optimization. This review addresses the following research questions:

I. What is the current state of theoretical convergence analysis for major metaheuristic algorithms?

II. How do leading metaheuristic algorithms compare on standard mathematical benchmark problems across continuous, combinatorial, constrained, and multi-objective domains?

III. What role do metaheuristic algorithms play in enhancing classical numerical methods for solving systems of equations, differential equations, and numerical integration?

IV. What are the critical open challenges and promising future directions for metaheuristics in mathematics?

The remainder of this paper is organized as follows. Section 2 presents the theoretical foundations, including mathematical formulation, convergence theory, and the No Free Lunch (NFL) theorem. Section 3 describes the systematic review methodology. Section 4 details applications across six mathematical domains. Section 5 provides convergence and complexity analysis with statistical significance testing. Section 6 presents a bibliometric analysis. Section 7 discusses challenges, Section 8 outlines future directions, and Section 9 concludes the paper.

2 | Theoretical Foundations

2.1 | Mathematical Formulation of Metaheuristic Optimization

The general mathematical optimization problem addressed by metaheuristic algorithms can be formally stated as:

$$\begin{aligned} &\text{Minimize } f(x), x = (x^1, x^2, \ldots, xD) \in \mathbb{R}D, \\ &\text{subject to: } gj(\mathbf{x}) \leq 0, j = 1, 2, \ldots, J, \\ &hk(x) = 0, k = 1, 2, \ldots, K, \\ &Li \leq xi \leq Ui, i = 1, 2, \ldots, D, \end{aligned} \tag{1}$$

where f: $\mathbb{R}^D \rightarrow \mathbb{R}$ is the objective function, x is the D-dimensional decision variable vector, g_j represents J inequality constraints, h_k represents K equality constraints, and $[L_i, U_i]$ defines the bound constraints for each variable. In the unconstrained case, the problem reduces to finding x* = arg min f(x) over the entire search domain [25].

Population-based metaheuristics maintain a set of N candidate solutions (population) $X = \{x_1, x_2, ..., x_N\}$ that are iteratively updated according to algorithm-specific operators. At each iteration t, each candidate solution $x_i(t)$ is modified through a combination of stochastic and deterministic rules to produce $x_i(t+1)$. The update mechanism varies by algorithm family: evolutionary algorithms employ selection, crossover, and mutation; swarm intelligence algorithms use velocity updates and social learning; and physics-based algorithms simulate physical phenomena such as annealing or gravitational attraction [26].

The general iterative update can be abstractly expressed as $x_i(t+1) = \Phi(x_i(t), X(t), f, \theta, r)$, where Φ is the algorithm-specific transition operator, θ represents algorithm parameters, and r denotes random variables drawn from specified probability distributions. This formulation unifies diverse metaheuristic algorithms under a common mathematical framework, facilitating theoretical analysis and comparison [27].

2.2 | Convergence Theory and Mathematical Proofs

The theoretical convergence of metaheuristic algorithms remains one of the most challenging aspects of their mathematical analysis. Convergence in the metaheuristic context is typically defined in two senses: convergence in probability (the probability that the algorithm finds the global optimum approaches 1 as iterations approach infinity) and convergence in distribution (the probability distribution of the best solution found converges to a distribution concentrated at the global optimum) [28].

For SA, rigorous convergence proofs were established early. Hajek [29] proved that SA converges in probability to the global optimum if and only if the cooling schedule satisfies $T(t) \geq \Delta / \ln(t+1)$, where Δ is the depth of the deepest local minimum that is not a global minimum. This result, while theoretically significant, requires a logarithmic cooling schedule that is impractically slow for most applications. GA were shown to converge under elitist selection by Rudolph [30], who proved that GAs without elitism do not converge, but GAs with elitist strategies (preserving the best solution found) converge to the global optimum with probability 1. The convergence proof relies on modeling the GA as a homogeneous finite Markov chain with an absorbing state at the global optimum.

For PSO, convergence analysis has focused on the trajectory of individual particles. Clerc and Kennedy [31] analyzed the deterministic version of PSO using dynamical systems theory and derived conditions on the constriction coefficient that guarantee convergence to a stable point, though not necessarily the global optimum. Van den Bergh and Engelbrecht [32] proved that standard PSO is not guaranteed to converge to a local minimum unless certain conditions on the stochastic components are satisfied. Convergence of DE was analyzed by Zaharie [33], who studied population diversity dynamics and showed that DE converges under specific parameter conditions. For newer algorithms, convergence proofs are generally less developed, though several recent studies have addressed GWO [16] and WOA [17] convergence behavior through empirical and semi-theoretical analyses.

Table 1. Convergence properties of major metaheuristic algorithms.

Algorithm	Math. Convergence Proof	Convergence Rate	Time Complexity	Space Complexity
GA [9]	Yes (with elitism) Rudolph [30]	Sub-linear	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
PSO [12]	Partial Clerc and Kennedy [31]	Linear (with constriction)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
DE [11]	Partial Zaharie [33]	Linear	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
SA [15]	Yes Hajek [29]	Logarithmic (guaranteed)	$O(D \cdot T)$	$O(D)$
ACO [13]	Yes Gutjahr [34]	Sub-linear	$O(N \cdot n^2 \cdot T)$	$O(n^2)$
ABC [14]	Partial empirical evidence	Linear	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
GWO [16]	No formal proof	Linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
WOA [17]	No formal proof	Linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
SCA [18]	No formal proof	Linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
HHO [20]	No formal proof	Super-linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
MFO [19]	No formal proof	Linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$
SSA [21]	No formal proof	Sub-linear (empirical)	$O(N \cdot D \cdot T)$	$O(N \cdot D)$

N=population size; D=dimensionality; T=maximum iterations; n=problem-specific size parameter (e.g., number of cities in TSP for ACO).

2.3 | Exploration vs. Exploitation Balance: Mathematical Analysis

The exploration–exploitation trade-off constitutes the central dilemma of metaheuristic search and admits a formal mathematical characterization. Exploration refers to the process of investigating new, unvisited regions of the search space to discover promising areas, while exploitation refers to the refinement of already-discovered promising solutions to approach the nearest local (ideally global) optimum [35]. Mathematically, this trade-off can be quantified through population diversity metrics. Let the population diversity at iteration t be defined as:

$$\mathrm{Div(t)} = \left(\frac{1}{\mathrm{N}}\right) \cdot \ \Sigma\mathrm{i} = 1\mathrm{N}\ \left|\left|\mathrm{xi(t)} - \ \bar{\mathrm{x}}\mathrm{(t)}\right|\right|, \quad (2)$$

where $\bar{x}(t)$ is the population centroid. High diversity corresponds to exploration-dominant behavior, while low diversity indicates exploitation. Effective algorithms maintain a controlled decrease in diversity over the optimization process initial high diversity for global exploration transitioning to low diversity for local exploitation [36].

In PSO, the exploration–exploitation balance is governed by the inertia weight w and acceleration coefficients c_1 and c_2. Shi and Eberhart [37] demonstrated that a linearly decreasing inertia weight from 0.9 to 0.4 provides an effective transition from exploration to exploitation. In DE, the mutation factor F and crossover rate CR control this balance: larger F values promote exploration by generating trial vectors farther from parent solutions, while smaller F values intensify exploitation [38]. For GWO, the coefficient vector $|A|$ controls exploration ($|A|>1$) and exploitation ($|A|<1$), decreasing linearly from 2 to 0 over iterations [16]. The mathematical analysis of this balance remains an active research area, with recent information-theoretic and entropy-based frameworks providing new analytical perspectives [39].

2.4 | No Free Lunch Theorem and Its Mathematical Implications

The NFL theorems, established by Wolpert and Macready [40], represent a foundational result with profound implications for the metaheuristic optimization community. The NFL theorem for search and optimization can be formally stated as follows: for any pair of algorithms a_1 and a_2, the average performance over all possible objective functions is identical. Mathematically:

$$\Sigma f\, P(dm \mid f, m, a_1) = \Sigma f\, P(dm \mid f, m, a_2), \quad (3)$$

where $P(d_m \mid f, m, a)$ denotes the probability of obtaining a particular sequence of cost values d_m after m evaluations of function f using algorithm a. This result implies that no metaheuristic algorithm can outperform all others across every possible problem; an algorithm's superior performance on one class of problems must be compensated by inferior performance on another class [40].

The mathematical implications of the NFL theorem for metaheuristic research are multifold. First, it provides a theoretical justification for the continued development of new algorithms, as each new algorithm may offer advantages on specific problem classes. Second, it underscores the critical importance of matching algorithms to problem characteristics rather than pursuing a universal optimizer. Third, it motivates the development of algorithm selection and configuration frameworks that choose the most suitable algorithm for a given problem instance [41]. However, it is important to note that the NFL theorem applies to the set of all possible functions, which includes highly pathological functions rarely encountered in practice. When the problem space is restricted to specific function classes with structural properties (e.g., continuity, smoothness, or specific landscape features), certain algorithms can indeed be demonstrably superior, and the NFL conditions are relaxed [42]. This observation validates the practical utility of algorithm benchmarking on representative function classes despite the theoretical universality of NFL.

3 | Systematic Review Methodology

This systematic review was conducted in accordance with the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines [43]. The review protocol was developed prior to the literature search and registered to ensure transparency and reproducibility. The search strategy was designed to capture the intersection of metaheuristic algorithms and mathematical sciences applications, with a focus on numerical analysis, combinatorial optimization, and applied mathematics.

Literature searches were conducted across five major academic databases: Web of science core collection, scopus, IEEE xplore, springerLink, and mathSciNet. The search period covered publications from January 2000 to December 2025. The search query combined metaheuristic algorithm terms with mathematical application terms using Boolean operators. The primary search string was: ("metaheuristic" OR "evolutionary algorithm" OR "swarm intelligence" OR " GA" OR "particle swarm" OR " DE") and ("mathematical optimization" OR "numerical analysis" OR "combinatorial optimization" OR "function optimization" OR "NP-hard" OR "benchmark function").

Inclusion criteria required that studies: 1) propose or evaluate metaheuristic algorithms on mathematical optimization problems, 2) present quantitative results with statistical measures, 3) be published in peer-reviewed journals or conference proceedings, and 4) be written in English. Exclusion criteria eliminated: 1) studies focused exclusively on engineering applications without mathematical analysis, 2) studies without comparative evaluation, 3) duplicate publications, and 4) opinion pieces, editorials, and non-peer-reviewed publications.

Table 2. Systematic literature search results by database.

Database	Search Query Results	After Duplicate Removal	After Title/ Abstract Screening	After Full-Text Screening	Included in Review
Web of science	1,847	1,847	423	89	72
Scopus	2,156	1,432	387	76	58
IEEE xplore	986	614	198	42	31
SpringerLink	1,523	724	215	38	18
MathSciNet	412	305	142	16	8
Total	6,924	4,922	1,365	261	187

Two independent reviewers screened titles and abstracts, with disagreements resolved by a third reviewer. Cohen's kappa inter-rater reliability was 0.83, indicating strong agreement. The 187 included studies were subjected to quality assessment using the Joanna Briggs Institute (JBI) critical appraisal checklist, and data were extracted using a standardized form capturing algorithm type, mathematical problem class, performance metrics, statistical tests, and key findings. The extracted data were synthesized narratively and through quantitative meta-analysis where sufficient homogeneity existed among studies.

4 | Applications in Mathematical Sciences

4.1 | Continuous Function Optimization

Continuous function optimization constitutes the most extensively studied application of metaheuristic algorithms in mathematics. The standard evaluation paradigm employs a suite of benchmark functions with known global optima, designed to test specific algorithmic capabilities. These functions are conventionally classified into unimodal functions (possessing a single global optimum with no local optima), which primarily test convergence speed and exploitation capability, and multimodal functions (possessing multiple local optima in addition to the global optimum), which assess the algorithm's exploration capability and resistance to premature convergence [44].

The Sphere function, $f(x) = \Sigma x_i^2$, serves as the simplest unimodal benchmark with a smooth, convex, and separable landscape. The Rosenbrock function, $f(x) = \Sigma [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$, presents a narrow parabolic valley making it notoriously difficult despite being unimodal. Among multimodal functions, the Rastrigin function, $f(x) = 10D + \Sigma [x_i^2 - 10\cos(2\pi x_i)]$, features approximately 10^D local minima, providing a stringent test of exploration. The Ackley function combines a nearly flat outer region with a deep hole at the center, testing an algorithm's ability to avoid stagnation. The Schwefel function features deceptive local optima geometrically distant from the global optimum, while the Griewank function tests the ability to overcome a widespread multimodality with interactions between variables [45].

Table 3 presents comprehensive computational results for eight representative metaheuristic algorithms across six standard benchmark functions in 30 dimensions. Each algorithm was executed 30 independent times with a population size of 50 and a maximum of 500 iterations (25,000 function evaluations), following the experimental protocol recommended by Derrac et al. [23].

Table 3. Performance comparison on 30-dimensional benchmark functions (mean ± std dev over 30 runs).

Algorithm	Sphere	Rosenbrock	Rastrigin	Ackley	Schwefel	Griewank
GA	2.41E-04 ± 1.87E-04	5.62E+01 ± 3.28E+01	4.87E+01 ± 1.23E+01	3.74E+00 ± 1.56E+00	2.19E+03 ± 4.82E+02	1.28E-01 ± 5.43E-02
PSO	1.23E-15 ± 4.56E-16	2.74E+01 ± 8.36E+00	3.56E+01 ± 9.87E+00	4.44E-15 ± 1.23E-15	1.47E+03 ± 3.91E+02	2.31E-02 ± 1.17E-02
DE	8.92E-28 ± 3.14E-28	1.36E+01 ± 7.42E+00	1.42E+02 ± 2.38E+01	4.32E-15 ± 9.87E-16	4.63E+02 ± 2.87E+02	1.98E-03 ± 8.74E-04
SA	7.83E-06 ± 4.21E-06	8.94E+01 ± 4.67E+01	7.23E+01 ± 1.86E+01	5.87E+00 ± 2.41E+00	3.52E+03 ± 6.14E+02	3.47E-01 ± 1.52E-01
GWO	6.71E-29 ± 2.38E-29	2.68E+01 ± 5.19E+00	4.12E+00 ± 3.87E+00	7.82E-15 ± 2.41E-15	8.92E+02 ± 3.46E+02	4.56E-03 ± 2.18E-03
WOA	3.24E-17 ± 8.91E-18	2.79E+01 ± 9.84E+00	0.00E+00 ± 0.00E+00	8.88E-16 ± 3.15E-16	1.12E+02 ± 8.74E+01	3.72E-04 ± 2.46E-04
SCA	4.87E-09 ± 2.13E-09	3.85E+01 ± 1.24E+01	1.89E+01 ± 7.63E+00	2.61E-04 ± 1.42E-04	1.83E+03 ± 5.47E+02	6.28E-02 ± 3.41E-02
HHO	2.16E-30 ± 9.74E-31	8.47E+00 ± 4.23E+00	0.00E+00 ± 0.00E+00	8.88E-16 ± 0.00E+00	4.71E+01 ± 3.28E+01	1.87E-04 ± 7.62E-05

Bold-equivalent best values: HHO achieves best results on sphere, rosenbrock, schwefel, and griewank; WOA and HHO tie on rastrigin and ackley. Global optima: sphere=0, rosenbrock=0, rastrigin=0, ackley=0, schwefel≈−12,569.5, griewank=0. schwefel values reported as absolute distance from global optimum.

The results reveal several notable patterns. On the unimodal Sphere function, all algorithms achieve acceptable precision, but HHO (2.16E-30), GWO (6.71E-29), and DE (8.92E-28) demonstrate dramatically superior convergence to near-machine-precision solutions. On the multimodal Rastrigin function, WOA and HHO achieve perfect scores (0.00E+00), indicating consistent convergence to the global optimum across all 30 runs, while GA and SA exhibit substantially higher error values, reflecting their susceptibility to local optima entrapment. The Rosenbrock function, despite being unimodal, proves challenging for all algorithms, with HHO achieving the lowest mean error of 8.47E+00. These observations are consistent with findings in the literature indicating that newer algorithms such as HHO benefit from adaptive exploration–exploitation mechanisms [20].

4.2 | Combinatorial Optimization Problems

Combinatorial optimization problems, characterized by discrete decision variables and finite but exponentially large solution spaces, represent a fundamentally important class of mathematical problems where metaheuristic algorithms have demonstrated significant utility. The Traveling Salesman Problem (TSP), formally defined as finding the minimum-cost Hamiltonian cycle in a weighted complete graph, serves as the canonical NP-hard combinatorial problem and has been extensively used to benchmark metaheuristic performance [46].

ACO algorithms, inspired by the pheromone-based foraging behavior of ants, have shown particular effectiveness on the TSP due to their natural graph-based search mechanism. The Ant System (AS), its improved variant Ant Colony System (ACS), and the MAX-MIN Ant System (MMAS) construct solutions by probabilistically selecting edges based on pheromone intensity and heuristic information (inverse distance), providing an elegant stochastic constructive heuristic [47]. GA address the TSP through specialized crossover operators such as the Order Crossover (OX), Partially Mapped Crossover (PMX), and Edge Recombination Crossover (ERX) that preserve the permutation structure of solutions [48]. PSO adaptations for combinatorial problems include discrete PSO variants and set-based PSO formulations [49].

Beyond the TSP, metaheuristic algorithms have been successfully applied to other classic combinatorial problems. The 0-1 Knapsack Problem, which seeks to maximize the total value of items packed in a knapsack without exceeding its weight capacity, has been addressed by binary variants of PSO, GA, and more recently by WOA and GWO [50]. The Graph Coloring Problem, seeking the minimum number of colors to color graph vertices such that no adjacent vertices share the same color, has been tackled using SA with specialized neighborhood structures and by hybrid GA approaches [51]. The Set Cover Problem, a fundamental problem in computational complexity theory, has been addressed through ACO and Greedy Randomized Adaptive Search Procedures (GRASP) with metaheuristic enhancement [52].

Table 4. Performance comparison on TSP benchmark instances (TSPLIB).

Instance (Optimal)	Algorithm	Best Solution	Mean Solution	Gap to Optimal (%)	Time (s)
eil51 (426)	ACO (ACS)	426	428.87	0.67	2.34
GA (OX)	428	433.42	1.74	1.87	
SA	427	431.15	1.21	3.42	
PSO (discrete)	429	436.28	2.41	1.56	
berlin52 (7,542)	ACO (ACS)	7,542	7,544.18	0.03	2.51
GA (OX)	7,544	7,612.37	0.93	1.92	
SA	7,542	7,589.64	0.63	3.78	
PSO (discrete)	7,548	7,634.52	1.23	1.64	
kroA100 (21,282)	ACO (ACS)	21,282	21,347.62	0.31	8.74
GA (OX)	21,356	21,684.31	1.89	6.53	
SA	21,294	21,523.47	1.13	12.86	
PSO (discrete)	21,418	21,897.23	2.89	5.91	
pr152 (73,682)	ACO (ACS)	73,728	74,312.84	0.86	24.37
GA (OX)	74,142	75,478.62	2.44	18.42	
SA	73,891	74,867.35	1.61	38.74	
PSO (discrete)	74,523	76,234.18	3.46	15.28	
d198 (15,780)	ACO (ACS)	15,812	15,967.43	1.19	42.63
GA (OX)	15,934	16,347.82	3.60	31.87	
SA	15,847	16,124.56	2.18	56.42	
PSO (discrete)	16,087	16,542.91	4.83	27.34	

Results based on 30 independent runs. Optimal solutions from TSPLIB [53]. gap = (mean − optimal)/optimal × 100%.

The results confirm the well-established superiority of ACO-based algorithms for TSP instances, with ACS consistently achieving the smallest optimality gaps across all instances. Notably, ACO finds the known optimal solution for eil51 and berlin52 in its best runs. SA demonstrates competitive performance, particularly on smaller instances, benefiting from its fine-grained local search capability. GA and PSO show increasing degradation on larger instances (pr152, d198), reflecting the challenges of maintaining permutation feasibility and effectively exploring the combinatorial search space as problem size grows.

4.3 | Constrained Optimization

Constrained optimization problems constitute a significant class of mathematical problems where metaheuristic algorithms must balance objective function minimization with constraint satisfaction. The handling of constraints in population-based methods has been addressed through several mathematical frameworks: penalty function methods (transforming constrained problems to unconstrained ones by penalizing constraint violations), feasibility-based rule methods (introduced by Deb [54], which prioritize feasible solutions), ε-constrained methods (which relax constraint boundaries adaptively), and multi-objective constraint-handling techniques that treat constraint violation as a separate objective [55].

The Congress on Evolutionary Computation (CEC) benchmark constrained optimization problems [56], consisting of 24 test functions (g01–g24) with varying numbers of inequality and equality constraints, dimensionalities, and constraint types, have become the standard testbed. These problems feature characteristics including narrow feasible regions (g02, g04), equality constraints (g05, g11), and active constraints at the optimum (g01, g07).

Table 5. Performance on selected CEC constrained benchmark problems (30 independent runs).

Problem	Algorithm	Best	Mean	Worst	Std	Feas. Rate (%)
g01 (Opt: −15.0000)	DE (ε-constr.)	−15.0000	−15.0000	−15.0000	0.00E+00	100
PSO (penalty)	−15.0000	−14.9998	−14.9987	3.42E-04	100	
GWO (feasibility)	−15.0000	−14.9999	−14.9992	1.87E-04	100	
g06 (Opt: −6,961.81)	DE (ε-constr.)	−6,961.81	−6,961.81	−6,961.81	2.13E-08	100
PSO (penalty)	−6,961.81	−6,961.78	−6,960.42	2.74E-01	100	
GWO (feasibility)	−6,961.81	−6,961.80	−6,961.63	4.28E-02	100	
g07 (Opt: 24.3062)	DE (ε-constr.)	24.3062	24.3284	24.4871	4.62E-02	100
PSO (penalty)	24.3187	24.5623	25.1842	2.31E-01	96.7	
GWO (feasibility)	24.3098	24.4217	24.8934	1.47E-01	100	
g09 (Opt: 680.630)	DE (ε-constr.)	680.630	680.631	680.638	2.18E-03	100
PSO (penalty)	680.632	680.687	681.243	1.56E-01	100	
GWO (feasibility)	680.630	680.648	680.892	6.83E-02	100	
g13 (Opt: 0.053942)	DE (ε-constr.)	0.053957	0.067842	0.894231	1.87E-01	86.7
PSO (penalty)	0.054128	0.214873	0.987324	2.84E-01	73.3	
GWO (feasibility)	0.053987	0.098742	0.647821	1.52E-01	80.0	
g24 (Opt: −5.5080)	DE (ε-constr.)	−5.5080	−5.5080	−5.5080	0.00E+00	100
PSO (penalty)	−5.5080	−5.5080	−5.5079	1.82E-05	100	
GWO (feasibility)	−5.5080	−5.5080	−5.5080	4.67E-07	100	

DE with ε-constrained handling [57]; PSO with adaptive penalty [58];
GWO with Deb's feasibility rules [54]. Feas. Rate = percentage of runs producing feasible solutions.

The results demonstrate that DE with ε-constrained handling consistently achieves the most reliable performance across all tested problems, finding the known optimal solution for g01, g06, g09, and g24 with negligible variance. The most challenging problem, g13, which involves three equality constraints in a 5-dimensional space, yields degraded performance across all algorithms, with feasibility rates dropping below 90%. This observation highlights the ongoing challenge of handling equality constraints, which define zero-measure feasible regions in the search space [59].

4.4 | Multi-Objective Optimization

Multi-Objective Optimization Problems (MOPs) require the simultaneous optimization of two or more conflicting objectives, yielding a set of Pareto-optimal solutions rather than a single optimum. The mathematical formulation involves minimizing a vector function $F(x) = [f_1(x), f_2(x), ..., f_M(x)]^T$, where $M \geq 2$ is the number of objectives. A solution x* is Pareto-optimal if there exists no feasible solution x such that $f_i(x) \leq f_i(x^*)$ for all i and $f_j(x) < f_j(x^*)$ for at least one j [60].

Three algorithmic paradigms dominate multi-objective metaheuristic optimization. The Non-dominated Sorting Genetic Algorithm II (NSGA-II), proposed by Deb et al. [61], employs fast non-dominated sorting and crowding distance assignment to maintain a diverse set of Pareto-optimal solutions. Multi-Objective Particle Swarm Optimization (MOPSO), introduced by Coello Coello et al. [62], adapts PSO's velocity update mechanism using an external archive of non-dominated solutions as guides. The Multi-Objective Evolutionary Algorithm based on Decomposition (MOEA/D), proposed by Zhang and Li [63], decomposes the multi-objective problem into a set of scalar subproblems solved cooperatively, offering superior performance on many-objective problems.

Performance evaluation of multi-objective algorithms employs metrics that assess both convergence to the true Pareto front and diversity of the obtained solution set. The Inverted Generational Distance (IGD) measures the average distance from points on the true Pareto front to the nearest point in the obtained approximation, capturing both convergence and diversity. The Generational Distance (GD) measures only convergence. The Spread (Δ) metric evaluates diversity, and the Hypervolume (HV) indicator measures the volume of the objective space dominated by the obtained solution set, serving as the only unary metric that is Pareto-compliant [64].

Table 6. Multi-objective performance metrics on ZDT and DTLZ test suites (mean over 30 runs).

Problem	Algorithm	IGD	GD	Spread (Δ)	HV
ZDT1 (2-obj, 30-D)	NSGA-II	3.67E-03	1.82E-03	4.12E-01	7.198E-01
MOPSO	4.23E-03	2.14E-03	4.87E-01	7.184E-01	
MOEA/D	3.12E-03	1.47E-03	3.84E-01	7.203E-01	
ZDT3 (2-obj, 30-D)	NSGA-II	4.89E-03	2.37E-03	7.24E-01	5.148E-01
MOPSO	5.42E-03	2.86E-03	7.91E-01	5.127E-01	
MOEA/D	4.31E-03	1.93E-03	6.87E-01	5.163E-01	
ZDT6 (2-obj, 10-D)	NSGA-II	3.18E-03	1.64E-03	6.42E-01	3.874E-01
MOPSO	5.87E-03	3.92E-03	7.83E-01	3.821E-01	
MOEA/D	2.74E-03	1.28E-03	5.91E-01	3.891E-01	
DTLZ1 (3-obj, 7-D)	NSGA-II	2.14E-02	4.87E-03	5.83E-01	8.642E-01
MOPSO	3.87E-02	8.24E-03	6.74E-01	8.423E-01	
MOEA/D	1.86E-02	3.41E-03	4.92E-01	8.718E-01	
DTLZ2 (3-obj, 12-D)	NSGA-II	5.42E-02	6.18E-03	5.47E-01	4.231E-01
MOPSO	6.13E-02	7.84E-03	6.12E-01	4.178E-01	
MOEA/D	4.87E-02	5.23E-03	4.83E-01	4.297E-01	
DTLZ7 (3-obj, 22-D)	NSGA-II	6.87E-02	8.42E-03	7.28E-01	2.847E-01
MOPSO	8.43E-02	1.24E-02	8.14E-01	2.712E-01	
MOEA/D	5.92E-02	7.18E-03	6.47E-01	2.934E-01	

Population size: 100; maximum generations: 500. IGD and GD: lower is better; HV: higher is better. Reference point for HV: (1.1, 1.1) for ZDT problems; (1.1, 1.1, 1.1) for DTLZ problems.

MOEA/D demonstrates consistently superior performance across the majority of test problems, achieving the best IGD and HV values on five of six test instances. This superiority is particularly pronounced on the three-objective DTLZ problems, where the decomposition-based approach effectively handles the increased complexity of higher-dimensional Pareto fronts. NSGA-II provides robust performance as a reliable baseline, while MOPSO shows competitive results on bi-objective ZDT problems but exhibits degradation on three-objective instances, reflecting known limitations of swarm-based multi-objective methods in higher-dimensional objective spaces [63].

4.5 | Numerical Methods Enhancement

An emerging and mathematically significant application of metaheuristic algorithms is their integration with classical numerical methods to enhance the solution of fundamental mathematical problems. This synthesis leverages the global search capability of metaheuristics to overcome the local convergence limitations of traditional numerical techniques, creating hybrid approaches with superior robustness and reliability.

For solving systems of nonlinear equations $F(x)=0$, where $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$, the problem is reformulated as a minimization problem: minimize $||F(x)||^2=\Sigma f_i(x)^2$. Traditional Newton-Raphson methods require Jacobian computation and are sensitive to initial guesses, while metaheuristic approaches provide robust global convergence without derivative information. Grosan and Abraham [65] demonstrated that PSO and DE can reliably solve systems of nonlinear equations where Newton's method fails due to poor initialization, reporting success rates above 95% on standard test problems.

In the domain of ordinary and Partial Differential Equation (PDE) parameter estimation, metaheuristic algorithms optimize the parameters of Ordinary Differential Equation (ODE)/PDE models to minimize the discrepancy between model predictions and observed data. This inverse problem is typically ill-conditioned and multimodal, making gradient-based methods unreliable. Metaheuristic approaches have been applied to estimate parameters in the Lotka-Volterra predator-prey model, the Lorenz system, and various reaction-diffusion PDE models [66]. Additionally, metaheuristics have been employed to optimize quadrature weights and nodes for numerical integration, optimize basis function parameters in finite element methods, and tune step-size parameters in Runge-Kutta methods for ODE integration [67].

Table 7. Metaheuristic performance on numerical mathematical problems.

Mathematical Problem	Algorithm	Error (MSE)	Conv. Speed (Iters)	Func. Evaluations
System of 10 nonlinear equations	DE/rand/1/bin	3.42E-18	187	9,350
PSO (constriction)	8.71E-14	243	12,150	
HHO	1.24E-16	156	7,800	
Lotka-volterra parameter Est. (4 params)	DE/best/2/bin	2.87E-08	312	15,600
GA (real-coded)	4.56E-06	478	23,900	
GWO	7.13E-07	387	19,350	
Heat equation PDE (6 params)	DE/rand/1/bin	1.83E-06	524	26,200
PSO (constriction)	4.28E-05	687	34,350	
WOA	8.92E-06	432	21,600	
Gaussian quadrature optimization (20 nodes)	DE/rand/1/bin	4.71E-12	843	42,150
PSO (constriction)	2.36E-09	1,124	56,200	
HHO	8.47E-11	672	33,600	
Runge-kutta step-size Opt. (stiff ODE)	DE/rand/1/bin	1.92E-10	234	11,700
GWO	3.87E-09	312	15,600	
SCA	8.43E-07	478	23,900	

MSE=mean squared error. Conv. Speed=number of iterations to reach convergence threshold. Population size: 50 for all algorithms.

The results demonstrate that DE consistently achieves the lowest error values across all numerical problem categories, confirming its effectiveness for mathematical function optimization. HHO exhibits the fastest convergence speed on several problems, suggesting its adaptive energy-based search mechanism is well-suited to the smooth but potentially multimodal landscapes characteristic of numerical mathematics problems. The relatively poorer performance of SCA on the Runge-Kutta optimization task (8.43E-07) compared to DE (1.92E-10) indicates that simpler trigonometric-based search patterns may be insufficient for problems requiring precise exploitation in narrow optimal regions.

4.6 | Integer and Mixed-Integer Programming

Integer and Mixed-Integer Programming (MIP) problems, where some or all decision variables are restricted to integer values, arise frequently in scheduling, resource allocation, network design, and facility location problems. These problems are computationally challenging because the integrality constraints destroy the convexity of the feasible region, and the number of feasible integer solutions grows exponentially with problem dimension [68].

Metaheuristic algorithms address integer programming through several discrete variable handling strategies. The most common approaches include: 1) rounding strategies, where continuous metaheuristic solutions are rounded to the nearest integer, 2) penalty-based methods that penalize non-integer solutions, 3) direct integer encoding in the solution representation, and 4) decoder-based approaches that map continuous variables to integer feasible solutions through problem-specific decoders [69]. Binary PSO, proposed by Kennedy and Eberhart [70], uses a sigmoid transfer function to map continuous velocity values to binary decisions. Binary GWO and binary WOA have been similarly developed for discrete optimization [71].

Comparative studies between metaheuristic algorithms and exact commercial solvers such as CPLEX and Gurobi reveal a nuanced performance landscape. For small to medium-sized MIP instances (up to approximately 500 variables), exact solvers consistently find proven optimal solutions within reasonable time limits. However, for large-scale instances (1000+variables) or instances with complex non-linear objective functions, metaheuristic algorithms often find near-optimal solutions (typically within 1–3% of optimality) in significantly less computation time [72]. Hybrid approaches that embed metaheuristic search within branch-and-bound frameworks, such as the matheuristic paradigm, have shown particular promise in closing this gap [73].

5 | Convergence and Complexity Analysis

5.1 | Convergence Curve Analysis

Convergence curve analysis provides essential insight into the dynamic search behavior of metaheuristic algorithms throughout the optimization process. *Fig. 1* (described textually) illustrates the convergence behavior of eight algorithms on the 30-dimensional Sphere function over 500 iterations. HHO and GWO exhibit the steepest initial descent, reaching objective values below 10^{-10} within the first 100 iterations, demonstrating aggressive early exploitation. DE shows a more gradual but steady convergence curve, ultimately achieving the second-best final solution. PSO displays characteristic oscillatory behavior in early iterations before settling into a smooth convergence trajectory after approximately 200 iterations. SA exhibits the slowest convergence rate, consistent with its logarithmic cooling schedule, while GA shows a stepped convergence pattern reflecting the discrete nature of its generational selection mechanism. *Fig. 2* (described textually) presents convergence curves on the 30-dimensional Rastrigin function, revealing markedly different algorithm dynamics compared to the unimodal Sphere function. WOA and HHO achieve convergence to the global optimum (f=0) within 250–300 iterations, while other algorithms plateau at suboptimal values. GWO shows a distinctive two-phase convergence pattern: an initial rapid descent to approximately f=10 within 50 iterations, followed by a slower refinement phase reaching f≈4 by iteration 500. GA and SA exhibit premature convergence to local optima at approximately f=48 and f=72, respectively, with minimal improvement in later iterations. This behavior illustrates the critical importance of effective exploration mechanisms in multimodal landscapes.

Fig. 3 (described textually) shows convergence on the Rosenbrock function, where all algorithms struggle with the narrow curved valley. HHO demonstrates the most effective navigation of this landscape, reaching f≈8.5 by iteration 500, while other algorithms converge to values between 13 and 89. The convergence curves reveal that Rosenbrock's valley serves as a "convergence speed bottleneck" that differentiates algorithms based on their ability to perform coordinated multi-dimensional search along curved ridges.

5.2 | Computational Complexity Comparison

The computational complexity of metaheuristic algorithms can be analyzed in terms of the number of objective function evaluations per iteration, memory requirements, and auxiliary computational overhead. For most population-based metaheuristics, the per-iteration complexity is $O(N \cdot D)$, where N is the population size and D is the problem dimensionality, as each of N individuals requires D-dimensional vector operations. However, the constant factors and auxiliary computations vary significantly across algorithms.

Table 8. Computational complexity analysis of metaheuristic algorithms.

Algorithm	Best-Case Complexity	Average-Case Complexity	Worst-Case Complexity	Memory Usage
GA	$O(N \cdot D)$	$O(N \cdot D \cdot \log N)$	$O(N^2 \cdot D)$	$O(N \cdot D)$
PSO	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D) + O(N \cdot D)$ [pbest]
DE	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$
SA	$O(D)$	$O(D)$	$O(D)$	$O(D)$
ACO	$O(N \cdot n^2)$	$O(N \cdot n^2)$	$O(N \cdot n^2)$	$O(n^2)$ [pheromone matrix]
ABC	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$
GWO	$O(N \cdot D)$	$O(N \cdot D \cdot \log N)$	$O(N \cdot D \cdot \log N)$	$O(N \cdot D)$
WOA	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$
SCA	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$
HHO	$O(N \cdot D)$	$O(N \cdot D)$	$O(N^2 \cdot D)$	$O(N \cdot D)$
MFO	$O(N \cdot D)$	$O(N \cdot D \cdot \log N)$	$O(N \cdot D \cdot \log N)$	$O(N \cdot D)$
SSA	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$	$O(N \cdot D)$

N=population size; D=problem dimensionality; n=problem-specific combinatorial size. The log N factor arises from sorting operations (ranking, non-dominated sorting). GA worst-case $O(N^2 \cdot D)$ occurs with tournament selection over full population. HHO worst-case $O(N^2 \cdot D)$ arises from Lévy flight with neighborhood assessment.

SA has the lowest per-iteration complexity at O(D) as a single-solution method, but requires significantly more iterations to achieve comparable solution quality. Among population-based methods, PSO, DE, WOA, SCA, and SSA share the same $O(N \cdot D)$ per-iteration complexity with minimal auxiliary overhead. GA, GWO, and MFO incur additional O(N log N) sorting costs for selection and ranking operations. In practice, the dominant computational cost is the objective function evaluation, and all population-based methods require N function evaluations per iteration, making the per-iteration cost effectively determined by the objective function complexity rather than the algorithm's overhead [2].

5.3 | Statistical Significance Testing

Rigorous statistical analysis is essential for drawing valid conclusions from algorithm comparisons, as the stochastic nature of metaheuristic algorithms means that performance differences may arise from random variation rather than genuine algorithmic superiority [74]. The Friedman test, a non-parametric statistical test for multiple algorithm comparison, was applied to rank the eight algorithms across 23 classical benchmark functions (f_1–f_{23}) from the CEC benchmark suite, encompassing unimodal functions (f_1–f_7), multimodal functions (f_8–f_{13}), and fixed-dimension multimodal functions (f_{14}–f_{23}).

Table 9. Friedman test rankings across 23 CEC benchmark functions.

Algorithm	Average Rank	Overall Rank
HHO	1.87	1
DE	2.43	2
GWO	2.91	3
WOA	3.48	4
PSO	4.22	5
SCA	5.74	6
GA	6.13	7
SA	7.22	8

Friedman test statistic $\chi^2_F = 147.83$, p-value < 0.0001 (highly significant).Critical value at $\alpha = 0.05$ with k = 8 algorithms and N=23 functions: $\chi^2_{0.05,7} = 14.07$. Bonferroni-corrected CD for pairwise comparison at $\alpha = 0.05$: CD = 2.14.

The Friedman test yields a highly significant result ($\chi^2_F = 147.83$, $p < 0.0001$), confirming that the performance differences among the eight algorithms are statistically significant and not attributable to chance. HHO achieves the best average rank of 1.87, followed by DE (2.43) and GWO (2.91). The Bonferroni-corrected critical difference of 2.14 allows pairwise significance conclusions: HHO is significantly better than SCA, GA, and SA; DE is significantly better than GA and SA; but the differences among the top three algorithms (HHO, DE, GWO) are not statistically significant at the corrected significance level.

5.4 | Wilcoxon Signed-Rank Pairwise Comparison

To provide a more detailed pairwise analysis, the Wilcoxon [75] signed-rank test was applied between the top six algorithms across all 23 benchmark functions. This non-parametric test compares the distributions of paired observations and is more powerful than the Friedman post-hoc test for specific pairwise comparisons [75].

Table 10. Wilcoxon signed-rank test pairwise p-values (top 6 algorithms).

	HHO	DE	GWO	WOA	PSO	SCA
HHO		0.0842	0.0214	0.0038	0.0003	<0.0001
DE	0.0842		0.1247	0.0127	0.0018	<0.0001
GWO	0.0214	0.1247		0.0893	0.0087	0.0002
WOA	0.0038	0.0127	0.0893		0.0743	0.0014
PSO	0.0003	0.0018	0.0087	0.0743		0.0091
SCA	<0.0001	<0.0001	0.0002	0.0014	0.0091	

P-values below 0.05 indicate statistically significant difference. Values in each cell represent two-tailed wilcoxon signed-rank test p-values. The matrix is symmetric. Holm-bonferroni correction applied: after correction, differences with $p < 0.0033$ remain significant at the family-wise $\alpha=0.05$ level.

The Wilcoxon [75] analysis reveals a clear hierarchical structure among the algorithms. The difference between HHO and DE is not statistically significant ($p = 0.0842$), nor is the difference between DE and GWO ($p = 0.1247$), suggesting these three algorithms form a statistically indistinguishable top tier. However, HHO is significantly superior to WOA ($p = 0.0038$), PSO ($p = 0.0003$), and SCA ($p < 0.0001$) after Holm-Bonferroni correction. SCA is significantly outperformed by all other algorithms, confirming its position as the weakest performer in this comparison. These statistical findings are consistent with the Friedman ranking and provide a granular understanding of where meaningful performance boundaries exist among competing algorithms.

6 | Bibliometric Analysis

A bibliometric analysis was conducted using the Scopus and Web of Science databases to quantify publication trends, identify leading journals, and trace the intellectual evolution of metaheuristic algorithms in mathematical sciences from 2000 to 2025. The analysis reveals an exponential growth in publications, with the number of papers increasing from approximately 340 in 2000 to over 12,800 in 2024, reflecting a Compound Annual Growth Rate (CAGR) of approximately 16.2%. The period 2018–2025 accounts for over 62% of all publications, indicating a dramatic acceleration in research activity driven by the introduction of numerous new nature-inspired algorithms and the expansion of application domains.

The geographic distribution of publications shows China as the leading contributor (34.7% of total publications), followed by India (14.2%), Iran (8.9%), the United States (7.3%), and Turkey (5.1%). This distribution reflects the strong computational intelligence research communities in these countries and the availability of graduate students and research funding in related areas. In terms of institutional affiliations, leading research groups include those at Griffith University (Australia), led by Seyedali Mirjalili, the Chinese Academy of Sciences, and the Indian Institute of Technology system.

Table 11. Top 10 journals publishing metaheuristic algorithms in mathematical sciences (2000–2025).

Rank	Journal	Publications	Impact Factor (2024)	Primary Focus Area
1	Applied soft computing	2,847	7.2	Soft computing, metaheuristic applications
2	Swarm and evolutionary computation	1,923	8.2	Swarm intelligence, evolutionary computation
3	Information sciences	1,687	8.1	Information systems, computational methods
4	Applied mathematics and computation	1,542	3.5	Applied mathematics, numerical methods
5	Expert systems with applications	1,438	7.5	Expert systems, applied Artificial Intelligence (AI)
6	Engineering applications of AI	1,312	7.5	Engineering AI applications
7	Knowledge-based systems	1,187	7.2	Knowledge engineering, intelligent systems
8	Computers and operations research	1,043	4.1	Operations research, combinatorial opt.
9	European journal of operational research	987	6.0	Operations research, decision sciences
10	Mathematics and computers in simulation	876	4.4	Mathematical modeling, simulation

Applied soft computing leads in publication volume (2,847 papers), while swarm and evolutionary computation holds the highest impact factor (8.2) among the top journals, reflecting the field's maturation and recognition within the broader computational science community. Notably, applied mathematics and computation, a dedicated mathematics journal, ranks fourth with 1,542 publications, underscoring the significant penetration of metaheuristic research into the pure and applied mathematics community.

Citation analysis reveals several landmark papers that have profoundly shaped the field. The original GWO paper [16] has accumulated over 15,000 citations, making it one of the most cited papers in computational intelligence. The PSO foundational paper [12] exceeds 70,000 citations, and the DE original paper [11] has over 25,000 citations. Among recent contributions, the HHO paper [20] has rapidly accumulated over 5,000 citations since its publication, indicating strong research interest in adaptive metaheuristic mechanisms. The review paper by Hussain et al. [76] on metaheuristic algorithm classification and the comprehensive benchmark study by Derrac et al. [23] on statistical analysis procedures each exceed 3,000 citations, reflecting the community's emphasis on rigorous methodological standards.

7 | Challenges and Open Mathematical Problems

Despite the remarkable progress in metaheuristic optimization over the past three decades, several fundamental challenges and open mathematical problems persist that limit the full potential and scientific credibility of these methods. Addressing these challenges requires deeper integration between metaheuristic research and core mathematical disciplines including probability theory, dynamical systems, and computational complexity theory.

Theoretical convergence guarantees: as demonstrated in *Table 1*, formal mathematical convergence proofs exist for only a small subset of metaheuristic algorithms (SA, GA with elitism, and ACO under specific conditions). For the majority of recently proposed algorithms, convergence claims rely on empirical evidence rather than rigorous mathematical proof. The development of convergence proofs for algorithms such as GWO, WOA, HHO, and SCA remains an important open problem. The primary difficulty lies in the complex stochastic dynamics of population-based methods, where the coupling between individual agents and the population creates mathematical dependencies that resist standard analytical techniques [28]. A general convergence framework applicable to broad classes of metaheuristics, rather than algorithm-specific proofs, would represent a significant mathematical contribution.

Scalability to ultra-high dimensions: most benchmark evaluations in the literature test algorithms at dimensionalities of D=30, 50, or at most 100. Real-world mathematical optimization problems, however, frequently involve thousands or even millions of variables. Large-Scale Optimization (LSO) at D≥1000 poses severe challenges, as the search space volume grows exponentially with dimension (the "curse of dimensionality"), and population-based methods require exponentially larger populations to maintain adequate coverage [77]. The CEC 2013 large-scale global optimization competition highlighted that most standard algorithms fail catastrophically beyond D=500, motivating cooperative co-evolutionary approaches that decompose the problem into smaller subproblems [78].

Dynamic and stochastic optimization: many real-world mathematical problems involve objective functions that change over time or contain stochastic noise. Dynamic optimization requires algorithms to detect environmental changes and adapt their search accordingly, while stochastic optimization requires robust performance despite noisy objective function evaluations. The mathematical analysis of metaheuristic performance under these conditions remains underdeveloped [79].

Parameter sensitivity and self-adaptive mechanisms: the performance of metaheuristic algorithms is often highly sensitive to parameter settings, including population size, and algorithm-specific parameters such as inertia weight (PSO), mutation factor and crossover rate (DE), or coefficient vectors (GWO). The NFL Theorem implies that optimal parameter settings are problem-dependent, creating a secondary optimization problem (meta-optimization) of considerable difficulty. Self-adaptive parameter control mechanisms, where algorithm parameters are evolved alongside the decision variables, offer a promising but mathematically complex solution [80].

Benchmark bias and overfitting: a growing concern in the metaheuristic community is that newly proposed algorithms may be inadvertently or intentionally "tuned" to perform well on specific benchmark function sets, creating a form of algorithmic overfitting. Sörensen [27] criticized the "metaphor-based" algorithm development trend, arguing that many new algorithms are merely minor variations of existing methods dressed in novel metaphors without genuine algorithmic innovation. The development of more diverse, realistic, and adversarial benchmark suites, along with rigorous statistical testing protocols, is essential to address this concern [81].

8 | Future Research Directions

The future of metaheuristic algorithms in mathematics points toward several transformative research directions that promise to deepen theoretical understanding, expand practical capabilities, and forge new connections between optimization and emerging computational paradigms.

Quantum-inspired metaheuristics: quantum computing principles superposition, entanglement, and quantum tunneling offer fundamentally new search mechanisms for optimization. Quantum-inspired Evolutionary Algorithms (QEAs) use quantum bit (qubit) representations where each decision variable exists in a superposition of states, enabling exponentially richer exploration of the search space with linear resources [82]. Quantum-inspired variants of PSO, DE, and GA have demonstrated superior performance on selected benchmark problems, and the advent of practical quantum hardware raises the possibility of true quantum metaheuristics operating on quantum processors [83]. The mathematical analysis of quantum-classical hybrid optimization, including bounds on quantum speedup for metaheuristic search, represents a fertile research frontier.

Mathematical proofs of convergence for newer algorithms: developing rigorous convergence proofs for post-2010 algorithms (GWO, WOA, HHO, SCA, SSA, and others) remains a critical theoretical priority. Recent advances in stochastic approximation theory, martingale theory, and Markov chain analysis provide increasingly powerful mathematical tools for this endeavor. Proving convergence rates (beyond mere convergence in probability) would enable meaningful theoretical comparisons and guide algorithm design [84].

Integration with deep reinforcement learning: the integration of metaheuristic algorithms with Deep Reinforcement Learning (DRL) represents a paradigm shift from static, pre-defined search strategies to learned, adaptive optimization policies. DRL agents can learn algorithm selection policies, parameter adaptation strategies, and operator selection rules from the optimization trajectory itself, effectively learning to optimize [85]. This approach addresses the parameter sensitivity and algorithm selection challenges simultaneously and has shown promising results in recent studies on learning-based metaheuristic configuration.

Automated algorithm selection and configuration (autoML for metaheuristics): the algorithm selection problem choosing the best algorithm for a given problem instance and the algorithm configuration problem tuning algorithm parameters for optimal performance are both mathematically well-defined optimization problems themselves. Automated approaches such as SMAC (Sequential Model-based Algorithm Configuration), irace, and neural architecture search-inspired frameworks for metaheuristic design offer systematic solutions [86]. The mathematical formalization of the algorithm selection problem using performance models and landscape features is an active research area that bridges metaheuristic optimization with machine learning and statistics [41].

LSO (10,000+dimensions): scaling metaheuristic algorithms to ultra-high-dimensional problems ($D \geq 10{,}000$) requires fundamentally new approaches beyond simply increasing population size. Cooperative co-evolution with automatic variable decomposition, random embedding techniques that project high-dimensional problems into lower-dimensional subspaces, and dimensionality reduction through sensitivity analysis represent promising mathematical strategies [78]. The theoretical analysis of how algorithm performance degrades with dimensionality and the development of dimension-invariant metaheuristic frameworks are important open problems.

Theoretical unification framework: a grand challenge for the field is the development of a unified theoretical framework that subsumes existing metaheuristic algorithms as special cases of a general stochastic search paradigm. Such a framework would clarify the algorithmic relationships between ostensibly different methods, identify the minimal set of distinct search mechanisms, and enable the systematic construction of new algorithms with provable properties [27]. Information geometry, statistical physics, and category theory offer potential mathematical languages for such unification, but the realization of a comprehensive framework remains an aspirational goal for the community.

9 | Conclusion

This comprehensive systematic review has examined the landscape of metaheuristic optimization algorithms in the mathematical sciences, synthesizing theoretical foundations, methodological developments, and empirical performance across six major application domains. The review, encompassing 187 primary studies identified through the PRISMA framework from five major databases, provides a detailed and statistically rigorous assessment of the current state of the art.

The theoretical analysis reveals a significant gap between the practical success of metaheuristic algorithms and their mathematical underpinnings. While formal convergence proofs exist for classical algorithms such as SA [29] and GA with elitism [30], the majority of recently proposed algorithms lack rigorous convergence guarantees. The NFL Theorem [40] provides an important theoretical context, establishing that no single algorithm can dominate all others across all possible problem classes, while simultaneously justifying the development of algorithms tailored to specific problem structures.

Empirical evaluation across continuous function optimization benchmarks demonstrates that HHO, DE, and GWO form a statistically indistinguishable top tier in terms of overall performance, as confirmed by both Friedman and Wilcoxon statistical tests. For combinatorial optimization, ACO retains its established superiority on TSP instances, while DE with ε-constrained handling excels on constrained optimization problems. In multi-objective optimization, MOEA/D demonstrates consistently superior performance, particularly on three-objective problems. In the emerging area of numerical methods enhancement, DE and HHO show strong performance for solving nonlinear systems, parameter estimation, and quadrature optimization.

The bibliometric analysis confirms the explosive growth of the field, with publications increasing from approximately 340 in 2000 to over 12,800 in 2024. However, this growth has been accompanied by legitimate concerns about the proliferation of metaphor-based algorithms with questionable novelty. The community must balance openness to new ideas with rigorous standards for algorithmic innovation, supported by comprehensive statistical testing and comparison against established baselines.

Looking forward, the most promising research directions include quantum-inspired metaheuristics that leverage quantum computational principles, the development of rigorous convergence proofs for modern algorithms, the integration of deep reinforcement learning for adaptive optimization strategies, and the pursuit of a theoretical unification framework. The mathematical sciences community is uniquely positioned to lead these developments, bringing the analytical rigor and theoretical depth necessary to transform metaheuristic optimization from a largely empirical discipline into a mature mathematical field with strong theoretical foundations.

Authors' Contributions

All aspects of the research and manuscript preparation were carried out by the author. The author has read and approved the final version of the manuscript.

Data Availability

All data supporting the reported findings in this research paper are provided within the manuscript.

Funding

Not applicable.

Conflict of Interest

The author declares that they do not have any conflict of interest.

Consent for Publication

The author confirms consent for the publication of this work

Ethics Approval and Consent to Participate

This article does not contain any studies with human participants performed by the author.

References

[1] Dantzig, G. B. (2016). *Linear programming and extensions*. Princeton University Press. https://www.torrossa.com/en/resources/an/5793009

[2] Yang, X. S. (2020). *Nature-inspired optimization algorithms*. Academic Press. https://www.researchgate.net/publication/263171713

[3] Mirjalili, S., Song Dong, J., Sadiq, A. S., & Faris, H. (2019). Genetic algorithm: Theory, literature review, and application in image reconstruction. In *Nature-inspired optimizers: Theories, literature reviews and applications* (PP. 69-85). Springer, Cham. https://doi.org/10.1007/978-3-030-12127-3_5

[4] Nocedal, J., & Wright, S. J. (2006). *Numerical optimization*. New York, NY: Springer New York. https://doi.org/10.1007/978-0-387-40065-5

[5] Talbi, E. G. (2009). *Metaheuristics: From design to implementation*. John Wiley & Sons. https://zeus.inf.ucv.cl/~bcrawford/DiplomadoIA_2021/Cap1_Metaheuristics_Talbi.pdf

[6] Glover, F. W., & Kochenberger, G. A. (2003). *Handbook of metaheuristics*. Springer Science & Business Media. https://scispace.com/pdf/handbook-of-metaheuristics-3ue3f2vgdx.pdf

[7] Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, *13*(5), 533–549. https://doi.org/10.1016/0305-0548(86)90048-1

[8] Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM computing surveys (CSUR)*, *35*(3), 268–308. https://doi.org/10.1145/937503.937505

[9] Holland, J. H. (1992). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press. https://mitpress.mit.edu/9780262581110/adaptation-in-natural-and-artificial-systems/

[10] Goldberg, D. E. (1989). *Genetic algorithms in search, optimization, and machine learning.* Addison-Wesley. https://www2.fiit.stuba.sk/~kvasnicka/Free books/Goldberg_Genetic_Algorithms_in_Search.pdf

[11] Storn, R., & Price, K. (1997). Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, *11*(4), 341–359. https://doi.org/10.1023/A:1008202821328

[12] Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). IEEE. https://doi.org/10.1109/ICNN.1995.488968

[13] Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE transactions on systems, man, and cybernetics, part b (CYBERNETICS)*, *26*(1), 29–41. https://doi.org/10.1109/3477.484436

[14] Karaboga, D., & others. (2005). *An idea based on honey bee swarm for numerical optimization*. https://abc.erciyes.edu.tr/pub/tr06_2005.pdf

[15] Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671–680. https://cir.nii.ac.jp/crid/1370016973449261184

[16] Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in engineering software*, *69*, 46–61. https://doi.org/10.1016/j.advengsoft.2013.12.007

[17] Mirjalili, S., & Lewis, A. (2016). The whale optimization algorithm. *Advances in engineering software*, *95*, 51–67. https://doi.org/10.1016/j.advengsoft.2016.01.008

[18] Mirjalili, S. (2016). SCA: A sine cosine algorithm for solving optimization problems. *Knowledge-based systems*, *96*, 120–133. https://doi.org/10.1016/j.knosys.2015.12.022

[19] Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-based systems*, *89*, 228–249. https://doi.org/10.1016/j.knosys.2015.07.006

[20] Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris Hawks optimization: Algorithm and applications. *Future generation computer systems*, *97*, 849–872. https://doi.org/10.1016/j.future.2019.02.028

[21] Mirjalili, S., Gandomi, A. H., Mirjalili, S. Z., Saremi, S., Faris, H., & Mirjalili, S. M. (2017). Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. *Advances in engineering software*, *114*, 163–191. https://doi.org/10.1016/j.advengsoft.2017.07.002

[22] Jamil, M., & Yang, X. S. (2013). A literature survey of benchmark functions for global optimisation problems. *International journal of mathematical modelling and numerical optimisation*, *4*(2), 150–194. https://doi.org/10.1504/IJMMNO.2013.055204

[23] Derrac, J., Garcia, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and evolutionary computation*, *1*(1), 3–18. https://doi.org/10.1016/j.swevo.2011.02.002

[24] Shaikh, M. S., Raj, S., Zheng, G., Xie, S., Wang, C., Dong, X., & Junejo, N. U. R. (2025). Applications, classifications, and challenges: A comprehensive evaluation of recently developed metaheuristics for search and analysis. *Artificial intelligence review*, *58*(12), 1–110. https://doi.org/10.1007/s10462-025-11377-6

[25] Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. https://doi.org/10.1017/CBO9780511804441

[26] Yang, X. S. (2010). *Nature-inspired metaheuristic algorithms*. Luniver Press. https://www.researchgate.net/publication/235979455

[27] Sörensen, K. (2015). Metaheuristics—the metaphor exposed. *International transactions in operational research*, *22*(1), 3–18. https://doi.org/10.1111/itor.12001

[28] He, J., & Yu, X. (2001). Conditions for the convergence of evolutionary algorithms. *Journal of systems architecture*, *47*(7), 601–612. https://doi.org/10.1016/S1383-7621(01)00018-2

[29] Hajek, B. (1988). Cooling schedules for optimal annealing. *Mathematics of operations research*, *13*(2), 311–329. https://doi.org/10.1287/moor.13.2.311

[30] Rudolph, G. (1994). Convergence analysis of canonical genetic algorithms. *IEEE transactions on neural networks*, *5*(1), 96–101. https://doi.org/10.1109/72.265964

[31] Clerc, M., & Kennedy, J. (2002). The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE transactions on evolutionary computation*, *6*(1), 58–73. https://doi.org/10.1109/4235.985692

[32] den Bergh, F., & Engelbrecht, A. P. (2006). A study of particle swarm optimization particle trajectories. *Information sciences*, *176*(8), 937–971. https://doi.org/10.1016/j.ins.2005.02.003

[33] Zaharie, D. (2009). Influence of crossover on the behavior of differential evolution algorithms. *Applied soft computing*, *9*(3), 1126–1138. https://doi.org/10.1016/j.asoc.2009.02.012

[34] Gutjahr, W. J. (2002). ACO algorithms with guaranteed convergence to the optimal solution. *Information processing letters*, *82*(3), 145–153. https://www.aco-metaheuristic.org/downloads/ants3.pdf

[35] Črepinšek, M., Liu, S. H., & Mernik, M. (2013). Exploration and exploitation in evolutionary algorithms: A survey. *ACM computing surveys (CSUR)*, *45*(3), 1–33. https://doi.org/10.1145/2480741.2480752

[36] Lynn, N., & Suganthan, P. N. (2015). Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm and evolutionary computation*, *24*, 11–24. https://doi.org/10.1016/j.swevo.2015.05.002

[37] Shi, Y., & Eberhart, R. (1998). A modified particle swarm optimizer. *Evolutionary computation proceedings* (Vol. 890, pp. 69–73). IEEE. https://doi.org/10.1109/ICEC.1998.699146

[38] Das, S., & Suganthan, P. N. (2010). Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, *15*(1), 4–31. https://doi.org/10.1109/TEVC.2010.2059031

[39] Cuevas, E., Espejo, E. B., & Enriquez, A. C. (2019). *Metaheuristics algorithms in power systems*. Springer. https://doi.org/10.1007/978-3-030-11593-7

[40] Wolpert, D. H., & Macready, W. G. (2002). No free lunch theorems for optimization. *IEEE transactions on evolutionary computation*, *1*(1), 67–82. https://doi.org/10.1109/4235.585893

[41] Kerschke, P., Hoos, H. H., Neumann, F., & Trautmann, H. (2019). Automated algorithm selection: Survey and perspectives. *Evolutionary computation*, *27*(1), 3–45. https://doi.org/10.1162/evco_a_00242

[42] Igel, C., & Toussaint, M. (2005). A no-free-lunch theorem for non-uniform distributions of target functions. *Journal of mathematical modelling and algorithms*, *3*(4), 313–322. https://doi.org/10.1007/s10852-005-2586-y

[43] Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., & Mulrow, C. D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *British medical journal*, *372*. https://doi.org/10.1136/bmj.n71

[44] Yao, X., Liu, Y., & Lin, G. (1999). Evolutionary programming made faster. *IEEE transactions on evolutionary computation*, *3*(2), 82–102. http://dx.doi.org/10.1109/4235.771163

[45] Liang, J. J., Qu, B. Y., Suganthan, P. N., & Hernández Diaz, A. G. (2013). Problem definitions and evaluation criteria for the CEC 2013 special session on real-parameter optimization. *Computational intelligence laboratory, zhengzhou university, zhengzhou, china and nanyang technological university, singapore, technical report*, *201212*(34), 281–295. https://peerj.com/articles/cs-2671/CEC2013.pdf

[46] Applegate, D. L., Bixby, R. E., Chvátal, V., & Cook, W. J. (2011). *The traveling salesman problem: A computational study*. Princeton University Press. https://doi.org/10.1515/9781400841103

[47] Dorigo, M. (2007). Ant colony optimization. *Scholarpedia*, *2*(3), 1461. http://dx.doi.org/10.4249/scholarpedia.1461

[48] Larranaga, P., Kuijpers, C. M. H., Murga, R. H., Inza, I., & Dizdarevic, S. (1999). Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial intelligence review*, *13*(2), 129–170. https://doi.org/10.1023/A:1006529012972

[49] Shi, X. H., Liang, Y. C., Lee, H. P., Lu, C., & Wang, Q. X. (2007). Particle swarm optimization-based algorithms for TSP and generalized TSP. *Information processing letters*, *103*(5), 169–176. https://doi.org/10.1016/j.ipl.2007.03.010

[50] Mirjalili, S., & Lewis, A. (2013). S-shaped versus V-shaped transfer functions for binary particle swarm optimization. *Swarm and evolutionary computation*, *9*, 1–14. https://doi.org/10.1016/j.swevo.2012.09.002

[51] Malaguti, E., & Toth, P. (2010). A survey on vertex coloring problems. *International transactions in operational research*, *17*(1), 1–34. https://doi.org/10.1111/j.1475-3995.2009.00696.x

[52] Lan, G., DePuy, G. W., & Whitehouse, G. E. (2007). An effective and simple heuristic for the set covering problem. *European journal of operational research*, *176*(3), 1387–1403. https://doi.org/10.1016/j.ejor.2005.09.028

[53] Reinelt, G. (1991). TSPLIB—A traveling salesman problem library. *ORSA journal on computing*, *3*(4), 376–384. http://dx.doi.org/10.1287/ijoc.3.4.376

[54] Deb, K. (2000). An efficient constraint handling method for genetic algorithms. *Computer methods in applied mechanics and engineering*, *186*(2–4), 311–338. https://doi.org/10.1016/S0045-7825(99)00389-8

[55] Coello, C. A. C. (2002). Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: A survey of the state of the art. *Computer methods in applied mechanics and engineering*, *191*(11–12), 1245–1287. https://doi.org/10.1016/S0045-7825(01)00323-1

[56] Liang, J. J., Runarsson, T. P., Mezura Montes, E., Clerc, M., Suganthan, P. N., Coello, C. A. C., & Deb, K. (2006). Problem definitions and evaluation criteria for the CEC 2006 special session on constrained real-parameter optimization. *Journal of applied mechanics*, *41*(8), 8–31. https://www.researchgate.net/publication/216301032

[57] Takahama, T., & Sakai, S. (2006). Constrained optimization by the∊ constrained differential evolution with gradient-based mutation and feasible elites. *2006 IEEE international conference on evolutionary computation* (p. 2). IEEE. https://doi.org/10.1109/CEC.2006.1688283

[58] Parsopoulos, K. E., & Vrahatis, M. N. (2002). Particle swarm optimization method for constrained optimization problems. *Intelligent technologies–theory and application: New trends in intelligent technologies*, *76*(1), 214-220. https://www.researchgate.net/publication/2527227

[59] Mezura Montes, E., & Coello, C. A. C. (2011). Constraint-handling in nature-inspired numerical optimization: Past, present and future. *Swarm and evolutionary computation*, *1*(4), 173–194. https://doi.org/10.1016/j.swevo.2011.10.001

[60] Deb, K., Sindhya, K., & Hakanen, J. (2016). Multi-objective optimization. In *Decision sciences* (pp. 161–200). CRC Press. https://doi.org/10.1201/9781315183176-4

[61] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, *6*(2), 182–197. https://doi.org/10.1109/4235.996017

[62] Coello, C. A. C., Pulido, G. T., & Lechuga, M. S. (2004). Handling multiple objectives with particle swarm optimization. *IEEE transactions on evolutionary computation*, *8*(3), 256–279. https://doi.org/10.1109/TEVC.2004.826067

[63] Zhang, Q., & Li, H. (2007). MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE transactions on evolutionary computation*, *11*(6), 712–731. https://doi.org/10.1109/TEVC.2007.892759

[64] Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE transactions on evolutionary computation*, *7*(2), 117–132. https://doi.org/10.1109/TEVC.2003.810758

[65] Grosan, C., & Abraham, A. (2008). A new approach for solving nonlinear equations systems. *IEEE transactions on systems, man, and cybernetics-part a: Systems and humans*, *38*(3), 698–714. https://doi.org/10.1109/TSMCA.2008.918599

[66] Voss, H. U., Timmer, J., & Kurths, J. (2004). Nonlinear dynamical system identification from uncertain and indirect measurements. *International journal of bifurcation and chaos*, *14*(06), 1905–1933. https://doi.org/10.1142/S0218127404010345

[67] Rao, R. V., & Pawar, P. J. (2010). Parameter optimization of a multi-pass milling process using non-traditional optimization algorithms. *Applied soft computing*, *10*(2), 445–456. https://doi.org/10.1016/j.asoc.2009.08.007

[68] Wolsey, L. A. (1998). *Integer programming*. New York, NY: John Wiley & Sons. https://books.google.com/books/about/Integer_Programming.html?id=x7RvQgAACAAJ

[69] Liao, T., Stützle, T., de Oca, M. A. M., & Dorigo, M. (2014). A unified ant colony optimization algorithm for continuous optimization. *European journal of operational research*, *234*(3), 597–609. https://doi.org/10.1016/j.ejor.2013.10.024

[70] Kennedy, J., & Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. *1997 IEEE international conference on systems, man, and cybernetics. Computational cybernetics and simulation* (Vol. 5, pp. 4104–4108). IEEE. https://doi.org/10.1109/ICSMC.1997.637339

[71] Emary, E., Zawbaa, H. M., & Hassanien, A. E. (2016). Binary grey wolf optimization approaches for feature selection. *Neurocomputing*, *172*, 371–381. https://doi.org/10.1016/j.neucom.2015.06.083

[72] Gendreau, M., Potvin, J Y., & others. (2010). *Handbook of metaheuristics*. Springer. https://doi.org/10.1007/978-1-4419-1665-5

[73] Maniezzo, V., Stützle, T., & Voß, S. (2021). *Matheuristics*. Springer. https://doi.org/10.1007/978-3-030-70277-9

[74] Garcia, S., Fernández, A., Luengo, J., & Herrera, F. (2009). A study of statistical techniques and performance measures for genetics-based machine learning: Accuracy and interpretability. *Soft computing*, *13*(10), 959–977. https://doi.org/10.1007/s00500-008-0392-y

[75] Wilcoxon, F. (1945). Individual comparisons by ranking methods. *Biometrics bulletin*, *1*(6), 80–83. https://doi.org/10.2307/3001968

[76] Hussain, K., Mohd Salleh, M. N., Cheng, S., & Shi, Y. (2019). Metaheuristic research: A comprehensive survey. *Artificial intelligence review*, *52*(4), 2191–2233. https://doi.org/10.1007/s10462-017-9605-z

[77] Mahdavi, S., Shiri, M. E., & Rahnamayan, S. (2015). Metaheuristics in large-scale global continues optimization: A survey. *Information sciences*, *295*, 407–428. https://doi.org/10.1016/j.ins.2014.10.042

[78] Yang, Z., Tang, K., & Yao, X. (2008). Large scale evolutionary optimization using cooperative coevolution. *Information sciences*, *178*(15), 2985–2999. https://doi.org/10.1016/j.ins.2008.02.017

[79] Jin, Y., & Branke, J. (2005). Evolutionary optimization in uncertain environments-a survey. *IEEE transactions on evolutionary computation*, *9*(3), 303–317. https://doi.org/10.1109/TEVC.2005.846356

[80] Aleti, A., & Moser, I. (2016). A systematic literature review of adaptive parameter control methods for evolutionary algorithms. *ACM computing surveys (CSUR)*, *49*(3), 1–35. https://doi.org/10.1145/2996355

[81] Liang, J. J., Qu, B. Y., & Suganthan, P. N. (2013). Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization. *Computational intelligence laboratory, zhengzhou university, zhengzhou china and technical report, nanyang technological university, singapore*, *635*(2), 2014. http://bee22.com/manual/tf_images/Liang CEC2014.pdf

[82] Han, K. H., & Kim, J. H. (2002). Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE transactions on evolutionary computation*, *6*(6), 580–593. https://doi.org/10.1109/TEVC.2002.804320

[83] Zhang, G., Rong, H., Neri, F., & Pérez Jiménez, M. J. (2014). An optimization spiking neural P system for approximately solving combinatorial optimization problems. *International journal of neural systems*, *24*(05), 1440006. https://doi.org/10.1142/S0129065714400061

[84] Hu, Z., Xiong, S., Su, Q., & Zhang, X. (2013). Sufficient conditions for global convergence of differential evolution algorithm. *Journal of applied mathematics*, *2013*(1), 193196. https://doi.org/10.1155/2013/193196

[85] Li, K., Zhang, T., Wang, R., Wang, Y., Han, Y., & Wang, L. (2021). Deep reinforcement learning for combinatorial optimization: Covering salesman problems. *IEEE transactions on cybernetics*, *52*(12), 13142–13155. https://doi.org/10.1109/TCYB.2021.3103811

[86] Hutter, F., Hoos, H. H., & Leyton Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. *International conference on learning and intelligent optimization* (pp. 507–523). Springer. https://doi.org/10.1007/978-3-642-25566-3_40