Assiut University Journal of Multidisciplinary Scientific Research (AUNJMSR)
Faculty of Science, Assiut University, Assiut, Egypt.
Printed ISSN 2812-5029
Online ISSN 2812-5037
Vol. 54(2): 358- 385 (2025)
https://aunj.journals.ekb.eg

Comparative Analysis of Meta-heuristic Algorithms for Unconstrained Optimization Problems

Alaa M.Asklany[1], Mohamed A.Tawhid [2], H. M. El-Hawary[1], and Amira A.Allam[1,3,*]

Affiliation 1; Department of Mathematics, Faculty of Science, Assiut University, Assiut, Egypt.
Affiliation 2; Department of Mathematics and Statistics, Faculty of Science, Thompson Rivers University, Kamloops, BC, Canada.
Affiliation 3; Department of Basic Science, Faculty of Engineering, Sphinx University, Assiut, Egypt.

[*]Corresponding Author: amirahm@science.aun.edu.eg

ARTICLE INFO

Article History:
Received: 2024-12-25
Accepted: 2025-03-05
Online: 2025-04-27

Keywords:
Metaheuristic, Population-Based Metaheuristics,Swarm Intelligence, Unconstrained Optimization Problem,

ABSTRACT

The use of metaheuristic algorithms in optimization has recently gained significant attention from researchers, often referred to as new or novel algorithms. These algorithms aim to efficiently solve complex optimization problems by mimicking natural processes or behaviors. This study explores the implementation of several recent meta-heuristic algorithms, such as the Hippopotamus Optimization Algorithm (HO), Puma Optimizer (PO), Spider Wasp Optimizer (SWO), Mountain Gazelle Optimizer (MGO), A Sinh Cosh Optimizer (SCHO), Kepler Optimization Algorithm (KOA), and Seahorse Optimizer (SHO). A comprehensive comparison is made between these meta-heuristic approaches using a set of 23 standard test functions, including both unimodal and multimodal functions that vary in complexity. The evaluation criteria include accuracy, convergence speed, and robustness. The results indicate that the Spider Wasp Optimizer (SWO) consistently outperforms other algorithms in terms of optimization performance. Additionally, two non-parametric statistical tests, the Friedman and the Wilcoxon Signed-Rank tests, have been employed to rigorously rank the performance of the algorithms. The findings provide valuable insights into the strengths and weaknesses of each algorithm and demonstrate the potential of SWO for addressing real-world optimization challenges.

1. INTRODUCTION

Optimization is critical in solving complex problems across various fields, including engineering, medicine, decision-making, and agriculture. Optimization problems may be used for a variety of real-world challenges, including engineering, health, decision-making, and agriculture.[1]. These types of problems are very challenging to solve. Several deterministic methods have been proposed to address such problems; however, these methods are not suitable for An extensive variety of real-world optimization problems due to challenges such as falling into local minima, requiring gradient information, and being time-consuming. [1], [2]. To overcome these challenges, metaheuristic algorithms have emerged as powerful tools for solving complex global optimization problems. These algorithms are designed to efficiently explore and exploit the search space without requiring gradient information, making them particularly suitable for non-differentiable, nonlinear, or multimodal problems [33]. Metaheuristics are stochastic in nature, relying on random processes to guide their search, which enables them to prevent being captured in local optima. Over the last few decades, a wide variety of metaheuristic algorithms inspired by natural phenomena, biological systems, and physical processes have been developed and applied successfully to a broad range of optimization problems [34]. Consequently, in recent decades, modern stochastic methods, referred to as metaheuristic algorithms (MAs), have been developed to deal with these challenges [3], [4], [5], [1] and [6]. In general, global optimization problems are essential in various application areas such as industry, engineering, and business. The main goal of optimization is to minimize or maximize specific objective functions. Mathematically, the continuous nonlinear global optimization problems described in this paper are represented as follows:

$$\min_{x \in \mathbb{R}^n} f(x)$$

Where: $f(x)$ is the objective function to be minimized, and $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$ represents the vector of decision variables that can take any real values. The goal is to find x^* such that:

$$f(x^*) \leq f(x) \quad \text{forall} \quad x \in \mathbb{R}^n$$

In this paper, seven recently representative metaheuristic algorithms have been selected and utilized to evaluate the performance of these metaheuristic algorithms. These algorithms include the Hippopotamus Optimization Algorithm (HO): Inspired by the aggressive behavior of hippopotamuses and their ability to navigate through water and land [7], Puma Optimizer (PO): Based on the hunting behavior of pumas, this algorithm focuses on speed and stealth, which helps in quickly narrowing down the search space.[8], Spider Wasp Optimizer (SWO): modeled after the predatory strategies of spider wasps. [9], Mountain Gazelle Optimizer (MGO): Inspired by the fast movements and survival strategies of mountain gazelles. [10], A Sinh Cosh Optimizer (SCHO): based on mathematical hyperbolic functions, providing a unique approach to optimization. [11], Kepler Optimization Algorithm (KOA): Inspired by Kepler's laws of planetary motion, this algorithm uses orbital mechanics to guide the search process. [12] and Sea-horse optimizer (SHO): Modeled after the social and hunting behavior of sea-horses. [13]. This study focuses on analyzing and comparing recently proposed metaheuristic algorithms advancements in optimization techniques. However, we would like to highlight that each of these algorithms has been previously compared with traditional optimization methods, such as the Genetic Algorithm (GA) particle swarm optimization (PSO) [35] in the original studies where they were introduced. Therefore, the main objective of this study is not to repeat these comparisons but rather to explore the specific advantages, challenges, and performance characteristics of these newer algorithms in a variety of benchmark problems. This kind of comparison is essential to identify which algorithm is best suited for different types of problems, especially when dealing with high-dimensional, multimodal, or complex problem landscapes. The primary objective of this work is to study in-depth characteristics and behaviors of each representative algorithm for an empirical study to gain a deeper understanding of them. Additionally, this study provides a methodology for effectively evaluating and comparing the performances of the different mentioned algorithms. For a fair comparison, all seven algorithms are implemented on the same platform and tested on the same set of benchmark problems with the same conditions and stopping criteria. The remainder of the paper is organized as follows: the next section provides related works and a brief discussion of the algorithms, outlining the advantages and disadvantages of each. Section 3 presents empirical performance comparisons of all the representative algorithms on the 23 continuous benchmark functions, their mathematical defections can be found in [11] and the summary is represented in Table 2. which is followed by a comprehensive discussion of the final results and statistical analysis. Section 4 concludes with final remarks.

2. Review of Related Work and the Apllied Algorithms

This section reviews the related work that inspired the current study. It also highlights the key achievements of the comparative methods in this paper by outlining the advantages, disadvantages, and design inspirations of the representative algorithms.

2.1 Related work

Many recent studies in this domain have focused on comparing the performance of various metaheuristic algorithms, especially regarding convergence speed, solution accuracy, and resilience in diverse optimization problem types. The use of statistical methods, such as the Friedman test and Wilcoxon Signed-Rank test, to rank the performance of algorithms, has become a standard practice.

• In [18] twelve global optimization metaheuristic algorithms' performances are investigated. These algorithms demonstrated a thorough numerical evaluation of twelve stochastic algorithms on specific continuous global optimization test problems, including particle swarm optimization (PSO), ant colony optimization (ACO), symbiotic organisms search (SOS), cuckoo search (CS), firefly algorithms (FA), artificial bee colony (ABC), and bat algorithms (BA).

• In [19], Several well-known evolutionary algorithms' characteristics were examined. To evaluate the optimization capabilities of genetic algorithms (GA), biogeography-based optimization (BBO), differential evolution (DE), evolution strategy (ES), and particle swarm optimization (PSO) on a set of real-world continuous optimization problems, the authors compared their basic and advanced versions. One of the main findings drawn from the experiment was that, under certain conditions, traditional versions of BBO, DE, ES, and PSO are equivalent to the genetic algorithm with global uniform recombination (GA/GUR). However this conceptual equivalency, their main contribution emphasizes that modifications to algorithms result in significantly differing performance levels. This result highlights the need for additional investigation, including a thorough examination and evaluation of the effectiveness of several metaheuristic algorithms, as explored in the current study..

• In a similar context, [20]compared the algorithmic equivalence of particle swarm optimization (PSO) with several more recent swarm intelligence algorithms, such as the artificial bee colony (ABC), firefly algorithm (FA), gravitational search algorithm (GSA), group search optimizer (GSO), and shuffled frog leaping algorithm (SFLA). The authors used combinatorial problems and continuous benchmark functions to quantitatively compare the basic and advanced versions of these algorithms. The research found that the standard versions of SFLA, GSO, FA, ABC, and GSA are algorithmically similar to basic PSO under certain experimental circumstances. However, the actual results also demonstrated how the basic and advanced algorithms performed differently for each of the two problem categories.

• A comparative analysis is demonstrated in [19] by evaluating the run-time complexity and the required function-evaluation number for obtaining the global minimizer of algorithmic ideas for many benchmark functions of continuous optimization problems including particle swarm optimization (PSO), artificial bee colony (ABC), differential evolution (DE), and cuckoo search (CK). According to empirical results, the CK algorithm and the DE algorithm have an extremely similar capacity for problem-solving.

• This study is motivated by the recent, and implementation of several new metaheuristic algorithms by different researchers such as [14], [15], [16], [18], and [17], They argue that their motivation is based on the widely accepted understanding that no one metaheuristic optimization algorithm can solve all types of optimization problems with varying structures. Therefore, there is a continuous need to improve existing algorithms or develop new metaheuristic optimization techniques capable of tackling increasingly complex and large-scale global optimization problems. Therefore, in line with this common belief, the current study seeks to investigate the algorithmic performance superiority of seven selected recent metaheuristic algorithms mentioned in subsection 2.2 and to also ascertain whether this set of algorithms has any similarity in their performance behaviors under the same experimental conditions. The next section provides a brief overview of the advantages and drawbacks of each representative algorithm to justify the selection of the aforementioned popular algorithms.

2.2 Representative Algorithms: Inspirations, Benefits and Challenges

Each of the seven metaheuristic algorithms that were previously described has a variety of advantages and disadvantages. However, the majority of these methods still have challenges, such as the inability to guarantee optimal solutions, the need for many iterations, and poor performance in the absence of adequate parameter tuning, they offer numerous advantages, such as flexibility and the ability to handle complex optimization problems.

Algorithm 1 General Metaheuristic Optimization Algorithm

1. Initialize the population (solutions) randomly within the defined search space.
2. Evaluate the fitness of each solution.
3. While the stopping condition is not met (e.g., max iterations or convergence threshold):
    a. Select the best solutions based on fitness.
    b. Update the position of the solutions using the algorithm-specific update rules.
    c. Evaluate the new fitness values of the updated solutions.
    d. Repeat until stopping condition is satisfied.
4. Output the best solution found.Output the best solution found.

Table 1 Abbreviationsa and the key characteristics of the seven algorithms

Algorithm	Inspiration	Selection Mechanism	Updating Strategy	Special Feature
Hippopotams Optimization Algorithm (HO)	Hippopotamus social behavior	Best fitness-based	Water movement strategy	Balances exploration & exploitation

Puma Optimizer (PO)	Puma hunting strategy	Adaptive selection	Distance-based updates	Dynamic learning mechanism
Spider Wasp Optimizer (SWO)	Spider wasp predatory behavior	Prey-tracking	Attack-based movement	High local search efficiency
Kepler Optimization Algorithm (KOA)	Kepler's planetary motion	Gravity-based	Orbital trajectory updates	Avoids local optima
Mountain Gazelle Optimizer (MGO)	Mountain gazelle escape behavior	Fittest individual tracking	Speed-based updates	Exploits survival instincts
A Sinh Cosh Optimizer (SCHO)	Sinh Cosh function properties	Rank-based	Logarithmic adjustments	Inspired by mathematical functions
Sea-horse optimizer (SHO)	Seahorse migration patterns	Group-based	Ocean current modeling	Strong adaptive mechanism

3. The Experimental Results

In this section, we investigate the efficiency of the proposed algorithms. Hippopotamus Optimization Algorithm (HO) [7], Puma Optimizer (PO) [8], Spider Wasp Optimizer (SWO) [9], Mountain Gazelle Optimizer (MGO) [10], A Sinh Cosh Optimizer (SCHO) [11], Kepler Optimization Algorithm (KOA) [12] and Sea-horse optimizer (SHO) [13] on unconstrained benchmark functions 23 benchmark functions at different dimensions. The algorithms are implemented in MATLAB, and their performance is evaluated against the 23 benchmark functions at different dimensions to determine their strengths and weaknesses. This suite comprises a diverse set of functions including unimodal, multimodal, hybrid, and composition functions. To statistically analyze the results obtained, two nonparametric statistical tests, Wilcoxon's rank sum test, and the Friedman test, are performed at the 5% significant level [32]. The statistical tests of the algorithms based on the results of Wilcoxon's rank sum test ($p - value/h$) are reported in tables and the symbols (+, - , $\approx$) indicate that a given algorithm performed significantly better (+), significantly worse (-) or not significantly different ($\approx$) compared to our algorithm SWO. Friedman rank test (Rank) is reported in the last row of each test problem 5. Table 3 details the parameter settings for the proposed algorithm, including population size, maximum iterations, dimension of the problems, and eps (acceptable error margin). To ensure a fair comparison, all algorithms are configured with a uniform population size of 30, and the termination criteria are an error gap smaller than 10^{-6} or the maximum allowable function evaluations are reached.

Table 2 Summary of Test Functions

Function	Range	Dimension	Global Minimum
F_1	[−100, 100]	30	0
F_2	[−10, 10]	30	0
F_3	[−100, 100]	30	0
F_4	[−100, 100]	30	0
F_5	[−30, 30]	30	0
F_6	[−100, 100]	30	0
F_7	[−1.28, 1.28]	30	0
F_8	[−500, 500]	30	-418.9829
F_9	[−5.12, 5.12]	30	0
F_{10}	[−32, 32]	30	0
F_{11}	[−600, 600]	30	0
F_{12}	[−50, 50]	30	0
F_{13}	[−50, 50]	30	0
F_{14}	[−65.536, 65.536]	2	1
F_{15}	[−5, 5]	4	0.00030
F_{16}	[−5, 5]	2	-1.0316
F_{17}	[−5, 10] × [0, 15]	2	0.3983
F_{18}	[−2, 2]	2	3
F_{19}	[0, 1]	3	-3.86
F_{20}	[0, 1]	6	-3.32

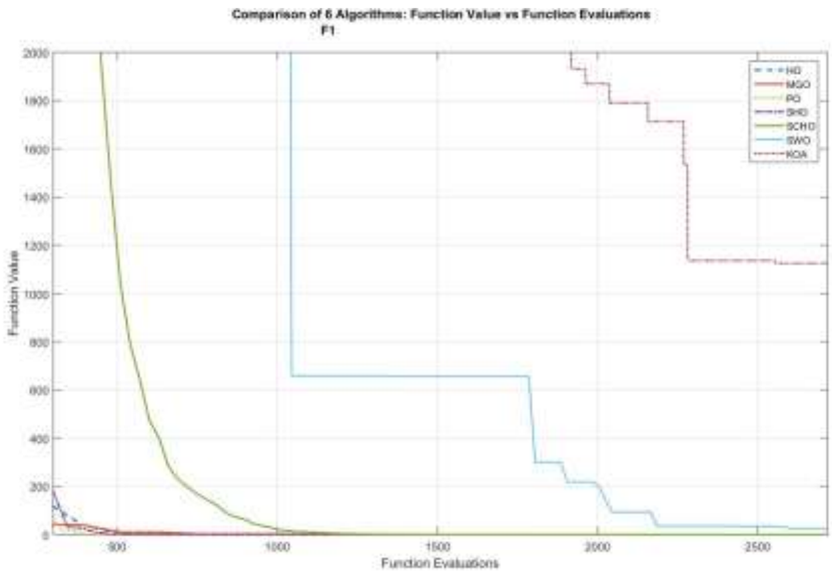
F_{21}	[0, 10]	4	-10.1532
F_{22}	[0, 10]	4	-10.4028
F_{23}	[0, 10]	4	-10.5363

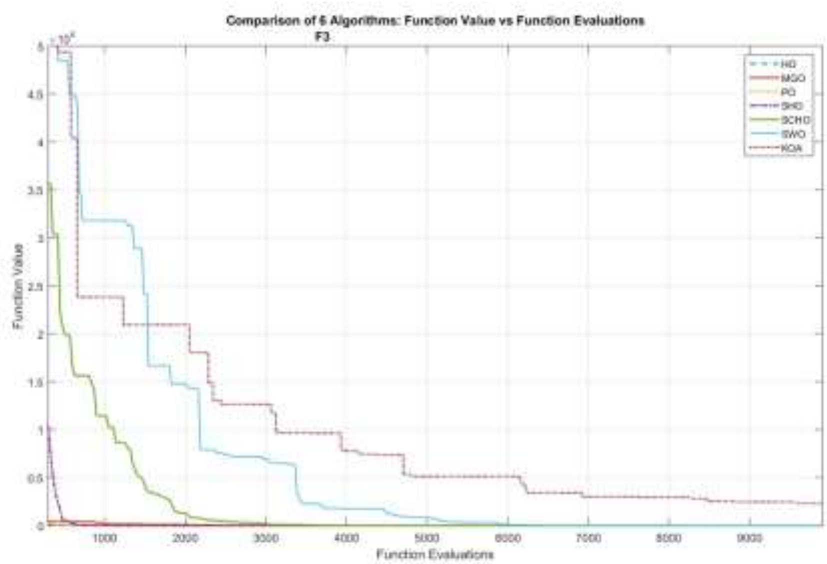
Table 3: Parameter settings.

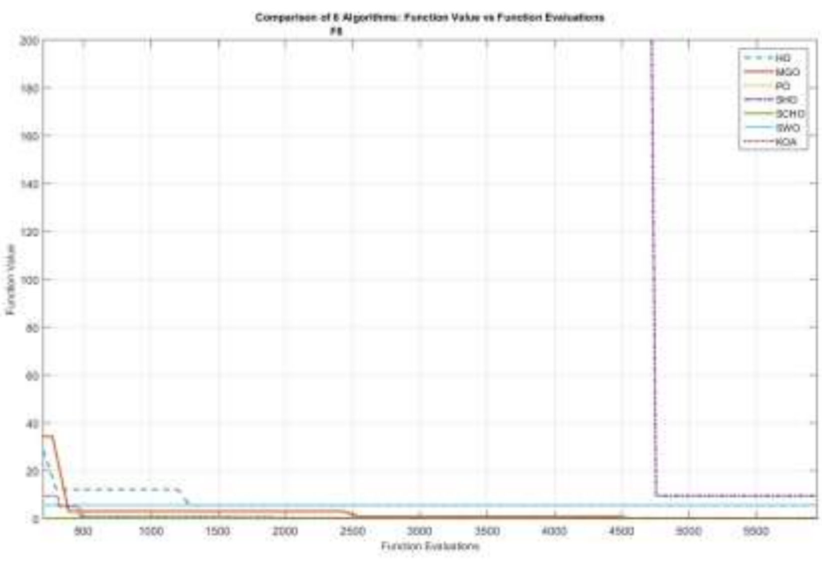
Parameters	Definitions	Values
Num	Population size	30
f_count	Maximum function evaluation	24000
Max_it	Maximum number of iterations	800
dim	Problem dimension	30
eps	The accepted error	E^{-6}

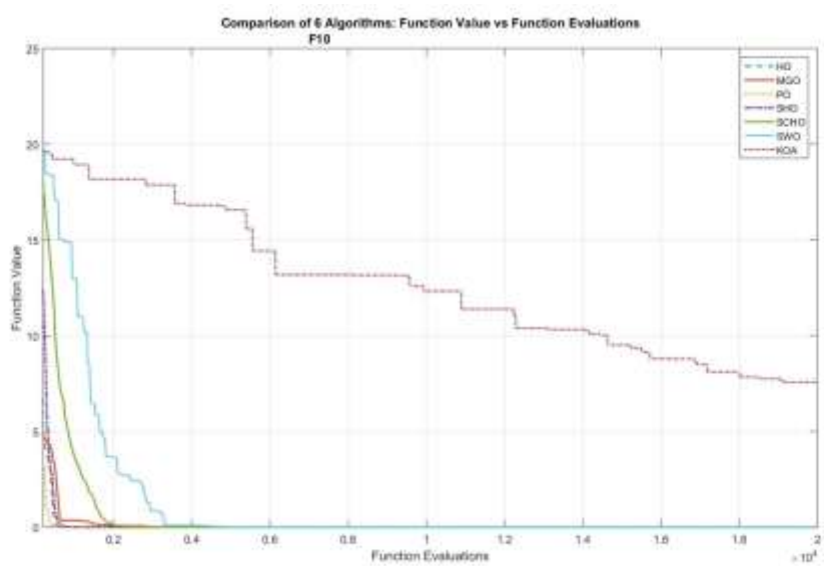
Figure 1 shows the performance of the compared algorithms over 8 functions with different features, these comparisons provide a clear view of how each algorithm handles problems helping to guide their selection and potential modifications for specific optimization tasks. The sub-figures of F_1 and F_2 show that Ho and SCHO algorithms demonstrated outstanding performance with rapid initial convergence. It consistently showcased rapid initial convergence, maintaining superior function value reduction across evaluations maintaining superior function value reduction.

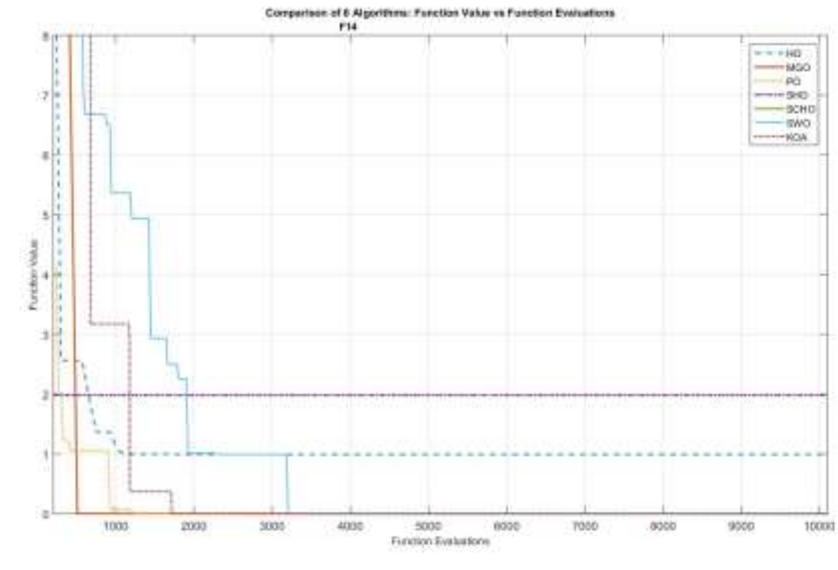
Fig. 1 The performance of the seven Algorithms

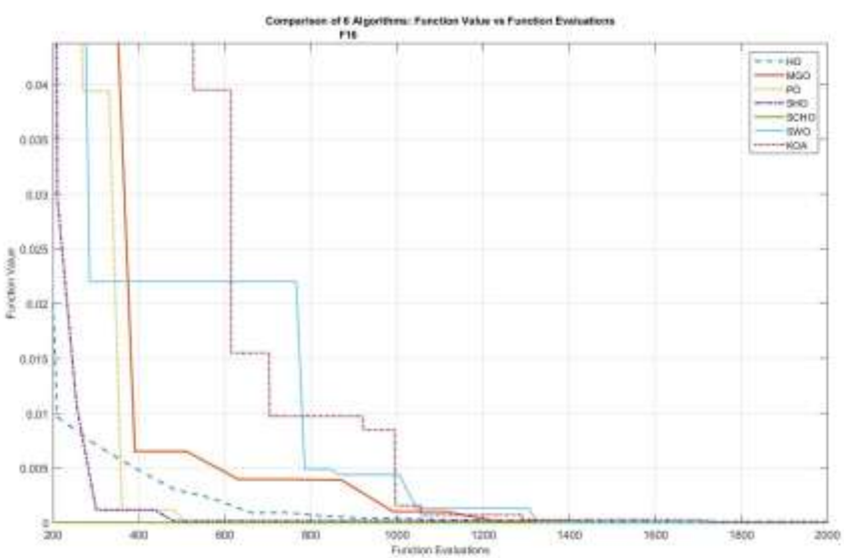


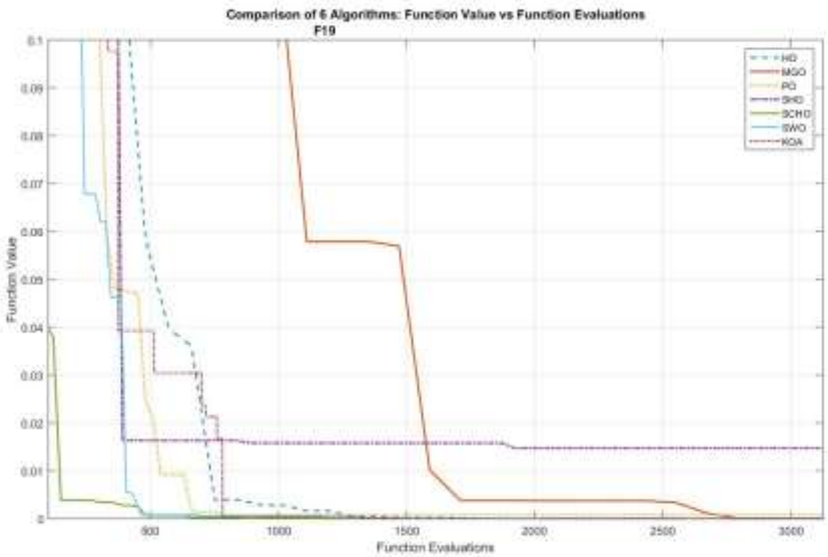


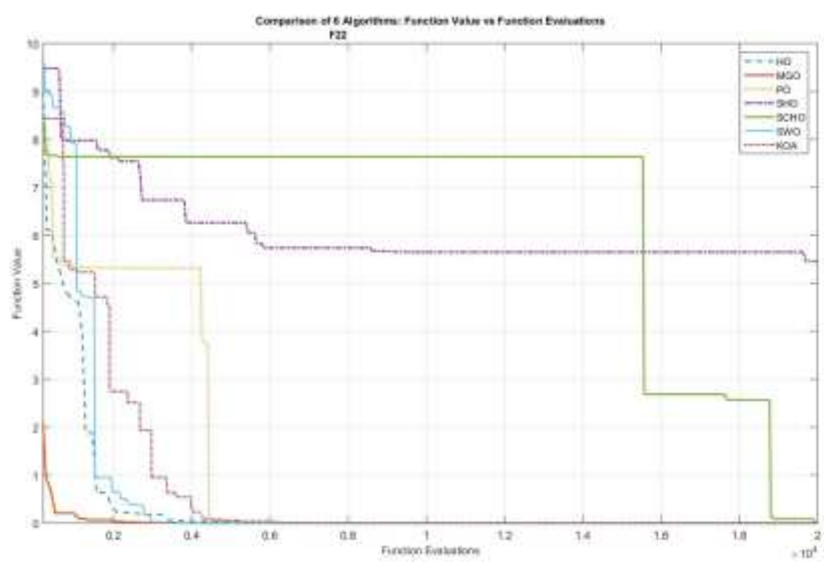

Comparison of 6 Algorithms: Function Value vs Function Evaluations
F8
HO
MGO
PO
SHO
SCHO
SWO
KOA
Function Value
Function Evaluations


Comparison of 6 Algorithms: Function Value vs Function Evaluations
F10
HO
MGO
PO
SHO
SCHO
SWO
KOA
Function Value
Function Evaluations


Comparison of 6 Algorithms: Function Value vs Function Evaluations
F14
HO
MGO
PO
SHO
SCHO
SWO
KOA
Function Value
Function Evaluations


Comparison of 6 Algorithms: Function Value vs Function Evaluations
F16
HO
MGO
PO
SHO
SCHO
SWO
KOA
Function Value
Function Evaluations





This highlights their robust exploitation capabilities, making it a reliable choice for unimodal problems where the global minimum is reached quickly. SWO and SHO displayed moderate convergence in both tests. These algorithms were consistent in reaching near-optimal solutions but with a more gradual approach compared to SCHO and HO. Their balanced exploration and exploitation capabilities make them dependable, although not the fastest solutions for unimodal optimization tasks. PO and MGO showed variable behavior, often starting strong but plateauing at certain points in both F_1 and F_3. KOA exhibited step-like convergence patterns, characterized by periods of stagnation interspersed with sudden improvements. This behavior indicates strong exploration capabilities but limited exploitation, leading to longer times to reach optimal solutions. The Behavior of the algorithms on Multimodal Test Functions F_8 and F_{10} shows that SCHO and HO consistently exhibited rapid initial convergence and maintained superior performance throughout the optimization process for both F8 and F10 with better performance for SCHO. This highlights its strong exploitation capabilities, making it an effective choice for multimodal problems where finding and maintaining optimal solutions amid many local optima is critical. SWO and SHO maintained consistent performance with gradual convergence. Their balanced approach allowed them to steadily approach lower function values without significant late-stage changes. KOA demonstrated a step-like convergence indicating strong exploratory behavior but slower overall progress. This suggests that KOA can explore effectively but struggles to refine solutions quickly after initial exploration. MGO and PO were able to make Early Convergence and rapid early progress but tended to plateau soon after. This suggests that these algorithms need enhancements to maintain steady progress throughout the optimization process, especially in complex landscapes. The performance of the algorithms across these four complex multimodal functions reveals varied strengths in exploration, exploitation, and long-term convergence. These functions, characterized by their numerous local optima, provide a challenging landscape for them. HO and SCHO are the top-performing algorithms, they showcase a strong balance between rapid initial convergence and steady long-term performance, effectively navigating multimodal landscapes and maintaining progress without prolonged plateaus. Especially noted for SCHO the robust initial exploitation capabilities. While it occasionally encounters plateaus (e.g., in F_{19} and F_{22}), it demonstrates the potential for significant late-stage improvements, highlighting its strong exploratory nature. SWO and SHO (Sea-horse Optimizer) maintain steady convergence patterns without aggressive improvements after the initial phase. Their behavior suggests a balanced approach to exploration and exploitation, making them reliable but not the fastest algorithms. SHO shows early convergence with limited further progress, whereas SWO maintains a consistent, moderate pace. PO and MGO exhibit strong early-stage convergence, rapidly reducing the function value but stabilizing quickly and often plateauing before reaching optimal solutions. This behavior indicates effective early exploitation with limited adaptability for continued progress. KOA is the exploration-oriented algorithm over all algorithms that stands out for its step-like convergence pattern, indicating strong exploration capabilities. However, its incremental improvements come at the cost of slower overall convergence. This makes KOA ideal for tasks that prioritize thorough exploration and global search but less suitable for situations demanding rapid results. To conclude the performance of all algorithms: SWO and SHO have a Balanced behavior, they are reliable for moderate optimization needs, providing stable performance without aggressive changes after the initial phase. While MGO and PO the initial fast convergence with quick early-stage optimization is needed, enhancements are recommended to sustain their performance over longer evaluations. HO and SCHO lead in terms of effective performance across multimodal functions with fixed dimensions, showcasing a strong balance of exploration and exploitation. KOA remains notable for its thorough exploration capabilities, though it lags in speed. We gave the average error, mean, the best solution f_{best}, the worst solution f_{worst}, and standard deviation (std) for all algorithms over 30 runs. The results in tables 4 and 5 demonstrate that the SWO algorithm exhibits the best performance and shows a lower average rank across the majority of functions, it can be considered a top performer.

Table 4: Statistical Results with Friedman's Rank Test for the proposed algorithms

Functions		KAO	MGO	PO	SWO	SCHO	SHO	HO
F1	mean	2.3E-06	7.5E-07	5.1E-07	4.3E-07	9.1E-07	6E-07	2.8E-07
	f_best	8.9E-07	4.8E-07	1.5E-07	1.3E-07	8.2E-07	3.5E-07	1.8E-10
	f_worst	7.7E-06	9.8E-07	9.9E-07	9.2E-07	9.9E-07	9.7E-07	8.8E-07
	std	2.1E-06	1.8E-07	2.8E-07	2.9E-07	5.3E-08	2E-07	3.4E-07
	rank	7	4	4	2.75	4.5	3.5	2.25
F2	mean	0.00054	7.5E-07	7.5E-07	5.9E-07	9.6E-07	8E-07	5.5E-07
	f_best	0.00023	3.7E-07	5E-07	4E-07	9.3E-07	5.7E-07	8.2E-08
	f_worst	0.00208	9.6E-07	9.8E-07	8.1E-07	1E-06	9.1E-07	1E-06
	std	0.00055	1.7E-07	1.5E-07	1.4E-07	1.8E-08	1.1E-07	3E-07
	rank	7	3.5	3.75	2.25	4.5	3.5	3.5
F3	mean	460.293	0.04417	4.5E-07	4.5E-07	3.5E-07	6.9E-07	3.5E-07
	f_best	206.834	3.5E-06	1.6E-07	6.8E-08	1.6E-11	4.9E-07	9.2E-09
	f_worst	814.951	0.41452	8.1E-07	8.2E-07	9.9E-07	9.2E-07	9E-07
	std	195.279	0.13022	2.6E-07	2.7E-07	4.4E-07	1.6E-07	3.1E-07
	rank	7	6	2.75	2.75	3	3.75	2.75

F4	mean	6.73214	7.6E-07	7E-07	6E-07	1.1E-07	7.7E-07	5.3E-07
	f_best	2.44092	4.3E-07	3.8E-07	2.9E-07	1.4E-09	4.2E-07	8E-08
	f_worst	12.4358	9.9E-07	9.5E-07	9E-07	3.1E-07	9.5E-07	9.4E-07
	std	3.05879	2.1E-07	2.1E-07	1.9E-07	1.1E-07	1.7E-07	3E-07
	rank	7	5.5	4.25	2.75	1	4.25	3.25
F5	mean	86.4129	1.4E-06	23.139	23.1845	27.8069	28.3353	0.13044
	f_best	31.1901	6.8E-07	0.00045	22.9331	26.2393	27.2813	0.02251
	f_worst	166.424	5.6E-06	26.183	23.6702	28.8272	28.8846	0.35617
	std	40.7119	1.5E-06	8.1403	0.26019	0.87245	0.50875	0.10728
	rank	7	1	3.75	3.5	5	5.5	2.25
F6	mean	1.8E-06	0.00105	2.1E-06	9.9E-07	1.18896	4.19554	0.0519
	f_best	7.1E-07	4.5E-05	7.2E-07	9.7E-07	1.8E-05	3.4762	0.00109
	f_worst	4.1E-06	0.00371	6.1E-06	1E-06	2.77861	4.75476	0.08094
	std	1.2E-06	0.00124	1.9E-06	1E-08	1.21751	0.3522	0.02932
	rank	1.75	4.25	2.75	1.5	5.75	6.75	5.25
F7	mean	0.03884	0.00167	0.00016	0.00066	4.5E-05	0.00013	0.00028
	f_best	0.02519	0.00041	5.5E-06	0.0003	2.8E-06	1.3E-05	7.9E-05
	f_worst	0.0531	0.00398	0.00032	0.00103	0.00016	0.00033	0.00076
	std	0.0097	0.00119	0.00011	0.00028	4.8E-05	9.8E-05	0.0002
	rank	7	6	2.5	5	1	2.5	4
F8	mean	0.0367	0.12012	1956.06	16.9621	0.09527	1.15761	8.66013
	f_best	-418.98	-418.97	-1821.7	-419.66	-418.98	-419.03	-418.81
	f_worst	-418.88	-418.51	-3191.2	-363.25	-419.26	-414.55	-385.81
	std	0.04708	0.1945	456.224	23.9156	0.13895	1.75083	13.1728
	rank	2.5	4	4	5.25	2.5	4	5.75
F9	mean	74.8427	7.1E-07	6.4E-07	5.9E-07	4.9E-08	5.9E-07	2.2E-07

	f_best	59.3395	4.9E-07	2.7E-07	8.8E-08	1.1E-10	2E-07	5.4E-09
	f_worst	98.3301	9.6E-07	9.2E-07	9.9E-07	2.8E-07	8.9E-07	9E-07
	std	11.6854	1.8E-07	2.2E-07	3E-07	8.6E-08	2.4E-07	2.8E-07
	rank	7	4.75	4.25	4.75	1	3.25	3
F10	mean	4.83773	8.6E-07	8.2E-07	8E-07	5E-07	7.8E-07	5.7E-07
	f_best	1.0193	4.8E-07	7.3E-07	6.1E-07	2.9E-09	3.9E-07	1.1E-07
	f_worst	18.3187	9.8E-07	9.6E-07	9.5E-07	9.9E-07	1E-06	9.5E-07
	std	5.37113	1.5E-07	6.6E-08	1.1E-07	5.1E-07	1.9E-07	2.7E-07
	rank	7	4.25	3.75	3.25	3.25	4	2.5
F11	mean	0.00888	7.1E-07	4.7E-07	5.6E-07	3.6E-07	6.9E-07	3.5E-07
	f_best	1.4E-06	3.1E-07	2E-07	8.8E-08	5.8E-10	3.7E-07	1.1E-09
	f_worst	0.04421	9.6E-07	9.3E-07	9.5E-07	9.7E-07	9.2E-07	9.1E-07
	std	0.01414	2E-07	2.3E-07	3.4E-07	4.7E-07	1.9E-07	3.4E-07
	rank	7	4.5	3.25	3.75	3.75	3.5	2.25
F12	mean	0.67213	1.7E-06	9.3E-07	9.7E-07	0.03169	0.42645	0.00331
	f_best	2.6E-06	4.3E-09	7.5E-07	9E-07	1.7E-08	0.36452	0.00249
	f_worst	3.69997	4.6E-06	1E-06	1E-06	0.31687	0.52747	0.00594
	std	1.21224	1.7E-06	7.2E-08	3.4E-08	0.1002	0.05077	0.00105
	rank	6.5	2.5	2	2	4.5	6	4.5
F13	mean	0.29082	7.2E-07	0.0022	9.6E-07	0.9494	2.4726	0.05777
	f_best	2.5E-06	3.4E-07	8.6E-07	8.8E-07	2.3E-07	2.23933	0.00105
	f_worst	2.90612	9.5E-07	0.01099	1E-06	2.05649	2.71279	0.13332
	std	0.91892	2.2E-07	0.00463	4.4E-08	0.86658	0.151	0.04299
	rank	6	1.5	3	2.25	4.5	6.25	4.5
F14	mean	6.6E-07	5.3E-06	0.00112	0.00024	4.09579	3.91257	0.19826
	f_best	1	1	1.00013	1.00007	1	0.99991	1

	f_worst	1	0.99999	0.998	0.99902	12.6705	12.6705	2.98211
	std	9E-07	7.2E-06	0.00103	0.00032	5.29815	4.58832	0.62679
	rank	2.5	3	4	3.5	6	4.75	4.25
F15	mean	0.0001	0.00022	9.9E-05	7.5E-06	2.6E-05	0.00212	8.9E-06
	f_best	0.00031	0.00031	0.00031	0.00031	0.00031	0.00031	0.00031
	f_worst	0.00074	0.00073	0.00122	0.00031	0.00038	0.02081	0.00032
	std	0.00017	0.00019	0.00029	7.5E-19	2.1E-05	0.00646	3.7E-06
	rank	4.25	4.75	4.5	1	3.75	7	2.75
F16	mean	2.8E-07	6.4E-07	9.1E-06	8.1E-07	4.9E-07	1.7E-06	5.1E-07
	f_best	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	f_worst	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316	-1.0316
	std	3.5E-07	7.4E-07	1.2E-05	8.9E-07	5.8E-07	2.4E-06	6.1E-07
	rank	2.75	4	5	4.75	2.75	5	3.75
F17	mean	5.9E-07	5.1E-07	4.2E-05	2.4E-06	1.6E-06	0.00184	8.3E-07
	f_best	0.398	0.398	0.398	0.398	0.398	0.398	0.398
	f_worst	0.398	0.398	0.39791	0.39801	0.398	0.40807	0.398
	std	6.4E-07	5.7E-07	5.1E-05	2.8E-06	2.1E-06	0.00305	9.8E-07
	rank	2.25	2.25	5	4.25	4.75	5.75	3.75
F18	mean	5.1E-07	4.8E-07	4.3E-07	6.2E-07	35.1669	8.1E-07	4.5E-07
	f_best	3	3	3	3	3	3	3
	f_worst	3	3	3	3	84.0001	3	3
	std	2.9E-07	3.5E-07	2.5E-07	3E-07	39.8283	9.2E-07	1.8E-07
	rank	3.75	3.5	1.5	4.5	7	5.25	2.5
F19	mean	6.9E-07	3.1E-06	0.00094	5.1E-05	2.6E-06	0.00324	1.8E-05
	f_best	-3.86	-3.86	-3.8601	-3.86	-3.86	-3.86	-3.86
	f_worst	-3.86	-3.86	-3.8623	-3.8602	-3.86	-3.853	-3.8601

	std	6.7E-07	4E-06	0.0012	8.3E-05	3.2E-06	0.00336	3E-05
	rank	2.5	4	3.5	3.5	3.25	6.75	4.5
F20	mean	3.32	3.32	4.67776	3.32046	3.32057	3.78893	3.32125
	f_best	-3E-08	-3E-08	-0.922	-3E-05	-3E-07	-0.0078	-6E-05
	f_worst	-3E-08	-3E-08	-2.2766	-0.0018	-0.0051	-1.6179	-0.0034
	std	3.5E-24	3.5E-24	0.36567	0.00063	0.0016	0.65583	0.00132
	rank	4	4	3.75	3.75	4.25	4.25	4
F21	mean	1E-06	8.5E-07	7.6E-07	7.7E-07	0.51243	5.74508	1.6E-05
	f_best	-10.153	-10.153	-10.153	-10.153	-10.153	-5.0823	-10.153
	f_worst	-10.153	-10.153	-10.153	-10.153	-5.0552	-0.8799	-10.153
	std	7.3E-07	1.5E-07	1.7E-07	1.4E-07	1.61121	1.25126	1.8E-05
	rank	3.5	3	1.5	2	6.25	6.75	5
F22	mean	5.4E-07	5.5E-07	0.76372	0.53151	2.65433	5.57419	2.7E-06
	f_best	-10.403	-10.403	-10.403	-10.403	-10.403	-5.549	-10.403
	f_worst	-10.403	-10.403	-2.7659	-5.0877	-5.0876	-4.3436	-10.403
	std	6.4E-07	6.2E-07	2.415	1.68079	2.79639	0.31742	5.6E-06
	rank	2	2.5	4.75	3.75	6	6	3
F23	mean	2.4E-05	9.6E-05	0.02855	0.00151	2.26072	6.28442	0.5289
	f_best	-10.403	-10.403	-10.407	-10.403	-10.403	-5.0899	-10.403
	f_worst	-10.403	-10.403	-10.473	-10.407	-2.8705	-0.945	-5.1285
	std	2.8E-05	0.00015	0.03613	0.00164	3.63777	1.6811	1.6675
	rank	2.75	2.75	2.5	2.75	5.25	6.75	5.25
	Av_rank	4.782609	3.717391	3.478261	3.282609	4.065217	5	3.673913

Compared to the other algorithms in this category, While these algorithms showed potential in breaking out of local optima, their slower convergence suggests the need for better exploitation mechanisms to enhance performance on unimodal functions. When examining the multimodal benchmark functions, the exploration abilities of the KAO algorithm demonstrated superior performance, particularly on functions $F_8, F_{14}, F_{16}, F_{19}, F_{20}, F_{22} and F_{23}$, especially in the fixed-dimension test cases. Its exploration capability makes it a strong candidate for problems that require extensive search space coverage at the expense of speed. Although SWO only achieved the best function evaluation on F_6, it provided steady and reliable performance without aggressive convergence and consistently ranked highest among all algorithms due to its stability and effective balance between exploitation and exploration processes. The convergence speeds of SCHO and HO emerging are the most efficient in both test functions due to their swift descent toward optimal values. SWO and SHO provided steady, reliable performance without aggressive convergence, while MGO and KOA required more time to achieve comparable results. that indicates SCHO and HO excel at exploitation, capitalizing quickly on promising areas of the search space. In contrast, MGO and PO showed stronger exploratory behavior, beneficial for more complex problems but less effective in the unimodal test cases analyzed, and could benefit from adaptive strategies that reintroduce exploration after reaching a plateau to avoid stagnation. KOA and SHO could incorporate mechanisms that encourage more aggressive exploitation once a promising region is found, improving their speed of convergence.

Table 5: Summary of Wilcoxon's rank sum at 5% significance level

SWO vs.	+ (better)	= (no sig.)	- (worse)
KOA	14	0	9
MGO	12	0	11
PO	17	0	6
SCHO	13	0	10
SHO	19	0	4
HO	8	0	15

From the Wilcoxon test results in Table 5, it can be observed that SWO has a higher + count than the other algorithms, particularly against SHO and SCHIO, suggesting that SWO is more effective in certain scenarios when compared to these algorithms. However, while SWO performs worse most of the time against HO, it achieves a better average rank than HO. This suggests that HO might be more robust or effective on certain functions in terms of average error, but not across all majorities.

4. CONCLUSION

In this study, a comparative analysis of seven recent metaheuristic algorithms; Hippopotamus Optimization Algorithm (HO), Puma Optimizer (PO), Spider Wasp Optimizer (SWO), Mountain Gazelle Optimizer (MGO), Sinh Cosh Optimizer (SCHO), Kepler Optimization Algorithm (KOA), and Sea-Horse Optimizer (SHO) were performed using 23 standard test functions. These functions included both unimodal and multimodal benchmark problems, allowing for a comprehensive evaluation of each algorithm's exploitation and exploration capabilities. The results demonstrated that for unimodal functions, the HO and SCHO algorithms excelled, indicating stronger exploitation abilities in navigating simpler landscapes. On the other hand, the KAO algorithm performed best on several multimodal functions, particularly those with fixed dimensions, showcasing its robust exploration abilities. Despite only achieving the best function evaluation on F_6, the Spider Wasp Optimizer (SWO) consistently maintained the highest overall rank due to its balanced performance across both unimodal and multimodal functions. Its stability and effective balance between exploitation and exploration highlight its potential as a versatile optimization tool. Statistical validation through the Friedman test and Wilcoxon Signed-Rank Test confirmed the significance of the differences in performance across the algorithms. These findings suggest that while certain algorithms may excel in specific problem types, a balance between exploration and exploitation is crucial for overall success across diverse optimization landscapes. Future research could focus on hybridizing these algorithms to leverage the strengths of multiple approaches, or exploring their performance on constrained optimization problems and real-world applications in areas like logistics, machine learning, energy systems, and engineering design.

A. Benchmark Functions

In this appendix, we present the mathematical expressions of the seven benchmark functions used in this study.

A.1 Sphere

$$F_1(x) = \sum_{i=1}^{n} x_i^2$$

A.2 Schwefel 2.22

$$F_2(x) = \sum_{i=0}^{n} |x_i| + \prod_{i=0}^{n} |x_i|$$

A.3 Shifted Schwefel's Problem

$$F_3(x) = \sum_{i=1}^{d} \left(\sum_{j=1}^{i} x_j\right)^2$$

A.4 Schwefel 2.21

$$F_4(x) = \max_i\{|x_i|, 1 \leq i \leq n\}$$

A.5 Rosenbrock

$$F_5(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$$

A.6 Step

$$F_6(x) = \sum_{i=1}^{n} ([x_i + 0.5])^2$$

A.7 Quartic

$$F_7(x) = \sum_{i=1}^{n} i x_i^4 + \text{random}[0,1)$$

A.8 Schwefel

$$F_8(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{|x_i|}\right)$$

A.9 Rastrigin

$$F_9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i + 10)\right]$$

A.10 Ackley

$$F_{10}(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) + \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i)\right)20 + e$$

A.11 Shifted Rotated Griewank's without Bounds

$$F_{11}(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

A.12 Penalized 1

$$F_{12}(x) = \frac{\pi}{n}\{10\sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$$

$$y_i = 1 + \frac{x_i + 1}{4}$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & \text{if} x_i > a \\ 0 & \text{if} - a \le x_i \le a \\ k(-x_i - a)^m & \text{if} x_i < -a \end{cases}$$

A.13 Penalized 2

$$F_{13}(x) = 0.1\{\sin^2(3\pi x_i) + \sum_{i=1}^{n} (x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)]$$

$$+(x_n - 1)^2[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$$

A.14 Foxholes

$$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^2}\right)^{-1}$$

A.15 Kowalik

$$F_{15}(x) = \sum_{i=1}^{11} \left(\frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_2 + x_3^2 + x_4^2} - a_i\right)^2$$

A.16 6 Hump Camel Back

$$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{x_1^6}{3} + x_1 x_2 + 4x_2^2 - 4x_2^4$$

A.17 Branin

$$F_{17}(x) = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10$$

A.18 Goldstein Price

$$F_{18}(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2) + 6x_1x_2 + 3x_2^2] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$$

A.19 Hartman3

$$F_{19}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{i=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$$

A.20 Hartman6

$$F_{20}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$$

A.21 Shekel5

$$F_{21}(x) = -\sum_{i=1}^{5} [(X - a_i)^{\mathsf{T}}(X - a_i) + c_i]^{-1}$$

A.22 Shekel7

$$F_{22}(x) = -\sum_{i=1}^{7} [(X - a_i)^{\mathsf{T}}(X - a_i) + c_i]^{-1}$$

A.23 Shekel10

$$F_{23}(x) = -\sum_{i=1}^{10} [(X - a_i)^{\mathsf{T}}(X - a_i) + c_i]^{-1}$$

5. REFERENCES

[1] Sahab, M. G., Toropov, V. V., Gandomi, A. H.: A review on traditional and modern structural optimization: problems and techniques. Metaheuristic Appl. Struct. Infrastruct. 25–47 (2013).

[2] Mirjalili, S., Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. 95, 51–67 (2016).

[3] Glover, F. W., Kochenberger, G. A.: Handbook of Metaheuristics. Springer Science & Business Media, vol. 57 (2003).

[4] Bianchi, L., Dorigo, M., Gambardella, L. M., Gutjahr, W. J.: A survey on metaheuristics for stochastic combinatorial optimization. Nat. Comput. 8, 239–287 (2009).

[5] Yang, X.-S.: Swarm-based metaheuristic algorithms and no-free-lunch theorems. Theory New Appl. Swarm Intell. (2012).

[6] Yang, X.-S., Ting, T. O., Karamanoglu, M.: Random walks, Lí©vy flights, Markov chains and metaheuristic optimization. Future Inf. Commun. Technol. Appl. ICFICE 2013, 1055–1064 (2013).

[7] Amiri, M. H., Mehrabi Hashjin, N., Montazeri, M., Mirjalili, S., Khodadadi, N.: Hippopotamus optimization algorithm: a novel nature-inspired optimization algorithm. Sci. Rep. 14(1), 5032 (2024).

[8] Abdollahzadeh, B., Khodadadi, N., Barshandeh, S., Trojovský, P., Gharehchopogh, F. S., El-kenawy, E.-S. M., Abualigah, L., Mirjalili, S.: Puma optimizer (PO): A novel metaheuristic optimization algorithm and its application in machine learning. *Cluster Computing*, 1–49 (2024).

[9] Abdel-Basset, M., Mohamed, R., Jameel, M., Abouhawwash, M.: Spider wasp optimizer: a novel meta-heuristic optimization algorithm. Artif. Intell. Rev. 56(10), 11675–11738 (2023).

[10] Abdollahzadeh, B., Gharehchopogh, F. S., Khodadadi, N., Mirjalili, S.: Mountain gazelle optimizer: a new nature-inspired metaheuristic algorithm for global optimization problems. Adv. Eng. Softw. 174, 103282 (2022).

[11] Bai, J., Li, Y., Zheng, M., Khatir, S., Benaissa, B., Abualigah, L., Wahab, M. A.: A sinh cosh optimizer. Knowl.-Based Syst. 282, 111081 (2023).

[12] Abdel-Basset, M., Mohamed, R., Azeem, S. A. A., Jameel, M., Abouhawwash, M.: Kepler optimization algorithm: A new metaheuristic algorithm inspired by Keplerâ€™s laws of planetary motion. Knowl.-Based Syst. 268, 110454 (2023).

[13] Zhao, S., Zhang, T., Ma, S., Wang, M.: Sea-horse optimizer: a novel nature-inspired meta-heuristic for global optimization problems. Appl. Intell. 53(10), 11833–11860 (2023).

[14] Cheng, M.-Y., Prayogo, D.: Symbiotic organisms search: A new metaheuristic optimization algorithm. Comput. Struct. 139, 98–112 (2014).

[15] Yang, X.-S.: Firefly algorithm, Levy flights and global optimization. Res. Dev. Intell. Syst. XXVI Incorporating Appl. Innov. Intell. Syst. XVII, 209–218 (2010).

[16] Yang, X.-S., Karamanoglu, M., He, X.: Flower pollination algorithm: A novel approach for multiobjective optimization. Eng. Optim. 46(9), 1222–1237 (2014).

[17] Civicioglu, P., Besdok, E.: A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. Artif. Intell. Rev. 39, 315–346 (2013).

[18] Ezugwu, A. E., Adeleke, O. J., Akinyelu, A. A., Viriri, S.: A conceptual comparison of several metaheuristic algorithms on continuous optimisation problems. Neural Comput. Appl. 32(10), 6207–6251 (2020).

[19] Ma, H., Simon, D., Fei, M., Chen, Z.: On the equivalences and differences of evolutionary algorithms. Eng. Appl. Artif. Intell. 26(10), 2397–2407 (2013).

[20] Ma, H., Ye, S., Simon, D., Fei, M.: Conceptual and numerical comparisons of swarm intelligence optimization algorithms. Soft Comput. 21, 3081–3100 (2017).

[21] Jiang, L., Zhang, Z., Lu, L., Shang, X., Wang, W.: Nonparametric Modelling of Ship Dynamics Using Puma Optimizer Algorithm-Optimized Twin Support Vector Regression. J. Mar. Sci. Eng. 12(5), 754 (2024).

[22] Mohamed, E. A., Braik, M. Sh., Al-Betar, M. A., Awadallah, M. A.: Boosted Spider Wasp Optimizer for High-dimensional Feature Selection. J. Bionic Eng. 1–36 (2024).

[23] Gao, Y., Li, Z., Wang, H., Hu, Y., Jiang, H., Jiang, X., Chen, D.: An Improved Spider-Wasp Optimizer for Obstacle Avoidance Path Planning in Mobile Robots. Math. 12(17), 2604 (2024).

[24] Mosavi, E., Shahzadeh Fazeli, S. A., Abbasi, E., Kaveh-yazdy, F.: Using Hybrid Mountain Gazelle Optimization and Particle Swarm Optimization Algorithms to Improve Clustering. 2024.

[25] Utama, D. M., Sanjaya, B. D., Nugraha, A.: A Modified Mountain Gazelle Optimizer for Minimizing Energy Consumption on No-Wait Permutation Flow Shop Scheduling Problem. Int. J. Intell. Eng. Syst. 17(2) (2024).

[26] Emam, M. M., Houssein, E. H., Abdel Samee, N., Alkhalifa, A. K., Hosney, M. E.: Optimizing cancer diagnosis: A hybrid approach of genetic operators and Sinh Cosh Optimizer for tumor identification and feature gene selection. Comput. Biol. Med. 180, 108984 (2024).

[27] Hu, G., Gong, C., Li, X., Xu, Z.: CGKOA: An enhanced Kepler optimization algorithm for multi-domain optimization problems. Comput. Methods Appl. Mech. Eng. 425, 116964 (2024).

[28] Li, J., Noto, M., Zhang, Y.: Improved Kepler Optimization Algorithm Based on Mixed Strategy. In: Proc. Int. Conf. Swarm Intell. Springer, pp. 157–170 (2024).

[29] Houssein, E. H., Abdalkarim, N., Abdel Samee, N., Alabdulhafith, M., Mohamed, E.: Improved Kepler Optimization Algorithm for enhanced feature selection in liver disease classification. Knowl.-Based Syst. 297, 111960 (2024).

[30] Hashim, F. A., Mostafa, R. R., Abu Khurma, R., Qaddoura, R., Castillo, P. A.: A new approach for solving global optimization and engineering problems based on modified sea horse optimizer. J. Comput. Des. Eng. 11(1), 73–98 (2024).

[31] Hasanien, H. M., Alsaleh, I., Tostado-Vأ©liz, M., Zhang, M., Alateeq, A., Jurado, F., Alassaf, A.: Hybrid particle swarm and sea horse optimization algorithm-based optimal reactive power dispatch of power systems comprising electric vehicles. Energy 286, 129583 (2024).

[32] Derrac, J., Garcأ-a, S., Molina, D., Herrera, F.: A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evol. Comput. 1(1), 3–18 (2011).

[33] Teixeira de Macedo, J. P. G.: NATURE INSPIRED ALGORITHMS FOR ROBOTIC ODOUR SEARCH. PhD thesis, Universidade de Coimbra (2023).

[34] Jiang, R., Crookes, D., Wei, H.-L., Zhang, L., Chazot, P.: Recent Advances in AI-enabled Automated Medical Diagnosis. CRC Press (2022).

[35] Liang, J., Ge, S., Qu, B., Yu, K., Liu, F., Yang, H., Wei, P., Li, Z.: Classified perturbation mutation-based particle swarm optimization algorithm for parameters extraction of photovoltaic models. Energy Conversion and Management, 203, 112138 (2020).