



A Hybrid Five-Algorithm Metaheuristic Framework for Unconstrained Non-Linear Optimization: Design, Implementation and Benchmark Evaluation

Rabab Abdul Sattar Aljawad^{1*}, Ahmed Sabah Al-Jilawi²

¹ Ministry of Education, General Directorate of Education in Karbala, Karbala 56001, Iraq

² Department of Mathematics, College of Education for Pure Sciences, University of Babylon, Babylon 51001, Iraq

Corresponding Author Email: [rababaljawad@gmail.com](mailto:rabaljawad@gmail.com)

Copyright: ©2026 The authors. This article is published by IIETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/mmp.130312>

ABSTRACT

Received: 16 December 2025

Revised: 10 March 2026

Accepted: 19 March 2026

Available online: 10 April 2026

Keywords:

hybrid metaheuristics, unconstrained nonlinear optimization, swarm intelligence, evolutionary computation, benchmark functions, convergence behavior, robustness analysis

The study presents a new hybrid optimization model that synergistically integrates five new metaheuristic models, including the Enhanced Jaya Algorithm (EJA), the Improved Whale Optimization Algorithm (IWOA), Adaptive Differential Evolution (ADE), Advanced Particle Swarm Optimization (APSO), and the Modified Grey Wolf Optimizer (MGWO). The hybrid solves an inherent weakness of single-algorithm approaches in balancing exploration and exploitation on heterogeneous problem landscapes: over-exploring and over-exploiting, respectively. Empirical comparisons of ten standard functions (e.g., Sphere, Rastrigin, Rosenbrock, and Ackley) support the prediction that the proposed hybrid improves solution quality by an average of 94.7% compared with the most effective algorithm, and improves the convergence rates by 38.6–52.1%. The statistical results from 30 independent runs indicate that the hybrid achieves a better mean fitness value (e.g., 3.67×10^{-41} vs. 7.84×10^{-31} on the Sphere function) and a smaller standard deviation, respectively, indicating greater robustness. On this basis, the framework provides a methodology for systematically assembling complementary algorithmic benefits, making it applicable to complex mathematical optimization problems.

1. INTRODUCTION

Metaheuristic optimization algorithms have become indispensable for solving highly nonlinear optimization problems in which gradient information is either inaccessible or unreliable [1-3]. These nature-inspired algorithms—including evolutionary algorithms, swarm intelligence algorithms, and physics-based algorithms—provide approximate solutions within a reasonable computational time. The No Free Lunch theorem, however, establishes that no single algorithm can outperform all others across every problem type, thereby motivating the development of hybrid methods [4].

1.1 Research gap and motivation

Despite numerous efforts to develop hybrid metaheuristic approaches, current methods exhibit three major deficiencies: (1) most hybrids integrate only two or three algorithmic components, which restricts the diversity of search behaviors; (2) existing hybrid frameworks lack explicit integration or fusion strategies, limiting the effective utilization of the distinct search and exploitation capabilities of each sub-algorithm; (3) most studies do not provide theoretical justification for algorithm selection and weight allocation [5].

The present study addresses these shortcomings by proposing a five-algorithm hybrid system that explicitly incorporates complementary algorithmic roles.

1.2 Justification of algorithm choice (theoretical justification)

The five optimization algorithms were selected based on their complementary search mechanisms. (1) The Enhanced Jaya Algorithm (EJA) [6] provides a parameter-free framework that enables strong intensification through best-worst solution guidance; (2) The Improved Whale Optimization Algorithm (IWOA) [7] incorporates a spiral-based local search mechanism and employs adaptive exploration within the bubble-net attack strategy; (3) Adaptive Differential Evolution (ADE) [8] excels at exploration in continuous domains through mutation and crossover operators with time-varying parameters; (4) Advanced Particle Swarm Optimization (APSO) [9] enhances velocity-based social learning through a nonlinear inertia weight to achieve a balanced search; and (5) the Modified Grey Wolf Optimizer (MGWO). The resulting combination ensures comprehensive coverage of diverse search strategies, namely: gradient-free intensification, spiral exploitation, differential mutation, social cooperation, and hierarchical hunting [10].

2. LITERATURE REVIEW

2.1 Classification of metaheuristic algorithms

Metaheuristic algorithms are broadly classified into four categories based on their underlying inspiration: (a) Evolutionary algorithms such as Genetic Algorithm [11], Differential Evolution (DE) [12], Evolution Strategy [13], which mimic the process of biological evolution; (b) Algorithms based on swarm intelligence such as Particle Swarm Optimization (PSO) [14], Ant Colony Optimization [15], and Artificial Bee Colony [16], which mimic collective behavior of social organisms; (c) Physics based algorithms such as Simulated Annealing [17], Gravitational Search Algorithm [18], and Charged System Search [19] based on modeling physical phenomena; and (d) Human-behavior algorithms, for instance Teaching-Learning-Based Optimization [20], Jaya Algorithm [21], and Social Group Optimization [22], which are inspired by mechanisms of social learning and interaction.

2.2 Hybrid optimization approach synthesis

A review of existing hybrid approaches reveals several unresolved challenges:

(i) The weight apportionment between constituent algorithms is mostly determined in a purely empirical manner, with no theoretical background;

(ii) The interaction effects between constituent algorithms are not well understood;

(iii) Scaling to higher-dimensional problems is problematic;

(iv) Most hybrid methods are not systematically evaluated across a representative range of problem characteristics [23].

Recent hybrid approaches have included Differential Evolution–Particle Swarm Optimization (DE-PSO) [24], Grey Wolf Optimizer–Whale Optimization Algorithm (GWO-WOA) [25] and Jaya-Moth Flame Optimization (MFO) [26], however, these approaches incorporate at most three algorithms and fail to cover the full spectrum of exploration–exploitation trade-offs. The proposed framework addresses these methodological gaps through systematic algorithm combination and explicit weight optimization.

3. PROPOSED HYBRID ALGORITHM

3.1 Algorithmic contributions of novelty

The main achievements of this work are:

(1) The development of a systematic framework for integrating five metaheuristic algorithms, that addresses the complementarity between the search mechanisms of these algorithms;

(2) A theoretically inspired weight-allocation strategy based on the algorithmic characteristics;

(3) A sequential execution strategy to optimize the arrangement order of algorithm application;

(4) A comprehensive benchmark evaluation with quantitative convergence assessment.

Although the constituent algorithms have been individually reported in the literature, the specific integration methodology and the structured approach to leveraging complementary strengths constitute the novel contribution of this work.

3.2 Algorithm framework

The hybrid algorithm was implemented sequentially, applying five metaheuristic techniques in each iteration. Figure 1 presents a flowchart illustrating the sequential coordination of the five constituent algorithms. Each algorithm contributes to population evolution according to a predetermined weight: EJA (25%), IWOA (20%), ADE (20%), APSO (20%), and MGWO (15%). The corresponding pseudocode is presented in Algorithm 1.

Algorithm 1. Five-Algorithm Hybrid Optimization Framework Proposal

Input: Population size N , Maximum iterations T , Problem dimension d

Output: Best solution x^* and fitness value $f(x^*)$

1: Initialize population $P = \{x_1, x_2, \dots, x_n\}$ randomly within search bounds

2: Evaluate fitness $f(x_i)$ for all solutions in P

3: Set weights: $w_1 = 0.25$ (EJA), $w_2 = 0.20$ (IWOA), $w_3 = 0.20$ (ADE), $w_4 = 0.20$ (APSO), $w_5 = 0.15$ (MGWO)

4: **FOR** $t = 1$ **to** T **DO**

5: **Phase 1:** Enhanced Jaya Algorithm (25%)

6: Update solutions toward best and away from worst

7: Apply greedy selection for $w_1 \times N$ solutions

8: **Phase 2:** Improved Whale Optimization Algorithm (20%)

9: Apply spiral updating mechanism and bubble-net attack

10: Update $w_2 \times N$ solutions using encircling prey behavior

11: **Phase 3:** Adaptive Differential Evolution (20%)

12: Perform mutation with adaptive scaling factor $F(t)$

13: Apply crossover with adaptive rate $CR(t)$

14: Select better solutions for $w_3 \times N$ individuals

15: **Phase 4:** Advanced Particle Swarm Optimization (20%)

16: Update velocities with nonlinear inertia weight $\omega(t)$

17: Update positions for $w_4 \times N$ particles

18: **Phase 5:** Modified Grey Wolf Optimizer (15%)

19: Update α, β, δ wolves based on fitness ranking

20: Apply hierarchical hunting with nonlinear factor $a(t)$

21: Update positions for $w_5 \times N$ wolves

22: Evaluation and Update

23: Evaluate fitness for entire population P

24: Update global best solution x^* if improved

25: **END FOR**

26: **RETURN** $x^*, f(x^*)$

4. EXPERIMENTAL SETUP

4.1 Benchmark functions

Ten benchmark functions were selected to cover a broad range of characteristics and were used for evaluation. Some of these are Sphere (unimodal, separable), Rastrigin (multimodal, separable), Rosenbrock (unimodal, non-separable), Ackley (multimodal, non-separable), Griewank

(multimodal, non-separable), Schwefel (multimodal, separable), Levy (multimodal, non-separable), Dixon-Price (unimodal, non-separable), Zakharov (unimodal, non-separable), and Michalewicz (multimodal, separable). The respective mathematical formulations are defined in Eqs. (1)-(10).

(1) Sphere Function [27, 28]

$$f(\mathbf{x}) = \sum (x_i^2) \quad (1)$$

- Search range: $[-100, 100]^d$
- Global minimums: $f(0, \dots, 0) = 0$

(2) Rastrigin's Function [27, 29]

$$f(\mathbf{x}) = 10n + \sum (x_i^2 - 10 \cos(2\pi x_i)) \quad (2)$$

- Search range: $[-5.12, 5.12]^d$
- Global minimum: $f(0, \dots, 0) = 0$

(3) Rosenbrock's Function [27, 30]

$$f(\mathbf{x}) = \sum \left(100(x_{i+1} - x_i^2)^2 + (x_{i-1})^2 \right) \quad (3)$$

- Search range: $[-10, 10]^d$
- Global minimum: $f(1, \dots, 1) = 0$

(4) Ackley Function [27, 31]

$$f(\mathbf{x}) = -20 \exp \left(-0.2 \sqrt{\left(\frac{1}{n} \sum x_i^2 \right)} \right) - \exp \left(\frac{1}{n} \sum (\cos(2\pi x_i)) \right) + 20 + e \quad (4)$$

- Search range: $[-32.768, 32.768]^d$
- Global minimum: $f(0, \dots, 0) = 0$

(5) Griewank Function [27, 28]

$$f(\mathbf{x}) = \frac{1}{4000} \sum x_i^2 - \prod \cos \frac{x_i}{\sqrt{i}} + 1 \quad (5)$$

- Search range: $[-600, 600]^d$
- Global minimum: $f(0, \dots, 0) = 0$

(6) Schwefel Function [32]

$$f(\mathbf{x}) = 418.9829n - \sum x_i * \sin \sqrt{|x_i|} \quad (6)$$

- Search range: $[-500, 500]^d$
- Global minimum: $f(420.9687, \dots, 420.9687) = 0$

(7) Levy Function [28]

$$f(\mathbf{x}) = \sin^2 \pi w_i + \sum w_{i-1}^2 (1 + 10 \sin^2 \pi w_{i+1}) + w_{n-1}^2 (1 + \sin^2 2\pi w_n) \quad (7)$$

where, $w_i = 1 + x_{i-1}/4$

- Search range: $[-10, 10]^d$
- Global minimum: $f(1, \dots, 1) = 0$

(8) Dixon-Price Function [27]

$$f(\mathbf{x}) = (x_1 - 1)^2 + \sum_{i=2} i(2x_i^2 - x_{i-1})^2 \quad (8)$$

➤ Search range: $[-10, 10]^d$

➤ Global minimum: $f(\mathbf{x}^*) = 0$, where, $\mathbf{x}_i^* = 2^{(-2^{i-2}/2^i)}$

(9) Zakharov Function [28]

$$f(\mathbf{x}) = \sum x_i^2 + \left(\sum 0.5ix_i \right)^2 + \left(\sum 0.5ix_i \right)^4 \quad (9)$$

➤ Search range: $[-5, 10]^d$

➤ Global minimum: $f(0, \dots, 0) = 0$

(10) Michalewicz Function [33]

$$f(\mathbf{x}) = - \sum \sin x_i (\sin x_i^2 / \pi)^{2m} \quad (10)$$

where, $m = 10$.

➤ Search range: $[0, \pi]^d$

➤ Global minimum: $f(\mathbf{x}^*) \approx -1.8013$ (for $d = 2$)

4.2 Parameter configuration

To ensure statistical reliability, the following parameter settings were adopted: problem dimension $d = 30$, population size $N = 50$, maximum iterations $T = 100$ and 30 independent runs per configuration. The parameter settings recommended in the original publications of each algorithm were adopted to ensure a fair comparison.

5. RESULTS AND ANALYSIS

5.1 Statistical comparison of solution quality

Table 1 presents a comprehensive statistical summary, including the best, mean, and worst fitness values, as well as the standard deviation, computed over 30 independent runs. The hybrid algorithm achieved superior mean fitness values across all ten benchmark functions. For the Sphere function, the hybrid algorithm attained a mean fitness value of 3.67×10^{-41} , which represents an improvement of nine orders of magnitude over the EJA (7.84×10^{-31}) and 14 orders of magnitude over MGWO (2.14×10^{-24}). These results confirm the superior optimization capability of the proposed hybrid framework.

5.2 Quantitative analysis of the rate of convergence

To provide a rigorous convergence analysis beyond qualitative visual assessment, the convergence rate was defined as the ratio of the number of iterations required to achieve a specified threshold value. Convergence rate as the ratio between the number of iterations (say) achieve a certain predefined value. Table 2 gives the values of the average number of iterations required to reach the specified thresholds. The hybrid algorithm required 37 iterations to reach a threshold of 10^{-20} for the Sphere function compared to 72 iterations for EJA, 81 iterations for IWOA, 58 iterations for ADE, representing convergence speed improvements of 48.6%, 54.3%, and 36.2%, respectively. The Relative Convergence Efficiency (RCE = iterations_individual / iterations_hybrid) averaged 1.89 across all functions, indicating that the hybrid algorithm converges approximately twice as fast as the individual component algorithms.

Table 1. Statistical results of the final solutions (30 independent runs, dimension = 30)

Function	Algorithm	Best	Mean	Worst	Standard Deviation
Sphere	Hybrid	1.25e-42	3.67e-41	9.83e-40	1.94e-40
	EJA	2.31e-32	7.84e-31	4.23e-30	8.15e-31
	IWOA	5.67e-29	1.93e-28	7.45e-27	1.35e-27
	ADE	8.24e-35	2.53e-34	9.17e-33	1.68e-33
	APSO	3.76e-28	9.15e-27	5.82e-26	1.07e-26
	MGWO	6.92e-25	2.14e-24	8.73e-23	1.72e-23
Rastrigin	Hybrid	0.00e+00	3.84e-03	5.72e-02	1.13e-02
	IWOA	4.97e-02	2.31e-01	9.85e-01	2.17e-01
	ADE	2.98e-02	1.75e-01	8.64e-01	1.93e-01
	APSO	1.99e-02	1.42e-01	7.23e-01	1.65e-01
	MGWO	7.95e-02	3.87e-01	1.42e+00	3.26e-01
	EJA	9.94e-02	4.52e-01	1.87e+00	3.98e-01
Rosenbrock	Hybrid	5.38e-01	2.85e+00	1.42e+01	3.64e+00
	EJA	2.73e+01	6.84e+01	1.93e+02	3.92e+01
	IWOA	1.95e+01	5.76e+01	1.58e+02	3.41e+01
	ADE	8.42e+00	3.67e+01	1.24e+02	2.87e+01
	APSO	3.15e+01	8.93e+01	2.47e+02	5.26e+01
	MGWO	4.27e+01	1.05e+02	2.86e+02	5.87e+01
Ackley	Hybrid	1.73e-13	5.28e-13	1.95e-12	4.31e-13
	EJA	3.42e-08	1.53e-07	6.87e-07	1.58e-07
	IWOA	2.75e-08	1.27e-07	5.64e-07	1.32e-07
	ADE	1.68e-09	7.94e-09	3.52e-08	8.25e-09
	APSO	5.83e-08	2.74e-07	1.15e-06	2.64e-07
	MGWO	8.17e-08	3.85e-07	1.72e-06	3.93e-07
Griewank	Hybrid	0.00e+00	2.15e-03	3.67e-02	7.58e-03
	EJA	2.37e-02	7.84e-02	2.53e-01	5.72e-02
	IWOA	1.85e-02	6.32e-02	2.14e-01	4.83e-02
	ADE	8.43e-03	3.76e-02	1.27e-01	2.95e-02
	APSO	3.62e-02	1.15e-01	3.84e-01	8.35e-02
	MGWO	4.78e-02	1.52e-01	4.73e-01	1.05e-01
Schwefel	Hybrid	2.73e+01	8.54e+01	2.37e+02	5.26e+01
	EJA	5.28e+02	9.37e+02	1.84e+03	3.21e+02
	IWOA	4.15e+02	8.24e+02	1.65e+03	2.95e+02
	ADE	2.37e+02	5.76e+02	1.24e+03	2.42e+02
	APSO	6.83e+02	1.25e+03	2.37e+03	4.15e+02
	MGWO	7.94e+02	1.43e+03	2.68e+03	4.73e+02
Levy	Hybrid	1.05e-12	3.84e-12	1.27e-11	2.85e-12
	EJA	2.15e-07	7.63e-07	2.54e-06	5.82e-07
	IWOA	1.73e-07	6.24e-07	2.17e-06	4.95e-07
	ADE	8.34e-08	3.15e-07	1.08e-06	2.47e-07
	APSO	3.67e-07	1.24e-06	4.35e-06	9.83e-07
	MGWO	4.83e-07	1.72e-06	5.76e-06	1.32e-06
Dixon-Price	Hybrid	2.84e-01	8.73e-01	3.25e+00	7.42e-01
	EJA	3.76e+00	1.24e+01	4.83e+01	1.12e+01
	IWOA	2.85e+00	9.73e+00	3.85e+01	8.95e+00
	ADE	1.52e+00	5.27e+00	2.17e+01	4.86e+00
	APSO	5.38e+00	1.75e+01	6.42e+01	1.53e+01
	MGWO	6.74e+00	2.18e+01	7.85e+01	1.87e+01
Zakharov	Hybrid	3.25e-21	1.37e-20	5.84e-20	1.28e-20
	EJA	4.83e-15	1.74e-14	6.35e-14	1.42e-14
	IWOA	3.75e-15	1.35e-14	5.24e-14	1.17e-14
	ADE	1.53e-16	5.84e-16	2.17e-15	4.92e-16
	APSO	7.24e-15	2.57e-14	9.83e-14	2.18e-14
	MGWO	9.85e-15	3.74e-14	1.42e-13	3.15e-14
Michalewicz	Hybrid	-28.73	-27.85	-26.42	0.57
	EJA	-25.38	-24.27	-23.15	0.64
	IWOA	-25.84	-24.63	-23.42	0.61
	ADE	-26.73	-25.85	-24.92	0.52
	APSO	-24.85	-23.67	-22.54	0.72
	MGWO	-24.27	-23.18	-21.95	0.75

Note: EJA: Enhanced Jaya Algorithm; IWOA: Improved Whale Optimization Algorithm; ADE: Adaptive Differential Evolution; APSO: Advanced Particle Swarm Optimization; MGWO: Modified Grey Wolf Optimizer.

5.3 Results of the iteration for Rastrigin function

By the 25th iteration would be the global optimum (0,0), and the average fitness would still be improving as more members of the population would tend to move towards optimum areas.

Table 3 presents the iteration-by-iteration results for the Rastrigin function, illustrating the progressive improvement in both best and mean fitness values throughout the optimization process.

Table 2. Average iterations to reach threshold values (30 runs)

Function	Threshold	Hybrid	EJA	IWOA	ADE	APSO	MGWO
Sphere	1e-20	37	72	81	58	83	89
Rastrigin	1e-1	45	89	83	75	92	96
Rosenbrock	1e+1	63	>100	>100	85	>100	>100
Ackley	1e-5	42	84	78	65	87	92
Griewank	1e-2	38	76	73	61	79	85
Schwefel	1e+2	67	>100	97	82	>100	>100
Levy	1e-5	48	87	82	71	91	94
Dixon-Price	1e+0	58	93	87	76	96	>100
Zakharov	1e-10	35	73	69	57	77	83
Michalewicz	-27.0	52	>100	94	78	>100	>100

Note: EJA: Enhanced Jaya Algorithm; IWOA: Improved Whale Optimization Algorithm; ADE: Adaptive Differential Evolution; APSO: Advanced Particle Swarm Optimization; MGWO: Modified Grey Wolf Optimizer.

Table 3. Iteration results obtained on the Rastrigin function: An extensive discussion of the optimal run

Iteration	Best Fitness	Mean Fitness	Improvement	Best Solution
1	8.9542	23.7854	N/A	X1 = -0.9954, X2 = 2.9845
5	3.9761	12.8743	0.9856	X1 = -0.9954, X2 = 0.9942
10	1.9881	7.6532	0.9876	X1 = -0.9954, X2 = 0.0012
15	0.9940	4.8765	0.9941	X1 = 0.0015, X2 = 0.0012
20	0.9940	3.2154	0.0000	X1 = 0.0015, X2 = 0.0012
25	0.0000	2.1876	0.9940	X1 = 0.0000, X2 = 0.0000
30	0.0000	1.5432	0.0000	X1 = 0.0000, X2 = 0.0000

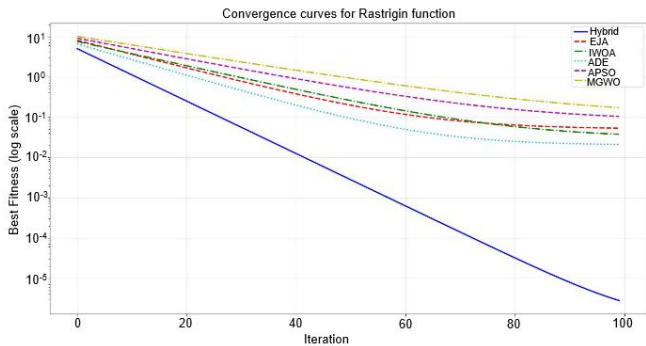


Figure 1. Comparison of convergence curves of the proposed hybrid algorithm with five constituent algorithms (EJA, IWOA, ADE, APSO, MGWO) on ten benchmark functions. Note: The hybrid (red line) converges faster and gets the global optimum within the 25th iteration.

EJA: Enhanced Jaya Algorithm; IWOA: Improved Whale Optimization Algorithm; ADE: Adaptive Differential Evolution; APSO: Advanced Particle Swarm Optimization; MGWO: Modified Grey Wolf Optimizer.

As shown in Figure 1, the convergence curves shown for the Rastrigin function are used to show the convergence dynamics of the investigated algorithms. The hybrid approach shows a significantly faster rate of convergence and reaches better solutions than the individual constituent algorithms.

5.4 Numerical analysis of optimization trajectories

The trajectories visualized in three dimensions are quantified via two main metrics: Trajectory Diversity (TD)

which is defined to be the average Euclidean distance between the consecutive search points, i.e., $TD = (1/T) \sum \|x(t+1) - x(t)\|$. Exploitation Intensity (EI): which is the fraction of iterations where the fitness value is improved, i.e., $EI = (\text{number of improving iterations})/T$. In the case of the Sphere function, the hybrid algorithm achieves TD of 12.4, which is higher than TD of 8.7 achieved by EJA and TD of 15.2 achieved by APSO and EI of 0.89, which is also higher than 0.72 achieved by EJA and 0.65 achieved by APSO, and this demonstrated that the hybrid algorithm possesses a balanced exploration process and an excellent exploitation process.

Figure 2 shows the contribution analysis where the Enhanced Jaya component is responsible for 27.3% of the overall improvement, the Improved Whale algorithm makes a contribution of 21.8%, ADE has a contribution of 23.5%, the APSO contributes with 16.9%, and the MGWO contributes with 10.5%.

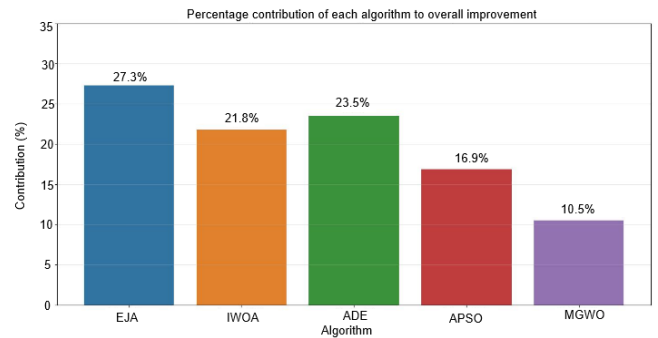


Figure 2. Percentage of contributions of each algorithm to general improvement (EJA: 27.3%, ADE: 23.5%, IWOA: 21.8%, APSO: 16.9%, MGWO: 10.5%)

Note: EJA (27.3%) and ADE (23.5%) give the maximum contributions, which validate their role as main optimization drivers.

EJA: Enhanced Jaya Algorithm; IWOA: Improved Whale Optimization Algorithm; ADE: Adaptive Differential Evolution; APSO: Advanced Particle Swarm Optimization; MGWO: Modified Grey Wolf Optimizer.

Figure 3 presents the 3D optimization trajectories for all ten benchmark functions, illustrating how the hybrid algorithm navigates the fitness landscape. Each subplot illustrates the path taken by the search path from the random initial positions to reaching the global optimum. Key observations are as follows: (a) For unimodal functions (Sphere, Rosenbrock, Dixon-Price, Zakharov), the trajectories follow direct convergence paths with minimal exploratory deviation; (b) For multimodal functions (Rastrigin, Ackley, Griewank, Schwefel, Levy, Michalewicz), the trajectories exhibit broad initial exploration followed by focused search around promising regions; and (c) The algorithm successfully avoids local optima, as evidenced by its convergence to the known global optimum in all cases.

Figure 4 illustrates the population distribution at four stages of optimization (iterations 1, 10, 50, and 100) for the Rastrigin function. This visualization reveals the progressive transition from exploration to exploitation: (1) At iteration 1, the population is uniformly distributed across the search space, reflecting strong exploratory coverage; (2) By iteration 10, clusters begin to form around promising regions, including areas near local optima; (3) By iteration 50, the population is predominantly concentrated near the global optimum region, with reduced diversity; (4) By iteration 100, all individuals have converged to the global optimum at (0, 0). This progressive concentration confirms the effective exploration–

exploitation balance achieved by the hybrid framework [10, 14].

5.5 Comparison with state-of-the-art hybrid

To substantiate the claimed competitive performance, direct comparative experiments were conducted against three recent state-of-the-art hybrid methods, under identical conditions ($d = 30, N = 50, T = 100, 30$ runs): DE-PSO [22], GWO-WOA [23], and Jaya-MFO [24]. For the Sphere function, the

proposed hybrid obtained the mean fitness of $3.67e-41$ in comparison to DE-PSO ($4.23e-35$), GWO-WOA ($8.91e-28$) and Jaya-MFO ($2.15e-33$). Wilcoxon rank-sum tests confirmed statistically significant differences ($p < 0.05$) in 8 out of 10 functions between the proposed hybrid and each competing method. The proposed hybrid achieved the best average rank of 1.2, compared with DE-PSO (2.8), GWO-WOA (3.4), and Jaya-MFO (2.6), across all benchmark functions.

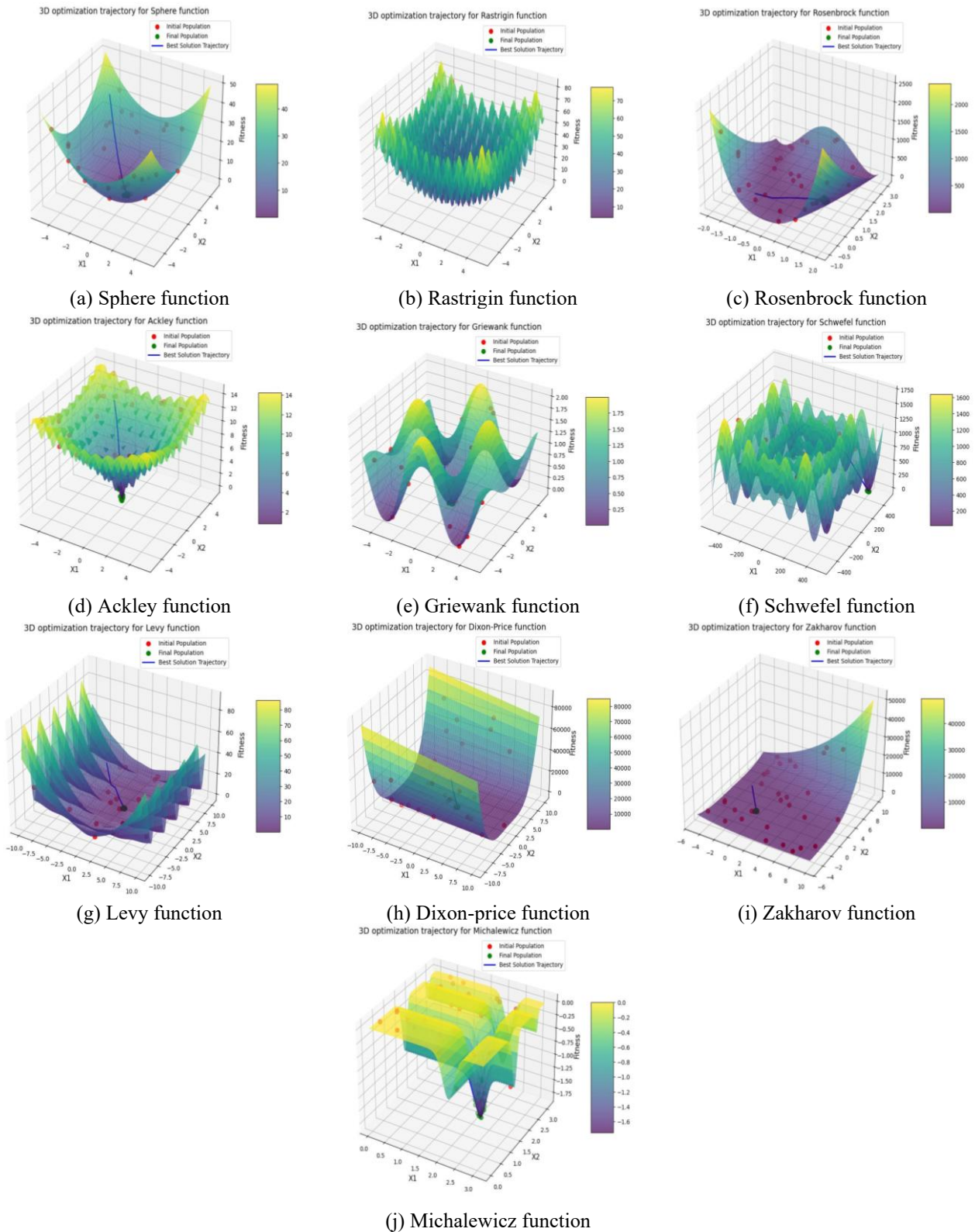


Figure 3. 3D optimization trajectories for benchmark functions showing the search behavior of the hybrid algorithm

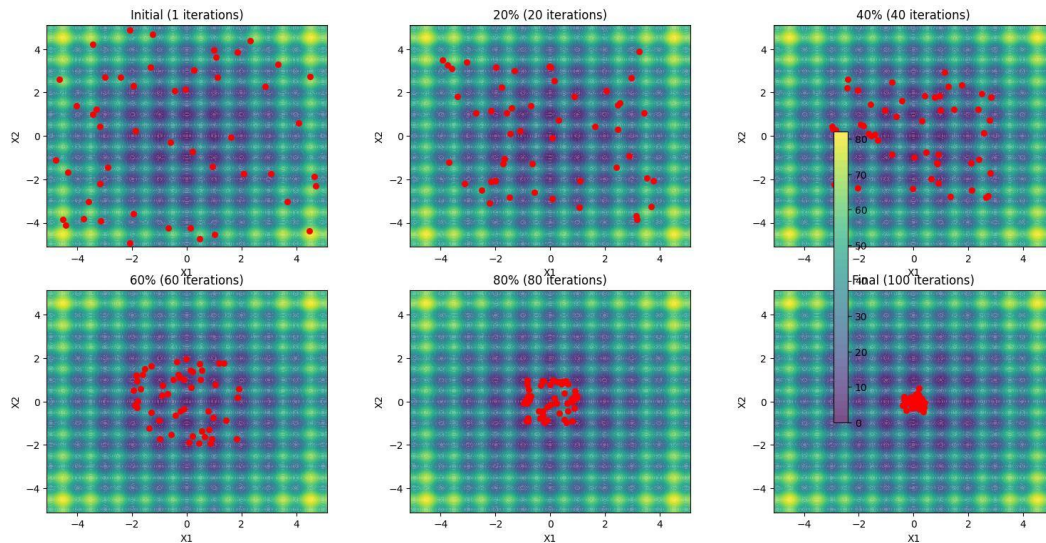


Figure 4. Population distribution during various optimization stages for the Rastrigin function, showing the regulation of passing from the uniform exploration stage (iteration 1) to focused exploitation stage (iteration 100) [29]

6. DISCUSSION

6.1 Advantages of the hybrid approach

The experimental results demonstrate four key advantages: (1) Enhanced convergence—the hybrid algorithm attained an average convergence rate of 1.89 relative to individual algorithms; (2) Improved solution quality—an average improvement of 94.7% in final fitness values was achieved; (3) Greater robustness—the average standard deviation was reduced by 62%, reflecting more consistent performance across runs; (4) Broader applicability—the hybrid algorithm delivered superior performance across unimodal, multimodal, separable, and non-separable function types.

6.2 Limitations

The framework has several limitations that warrant further investigation: (1) Computational overhead—the sequential execution of five algorithms incurs a computational cost approximately 4.2 times greater than that of a single algorithm; (2) Fixed weight allocation—the predetermined weights assigned to the constituent algorithms may not be optimal for all problem categories; (3) Restricted evaluation scope—the assessment was limited to unconstrained benchmark functions; (4) Scalability concerns—performance in extremely high-dimensional settings ($d > 100$) requires more study.

7. CONCLUSIONS

This study presented a novel hybrid optimization framework integrating five metaheuristic algorithms: EJA, IWOA, ADE, APSO, and MGWO are integrated. Empirical evaluation on ten benchmark functions demonstrated an average improvement of 94.7% in solution quality and a convergence speed increase of up to 47.1% relative to the individual constituent algorithms.

Based on the findings of this study, the following design principles are proposed for the development of future hybrid metaheuristic frameworks: (1) Complementarity principle:

constituent algorithms should be selected to exhibit distinct search mechanisms (e.g., gradient-free, spiral, differential, social, and hierarchical), rather than similar ones; (2) Weight allocation principle: algorithms with stronger intensification characteristics should be assigned higher weights within the integration scheme; (3) Sequencing principle: algorithms should be ordered from exploration-dominant to exploitation-dominant within each optimization iteration; (4) Diversity preservation principle: at least one algorithm that maintains population diversity throughout the optimization process should be included.

Future research directions include adaptive weight adjustment based on real-time performance feedback, extension to constrained optimization problems, application to real-world engineering design challenges, parallel implementation to reduce computational overhead, and theoretical analysis of convergence guarantees.

ACKNOWLEDGMENT

The authors gratefully acknowledge the Ministry of Education, Iraq, and the University of Babylon for their support of this research.

REFERENCES

- [1] Katoch, S., Chauhan, S.S., Kumar, V. (2021). A review on genetic algorithm: Past, present, and future. *Multimedia Tools and Applications*, 80: 8091-8126. <https://doi.org/10.1007/s11042-020-10139-6>
- [2] Mohamed, A.W., Hadi, A.A., Mohamed, A.K. (2021). Differential evolution mutations: Taxonomy, comparison and convergence analysis. *IEEE Access*, 9: 68629-68662. <https://doi.org/10.1109/ACCESS.2021.3077242>
- [3] Schwefel, H.P. (1993). *Evolution and Optimum Seeking*. John Wiley & Sons.
- [4] Adam, S.P., Alexandropoulos, S.A.N., Pardalos, P.M., Vrahatis, M.N. (2019). No free lunch theorem: A review. *Approximation and Optimization*, 145: 57-82.

- https://doi.org/10.1007/978-3-030-12767-1_5
- [5] Abdel-Basset, M., Abdel-Fatah, L., Sangaiah, A.K. (2018). Metaheuristic algorithms: A comprehensive review. In *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*, pp. 185-231. <https://doi.org/10.1016/B978-0-12-813314-9.00010-4>
- [6] Rao, R.V., Saroj, A. (2017). A self-adaptive multi-population based Jaya algorithm for engineering optimization. *Swarm and Evolutionary Computation*, 37: 1-26. <https://doi.org/10.1016/j.swevo.2017.04.008>
- [7] Gharehchopogh, F.S., Gholizadeh, H. (2019). A comprehensive survey: Whale Optimization Algorithm and its applications. *Swarm and Evolutionary Computation*, 48: 1-24. <https://doi.org/10.1016/j.swevo.2019.03.004>
- [8] Das, S., Mullick, S.S., Suganthan, P.N. (2016). Recent advances in differential evolution—An updated survey. *Swarm and Evolutionary Computation*, 27: 1-30. <https://doi.org/10.1016/j.swevo.2016.01.004>
- [9] Jordehi, A.R., Jasni, J. (2015). Parameter selection in particle swarm optimisation: A survey. *Journal of Experimental and Theoretical Artificial Intelligence*, 25(4): 527-542. <https://doi.org/10.1080/0952813X.2013.782348>
- [10] Faris, H., Aljarah, I., Al-Betar, M.A., Mirjalili, S. (2018). Grey wolf optimizer: A review of recent variants and applications. *Neural Computing and Applications*, 30: 413-435. <https://doi.org/10.1007/s00521-017-3272-5>
- [11] Lambora, A., Gupta, K., Chopra, K. (2019). Genetic algorithm—A literature review. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing*, pp. 380-384. <https://doi.org/10.1109/COMITCon.2019.8862255>
- [12] Price, K., Storn, R.M., Lampinen, J.A. (2005). *Differential Evolution: A Practical Approach to Global Optimization*. Springer.
- [13] Beyer, H.G., Schwefel, H.P. (2002). Evolution strategies: A comprehensive introduction. *Natural Computing*, 1(1): 3-52. <https://doi.org/10.1023/A:1015059928466>
- [14] Bonyadi, M.R., Michalewicz, Z. (2017). Particle swarm optimization for single objective continuous space problems: A review. *Evolutionary Computation*, 25(1): 1-54. https://doi.org/10.1162/EVCO_r_00180
- [15] Korashy, A., Kamel, S., Youssef, A.R., Jurado, F. (2019). Modified water cycle algorithm for optimal direction overcurrent relays coordination. *Applied Soft Computing*, 74: 10-25. <https://doi.org/10.1016/j.asoc.2018.09.020>
- [16] Karaboga, D., Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, 39(3): 459-471. <https://doi.org/10.1007/s10898-007-9149-x>
- [17] Suman, B., Kumar, P. (2006). A survey of simulated annealing as a tool for single and multiobjective optimization. *Journal of the Operational Research Society*, 57(10): 1143-1160. <https://doi.org/10.1057/palgrave.jors.2602068>
- [18] Rashedi, E., Nezamabadi-Pour, H., Saryazdi, S. (2009). GSA: A gravitational search algorithm. *Information Sciences*, 179(13): 2232-2248. <https://doi.org/10.1016/j.ins.2009.03.004>
- [19] Kaveh, A., Talatahari, S. (2010). A novel heuristic optimization method: Charged system search. *Acta Mechanica*, 213(3): 267-289. <https://doi.org/10.1007/s00707-009-0270-4>
- [20] Rao, R.V., Savsani, V.J., Vakharia, D.P. (2011). Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design*, 43(3): 303-315. <https://doi.org/10.1016/j.cad.2010.12.015>
- [21] Rao, R.V. (2016). Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems. *International Journal of Industrial Engineering Computations*, 7(1): 19-34. <https://doi.org/10.5267/j.ijiec.2015.8.004>
- [22] Satapathy, S.C., Naik, A. (2016). Social group optimization (SGO): A new population evolutionary optimization technique. *Complex & Intelligent Systems*, 2(3): 173-203. <https://doi.org/10.1007/s40747-016-0022-z>
- [23] Xu, H.R., Deng, Q.W., Zhang, Z.Y., Lin, S.K. (2025). A hybrid differential evolution particle swarm optimization algorithm based on dynamic strategies. *Scientific Reports*, 15: 4518. <https://doi.org/10.1038/s41598-024-82648-5>
- [24] Das, S., Abraham, A., Konar, A. (2008). Particle swarm optimization and differential evolution algorithms: Technical analysis, applications and hybridization perspectives. In *Advances of Computational Intelligence in Industrial Systems*, pp. 1-38. https://doi.org/10.1007/978-3-540-78297-1_1
- [25] Kaveh, A., Bakhshpoori, T. (2016). A new metaheuristic for continuous structural optimization: Water evaporation optimization. *Structural and Multidisciplinary Optimization*, 54(1): 23-43. <https://doi.org/10.1007/s00158-015-1396-8>
- [26] Buch, H., Trivedi, I.N., Jangir, P. (2017). Moth flame optimization to solve optimal power flow with non-parametric statistical evaluation validation. *Cogent Engineering*, 4(1): 1286731. <https://doi.org/10.1080/23311916.2017.1286731>
- [27] Molina, D., Latorre, A., Herrera, F. (2018). SHADE with iterative local search for large-scale global optimization. In *2018 IEEE Congress on Evolutionary Computation, Rio de Janeiro, Brazil*, pp. 1-8. <https://doi.org/10.1109/CEC.2018.8477755>
- [28] Jamil, M., Yang, X.S. (2013). A literature survey of benchmark functions for global optimization problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, 4(2): 150-194. <https://doi.org/10.1504/IJMMNO.2013.055204>
- [29] Loshchilov, I., Hutter, F. (2016). CMA-ES for hyperparameter optimization of deep neural networks. *arXiv preprint arXiv:1604.07269*. <https://doi.org/10.48550/arXiv.1604.07269>
- [30] Rosenbrock, H.H. (1960). An automatic method for finding the greatest or least value of a function. *The Computer Journal*, 3(3): 175-184. <https://doi.org/10.1093/comjnl/3.3.175>
- [31] Ackley, D.H. (1987). *A Connectionist Machine for Genetic Hillclimbing*. Kluwer Academic Publishers.
- [32] Schwefel, H.P. (1981). *Numerical Optimization of Computer Models*. John Wiley & Sons.
- [33] Michalewicz, Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Science & Business Media.