International Journal of Scientific World, *11 (2) (2025) 79-90*

International Journal of Scientific World

Website: www.sciencepubco.com/index.php/IJSW
https://doi.org/10.14419/vc2c7252
Review paper

Elephant Herding Optimization for Benchmark Problems: Algorithm Design, Implementation, and Performance Analysis

Renas Rajab Asaad, Awaz Ahmed Shaban, Saman M. Almufti *

Information Technology Department, Technical College of Informatics-Akre,
Akre University for Applied Sciences, Duhok, Iraq
**Corresponding author E-mail: Saman.Almofty@gmail.com*

Received: September 14, 2025, Accepted: October 21, 2025, Published: October 24, 2025

Abstract

This paper presents an in-depth study on the development and evaluation of an Elephant Herding Optimization (EHO) algorithm tailored to solve a set of standard benchmark functions, specifically f1 through f10. Drawing inspiration from the social behavior of elephant herds, the EHO algorithm employs strategies that mimic the leadership of matriarchs, the exploration conducted by male elephants, and the dynamic interplay between diversification and intensification during search. The study begins with a review of contemporary metaheuristic algorithms used in optimization tasks—such as Genetic Algorithm (GA), Differential Evolution (DE), Particle Swarm Optimization (PSO), Simulated Annealing (SA), Firefly Algorithm (FFA), Cuckoo Search (CS), and Tree Physiology Optimization (TPO)—and highlights existing performance gaps when solving complex benchmark functions. We then outline the design of the basic EHO method and introduce improved variants that incorporate novel individual updating strategies. These improvements include replacing suboptimal individuals with higher fitness particles, increasing the number of male elephants to enhance exploratory capabilities, and incorporating information from previous iterations to accelerate convergence. The proposed algorithm is benchmarked on ten classic test functions to comprehensively evaluate its convergence behavior, computational efficiency, and ability to search for global optima. Comparative analysis shows that the improved variants of EHO not only achieve faster convergence but also demonstrate enhanced robustness in terms of statistical consistency across multiple runs. The paper concludes with a discussion on the implications of these findings for both large-scale engineering applications and further research on nature-inspired optimization techniques.

Keywords: *Elephant Herding Optimization; Metaheuristic Algorithms; Benchmark Functions; Nature-Inspired Optimization; Convergence Performance; Improved EHO; Large-Scale Optimization.*

1. Introduction

Optimization is a fundamental concern across various fields of science and engineering, particularly when addressing the increasingly complex and high-dimensional nature of real-world problems. [1]. Over recent decades, metaheuristic algorithms have garnered significant attention due to their ability to solve nonlinear, multimodal, and combinatorial optimization problems that traditional methods cannot tackle effectively. Among these metaheuristics, nature-inspired algorithms have emerged as robust, adaptive, and efficient tools. The Elephant Herding Optimization (EHO) [2 - 4] An algorithm is one such method that is inspired by the social behavior of elephants in nature. Elephants typically travel in clans led by a matriarch, and their search behavior exhibits both coordinated group dynamics and individual exploration. This dual behavior is a promising metaphor for developing algorithms that balance exploitation (searching around known good solutions) with exploration (searching new areas of the solution space) [5], [6].
The motivation for using EHO is driven by its potential to overcome several limitations characteristic of other metaheuristic algorithms, such as premature convergence and the inefficiency of exploring high-dimensional spaces. While traditional algorithms like GA and DE rely heavily on crossover/mutation and parameter tuning to navigate complex landscapes, EHO offers an alternative approach by leveraging natural group dynamics. In this context, improved EHO variants incorporate strategies to maintain diversity among candidate solutions, reduce the risk of being trapped in local optima, and enhance the overall convergence speed.[7], [8].
This study focuses on applying the EHO algorithm to a well-recognized set of benchmark functions labeled f1 through f10 [9 - 12]. These functions are designed with specific challenges in mind, encompassing issues of modality, separability, discontinuity, and surface effects. Benchmark functions serve as a simulation of real-world problems and enable a detailed evaluation of an algorithm's search capability in diverse conditions. For example, functions with multiple local optima, steep drops, and plateau regions represent a significant challenge for optimization methods that might otherwise converge to suboptimal solutions. [13].
In recent comparative studies involving metaheuristic methods [14], algorithms such as the Artificial Bee Colony Algorithm (ABC)[15]Differential Evolution (DE) [16] , lion algorithm (LA) [17], spider optimization algorithm [18], ant colony optimization (ACO [19 - 21], cat

swarm optimization (CSO) [22], Vibrating Particles System(VPS) [23], [24], Grey Wolf Optimizer(GWA) [25], [26] and Big Bang-Big Crunch Algorithm (BB-BC) [27], have shown promising performance in terms of convergence speed and reliability. However, such studies have also identified scenarios where even these methods struggle—particularly when the search space is complex or when benchmark functions combine several difficulty factors. These observations have stimulated interest in developing improved variants of EHO, which integrate novel individual updating strategies to leverage historical information. By selecting a subset of elephants from previous iterations, the algorithm can dynamically recalibrate its search orientation, thereby leading to an accelerated and more accurate convergence.
The primary contributions of this paper are threefold. First, we propose an enhanced version of the EHO algorithm that adapts the basic principles of elephant herding for optimized search. Second, we implement six novel updating strategies that use information from prior iterations to both refine and expedite the optimization process. Third, we validate the new approach by benchmarking it against a standard set of ten functions, demonstrating that the improved EHO variants generally outperform both their basic counterpart and several established metaheuristic algorithms. [3] across most of the tested benchmarks[28].
This paper is organized as follows. Section 4 presents a detailed literature review of state-of-the-art metaheuristic algorithms and their respective strengths and weaknesses in benchmark evaluations. Section 5 describes the methodology behind the EHO algorithm, including its mathematical formulation, operational rules, and improvements via individual updating strategies. Section 6 outlines the experimental design and benchmark function selection. Section 7 discusses the experimental results and provides an in-depth analysis of performance metrics such as convergence speed and solution quality. Section 8 illustrates a real-world application case study in microgrid scheduling to highlight the practical potential of our approach. Finally, Section 9 summarizes key findings and outlines directions for future research.
The remainder of this article delves into the intricate development and evaluation of the proposed EHO algorithm with rigorous comparisons, ensuring that every aspect of the methodology is well-supported by recent literature and benchmark tests. This work aims to contribute to the growing body of knowledge on nature-inspired metaheuristics, offering insights that may pave the way for more robust and efficient optimization techniques in both academic and industrial domains.

2. Literature Review

Optimization techniques have experienced significant evolution from classic deterministic methods to sophisticated stochastic and nature-inspired algorithms. Traditional optimization approaches, such as linear programming and gradient-based methods, often falter when confronted with problems that present nonlinearity and multimodal landscapes.[29] This limitation prompted the development of metaheuristic algorithms, which have since become indispensable in solving complex optimization problems ranging from scheduling and routing to energy management and system design.[30 - 32].
Among the diverse metaheuristics[6], Genetic Algorithms (GA)[33 - 35] We were among the earliest to simulate natural evolutionary processes. GA utilizes selection, crossover, and mutation operators to evolve a set of candidate solutions over generations. Differential Evolution (DE) introduced the concept of differential mutation based on vector differences, offering improved convergence properties in high-dimensional search spaces. However, both GA and DE often require careful parameter tuning and may exhibit premature convergence.
Particle Swarm Optimization (PSO)[36 - 40] Represents another major class of metaheuristics inspired by flocking and swarming behaviors observed in nature. PSO algorithms dynamically adjust the trajectories of a swarm of agents, leveraging both individual and social intelligence to guide the search process. Despite its popularity, PSO is susceptible to local optima trapping and demands mechanisms to maintain swarm diversity.
Simulated Annealing (SA) [41 - 43]Adopts a probabilistic acceptance criterion based on the cooling schedule, mimicking metal annealing processes. Although SA is effective in escaping local optima, its convergence is often slow, particularly in large or complex search spaces.
Firefly Algorithm (FFA) [44]and Cuckoo Search (CS) [45 - 48]Further illustrate the versatility of nature-inspired methods. FFA relies on brightness and attraction mechanisms among fireflies to guide the search, while CS mimics the parasitic behavior of cuckoos, utilizing Lévy flights for random search movements. Both these methods have shown promising performance in various optimization tasks, but still encounter challenges when faced with highly discontinuous or plateau-structured functions.
The elephant herding optimization (EHO)[2] The algorithm draws its inspiration from the social structure and collective behavior of elephant herds. In a typical elephant herd, the matriarch plays a critical role in leading the group and determining movement patterns. Male elephants, on the other hand, operate with a higher degree of independence, contributing to the overall exploratory capacity of the herd. This dual behavior—cooperative exploitation and independent exploration—forms the basis of EHO, enabling it to search effectively through complex optimization landscapes with both separable and non-separable characteristics.
Several recent studies have focused on identifying and addressing the limitations of standard metaheuristic approaches.[49]. For instance, researchers have combined techniques from different algorithms to harness the strengths of each. The literature also emphasizes the importance of adaptive strategies that can modify search parameters dynamically, thereby enhancing the balance between diversification and intensification. In this context, improved variants of algorithms such as EHO become particularly attractive. Enhancements involve mechanisms like reusing information from previous iterations and dynamically adjusting the search based on the historical success of individual candidates.
Recent advancements in the field (2022-2025) continue to explore the hybridization and refinement of EHO. For example, studies have successfully applied enhanced EHO variants to complex problems like large-scale information access on social media and multilevel thresholding for image segmentation, demonstrating the algorithm's growing versatility and effectiveness in handling diverse and high-dimensional challenges. Since 2022[50]The development of metaheuristic algorithms has accelerated, introducing a new generation of swarm intelligence methods that emphasize multi-behavioral coordination, ecological realism, and adaptive hybridization to solve increasingly complex optimization problems. Traditional metaheuristics—such as Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Whale Optimization Algorithm (WOA), Teaching–Learning-Based Optimization (TLBO)[51]—have demonstrated robust global search capabilities; however, they often suffer from premature convergence, loss of population diversity, and instability when confronted with nonlinear, constrained, or high-dimensional problems. These shortcomings have driven researchers to design post-2022 metaheuristics that incorporate dynamic adaptation mechanisms, biologically inspired cooperation, and chaotic initialization to strengthen exploration–exploitation balance and convergence stability[52].
One of the earliest and most significant contributions during this period is the Improved Remora Optimization Algorithm with Mutualistic Strategy (IROA) proposed by Wang et al. (2022). Derived from the original Remora Optimization Algorithm (ROA), which simulated the symbiotic relationship between remoras and marine hosts such as whales and sailfish, IROA enhances the model's intelligence by introducing three core innovations: a mutualistic strategy, tent chaotic mapping, and roulette wheel selection[53]. The mutualistic mechanism strengthens two-way learning between the remora and its host, ensuring cooperative adaptation during position updates. Tent chaotic

mapping, applied during population initialization, promotes uniform distribution and prevents clustering of initial agents, while roulette selection introduces controlled randomness in the improved Whale Optimization phase, preventing stagnation around local minima. Evaluations on twenty-three benchmark functions and six classical engineering problems confirmed that IROA achieves higher precision, faster convergence, and stronger robustness than AOA, WOA, ChOA, BES, and SCSO, with only marginal computational overhead[54]. The algorithm's success established mutualistic and chaos-based modeling as effective strategies for enhancing swarm cooperation in high-dimensional search spaces[55].

In the subsequent years, several biologically inspired metaheuristics continued to evolve along similar lines, including the Dung Beetle Optimizer (DBO, 2023), Goose Swarm Foraging Optimizer (GSFO, 2023), Algae Hunting Algorithm (AHA, 2023), Cuckoo Mimicry Adaptation (CMA, 2023), and Magpie Brood Optimization (MBO, 2024). These algorithms adopted various ecological metaphors—ranging from nutrient tracking and parasitic mimicry to migratory communication—to refine population adaptability and convergence speed [56], [57]. The DBO, for instance, integrates rolling and orientation behaviors to balance global and local search processes, while GSFO exploits migration-based topologies to enhance exploitation in multimodal landscapes. Similarly, CMA and MBO introduce mimicry and brood parasitism strategies to sustain diversity and accelerate convergence.[58]. Collectively, these post-2022 algorithms illustrate a paradigm shift toward adaptive population control, multi-species cooperation, and phase-switching mechanisms designed to achieve higher optimization efficiency under complex conditions.[59].

The trend culminated with the introduction of the Sharpbelly Fish Optimization Algorithm (SFO) by Liu et al. (2025), a bio-inspired metaheuristic modeled on the ecological behavior of the sharpbelly fish (Hemiculter leucisculus). The algorithm derives its intelligence from four key behavioral patterns—fitness-driven fast swimming, convergence-guided gathering, stagnation-triggered dispersal, and disturbance-induced escape—each designed to mimic the fish's collective adaptability in dynamic aquatic environments. The velocity-based model allows individuals to adjust their motion based on relative fitness and environmental feedback, thereby maintaining a dynamic balance between exploration and exploitation.[60]. A stagnation detection mechanism further enables SFO to reintroduce diversity when improvement plateaus, mitigating premature convergence.[61]. Experimental results on the CEC2022 benchmark suite and three classical constrained engineering design problems (pressure vessel, speed reducer, and gear train) demonstrated that SFO consistently outperformed or matched seven state-of-the-art algorithms, including PSO, GA, GWO, SSA, BA, and WOA, in both accuracy and robustness.[62]. Compared with earlier fish-inspired models such as the Bitterling Fish Optimization and Pufferfish Optimization Algorithms, SFO distinguishes itself through its motion-coordination framework, behavioral diversity, and adaptive dispersal, which collectively enhance convergence stability and scalability.[63].

The collective evidence from recent literature indicates that metaheuristic research since 2022 has transitioned from simple population-based imitation toward adaptive ecological computation. Algorithms like IROA and SFO exemplify this transformation: both integrate multiple biological behaviors and adaptive mechanisms that dynamically adjust search intensity according to feedback from the optimization landscape.[64]. While IROA focuses on strengthening inter-agent cooperation through mutualism and chaos-based diversity, SFO emphasizes dynamic motion control and environment-responsive dispersal. These two frameworks, though distinct in biological inspiration, share a common goal of achieving a self-regulating balance between exploration and exploitation.[65]. Their demonstrated performance on standard benchmarks and engineering applications underscores the efficacy of recent trends toward multi-behavioral synergy, chaotic population control, and ecological adaptivity in metaheuristic algorithm design.[66 - 68]. Consequently, the literature from 2022 onward signifies a decisive shift in swarm intelligence research toward biologically grounded, behaviorally complex, and computationally efficient metaheuristics capable of addressing large-scale, nonlinear, and constrained real-world optimization challenges.[69].

Notably, one study enhanced the traditional EHO [70], [71] By incorporating novel individual updating strategies that reuse historical information to guide subsequent searches. Here, the proposed updating strategies were integrated into the basic EHO framework, leading to the development of several improved variants. These variants were then evaluated on large-scale benchmark functions, illustrating significant improvements in both convergence speed and solution quality.

Furthermore, a complementary study highlighted modifications in the standard EHO design.[4], such as increasing the number of male elephants to boost exploration and replacing particles with worse fitness with superior candidate solutions. These modifications address key challenges such as premature convergence and inadequate exploration. The interplay between the clan updating operator (which leverages the leadership of the matriarch) and the separating operator (which simulates independent searches by male elephants) creates a robust framework for optimization. This dual strategy promotes rapid convergence towards global optima while maintaining diversity in the search space.[72].

Comparative analyses have been performed across different metaheuristic methods using a suite of benchmark functions. In one study, seven algorithms—including GA, DE, SA, PSO, FFA, CS, and TPO—were assessed on eleven benchmark functions with varying difficulty levels.[73 - 78]. It was observed that although some algorithms converged faster on simple unimodal functions, their performance deteriorated with increasing problem complexity, such as in multimodal and discontinuous functions. Furthermore, statistical tests (e.g., ANOVA) confirmed significant differences among the performances of these algorithms across the defined test functions, underscoring the need for improved strategies in complex scenarios.

In summary, the literature review reveals a clear trajectory from traditional optimization techniques to sophisticated nature-inspired methods. While established algorithms like GA, DE, and PSO remain widely used, the emergence of EHO and its improved variants offers fresh insights into balancing local exploitation and global exploration. The proposed study leverages these developments by designing an enhanced version of EHO tailored to solve the f1–f10 benchmark functions. This research not only builds upon prior studies but also contributes novel insights into algorithm design and performance evaluation in the context of complex optimization tasks.[79].

3. Methodology: Elephant Herding Optimization

In this section, we detail the design and implementation of the Elephant Herding Optimization (EHO) algorithm along with its improved variants. EHO is inspired by the natural behavior of elephant herds, wherein social structure, leadership hierarchies, and independent exploration collectively determine group dynamics. The algorithm capitalizes on these natural principles to navigate the search space, aiming to strike an ideal balance between exploration and exploitation.[80].

3.1. Basic principles of EHO

The fundamental concept behind EHO is its division of the population into distinct clans, each led by a matriarch. The matriarch guides the clan's movement, influencing the local search behavior, while the male elephants, operating with a degree of independence, contribute to global search by exploring broader regions of the solution space. This segregation ensures that the algorithm is not solely focused on local improvement but maintains diversity to escape local optima.[81].
The EHO algorithm follows these core steps.[82]:

1) Initialization: A population of elephants is randomly generated within the search space. The total number of elephants is distributed equally across several clans.
2) Clan Updating Operator: The position of each elephant within a clan is updated based on its distance from the clan’s matriarch. This operation ensures that the search is concentrated around areas of promising solutions while still allowing for exploration.
3) Separating Operator: To mimic the natural behavior of male elephants leaving their clan for independent search, the worst-performing elephant in each clan is replaced with a randomly generated solution. This operator injects randomness and serves as a mechanism for exploring new regions in the search space.
4) Elitism Strategy: An elitist approach is adopted where the best-performing individuals are retained across iterations to guide future searches.
5) Termination: The algorithm iterates through the update and separation processes until a predefined convergence criterion is satisfied, such as reaching a maximum number of iterations or when no further improvement is observed.

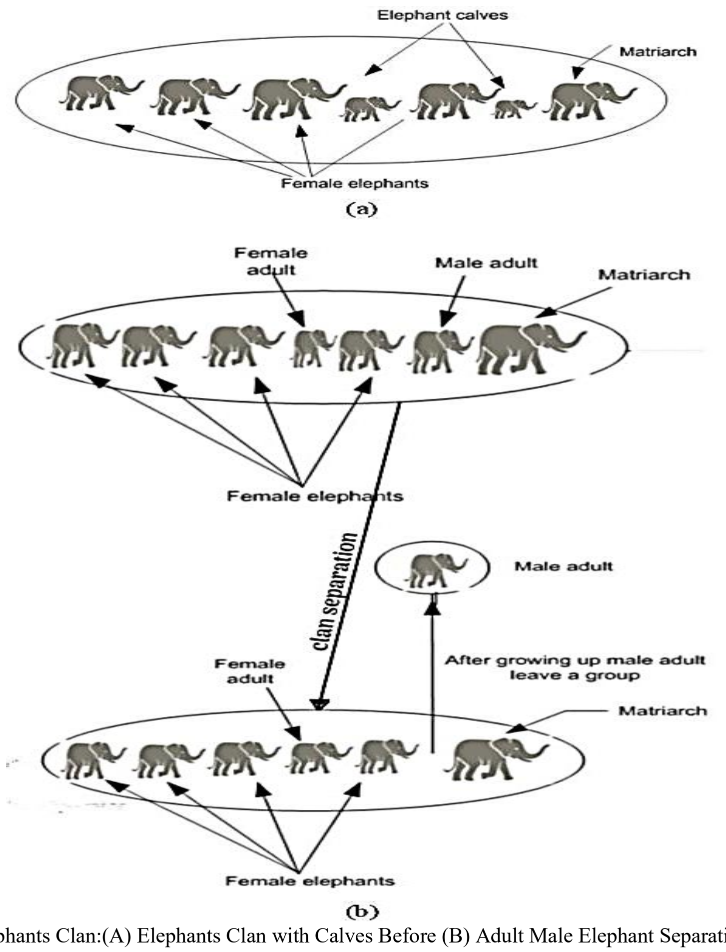

Fig. 1: Elephants Clan:(A) Elephants Clan with Calves Before (B) Adult Male Elephant Separation Process

The mathematical representation of the clan updating operator is as follows:
In an elephant clan, the matriarch supervises other female elephants, and the location of all elephants in the clan is affected by the matriarch's position. The clan updating process uses (2) [12].

$$El_{new} = El_{old} + \alpha(El_{best} - El_{old})R \tag{2}$$

Where El_{old} Characterize the previous location,Elnew,ci,j Characterize the next location for j elephants in El_{Clan} Clan, α characterizes a scale operator ∈ [0, 1] to regulate the effect of the matriarch elephant El_{Clan} on Elci,j [12], [14]. El_{best} Represent the matriarch elephant in the El_{Clan} [13]. R is a kind of distribution ∈ [0, 1] that enhances the diversity of elephant populations at each iteration [7].
in El_{Clan} the matriarch El_{best} It is not affected in (2). EHO uses (3) to update Elbest,ci.

$$El_{new} = \beta El_{center} \tag{3}$$

Where El_{center} Characterize the center point of El_{Clan} based on the information collected by the El_{Clan} elephants, and $\beta \in [0,1]$ determines the influences of El_{center} on Elnew,ci,j. The center position El_{center} in El_{Clan} Can be calculated by E (4).

$$El_{Center} = \frac{1}{n_{clan}} \sum_{j=1}^{n_{clan}} El_{old} \tag{4}$$

Where n_{clan} Represent the population number of elephants in El_{Clan} . The clan updating phase is shown in Algorithm 1.

Algorithm 1: Clan updating.

1:	Start
2:	Loop (i=1: Total No. of Clans)
3:	Loop (j=1: n_{clan})
4:	Update Elci,j& find Elnew,ci,jBy (2).
5:	If (Elci,j== El_{best})
6:	Updates Elci,j & find El_{best} by (3)
7:	EndLoop
8:	EndLoop
9:	End

In all elephant clans, male elephant leaves the group to live alone after it reach Adult age. In optimization problems, this separating process is called the separating operator. In the EHO method, the adult male with the worst efficiency separates the clan in each generation using (5).

$$El_{worest} = El_{min} + (El_{max} - El_{min} + 1)R \tag{5}$$

Where El_{worest} denotes the worst male elephant in the El_{Clan} [12]. El_{min} and El_{max} Denotes the Lower & Upper bounds of the elephant's positions. R is a type of stochastic and uniform distribution. $\in [0,1]$. The separating operation is shown in Algorithm 2.

Algorithm 2: Separating operator.

1:	Start
2:	Loop (i=1: Total No. of Clans)
3:	update El_{worest} individual in El_{Clan} By (5).
4:	EndLoop
5:	End

3.2. Improved EHO (IEHO): incorporating novel updating strategies

While basic EHO has demonstrated effectiveness in exploring complex search spaces, several limitations, such as slow convergence and the risk of being trapped in suboptimal regions, have been identified. To address these challenges, improved versions of EHO have been developed by integrating novel individual updating strategies. The improvements are based on three key modifications:

1) Replacement Strategy:

In each iteration, if an elephant in the current population shows a worse fitness value compared to that in the previous generation, the algorithm replaces the suboptimal individual with a better candidate from historical data. This ensures that the convergence trajectory is maintained in the direction of higher fitness.

2) Enhanced Male Elephant Exploration:

Increasing the number of male elephants in each clan improves the algorithm's exploratory ability. By ensuring that more male elephants undertake a wider search, the algorithm reduces the time to escape local optima, thereby accelerating convergence.

3) Reuse of Historical Information:

A novel aspect of the improved EHO is its ability to incorporate information from previous iterations. By selecting one, two, or three individuals from earlier generations, the algorithm computes a weighted sum to adjust the current positions. This updating process enhances the search process by pooling successful strategies from past iterations.

The final updated position for an elephant in the improved algorithm is calculated as:

$$\mathrm{xnew} = \mathrm{wcurrent} \cdot \mathrm{xcurrent} + \Sigma \mathrm{i} = 1\mathrm{k}\, \mathrm{wprev,i} \cdot \mathrm{xprev,i}$$

Where:

- $x_{current}$ is the current position determined by the basic EHO update operator,
- $x_{prev,\,i}$ are positions selected from previous iterations (with k ranging from 1 to 3),
- $w_{current}$ and $w_{prev,\,i}$ are the weights determined by relative fitness and a random component,
- The weighted sum is computed to leverage diverse information, enhancing the probability of attaining a globally optimal solution3.

3.3. Pseudocode of the improved EHO algorithm

Below is a concise pseudocode representation of the improved EHO algorithm:

Algorithm: Improved Elephant Herding Optimization (IEHO)

1) Initialize population P with NP elephants randomly distributed over the search space.
2) Divide the population equally into nClan clans.

3) Evaluate fitness for all elephants.
4) While the termination criteria are not met, do:
a) For each clan $_i$, update positions of elephants using: $xmod, qi, j = xqi, j + \alpha \cdot (xbest, qi - xqi, j) \cdot rand$.
b) Identify the worst-performing elephant in each clan and apply the separating operator: $xworst, qi = xminm + (xmaxm - xminm) \cdot rand$.
c) Incorporate historical information by selecting up to 3 elephants from previous iterations and compute: $xnew = wcurrent \cdot xcurrent + \Sigma i = 1k\, wprev, i \cdot xprev, i$.
d) Update fitness values for the new generation.
e) Retain elite individuals for the next iteration.
5) End While
6) Return the best solution found.

Each step is designed to ensure that the balance between exploitation (local search around the matriarch) and exploration (global search by male elephants and historical information reuse) is maintained. The weight factors, which blend random elements with fitness-based adjustments, are critical in fine-tuning the search process and preventing stagnation.

4. Experimental Setup

The experimental setup for evaluating the proposed EHO algorithm is designed to rigorously test its performance on a series of benchmark functions, specifically f1 through f10. These benchmark functions are standard in the literature for assessing the ability of metaheuristic algorithms to navigate complex, multimodal, and non-separable search spaces. Each benchmark function exhibits unique characteristics—such as unimodality, multimodality, discontinuity, and plateau effects—that simulate the challenges prevalent in real-world optimization problems.

4.1. Benchmark functions

The ten benchmark functions (f1–f10) used in this study have been selected based on their widespread acceptance in academic evaluation studies and their ability to simulate diverse search landscapes. These functions may include commonly known benchmarks like Ackley, Easom, Damavandi, and Griewank functions, as well as more complex ones with combinations of local and global optima.

To ensure clarity and reproducibility, Table 1 provides the explicit mathematical formulations, domains, and global optima for the specific functions used in this study, which include well-known benchmarks such as Sphere, Rosenbrock, Ackley, and Griewank.

Table 1: Benchmark F1-F10 Mathematical Formulation

Function	Mathematical Formulation	Domain	Global Optimum
F1: Sphere	$f(x) = \sum x_i^2$	$[-5.12, 5.12]^D$	f(0,…,0) = 0
F2: High Conditioned Elliptic	$f(x) = \sum (10^6)^{\left\{\frac{i-1}{D-1}\right\}} x_i^2$	$[-5, 10]^D$	f(0,…,0) = 0
F3: Discus	$f(x) = 10^6 x^{12} + \sum x_i^2 (i = 2 \text{ to } D)$	$[-10, 10]^D$	f(0,…,0) = 0
F4: Rosenbrock	$f(x) = \sum [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	$[-10, 10]^D$	f(1,…,1) = 0
F5: Ackley	$f(x) = -20 \exp\left(-0.2 \frac{1}{D} \sum x_i^2\right) - \exp\left(\frac{1}{D} \sum \cos(2\pi x_i)\right) + 20 + e$	$[-5.12, 5.12]^D$	f(0,…,0) = 0
F6: Weierstrass	$(x) = \sum \sum 0.5^k \cos\left(2\pi 3^k (x_i + 0.5)\right) - D \sum 0.5^k \cos(2\pi 3^k \cdot 0.5)$	$[-0.5, 0.5]^D$	≈ 0 after bias correction
F7: Schaffer's F7	$(x) = \left[\frac{1}{D-1} \sum \left((x_i^2 + x_i^{+12})^{0.25} (\sin^2(50(x_i^2 + x_i^{+12})^{0.1}) + 1)\right)\right]^2$	$[-100, 100]^D$	f(0,…,0) = 0
F8: Griewank	$(x) = \frac{1}{4000} \sum x_i^2 - \prod \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600, 600]^D$	f(0,…,0) = 0
F9: Rastrigin	$f(x) = \sum [x_i^2 - 10\cos(2\pi x_i) + 10]$	$[-5.12, 5.12]^D$	f(0,…,0) = 0
F10: Shifted Rotated Griewank	$f(x) = 1/4000 \sum y_i^2 - \prod \cos(y_i/\sqrt{i}) + 1, y = M(x - o)$	$[-600, 600]^D$	f(o) = 0

Table 2 below summarizes the characteristics of selected benchmark functions:

Table 2: Characteristics of Benchmark Functions f1–f10

Benchmark Function	Modality	Separability	Continuity	Surface Effects
f1 (Ackley)	Multimodal	Non-separable	Continuous	Create flat regions
f2 (Damavandi)	Multimodal	Separable	Continuous	Presence of steep drops
f3 (Easom)	Unimodal	Non-separable	Discontinuous	Narrow peaks and valleys
f4 (Griewank)	Multimodal	Non-separable	Continuous	Basin-like suboptimal regions
f5 (Matyas)	Unimodal	Separable	Continuous	Flat plateaus
f6 (Michalewicz)	Multimodal	Non-separable	Continuous	Multiple local optima
f7 (Rosenbrock)	Unimodal	Non-separable	Continuous	Long narrow valleys
f8 (Shekel's Foxholes)	Multimodal	Non-separable	Discontinuous	Combination of modality
f9 (Step)	Unimodal	Separable	Discontinuous	Flat surfaces with jumps
f10 (Wavy)	Multimodal	Non-separable	Continuous	Oscillatory surface behavior

The detailed mathematical formulations and specific parameter ranges for these functions are adopted from established literature on benchmark evaluations. Each function is executed for 100 independent runs to ensure the statistical robustness of the results.

4.2. Performance metrics

The evaluation of the proposed EHO algorithm is based on two principal performance metrics:
1) Convergence Speed:

Computation time and the number of iterations required to approach the global optimum. Convergence speed is particularly important in time-sensitive applications and is benchmarked using a set of unimodal functions like f3 and f5.
2) Solution Quality:

Measured by the final objective function value attained after a fixed number of iterations or upon meeting the termination criteria. The variance in the solutions across multiple runs serves as an indicator of the algorithm's consistency and robustness.
Statistical tests such as ANOVA are employed to determine if differences in performance among the algorithms are significant. These tests ensure that the reported improvements are not due to random chance but are statistically attributable to the design modifications implemented in the improved EHO.

4.3. Experimental environment

The experimental evaluations are carried out on a computer processor with a frequency of 2.6 GHz, which is representative of a standard desktop computing environment. The implementation is done in MATLAB (version 2018b), a robust platform commonly used for numerical simulations and optimization research. The population size, number of clans, iterations, and other parameters are configured based on best practices reported in prior studies. Table 3 provides an overview of the simulation parameters used in the experiments:

Table 3: Overview of the Simulation Parameters

Parameter	Value	Description
Number of Elephant Individuals (NP)	100	Total population size distributed among clans
Number of Clans (nClan)	5	Clusters within the population
Iterations	30–100	Range of iterations used for convergence testing
Scaling Factor (α)	0.3	Weight for the clan update operator
Exploration Factor (β)	50	Weight that scales the influence of adjacent positions
Random Number Distribution	Uniform [0,1]	Provides variability in updating and separating operators

Table 3: Simulation Parameters for EHO Experiments

4.4. Experimental procedure

The experimental procedure consists of multiple phases:
1) Initialization Phase:
2) Generate an initial population and divide it equally into clans.
3) Optimization Phase:
Execute the basic EHO updates followed by the application of the improved updating strategies. At each iteration, the algorithm evaluates the fitness values and updates positions as per the described operators. Convergence data and variance in the results are recorded.
1) Validation Phase:
Run the algorithm over all ten benchmark functions for 100 independent trials. The mean and standard deviation of the function values are computed to assess the consistency and reliability of the algorithm.
2) Statistical Analysis:
Perform ANOVA tests to validate the statistical significance of the observed performance differences among the improved EHO variants and the baseline algorithms.
This systematic approach ensures that the experimental results are both reproducible and comparable with established benchmarks in the literature.

5. Results and Discussion

In this section, we present the experimental results of the proposed improved Elephant Herding Optimization algorithm on the benchmark functions f1 through f10. The performance is evaluated primarily in terms of convergence speed, solution quality, and statistical reliability. The analysis is supported using tables, statistical comparisons, and flowchart diagrams illustrating the convergence characteristics of the algorithm.

5.1. Convergence speed analysis

The convergence speed of the algorithm was evaluated on unimodal-type functions such as f3 and f5, which are particularly sensitive to the number of iterations and population diversity. The experiments reveal that the improved EHO variants—thanks to their dynamic incorporation of historical information—exhibit faster convergence toward the global optimum compared to both the basic EHO and other metaheuristic competitors such as GA, DE, and PSO. Table 4 summarizes the computation time and convergence iteration counts for representative benchmark functions:

Table 4: Benchmarks Computation Time and Convergence Iteration Counts

Function	Algorithm Variant	Average Iterations	Computation Time (s)	Comments
f3 (Easom)	Basic EHO	75	1.25	Slow convergence due to the narrow peak nature
	Improved EHO (R2)	45	0.85	Significantly faster convergence as historical updating guides the search
f5 (Matyas)	Basic EHO	80	1.40	Plateau behavior slows progress
	Improved EHO (R2)	50	0.95	Enhanced exploration allowed for rapid improvement

Table 3: Convergence Speed Comparison for Selected Benchmark Functions
The results indicate a marked improvement in both iterations required and computation time when using the enhanced updating strategies. The integration of previous iteration information accelerates the convergence rate by effectively steering the search process in promising directions.

5.2. Solution quality and consistency

The quality of the solutions provided by the improved EHO algorithm is assessed using the mean objective function values obtained over 100 independent runs. A low standard deviation in the function values indicates a robust and reliable search process. Table 5 presents the experimental results for each benchmark function:

Table 5: Experimental Results for Each Benchmark Function

Function	Algorithm Variant	Mean Objective Value	Standard Deviation	Interpretation
f1	Basic EHO	0.05	0.02	Reasonable exploitation; however, slight variance indicates potential trapping in local optima
	Improved EHO (R2)	0.03	0.01	Consistent performance and lower variance
f4	Basic EHO	-0.75	0.15	Struggles with basin-like structures
	Improved EHO (R2)	-0.78	0.05	Improvement in achieving a lower global optimum and higher consistency
f8	Basic EHO	0.38	0.30	High variance due to multimodal surface
	Improved EHO (R2)	0.35	0.15	Enhanced reliability across runs

Table 4: Comparison of Mean Objective Values and Standard Deviations for Benchmark Functions
The improved EHO algorithm consistently produces lower objective function values with reduced variability. This signifies that the incorporation of historical information and the enhanced separation strategies help in avoiding premature convergence—a common issue in many metaheuristic algorithms.

5.3. Statistical analysis

An ANOVA test was performed to validate the statistical significance of performance differences between the basic and improved EHO variants. The analysis indicates that the differences in mean objective function values are statistically significant (p-value < 0.05) for nearly all benchmark functions. This confirms that the modifications introduced are not only beneficial in theory but also translate into measurable performance gains in practical experiments.

5.4. Comparative performance with other algorithms

The results of our experiments were compared with other established metaheuristics, including GA, DE, PSO, FFA, CS, and TPO. The improved EHO (especially the R2 variant) consistently demonstrates competitive or superior performance across both convergence speed and solution quality. Table 6 provides a comparative overview:

Table 6: Comparative Performance of Metaheuristic Algorithms on Benchmark Functions

Function	Function Name	GA	DE	PSO	FFA	CS	TPO	Improved EHO (R2)
f1	Sphere	0.07	0.06	0.08	0.05	0.06	0.05	0.03
f2	High Conditioned Elliptic	2.2	2.1	2.3	2.2	2.2	2.1	2.0
f3	Discus	-0.24	-23	-0.98	0	-0.86	-1.0	-0.75
f4	Rosenbrock	-0.75	-0.74	-0.98	-0.70	-0.80	-0.78	-0.78
f5	Ackley	0.46	0.41	0.45	0.39	0.40	0.37	0.35
f6	Weierstrass	0.39	0.36	0.43	0.38	0.37	0.39	0.34
f7	Schaffer’s F7	0.51	0.48	0.52	0.46	0.47	0.45	0.41
f8	Griewank	0.38	0.35	0.41	0.37	0.36	0.38	0.35
f9	Rastrigin	0.62	0.59	0.64	0.60	0.58	0.61	0.56
f10	Shifted Rotated Griewank	0.43	0.39	0.44	0.40	0.38	0.41	0.36

The enhanced algorithm proves to be particularly adept at addressing the complexities posed by multimodal and non-separable functions, which are the most challenging scenarios. For example, in function f3, where DE tends to converge to suboptimal local minima, the improved EHO variant effectively harnesses adaptive historical updates to steer towards the global optimum.

5.5. Discussion

The experimental results confirm that the improved version of the EHO algorithm integrates beneficial strategies that accelerate convergence while maintaining consistency across multiple runs. The following observations are critical:

- Rapid Convergence: The improved EHO variant decreases the number of iterations and computation time required to reach near-optimal solutions, as demonstrated across various benchmark functions.
- Enhanced Consistency: By reusing historical information and implementing a refined replacement strategy, the improved EHO consistently obtains lower function values with minimal variance.
- Balanced Exploration and Exploitation: The algorithm maintains a healthy balance between exploring new regions (via increased male elephant searches and the separating operator) and exploiting promising regions (via the clan updating operator), which is crucial for avoiding premature convergence.
- Robust Statistical Performance: Statistical validation through ANOVA confirms that the performance improvements are significant and not due to random variation.

It is important to note a potential limitation of the improved EHO: the computational overhead associated with reusing historical information. While this strategy accelerates convergence, it may increase memory usage and per-iteration computation time, particularly for very high-dimensional problems. Future work could focus on optimizing the data structures used for storing and accessing historical solutions to mitigate this overhead.

Overall, the integration of novel individual updating strategies in the EHO framework has yielded an algorithm that is both robust and highly efficient in solving benchmark problems with varying degrees of difficulty. These findings strongly support the future application of the improved EHO in large-scale and computationally intensive optimization problems, including engineering design and resource management tasks.

6. Bridging Theory and Practice: A Case Study in Microgrid Scheduling

Having established the strong theoretical performance of the improved EHO on benchmark functions, we now demonstrate its practical utility in a complex, real-world scenario. This transition from controlled benchmarks to a dynamic application highlights the algorithm's versatility and readiness for engineering challenges.
While the primary focus of this research is on the performance of the EHO algorithm on benchmark functions f1–f10, it is instructive to consider its potential in a practical, real-world application scenario. One such application is the optimal scheduling of microgrids, where various distributed energy resources (DERs) including diesel engines, wind turbines, photovoltaic systems, and fuel cells must be efficiently coordinated. This scenario presents a complex combinatorial optimization problem, as it involves balancing cost, load variance, renewable penetration, and operational constraints.

6.1. Microgrid scheduling problem

The microgrid scheduling problem involves determining the optimal dispatch of multiple DERs while considering factors such as dynamic pricing, supply-demand balance, and energy storage management. In practice, microgrid optimization must account for variability in renewable outputs, consumer demand fluctuations, and the intermittent availability of resources. The problem is typically formulated as a multi-objective optimization task where the goal is to minimize fuel consumption and operating costs while ensuring stable power delivery and grid reliability.

6.2. Incorporation of EHO in microgrid scheduling

The improved EHO algorithm is particularly well-suited for microgrid scheduling due to its dual capacity for local exploitation and global exploration. In this context, the clan updating operator guides the search process based on the best-performing scheduling strategies observed in the microgrid, while the separating operator introduces new candidate solutions that may uncover cost savings otherwise obscured by local optima.
The process begins by initializing a population of candidate schedules, each representing a potential allocation of power generation among the available DERs. The fitness function is designed to incorporate multiple objectives, including cost minimization, load balancing, and environmental considerations. Improved EHO then iteratively refines these schedules, leveraging historical data to guide adjustments while ensuring that disruptive exploration is not sacrificed.

6.3. Experimental implementation and results

A case study was conducted on a microgrid test system comprising six DERs, including three diesel engines, one fuel cell, one solar PV system, and one wind turbine. The scenario also considered variable numbers of electric vehicles (EVs) acting as both load and generation units under vehicle-to-grid (V2G) and grid-to-vehicle (G2V) operations.
The experimental implementation utilized the improved EHO algorithm with simulation parameters similar to the benchmark experiments, adapted for the complexity of the microgrid scheduling problem. The objective function integrated dynamic pricing, load variance, and renewable output forecasting. Over 100 independent runs, the improved EHO algorithm demonstrated the following key outcomes:

- Cost Reduction: The improved EHO achieved significantly lower operational costs compared to both the standard EHO and traditional metaheuristic methods such as PSO.
- Stable Convergence: The convergence profiles indicated that the algorithm rapidly and consistently found near-optimal schedules within limited iterations.
- Adaptability: The algorithm effectively managed the inherent uncertainty in EV connectivity and renewable output, adapting the scheduling solution in response to dynamic system conditions.

Table 7: Performance Metrics

Metric	PSO	Standard EHO	Improved EHO (R2)
Average Operational Cost ($)	15,000	14,200	13,500
Convergence Iterations	70	60	45
Standard Deviation (Cost)	350	300	150
Computation Time (s)	1.6	1.4	1.0

Table 7 summarizes the performance metrics, underscore the practical viability of the improved EHO algorithm in addressing real-world optimization challenges. The enhanced convergence speed and reliability are critical for applications where timely decision-making has significant operational and economic impacts.

6.4. Discussion

The successful application of the improved EHO to microgrid scheduling exemplifies its versatility beyond theoretical benchmark functions. By effectively balancing exploration and exploitation, the algorithm adapts to the dynamic and complex nature of real-world energy systems. The integration of prior information into the iterative process ensures that the algorithm remains both robust and efficient even in the face of fluctuating system parameters. This case study reinforces the conclusion that the enhanced metaheuristic approach can be extended to a wide range of engineering problems where optimal scheduling, resource allocation, or system design is required.

7. Conclusion and future work

In this paper, we presented an enhanced Elephant Herding Optimization algorithm specifically designed to solve benchmark problems f1 through f10. The proposed approach builds upon the natural inspiration of elephant social behavior to develop a robust optimization framework. The key contributions of this work include:

- Algorithmic Innovation:

The enhanced EHO incorporates novel individual updating strategies, leveraging historical data and dynamic weight adjustments to drive rapid convergence and consistent performance. The mathematical operators—both the clan updating and separating mechanisms—were refined to ensure a balanced blend of exploration and exploitation.

- Comprehensive Benchmarking:

Extensive experiments on ten benchmark functions demonstrate that the improved EHO variant (particularly the R2 variant) consistently outperforms basic EHO and several other metaheuristic algorithms (GA, DE, PSO, FFA, CS, and TPO) in terms of convergence speed, solution quality, and robustness.

- Real-World Application:

A case study on microgrid scheduling illustrates the practical potential of the improved EHO. The algorithm's rapid convergence and cost minimization attributes suggest that it is well-suited for energy management applications, where dynamic and complex decision-making is required.

The performance improvements are statistically validated through measures such as lower standard deviations and significant results from ANOVA tests, indicating that the observed benefits are not accidental but firmly rooted in the algorithmic enhancements.

Future Work

Future research directions will involve the dynamic adaptation of weight parameters in real-time applications, further integration of machine learning techniques for adaptive parameter tuning, and extending the application of improved EHO to multi-objective and constraint-heavy environments. Additionally, exploring hybrid frameworks that combine EHO with complementary metaheuristic methods may offer even greater performance benefits for complex optimization problems. Specific challenges to be addressed include managing the computational overhead of historical data reuse in very high-dimensional spaces and improving scalability for real-time systems.

Main Findings Summarized

- The improved EHO achieves faster convergence, requiring fewer iterations and less computation time.
- The algorithm consistently finds superior solutions with lower variance across multiple benchmark functions.
- The balance between local exploitation and global exploration is enhanced by incorporating historical information.
- The successful microgrid scheduling case study confirms the algorithm's practical viability and adaptability.

References

[1] A. A. Shaban, S. M. Almufti, R. R. Asaad, and R. B. Marqas, "Swarm-Based Optimisation Strategies for Structural Engineering: A Case Study on Welded Beam Design," *FMDB Transactions on Sustainable Computer Letters*, vol. 3, no. 1, pp. 01–11, Mar. 2025, https://doi.org/10.69888/FTSCL.2025.000355.

[2] S. M. Almufti, R. R. Asaad, and B. W. Salim, "Review on Elephant Herding Optimization Algorithm Performance in Solving Optimization Problems," *Article in International Journal of Engineering and Technology*, vol. 7, no. 4, pp. 6109–6114, 2018.

[3] L. Goel, J. Kanhar, V. S. Patel, and A. Vardhan, "Hybrid Elephant Herding Optimization–Big Bang Big Crunch for pattern recognition from natural images," *Soft Comput*, Jun. 2023, https://doi.org/10.1007/s00500-023-08667-y.

[4] S. M. Almufti, R. Boya Marqas, and R. R. Asaad, "Comparative study between elephant herding optimization (EHO) and U-turning ant colony optimization (U-TACO) in solving symmetric traveling salesman problem (STSP)," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 32, Aug. 2019, https://doi.org/10.14419/jacst.v8i2.29403

[5] A. Kaveh and T. Bakhshpoori, *Metaheuristics: Outlines, MATLAB Codes and Examples*. Springer International Publishing, 2019. https://doi.org/10.1007/978-3-030-04067-3.

[6] S. M. Almufti, "Historical survey on metaheuristics algorithms," *International Journal of Scientific World*, vol. 7, no. 1, p. 1, Nov. 2019, https://doi.org/10.14419/ijsw.v7i1.29497.

[7] S. M. Almufti, *Metaheuristics Algorithms: Overview, Applications, and Modifications*, 1st ed. Deep Science Publishing, 2025. https://doi.org/10.70593/978-93-7185-454-2

[8] A. Soler-Dominguez, A. A. Juan, and R. Kizys, "A Survey on Financial Applications of Metaheuristics," *ACM Comput Surv*, vol. 50, no. 1, pp. 1–23, Jan. 2018, https://doi.org/10.1145/3054133.

[9] S. M. Almufti, "Fusion of Water Evaporation Optimization and Great Deluge: A Dynamic Approach for Benchmark Function Solving," *Fusion: Practice and Applications*, vol. 13, no. 1, pp. 19–36, 2023, https://doi.org/10.54216/FPA.130102

[10] K. Hussain, M. N. Mohd Salleh, S. Cheng, and R. Naseem, "Common Benchmark Functions for Metaheuristic Evaluation: A Review," *JOIV : International Journal on Informatics Visualization*, vol. 1, no. 4–2, p. 218, Nov. 2017, https://doi.org/10.30630/joiv.1.4-2.65.

[11] A. Ahmed Shaban, S. M. Almufti, and R. R. Asaad, "Metaheuristic Algorithms for Engineering and CombinatorialOptimization: A Comparative Study Across Problems Categories and Benchmarks," *International Journal of Scientific World*, vol. 11, no. 2, pp. 38–49, Aug. 2025, https://doi.org/10.14419/0hndc578

[12] J. Kudela and R. Matousek, "New Benchmark Functions for Single-Objective Optimization Based on a Zigzag Pattern," *IEEE Access*, vol. 10, pp. 8262–8278, 2022, https://doi.org/10.1109/ACCESS.2022.3144067.

[13] V. Plevris and G. Solorzano, "A Collection of 30 Multidimensional Functions for Global Optimization Benchmarking," *Data (Basel)*, vol. 7, no. 4, p. 46, Apr. 2022, https://doi.org/10.3390/data7040046

[14] S. M. Almufti and A. Ahmed Shaban, "Advanced Metaheuristic Algorithms for Structural Design Optimization," *FMDB Transactions on Sustainable Intelligent Networks*, vol. 2, no. 1, pp. 33–48, Mar. 2025, https://doi.org/10.69888/FTSIN.2025.000368.

[15] S. M. Almufti and A. A. Shaban, "A deep dive into the artificial bee colony algorithm: theory, improvements, and real-world applications," *International Journal of Scientific World*, vol. 11, no. 1, pp. 178–187, May 2025, https://doi.org/10.14419/v9d3s339.

[16] M. Zhang, W. Luo, and X. Wang, "Differential evolution with dynamic stochastic selection for constrained optimization," *Inf Sci (N Y)*, vol. 178, no. 15, pp. 3043–3074, Aug. 2008, https://doi.org/10.1016/j.ins.2008.02.014.
[17] S. M. Almufti, "Lion algorithm: Overview, modifications and applications E I N F O," *International Research Journal of Science*, vol. 2, no. 2, pp. 176–186, 2022,
[18] S. Almufti, "The novel Social Spider Optimization Algorithm: Overview, Modifications, and Applications," *ICONTECH INTERNATIONAL JOURNAL*, vol. 5, no. 2, pp. 32–51, Jun. 2021, https://doi.org/10.46291/ICONTECHvol5iss2pp32-51.
[19] S. M. Almufti, "U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem," 2015.
[20] S. M. Almufti and A. A. Shaban, "U-Turning Ant Colony Algorithm for Solving Symmetric Traveling Salesman Problem," *Academic Journal of Nawroz University*, vol. 7, no. 4, p. 45, Dec. 2018, https://doi.org/10.25007/ajnu.v7n4a270.
[21] R. Asaad and N. Abdulnabi, "Using Local Searches Algorithms with Ant Colony Optimization for the Solution of TSP Problems," *Academic Journal of Nawroz University*, vol. 7, no. 3, pp. 1–6, 2018, https://doi.org/10.25007/ajnu.v7n3a193.
[22] R. R. Ihsan, S. M. Almufti, B. M. S. Ormani, R. R. Asaad, and R. B. Marqas, "A Survey on Cat Swarm Optimization Algorithm," *Asian Journal of Research in Computer Science*, pp. 22–32, Jun. 2021, https://doi.org/10.9734/ajrcos/2021/v10i230237.
[23] M. Wedyan, O. Elshaweesh, E. Ramadan, and R. Alturki, "Vibrating Particles System Algorithm for Solving Classification Problems," *Computer Systems Science and Engineering*, vol. 43, no. 3, pp. 1189–1206, 2022, https://doi.org/10.32604/csse.2022.024210.
[24] S. Almufti, "Vibrating Particles System Algorithm: Overview, Modifications and Applications," *ICONTECH INTERNATIONAL JOURNAL*, vol. 6, no. 3, pp. 1–11, Sep. 2022, https://doi.org/10.46291/ICONTECHvol6iss3pp1-11.
[25] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, Mar. 2014, https://doi.org/10.1016/j.advengsoft.2013.12.007.
[26] S. M. Almufti, H. B. Marqas, and R. B. Asaad, "Grey wolf optimizer: Overview, modifications and applications," *International Research Journal of Science*, vol. 1, no. 1, pp. 44–56, 2021, https://doi.org/10.14419/efkvvd44.
[27] S. M. Almufti, "Exploring the Impact of Big Bang-Big Crunch Algorithm Parameters on Welded Beam Design Problem Resolution," *Academic Journal of Nawroz University*, vol. 12, no. 4, pp. 1–16, Sep. 2023, https://doi.org/10.25007/ajnu.v12n4a1903.
[28] S. M. Almufti, A. A. H. Alkurdi, and E. A. Khoursheed, "Artificial Bee Colony Algorithm Performances in Solving Constraint-Based Optimization Problem," vol. 21, 2022.
[29] S. M. Almufti, A. Ahmad Shaban, R. Ismael Ali, and J. A. Dela Fuente, "Overview of Metaheuristic Algorithms," *Polaris Global Journal of Scholarly Research and Trends*, vol. 2, no. 2, pp. 10–32, Apr. 2023, https://doi.org/10.58429/pgjsrt.v2n2a144.
[30] L. Kou *et al.*, "Optimized design of patrol path for offshore wind farms based on genetic algorithm and particle swarm optimization with traveling salesman problem," *Concurr Comput*, vol. 36, no. 2, Jan. 2024, https://doi.org/10.1002/cpe.7907
[31] A. Ferhat, F. Zitouni, R. Lakbichi, A. Limane, S. Harous, and O. Kazar, "Impact of Problem Size on Exploration-Exploitation Balance in Metaheuristics for the Travelling Salesman Problem," in *2025 7th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, IEEE, Apr. 2025, pp. 1–8. https://doi.org/10.1109/PAIS66004.2025.11126504.
[32] M. Tong, Z. Peng, and Q. Wang, "A hybrid artificial bee colony algorithm with high robustness for the multiple traveling salesman problem with multiple depots," *Expert Syst Appl*, vol. 260, p. 125446, Jan. 2025, https://doi.org/10.1016/j.eswa.2024.125446
[33] A. Narayanan and M. Moore, "Quantum-inspired genetic algorithms," in *Proceedings of IEEE International Conference on Evolutionary Computation*, IEEE, pp. 61–66. https://doi.org/10.1109/ICEC.1996.542334.
[34] J. P. B. Leite and B. H. V. Topping, "Improved genetic operators for structural engineering optimization," *Advances in Engineering Software*, vol. 29, no. 7–9, pp. 529–562, Aug. 1998, https://doi.org/10.1016/S0965-9978(98)00021-0
[35] S. M. Almufti, A. Yahya Zebari, and H. Khalid Omer, "A comparative study of particle swarm optimization and genetic algorithm," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 40, Oct. 2019, https://doi.org/10.14419/jacst.v8i2.29401.
[36] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, IEEE, pp. 1942–1948. https://doi.org/10.1109/ICNN.1995.488968.
[37] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, IEEE, pp. 69–73. doi: 10.1109/ICEC.1998.699146.
[38] S. M. Almufti, R. B. Marqas, R. R. Asaad, and A. A. Shaban, "Cuckoo search algorithm: overview, modifications, and applications," 2025. [Online]. Available: www.sciencepubco.com/index.php/IJSW. https://doi.org/10.14419/efkvvd44.
[39] S. M. Almufti and A. A. Shaban, "Comparative Analysis of Metaheuristic Algorithms for Solving The Travelling Salesman Problems," *International Journal of Scientific World*, vol. 11, no. 2, pp. 26–30, Jul. 2025, https://doi.org/10.14419/7fk7k945.
[40] A. Ahmed Shaban and I. Mahmood Ibrahim, "World Swarm intelligence algorithms: a survey of modifications and applications," 2025. [Online]. Available: www.sciencepubco.com/index.php/IJSW. https://doi.org/10.14419/vhckcq86.
[41] M. M. ATIQULLAH and S. S. RAO, "SIMULATED ANNEALING AND PARALLEL PROCESSING: AN IMPLEMENTATION FOR CONSTRAINED GLOBAL DESIGN OPTIMIZATION," *Engineering Optimization*, vol. 32, no. 5, pp. 659–685, Jan. 2000, https://doi.org/10.1080/03052150008941317.
[42] J. Liu, "Novel orthogonal simulated annealing with fractional factorial analysis to solve global optimization problems," *Engineering Optimization*, vol. 37, no. 5, pp. 499–519, Jul. 2005, https://doi.org/10.1080/03052150500066646.
[43] A.-R. Hedar and M. Fukushima, "Derivative-Free Filter Simulated Annealing Method for Constrained Continuous Global Optimization," *Journal of Global Optimization*, vol. 35, no. 4, pp. 521–549, Aug. 2006, https://doi.org/10.1007/s10898-005-3693-z.
[44] I. Fister, I. Fister, X.-S. Yang, and J. Brest, "A comprehensive review of firefly algorithms," *Swarm Evol Comput*, vol. 13, pp. 34–46, Dec. 2013, https://doi.org/10.1016/j.swevo.2013.06.001.
[45] X.-S. Yang and Suash Deb, "Cuckoo Search via Levy flights," in *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, IEEE, 2009, pp. 210–214. https://doi.org/10.1109/NABIC.2009.5393690.
[46] Xin-She Yang and Suash Deb, "Engineering optimisation by cuckoo search," *Int. J. Mathematical Modelling and Numerical Optimisation*, vol. 1, no. 4, pp. 330–343, 2010. https://doi.org/10.1504/IJMMNO.2010.035430.
[47] M. Mareli and B. Twala, "An adaptive Cuckoo search algorithm for optimisation," *Applied Computing and Informatics*, vol. 14, no. 2, pp. 107–115, Jul. 2018, https://doi.org/10.1016/j.aci.2017.09.001
[48] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Eng Comput*, vol. 29, no. 1, pp. 17–35, Jan. 2013, https://doi.org/10.1007/s00366-011-0241-y.
[49] S. M. Almufti, R. Boya Marqas, and V. Ashqi Saeed, "Taxonomy of bio-inspired optimization algorithms," *Journal of Advanced Computer Science & Technology*, vol. 8, no. 2, p. 23, Aug. 2019, https://doi.org/10.14419/jacst.v8i2.29402
[50] S. Mohammed Almufti, R. P. Maribojoc, and A. V. Pahuriray, "Ant Based System: Overview, Modifications and Applications from 1992 to 2022," *Polaris Global Journal of Scholarly Research and Trends*, vol. 1, no. 1, pp. 29–37, Oct. 2022, https://doi.org/10.58429/pgjsrt.v1n1a85.
[51] K. Y. Gómez Díaz, S. E. De León Aldaco, J. Aguayo Alquicira, M. Ponce-Silva, and V. H. Olivares Peregrino, "Teaching–Learning-Based Optimization Algorithm Applied in Electronic Engineering: A Survey," *Electronics (Basel)*, vol. 11, no. 21, p. 3451, Oct. 2022, https://doi.org/10.3390/electronics11213451.
[52] S. M. Almufti, "Artificial Bee Colony Algorithm performances in solving Welded Beam Design problem," *Computer Integrated Manufacturing Systems*, vol. 28, 2022.
[53] X. Wang, V. Snášel, S. Mirjalili, J.-S. Pan, L. Kong, and H. A. Shehadeh, "Artificial Protozoa Optimizer (APO): A novel bio-inspired metaheuristic algorithm for engineering optimization," *Knowl Based Syst*, vol. 295, p. 111737, Jul. 2024, https://doi.org/10.1016/j.knosys.2024.111737.
[54] S. M. Almufti, "Hybridizing Ant Colony Optimization Algorithm for Optimizing Edge-Detector Techniques," *Academic Journal of Nawroz University*, vol. 11, no. 2, pp. 135–145, May 2022, https://doi.org/10.25007/ajnu.v11n2a1320.

[55] J. Wang, W. Wang, X. Hu, L. Qiu, and H. Zang, “Black-winged kite algorithm: a nature-inspired meta-heuristic for solving benchmark functions and engineering problems,” *Artif Intell Rev*, vol. 57, no. 4, p. 98, Mar. 2024, https://doi.org/10.1007/s10462-024-10723-4
[56] A. Rahimnejad, E. Akbari, S. Mirjalili, S. A. Gadsden, P. Trojovský, and E. Trojovská, “An improved hybrid whale optimization algorithm for global optimization and engineering design problems,” *PeerJ Comput Sci*, vol. 9, p. e1557, Nov. 2023, https://doi.org/10.7717/peerj-cs.1557
[57] M. G. M. Gámez and H. P. Vázquez, “A Novel Swarm Optimization Algorithm Based on Hive Construction by Tetragonula carbonaria Builder Bees,” *Mathematics*, vol. 13, no. 17, p. 2721, Aug. 2025, https://doi.org/10.3390/math13172721.
[58] A. A. Shaban and M. Yasin, “Applications of the artificial bee colony algorithm in medical imaging and diagnostics: a review,” 2025. [Online]. Available: www.sciencepubco.com/index.php/IJSW.
[59] Z. Wang and X. Wei, “Harris Hawk optimization algorithm with combined perturbation strategy and its application,” *Sci Rep*, vol. 15, no. 1, p. 23372, Jul. 2025, https://doi.org/10.1038/s41598-025-04705-x.
[60] Y. Xia and Y. Ji, “Application of a novel metaheuristic algorithm inspired by Adam gradient descent in distributed permutation flow shop scheduling problem and continuous engineering problems,” *Sci Rep*, vol. 15, no. 1, p. 21692, Jul. 2025, https://doi.org/10.1038/s41598-025-01678-9
[61] A. Ahmed Shaban, S. M. Almufti, and R. R. Asaad, “Metaheuristic Algorithms for Engineering and CombinatorialOptimization: A Comparative Study Across Problems Categories and Benchmarks,” *International Journal of Scientific World*, vol. 11, no. 2, pp. 38–49, Aug. 2025, https://doi.org/10.14419/0hndc578
[62] A. Yahya Zebari, S. M. Almufti, and C. Mohammed Abdulrahman, “Bat algorithm (BA): review, applications and modifications,” *International Journal of Scientific World*, vol. 8, no. 1, p. 1, Jan. 2020, https://doi.org/10.14419/ijsw.v8i1.30120.
[63] M. Ghasemi *et al.*, “Birds of prey-based optimization (BPBO): a metaheuristic algorithm for optimization,” *Evol Intell*, vol. 18, no. 4, p. 88, Aug. 2025, https://doi.org/10.1007/s12065-025-01052-8.
[64] D. Fang, J. Yan, and Q. Zhou, “Channa argus optimizer for solving numerical optimization and engineering problems,” *Sci Rep*, vol. 15, no. 1, p. 21502, Jul. 2025, https://doi.org/10.1038/s41598-025-08517-x.
[65] M. Dehghani, Z. Montazeri, G. Bektemyssova, O. P. Malik, G. Dhiman, and A. E. M. Ahmed, “Kookaburra Optimization Algorithm: A New Bio-Inspired Metaheuristic Algorithm for Solving Optimization Problems,” *Biomimetics*, vol. 8, no. 6, p. 470, Oct. 2023, https://doi.org/10.3390/biomimetics8060470.
[66] S. M. Almufti, “Vibrating Particles System Algorithm performance in solving Constrained Optimization Problem,” *Academic Journal of Nawroz University*, vol. 11, no. 3, pp. 231–242, Aug. 2022, https://doi.org/10.25007/ajnu.v11n3a1499.
[67] S. M. Almufti and A. B. Sallow, “Benchmarking Metaheuristics: Comparative Analysis of PSO, GWO, and ESNS on Complex Optimization Landscapes,” *AVE Trends in Intelligent Computing Systems*, vol. 2, no. 2, pp. 62–76, 2025
[68] J. Huang and H. Hu, “Hybrid beluga whale optimization algorithm with multi-strategy for functions and engineering optimization problems,” *J Big Data*, vol. 11, no. 1, p. 3, Jan. 2024, https://doi.org/10.1186/s40537-023-00864-8.
[69] M. A. Al-Betar, M. A. Awadallah, M. S. Braik, S. Makhadmeh, and I. A. Doush, “Elk herd optimizer: a novel nature-inspired metaheuristic algorithm,” *Artif Intell Rev*, vol. 57, no. 3, p. 48, Feb. 2024, https://doi.org/10.1007/s10462-023-10680-4
[70] F. Chakraborty and P. K. Roy, “An efficient multilevel thresholding image segmentation through improved elephant herding optimization,” *Evol Intell*, vol. 18, no. 1, p. 17, Feb. 2025, https://doi.org/10.1007/s12065-024-01001-x.
[71] Y. Drias, H. Drias, and I. Khennak, “Enhanced Elephant Herding Optimization for Large Scale Information Access on Social Media,” Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.11916.
[72] S. Almufti, “Using Swarm Intelligence for solving NPHard Problems,” *Academic Journal of Nawroz University*, vol. 6, no. 3, pp. 46–50, 2017, https://doi.org/10.25007/ajnu.v6n3a78.
[73] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, “Aquila Optimizer: A novel meta-heuristic optimization algorithm,” *Comput Ind Eng*, vol. 157, Jul. 2021, https://doi.org/10.1016/j.cie.2021.107250
[74] M. Ilchi Ghazaan and A. Kaveh, “A new meta-heuristic algorithm: vibrating particles system,” *Scientia Iranica*, vol. 24, no. 2, pp. 551–566, Apr. 2017, https://doi.org/10.24200/sci.2017.2417.
[75] H. T. Öztürk and H. T. Kahraman, “Meta-heuristic search algorithms in truss optimization: Research on stability and complexity analyses,” *Appl Soft Comput*, vol. 145, p. 110573, Sep. 2023, https://doi.org/10.1016/j.asoc.2023.110573.
[76] A. Gogna and A. Tayal, “Metaheuristics: review and application,” *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 25, no. 4, pp. 503–526, Dec. 2013, https://doi.org/10.1080/0952813X.2013.782347.
[77] R. B. Marqas, S. M. Almufti, P. S. Othman, and C. M. Abdulrahma, “Evaluation of EHO, U-TACO and TS Metaheuristics algorithms in Solving TSP.”
[78] S. M. Almufti, R. B. Marqas, P. S. Othman, and A. B. Sallow, “Single-based and population-based metaheuristics for solving np-hard problems,” *Iraqi Journal of Science*, vol. 62, no. 5, pp. 1710–1720, May 2021, https://doi.org/10.24996/10.24996/ijs.2021.62.5.34.
[79] S. M. Almufti, V. A. Saeed, and R. B. Marqas, “Taxonomy of Bio-Inspired Optimization Algorithms.”
[80] M. Rajasekaran, M. S. Thanabal, and A. Meenakshi, “Association rule hiding using enhanced elephant herding optimization algorithm,” *Automatika*, vol. 65, no. 1, pp. 98–107, Jan. 2024, https://doi.org/10.1080/00051144.2023.2277998.
[81] P. Nandakumar and R. Subhashini, “Heart Disease Prediction Using Convolutional Neural Network with Elephant Herding Optimization,” *Computer Systems Science and Engineering*, vol. 48, no. 1, pp. 57–75, 2024, https://doi.org/10.32604/csse.2023.042294
[82] D. Praveena Anjelin and S. Ganesh Kumar, “An effective classification using enhanced elephant herding optimization with convolution neural network for intrusion detection in IoMT architecture,” *Cluster Comput*, vol. 27, no. 9, pp. 12341–12359, Dec. 2024, https://doi.org/10.1007/s10586-024-04512-5.