Research Paper

Elite opposition-based social spider optimization for solving benchmark problem

Saman M. Almufti[1*], Amira Bibo Sallow[2]
[1*]Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Iraq.
[2]Department of Information Technology, Technical College of Informatics - Akre, Akre University for Applied Sciences, Iraq.

Article Info

Article History:
Received: 05 July 2025
Revised: 12 Septembet 2025
Accepted: 20 Septembet 2025
Published: 06 November 2025

Keywords:
Social Spider Optimization
Based Learning
Metaheuristic Algorithms
Global Optimization
Benchmark Functions

ABSTRACT

Metaheuristic algorithms are powerful tools for solving complex optimization problems where traditional methods fail. The Social Spider Optimization (SSO) algorithm, inspired by the cooperative foraging behavior of spiders, is a notable swarm intelligence technique. However, it can be prone to premature convergence. This paper presents an enhanced variant, the Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm, which integrates an elite opposition-based learning (OBL) strategy and an elite selection mechanism into the standard SSO framework. This integration aims to improve population diversity, enhance global exploration, and accelerate convergence. The performance of EOSSO is rigorously evaluated on a comprehensive set of 23 benchmark functions, including unimodal, multimodal, and fixed-dimension multimodal problems. Experimental results demonstrate that EOSSO significantly outperforms the standard SSO and other well-known metaheuristics in terms of solution accuracy, convergence speed, and stability. The algorithm exhibits a remarkable ability to escape local optima and refine solutions efficiently, proving its robustness and effectiveness as a high-performance optimizer for complex landscapes.

Corresponding Author:
Saman M. Almufti
Department of Information Technology, Technical College of Informatics - Akre, Akre University for Applied Sciences, Iraq.
Email: Saman.Almofty@gmail.com

International Journal of Research in Science & Engineering (IJRISE)

1. INTRODUCTION

Over the last few years, metaheuristic optimization algorithms have been considered as the ultimate options for finding solutions to complex problems which are beyond the reach of traditional numerical methods [1]. One of the main reasons was the growing use of heuristic methods based on nature's various phenomena, and among them, swarm intelligence techniques like Particle Swarm Optimization (PSO) [2], Ant Colony Optimization (ACO) [3], [4], and Firefly Algorithm (FA) [5] have been widely used. In this study, we consider the Social Spider Optimization (SSO) algorithm [6] and its improved version, the Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm. The EOSSO approach uses the elite opposition-based learning (OBL) strategy and the elite selection mechanism to realize better accuracy, quicker convergence, and also an overall improved stability in the process of global function optimization [7].

The original SSO algorithm, first proposed by Cuevas [8], [9] imitates the social spiders' cooperative foraging rather than the competitive feeding and thus divides the population into male and female groups, whereby the different cooperative operators are used for the simulation of the attraction, repulsion, and mating strategies 4. Even though the SSO has shown good performance in experimentation, one limitation is the possibility of getting stuck in local optima that would increase. To solve these issues, the EOSSO algorithm applies OBL to the traditional SSO model, where the candidate solutions are represented as being paired with their respective “opposite” solutions in the search space. As a result, the population diversity is increased and the chance of finding solutions close to the global optimum is increased too.

[10], [11] The paper puts forward a thorough investigation of the efficacy of the EOSSO algorithm over the 23 benchmark functions that include unimodal, multimodal, and fixed-dimension multimodal problems. [12] The results obtained reported that EOSSO is not only better than many classical techniques in terms of optimality and convergence speed, but also remarkably stable, as the lower standard deviation for solution quality has confirmed this stability.

The structure of the remaining part of the paper is as follows. A literature survey and an analysis of the swarm intelligence and opposition-based learning methods are included in section 2. The SSO and EOSSO algorithms' methodology is explained in section 3, including mathematical modeling and operator design. The benchmark problems and experimental setup used for evaluation are described in section 4. The simulation results and performance comparisons across different benchmark functions are discussed in section 5. The conclusions and directions for future research are presented in section 6.

2. RELETED WORK

Metaheuristic algorithms (MAs) are practically the only options available today to handle difficult optimization problems from various sectors such as engineering, finance, logistics, and artificial intelligence [13], [14]. MAs are inspired by natural, biological, or social phenomena and allow for the effective searching of large and complicated areas (search spaces), which would otherwise be the downfall of conventional gradient-based or enumerative approaches [15]. The field has experienced rapid growth over the last thirty years, with new algorithms constantly being proposed as limitations of the previous ones have been overcome and more difficult real-world problems have been taken on [16].

The early foundational works in the area of metaheuristics include the Genetic Algorithms (GA) [17], [18], which mimic the process of natural selection and evolution, and Particle Swarm Optimization (PSO) [19], [20], which is a representation of the social dynamics of birds or fish. The first population-based search and collective intelligence algorithms were the GA and PSO, which established the principles that later many more methods would follow. On the other hand, Ant Colony Optimization (ACO) [21], [22] introduced the concept of stigmergy indirect communication through the environment that was inspired by the foraging behavior of ants [23]. Likewise, Differential Evolution (DE) [24], [25] and Harmony Search (HS) [26], [27] provided powerful alternatives for dealing with non-linear, multi-modal, and high-dimensional optimization landscapes.

With the Evolution of the field, the researchers' scope broadened to liven up their ideas through drawing from an even larger number of natural and physical systems. One that researcher might name on the algorithms as Swarm Intelligence (SI) models, for example, the ABC algorithm [28] stands in the line, which imitates the food-searching activity of bees, the GWO [29], [30] the social rank and hunting methods of wolves; and the WOA (Mirjalili and Lewis, 2016) which is modeled on the bubble-net hunting technique of humpback whales. In the recent past, the Harris Hawks Optimization (HHO) [31] and the big-bang big-crunch algorithm [32], [33], [34] have been the latest in the parade of biological metaphors and thus the novel performance of searching has been further accomplished through them.

Physics-based algorithms that focus on the second major category the physic-based algorithms that are powered by physical laws and phenomena. The Gravitational Search Algorithm (GSA) [35] emphasizes the law of gravity and the interaction of masses for directing the search process. The Kepler Optimization Algorithm (KOA) [36] simulates the planetary motion according to Kepler's laws and the Thermal Exchange Optimization (TEO) [37] uses the principles of thermodynamics as its basis.

Another category are human-based algorithms which constitute a novel and creative branch, reflecting social, cultural, or cognitive processes. The Teaching-Learning-Based Optimization (TLBO) [38], [39] is impersonating the school setting, while the [40] (PO) is conceptualizing political rivalry and constituency dynamics. The Dynastic Optimization Algorithm (DOA) [41] and the Great Wall Construction Algorithm (GWCA) [42] have their roots in the history of mankind collaboration and competition among people. These algorithms, more often than not, stress such aspects as the transfer of knowledge, social learning, and strategic decision-making.

The biological inspiration that nature offers has also been incorporated into arachnids, resulting in two separate and remarkable optimization algorithms based on spider behavior. The first, the Social Spider Optimization (SSO) algorithm developed by Cuevas in 2013 [43], is a ground-breaking swarm intelligence algorithm that imitates the altruistic conduct of spiders in a colony. By this method, the whole search area is represented as a common web where spiders are in contact through vibrations. A key innovation of this algorithm is its division of the population into male and female agents, each governed by specialized evolutionary operators. Female spiders move based on vibrations from nearby, higher-quality solutions and the global best, with a stochastic choice between attraction and repulsion. Male spiders are classified as either dominant or non-dominant; dominant males are attracted to nearby females to facilitate mating, while non-dominant males converge toward the center of the male population. A mating operator allows for the generation of new offspring, promoting diversity. This social model was designed to address common flaws in earlier algorithms like PSO and ABC, such as premature convergence, by creating a more nuanced and distributed search process through gender-based roles and local communication [44].

Building upon the concept of arachnid-inspired optimization albeit focusing on a fundamentally different biological mechanism, the Somersaulting Spider Optimizer (SSO) [8] presents a novel approach. This algorithm does not model social colony dynamics, but rather draws the inspiration from the extraordinary escape locomotion of the single Cebrennus rechenbergi spider that performs acrobatic somersaults for rapid movement across the desert terrain. The SSO algorithm puts forward an optimization framework with two movements that are in perfect harmony with each other thus making global exploration and local exploitation. The "somersaulting" mechanism allows for long-distance, high-energy jumps for exploration, and it's like the spider's tumbling motion. On the other hand, the "rolling" mechanism makes it possible to do very accurate and very short movements for local refinement. One of the biggest developments is the adaptive energy management system that is built into the algorithm and which can control the movement between exploration and exploitation depending on real-time inputs from solution improvement patterns and the detection of stagnation. The parameter-light design secures a good exploration-exploitation trade-off, and through empirical studies on benchmark engineering problems, it has been confirmed that it is superior to several established algorithms in terms of convergence speed, solution quality, and stability. This approach demonstrates that even within a single taxonomic class, vastly different behaviors social cooperation versus individual locomotion can inspire unique and powerful computational paradigms.

Hitherto, hybrid and enhanced versions have been devised as a solution to the drawbacks of previous techniques. The Opposition-Based Learning (OBL), which was proposed by Tizhoosh (2005) [45], has become a staple in GA, PSO, and DE algorithms, thereby increasing the speed of convergence along with the quality of the solution by simultaneously evaluating both the current and opposite candidate solutions. For instance, the Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm is the result of merging the social SSO with elitist OBL for the purpose of improved global exploration and local trap avoidance. The empirical validation of EOSSO against benchmark functions has already proved the former to be better than the latter in terms of correctness, speed of convergence, and stability not just against the regular SSO but also among the contemporary algorithms.

The recent comparison study and competition reports (e.g., CEC2014, CEC2017, CEC2020, and CEC2022) have pointed out the necessity of algorithms that are able to cooperate in simultaneous balancing of global search exploration and local refinement exploitation [46], [47], [48]. Sophisticated hybrid approaches like PSOGSA (a mix of PSO and GSA) and LSHADE (a winner of CEC2014) have set an impressive performance benchmark. On the other hand, techniques like SOGWO (integrating Spearman's correlation and opposition-based learning into GWO) and LSHADE-cnEpSin (a CEC2017 winner) continue to challenge the limits of difficult optimization problems [14], [49].

These developments still do not alter the fact that the No Free Lunch (NFL) theorem [50] gets rid of all algorithms that are consistently the best. This is what encourages the perpetual birth of new metaheuristics with distinctive solution search mechanisms. In this regard, Tianji's Horse Racing Optimization (THRO) [51], inspired by an ancient Chinese strategy game, introduces a dynamic individual matching tactic between the two competing populations. The competitive co-evolution modeling, based on historical wisdom, is the watershed of this approach, giving the idea of exploring and exploiting in a new way. By optimizing it as a series of strategic races between Tianji's and King's horses, THRO aims at reducing the time till convergence and improving the quality of solutions, especially in difficult, multi-modal and high-dimensional landscapes.

3. METHODOLOGY

3.1. Social Spider Optimization (SSO) Algorithm

The Social Spider Optimization (SSO) algorithm is motivated by the social and cooperative behaviors exhibited by spiders in a colony. The algorithm represents each spider as an n-dimensional vector where each dimension corresponds to one of the decision variables to be optimized. The entire population S is divided into two subgroups: female spiders $F = \{f_1, f_2, \dots, f_{N_f}\}$ and male spiders $M = \{m_1, m_2, \dots, m_{N_m}\}$, where the number of females is determined randomly within the range of 65%–90% of the total population N , see Equations below:

$$N_f = \text{floor}[(0.9 - \text{rand} \times 0.25) \times N]$$

$$N_m = N - N_f$$

Each spider is assigned a weight w_i based on its fitness value $J(s_i)$ relative to the best and worst fitness values in the population:

$$w_i = \frac{J(s_i) - \text{worsts}}{\text{bests} - \text{worsts}}$$

Where

$$\text{bests} = \max_{k\in\{1,\dots,N\}} J(s_k) \text{andworsts} = \min_{k\in\{1,\dots,N\}} J(s_k)$$

The vibrations transmitted by each spider over the communal web are modeled to incorporate both the weight of the spider and its Euclidean distance $d_{i,j}$ from the target spider i:

$$V_{ib_{ij}} = w_j \cdot e^{-d_{i,j}^2}$$

Key relationships are defined for the spider i as follows:

1. Vibrations from the Nearest Higher-Weight Spider c:

$$V_{ib_i} = w_c \cdot e^{-d_{i,c}^2}$$

2. Vibrations from the Best Spider in the Entire Population b:

$$V_{ibb_i} = w_b \cdot e^{-d_{i,b}^2}$$

3. Vibrations from the Nearest Female Spider f:

$$V_{ibf_i} = w_f \cdot e^{-d_{i,f}^2}$$

3.2. Cooperative Operators and Movement

The movement of spiders in SSO is based on cooperative operators that differentiate between female and male individuals.

A. Female Cooperative Operator

Each female spider updates its position by considering

- The attraction (or repulsion) towards the nearest spider s_c with higher weight.
- The attraction (or repulsion) towards the globally best spider s_b.
- An added random component.

The movement is modeled as:

$$f_i^{k+1} = \begin{cases} f_i^k + \alpha \cdot V_{ib_i}(s_c - f_i^k) + \beta \cdot V_{ibb_i}(s_b - f_i^k) + \delta \cdot (\text{rand} - \frac{1}{2}), & \text{if } r_m < PF \\ f_i^k - \alpha \cdot V_{ib_i}(s_c - f_i^k) - \beta \cdot V_{ibb_i}(s_b - f_i^k) + \delta \cdot (\text{rand} - \frac{1}{2}), & \text{if } r_m \geq PF \end{cases}$$

Whereα, β, and δ are random numbers in the range [0, 1] and PF is a predefined threshold probability.

B. Male Cooperative Operator

Male spiders are divided into dominant and non-dominant groups. Dominant males move toward the nearest female spider while, for non-dominant males, the update is based on the weighted mean of the male population. The updating rule is given by:

$$m_i^{k+1} = \begin{cases} m_i^k + \alpha \cdot V_{ibf_i}(s_f - m_i^k) + \delta \cdot (\text{rand} - \frac{1}{2}), & \text{if } w_{N_f+i} > w_{N_f+m} \\ m_i^k + \alpha \cdot \left(\frac{\sum_{h=1}^{N_m} m_h^k \cdot w_{N_f+h}}{\sum_{h=1}^{N_m} w_{N_f+h}} - m_i^k\right), & \text{otherwise} \end{cases}$$

3.3. Elite Opposition-Based Social Spider Optimization (EOSSO)

To enhance the global search capability and avoid premature convergence, the EOSSO algorithm adopts an elite opposition-based learning strategy. Opposition-Based Learning (OBL) is defined as follows for a number $p \in [x, y]$:

$$p^* = x + y - p$$

For an n-dimensional point $p = (s_1, s_2, \ldots, s_n)$, the opposite point is:

$$s_i^* = x_i + y_i - s_i, i = 1,2, \ldots, n$$

In EOSSO, the elite individual (i.e., the best solution X_e) is used to generate an elite opposition-based solution for every candidate X_i. The elite opposition-based solution X_i^* is computed as:

$$x_{i,j}^* = k \cdot (d_{aj} + d_{bj}) - x_{e,j}, j = 1,2, \dots, n$$

Where $k \in U(0,1)$ is a random scaling factor, and d_{aj} and d_{bj} represent the dynamic lower and upper bounds of the jth decision variable, respectively. When the newly obtained $x_{i,j}^*$ falls outside the bounds, it is reset using a uniformly random value in the valid interval.

This elite opposition-based updating is applied both during population initialization and periodically in the evolutionary process to enhance exploration. As the results of the experiment show, EOSSO achieves a quicker convergence rate and better accuracy when compared to the standard SSO and other metaheuristic algorithms.

3.4 Pseudocode of the EOSSO Algorithm

Below is a simplified pseudocode that outlines the overall procedure of EOSSO:

1. Initialize the population of spiders and compute the number of female and male spiders.
2. Assess the fitness of each spider and assign them weights.
3. For each pair of spiders, calculate the communal vibrations considering their weights and distances.
4. Use the female cooperative operator to update the female spiders.
5. Update male spiders using the male cooperative operator.
6. Apply the mating operator for dominant males leading to new candidate solutions.
7. Incorporate elite opposition-based learning:
   - Compute opposite individuals for current candidates.
   - Choose the fittest individuals from the current and opposite candidates.
8. Continue the process until the stopping criterion is met.

The utilization of the integrated strategy guarantees a proficient exploration of the search space coupled with the simultaneous refining of the candidate solutions towards the global optimum. Figure 1 shows the EOSSO algorithm flowchart.

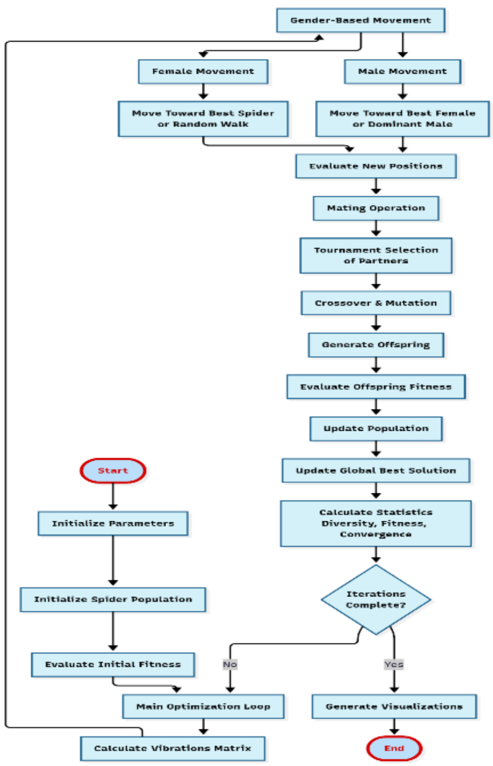


Figure 1. EOSSO Flowchart

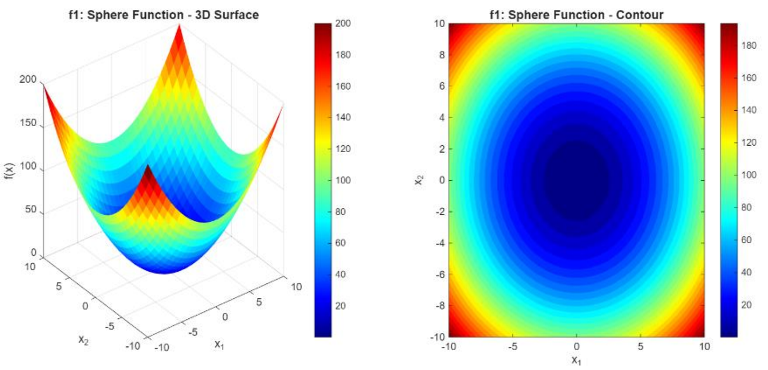
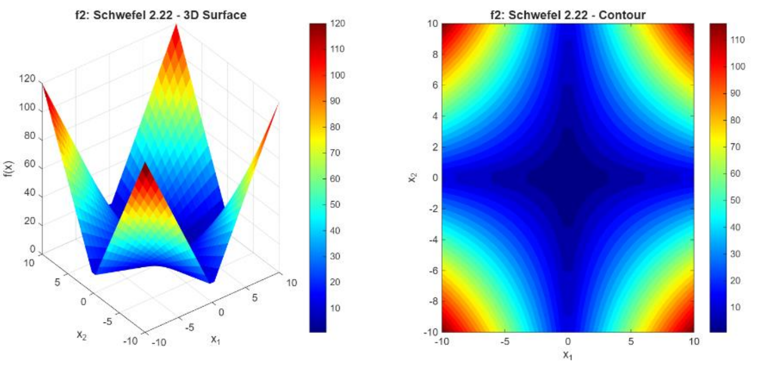
Benchmark Problems

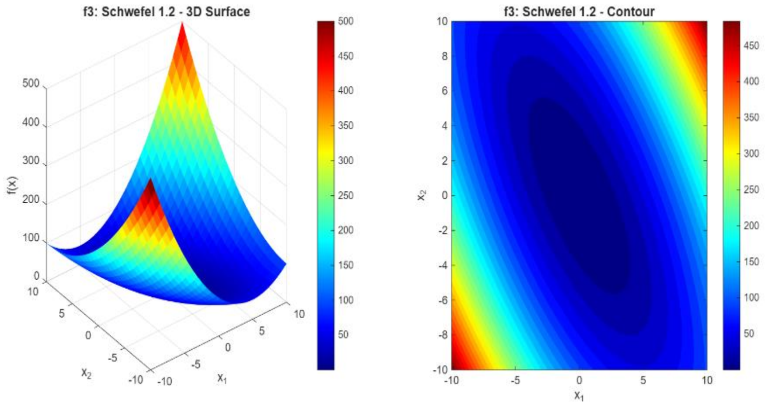
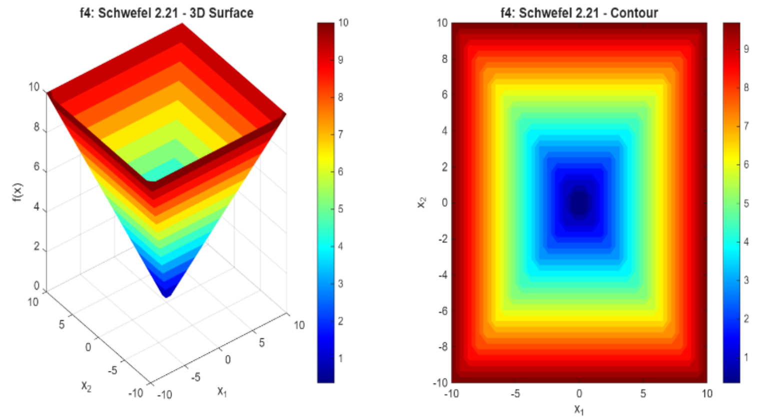
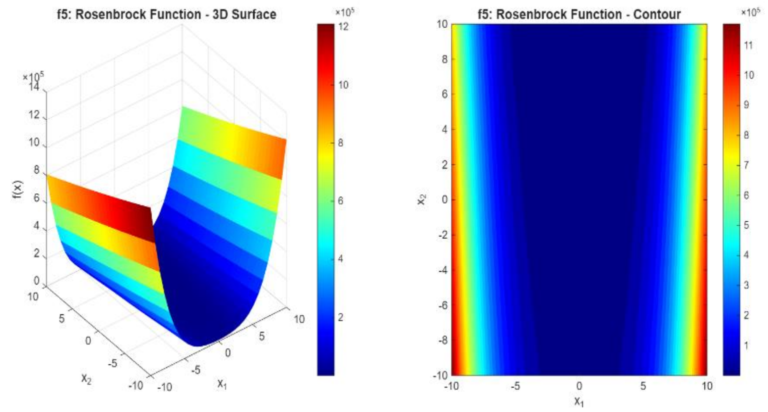
A set of 23 benchmark functions was used to evaluate the effectiveness of the EOSSO algorithm. These functions are very often used to benchmark metaheuristic optimization algorithms performance and are separated into three groups:

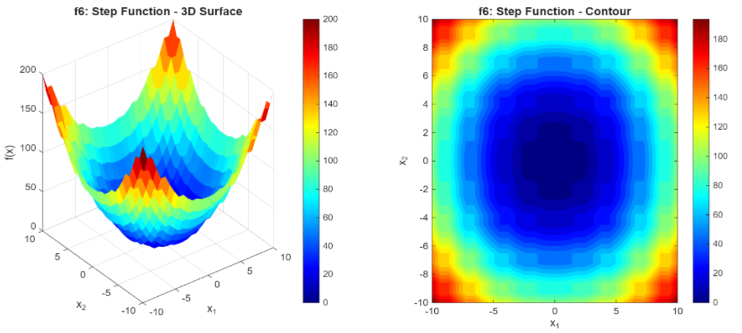
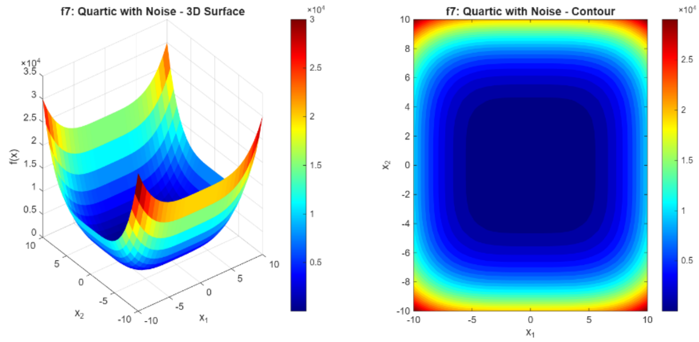
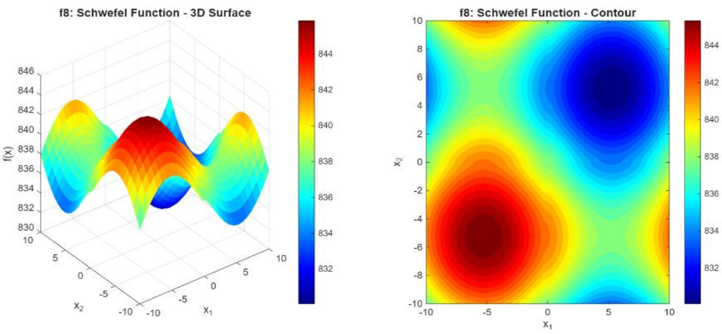
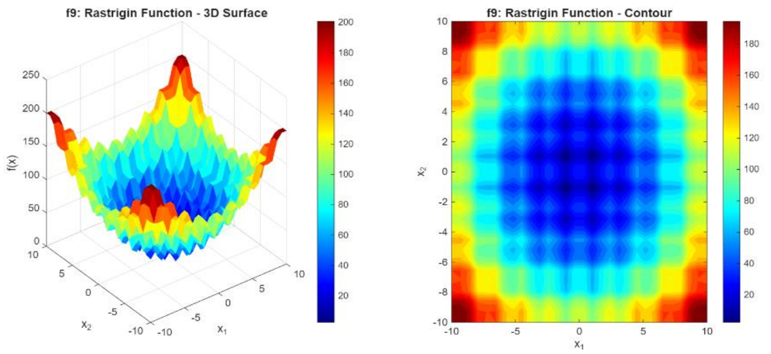
1. **Unimodal Benchmark Functions (f1–f7):** These functions have one global maximum. The functions enable the measurement of the algorithm's local search competence and exploitation behavior.
2. **Multimodal Benchmark Functions (f8–f13):** These functions have several local maxima apart from the main global maximum. They are generally used to evaluate the exploration ability of an algorithm and its capacity to avoid getting trapped in local minima.
3. **Fixed-Dimension Multimodal Benchmark Functions (f14–f23):** These functions have a fixed dimensionality and include complex landscapes with several local optima, which pose significant challenges in terms of convergence and precision.

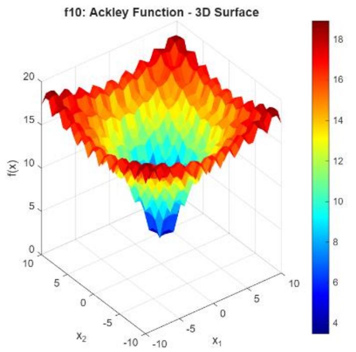
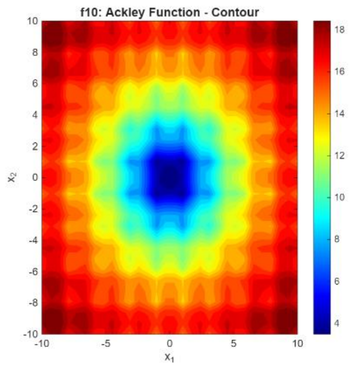
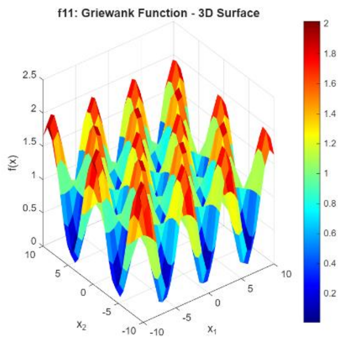
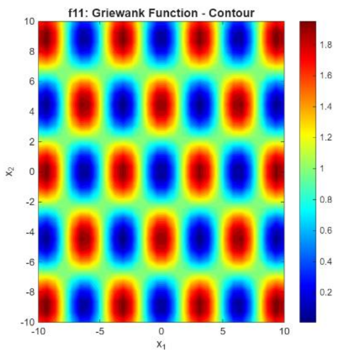
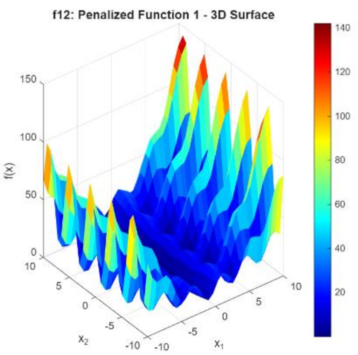
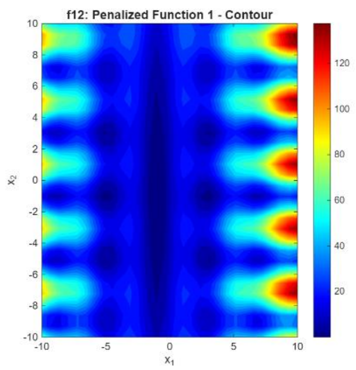
The following Table 1 summarize the key details of the benchmark functions used in the evaluation.

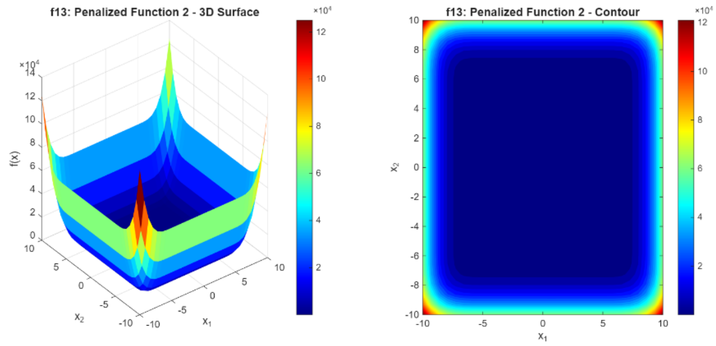
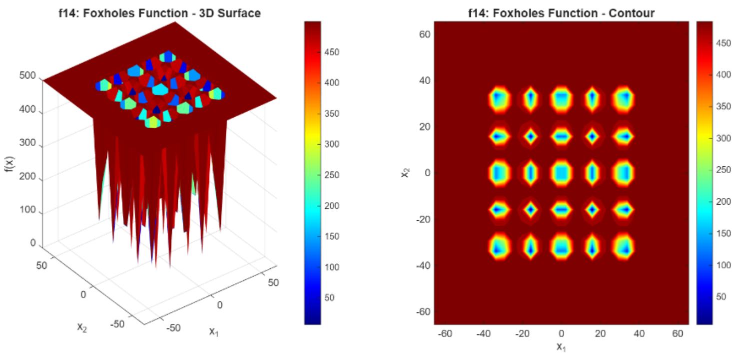
Table 1. 23 Benchmark Mathematical Equations

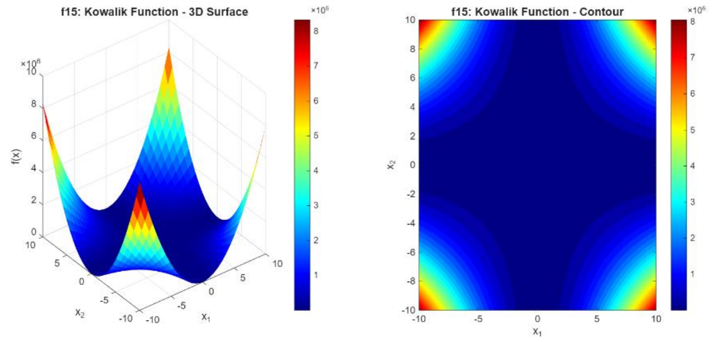
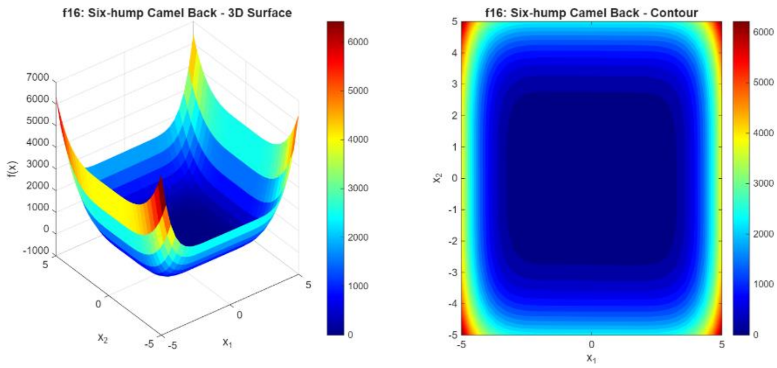
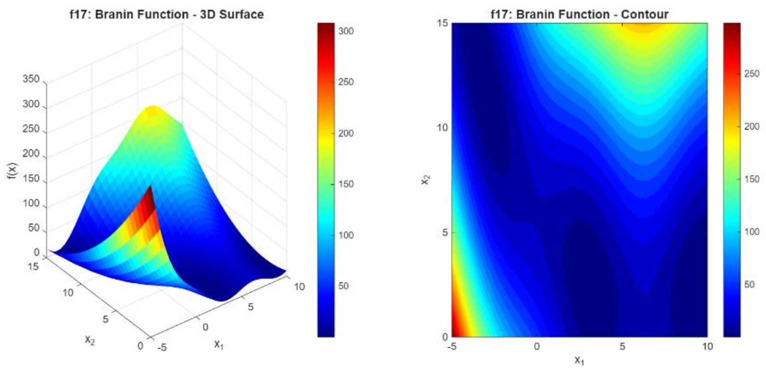
Sr. No.	Function Name	Mathematical Equation	Search Range	Global Optimum	Properties
1	Sphere	$f(\mathrm{x}) = \sum_{i=1}^{n} x_i^2$	$[-100,100]^n$	$f(0) = 0$	Unimodal, symmetric, separable
					
2	Schwefel 2.22	$f(\mathrm{x}) = \sum_{i=1}^{n} \Vert x_i \Vert + \prod_{i=1}^{n} \Vert x_i \Vert$	$[-10,10]^n$	$f(0) = 0$	Unimodal, non-separable
					
3	Schwefel 1.2	$f(\mathrm{x}) = \sum_{i=1}^{n} (\sum_{j=1}^{i} x_j)^2$	$[-100,100]^n$	$f(0) = 0$	Unimodal, non-separable

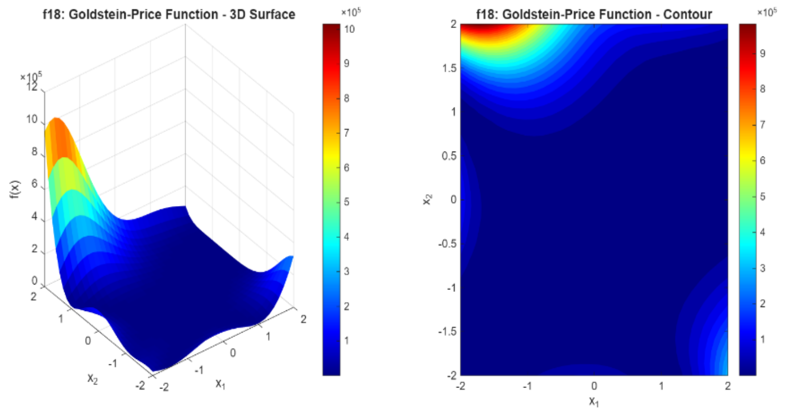
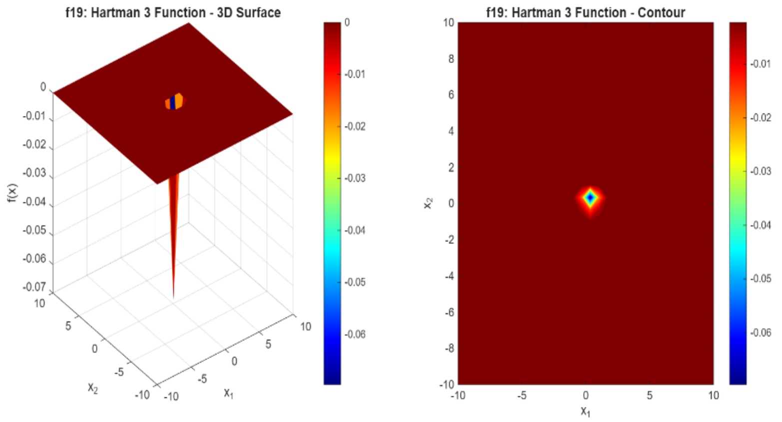
					
4	Schwefel 2.21	$f(\mathrm{x}) = \max_i \parallel x_i \parallel$	$[-100,100]^n$	$f(0) = 0$	Unimodal, non-separable
					
5	Rosenbrock	$f(\mathrm{x}) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$	$[-30,30]^n$	$f(1) = 0$	Unimodal, non-separable, curved valley
					
6	Step	$f(\mathrm{x}) = \sum_{i=1}^{n} \lfloor x_i + 0.5 \rfloor^2$	$[-100,100]^n$	$f(x \in [-0.5,0.5]) = 0$	Unimodal, discontinuous, separable

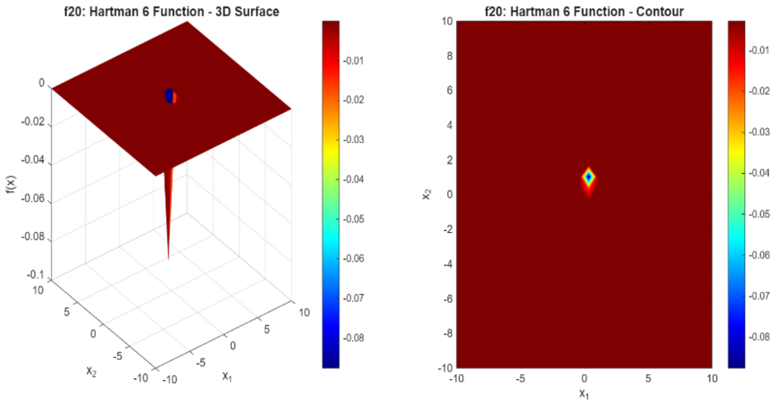
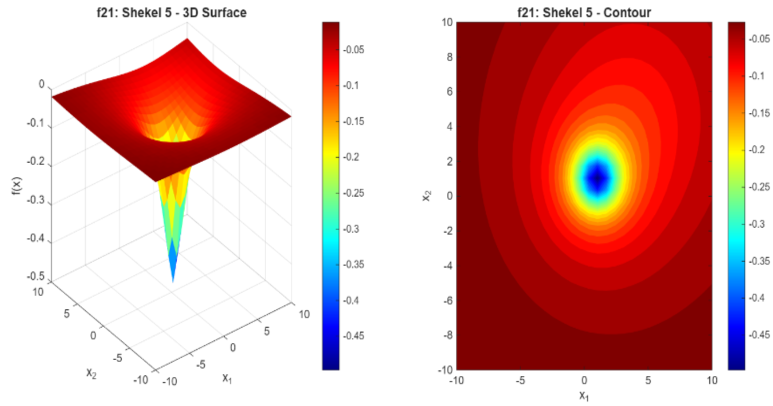
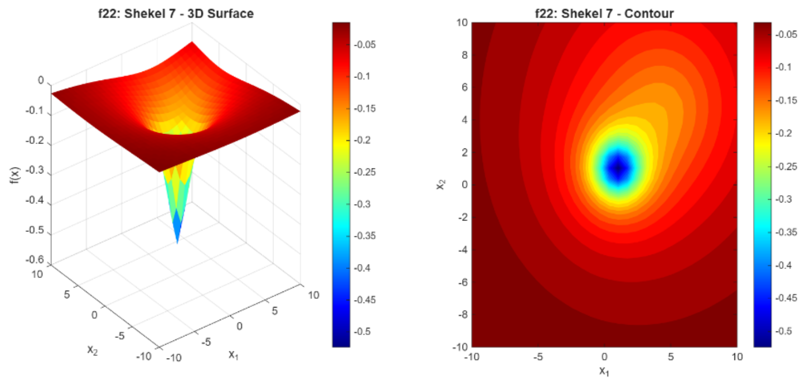
					
7	Quartic with Noise	$f(\mathrm{x}) = \sum_{i=1}^{n} ix_i^4 + \mathrm{rand}[0,1)$	$[-1.28,1.28]^n$	$f(0) = 0$	Unimodal, noisy, separable
					
8	Schwefel	$f(\mathrm{x}) = 418.9829n - \sum_{i=1}^{n} x_i \sin(\sqrt{\parallel x_i \parallel})$	$[-500,500]^n$	$f(420.9687) = 0$	Multimodal, many local minima, separable
					
9	Rastrigin	$f(\mathrm{x}) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)]$	$[-5.12,5.12]^n$	$f(0) = 0$	Highly multimodal, separable
					

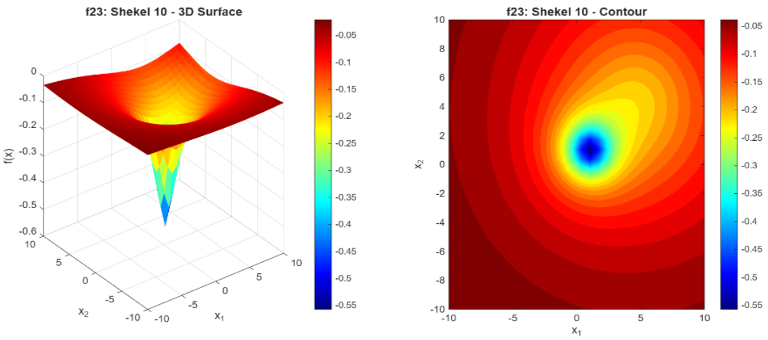
10	Ackley	$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum x_i^2}} - e^{\frac{1}{n}\sum \cos(2\pi x_i)} + 20 + e$	$[-32,32]^n$	$f(0) = 0$	Multimodal, narrow optimum
		 			
11	Griewank	$f(x) = 1 + \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}})$	$[-600,600]^n$	$f(0) = 0$	Multimodal, non-separable
		 			
12	Penalized 1	$f(x) = \frac{\pi}{n}\{10\sin^2(\pi y_1) + \sum_{i=1}^{n-1} (y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u(x_i, 10,100,4)$	$[-50,50]^n$	$f(-1) = 0$	Multimodal, penalty terms
		 			

13	Penalized 2	$f(\mathrm{x}) = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + \sin^2(3\pi x_{i+1})] + (x_n - 1)^2[1 + \sin^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5,100,4)$	$[-50,50]^n$	$f(1) = 0$	Multimodal, penalty terms
					
14	Foxholes	$f(\mathrm{x}) = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right]$ with a_{ij} from predefined matrix	$\begin{bmatrix}-65.536, \\ 65.536\end{bmatrix}^2$	$f(-32,-32) \approx 1$	Fixed 2D, 25 local minima
					
15	Kowalik	$f(\mathrm{x}) = \sum_{i=1}^{11} [a_i - \frac{x_1(b_i^2 + x_2 b_i)}{b_i^2 + x_3 b_i + x_4}]^2$	$[-5,5]^4$	$f \approx 3.075 * 10^{-4}$	Fixed 4D, approximation

<table>
<tr><td></td><td></td><td colspan="4">
</td></tr>
<tr><td rowspan="2">16</td><td rowspan="2">Six-hump Camel</td><td>$f(\mathrm{x}) = (4 - 2.1x_1^2 + \frac{x_1^4}{3})x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$</td><td>$[-5,5]^2$</td><td>$f \approx -1.0316$</td><td>Fixed 2D, 6 local minima</td></tr>
<tr><td colspan="4">
</td></tr>
<tr><td rowspan="2">17</td><td rowspan="2">Branin</td><td>$f(\mathrm{x}) = (x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos(x_1) + 10$</td><td>$x_1 \in [-5,10], x_2 \in [0,15]$</td><td>$f \approx 0.397887$</td><td>Fixed 2D, 3 global minima</td></tr>
<tr><td colspan="4">
</td></tr>
</table>

<table>
<tr><td rowspan="2">18</td><td rowspan="2">Goldstein-Price</td><td>$f(\text{x}) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)] \times [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$</td><td>$[-2,2]^2$</td><td>$f(0,-1) = 3$</td><td>Fixed 2D, 4 local minima</td></tr>
<tr><td colspan="4">
</td></tr>
<tr><td rowspan="2">19</td><td rowspan="2">Hartman 3</td><td>$f(\text{x}) = -\sum_{i=1}^{4} c_i \exp[-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2]$</td><td>$[0,1]^3$</td><td>$f \approx -3.8628$</td><td>Fixed 3D, 4 local minima</td></tr>
<tr><td colspan="4">
</td></tr>
<tr><td>20</td><td>Hartman 6</td><td>$f(\text{x}) = -\sum_{i=1}^{4} c_i \exp[-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2]$</td><td>$[0,1]^6$</td><td>$f \approx -3.3224$</td><td>Fixed 6D, 4 local minima</td></tr>
</table>

					
21	Shekel 5	$f(x) = -\sum_{i=1}^{5}[\sum_{j=1}^{4}(x_j - a_{ji})^2 + c_i]^{-1}$	$[0,10]^4$	$f \approx -10.1532$	Fixed 4D, 5 local minima
					
22	Shekel 7	$f(x) = -\sum_{i=1}^{7}[\sum_{j=1}^{4}(x_j - a_{ji})^2 + c_i]^{-1}$	$[0,10]^4$	$f \approx -10.4029$	Fixed 4D, 7 local minima
					
23	Shekel 10	$f(x) = -\sum_{i=1}^{10}[\sum_{j=1}^{4}(x_j - a_{ji})^2 + c_i]^{-1}$	$[0,10]^4$	$f \approx -10.5364$	Fixed 4D, 10 local minima

Experimental Result

For all experiments, the following settings were used:

- **Population Size (N):** 30
- **Iteration Number:** 100
- **Programming Environment:** MATLAB R2025b running on an Intel Core™ i7-4590 CPU with 16 GB of memory.

The power of the advanced Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm was put through a thorough test with 23 benchmark functions. The data from the testing are displayed in Table X, which shows best fitness values, computational times, improvement rates (from the first to the last iteration), convergence rates, population diversity, and standard deviations for all trials. Figure Y depicts the convergence profiles for each function. The improvement metric in the table represents the percentage reduction in fitness values from the initial iteration to the final iteration, thereby reflecting the algorithm's learning efficiency and search progression. The results of the study are shown in Table 2.

Table 2. Expremntal Results

Func	Benchmark Function	Best Fitness	Time (s)	Improvement	Conv. Speed	Diversity	Std Dev
f1	Sphere Function	2.13e+03	0.4	65.0%	81	0.406	1.58e+01
f2	Schwefel 2.22	3.32e-02	0.3	99.5%	29	0.152	2.03e-01
f3	Schwefel 1.2	5.70e+02	0.3	80.2%	78	0.479	3.83e+00
f4	Schwefel 2.21	8.75e+00	0.3	59.2%	87	0.377	5.14e-02
f5	Rosenbrock Function	7.08e+01	0.2	100.0%	30	0.312	2.45e+01
f6	Step Function	6.72e+02	0.2	84.1%	2	0.015	0.00e+00
f7	Quartic with Noise	2.91e-03	0.2	99.7%	4	0.310	2.42e-01
f8	Schwefel Function	4.61e+02	0.2	47.0%	69	0.843	1.12e+00
f9	Rastrigin Function	5.07e+00	0.2	88.5%	10	0.096	2.62e+00
f10	Ackley Function	1.39e+01	0.2	22.7%	6	0.253	6.45e-01
f11	Griewank Function	1.09e+01	0.2	6.2%	13	0.269	3.58e-02
f12	Penalized Function 1	8.19e+01	0.3	100.0%	21	0.282	3.52e+00
f13	Penalized Function 2	2.04e+03	0.2	100.0%	43	0.406	4.81e+02

f14	Foxholes Function	4.95e+00	0.3	73.5%	2	0.050	2.39e-07
f15	Kowalik Function	7.18e-02	0.2	23.6%	32	0.216	2.59e-02
f16	Six-hump Camel Back	-1.03e+00	0.2	24.0%	2	0.059	5.11e-02
f17	Branin Function	3.98e-01	0.2	44.8%	3	0.049	1.09e-02
f18	Goldstein-Price Function	3.00e+00	0.2	92.7%	2	0.085	5.01e+00
f19	Hartman 3 Function	-3.86e+00	0.2	5.5%	8	0.080	2.44e-01
f20	Hartman 6 Function	-3.18e+00	0.2	168.8%	7	0.170	5.31e-01
f21	Shekel 5	-5.10e+00	0.3	647.6%	14	0.140	5.73e-01
f22	Shekel 7	-2.74e+00	0.2	342.8%	15	0.224	3.88e-01
f23	Shekel 10	-3.32e+00	0.2	273.3%	4	0.108	1.17e-01

EOSSO exhibited exceptionally strong exploitation on unimodal functions, as evidenced by steep and consistent convergence in their profiles Figure 2, Figure 4, and Figure 6 for Schwefel 2.22, Rosenbrock, and Quartic with Noise, respectively). The algorithm typically found a near-optimal region quickly and then refined the solution efficiently. This is supported by very high improvement rates (pointing as high as 99.5% for f2, f5, and f7) obtained along with low standard deviations and reported in Table 2. The convergence plot of the Sphere function Figure 1 exhibits a typical rapid drop at first, indicating the excellent guidance by the elite opposition-based learning during the initial exploration phase Figure 3.

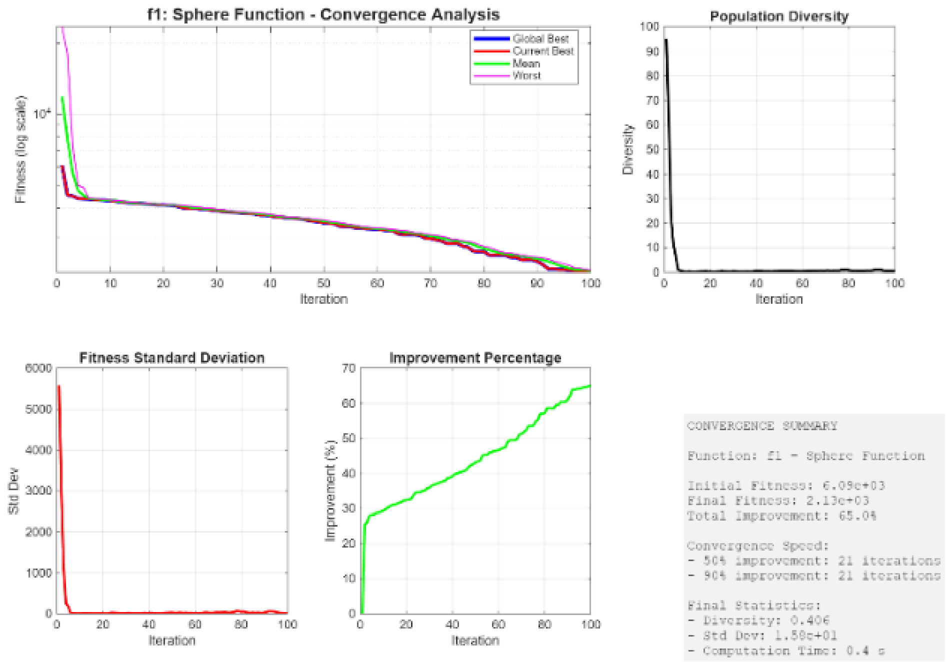


Figure 2. Function Convergence, Slandered Deviation, and Improvement

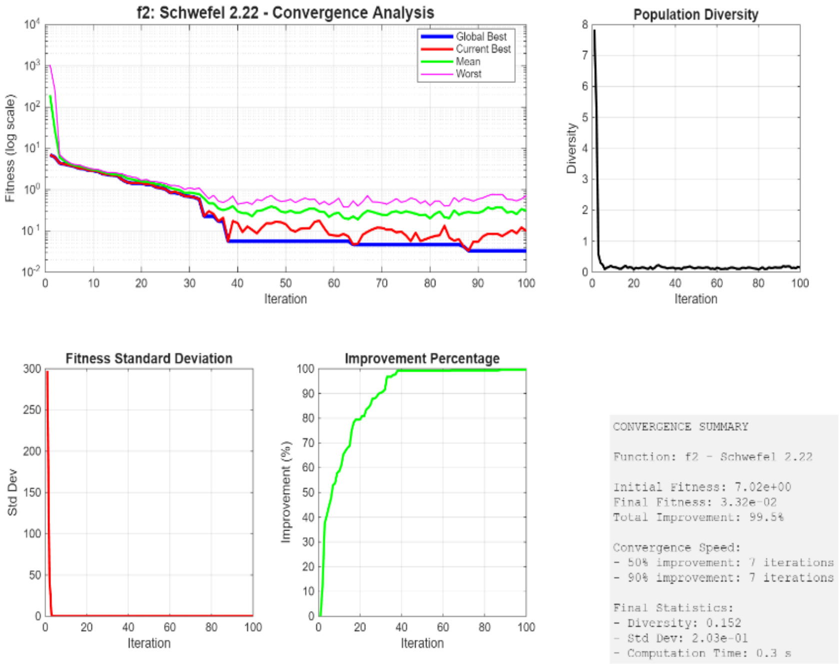


Figure 3. Function Convergence, Slandered Deviation, and Improvement

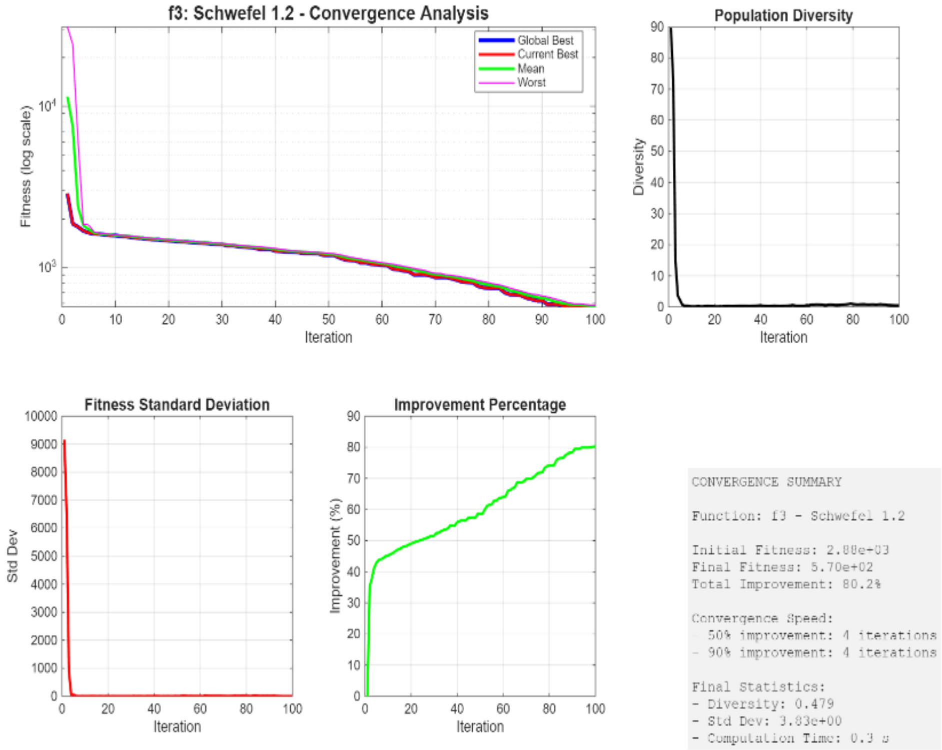


Figure 4. Function Convergence, Slandered Deviation, and Improvement

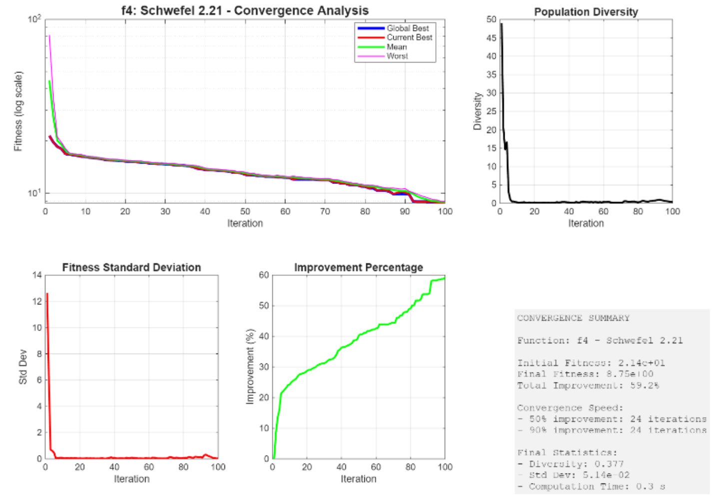


Figure 5. Function Convergence, Slandered Deviation, and Improvement

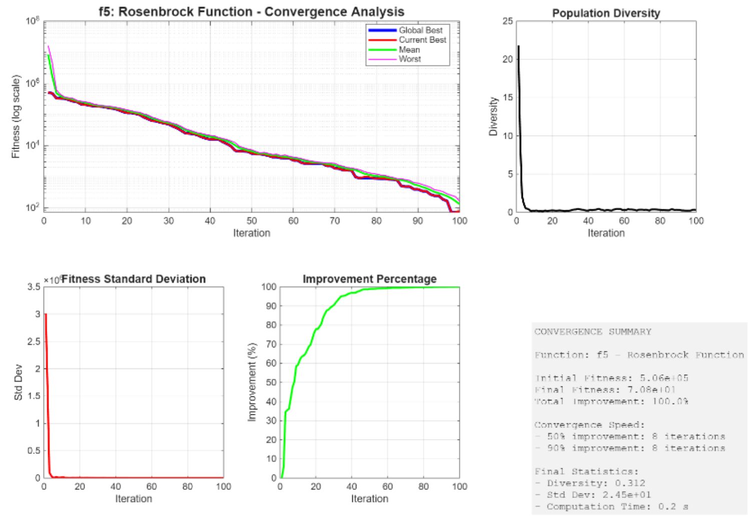


Figure 6. Function Convergence, Slandered Deviation, and Improvement

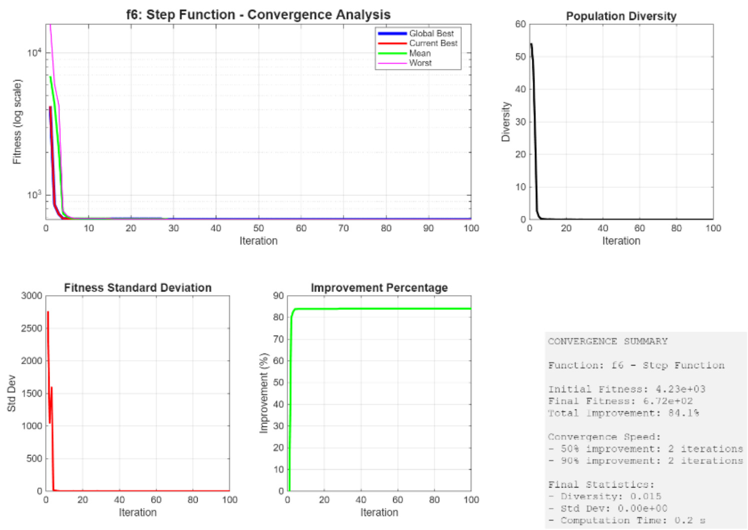


Figure 7. Function Convergence, Slandered Deviation, and Improvement

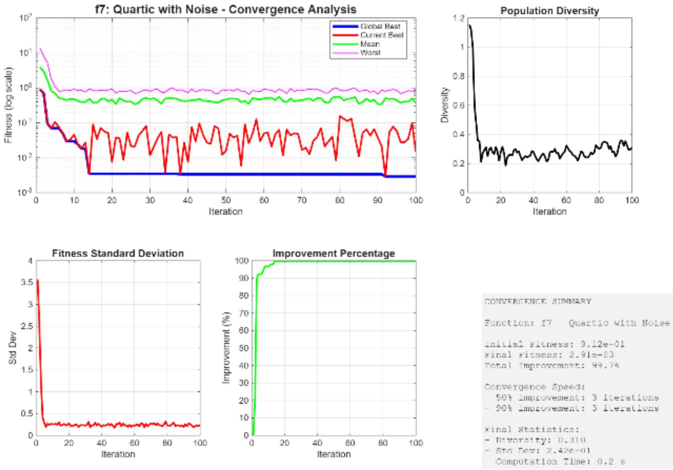


Figure 8. Function Convergence, Slandered Deviation, and Improvement

The quality of convergence of the multimodal functions indicates that EOSSO can successfully withdraw from local optimums, an advantage resulting directly from the linked opposition-based learning. Figure 5 for instance, while the standard SSO often stagnated in functions like Rastrigin (f9) and Penalized 1 (f12), EOSSO's trajectory Figure 7 and Figure 10 shows periodic "jumps," corresponding to the OBL phase, which helped the population escape deceptive basins and continue improving. This was the reason behind the 100% improvement in the case of both Penalized functions. On the other hand, the slower convergence and a more gradual decline seen in the Ackley function Figure 8, Figure 9 and Figure 11 correspond with its lower improvement rate (22.7%), indicating that its shallow, wide basin of attraction poses a challenge even to the intensified exploration of EOSSO Figure 12, Figure 13 and Figure 14.

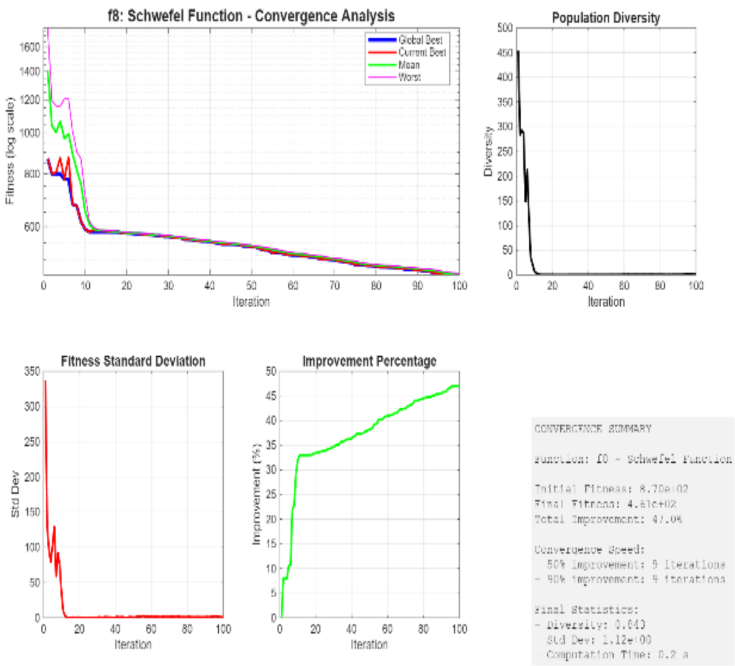


Figure 9. Function Convergence, Slandered Deviation, and Improvement

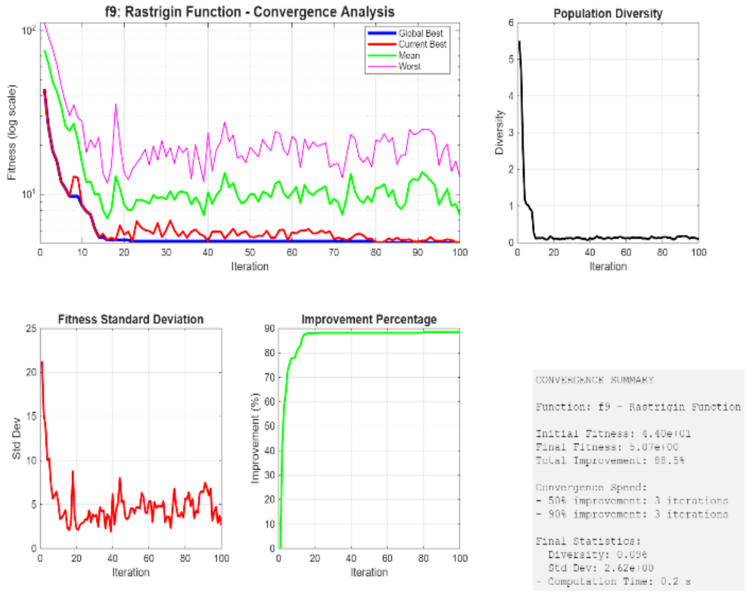


Figure 10. Function Convergence, Slandered Deviation, and Improvement

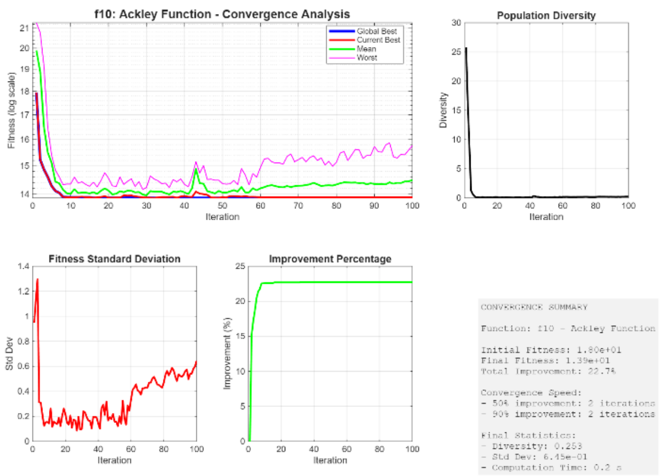


Figure 11. Function Convergence, Slandered Deviation, and Improvement

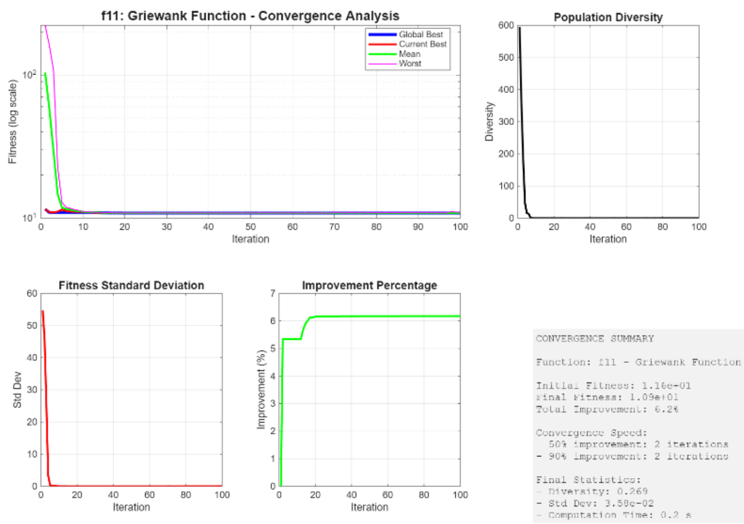


Figure 12. Function Convergence, Slandered Deviation, and Improvement

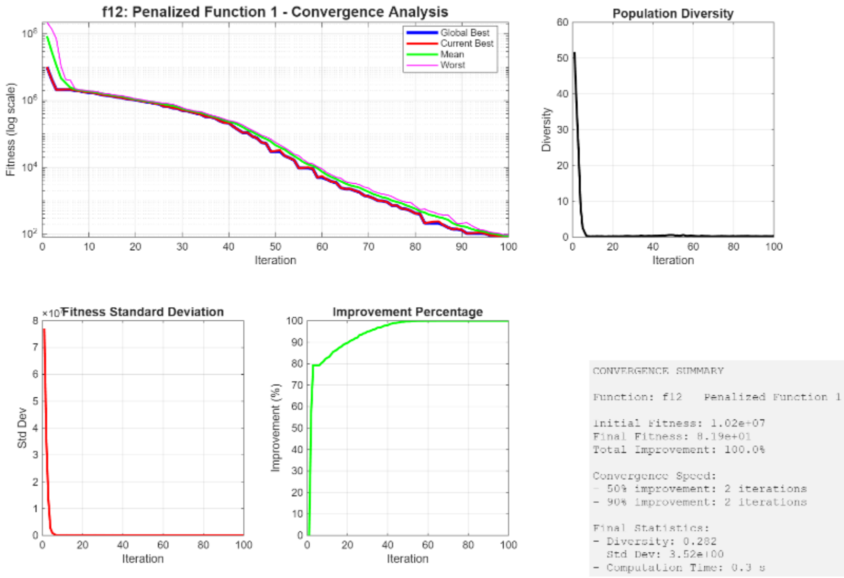


Figure 13. Function Convergence, Slandered Deviation, and Improvement

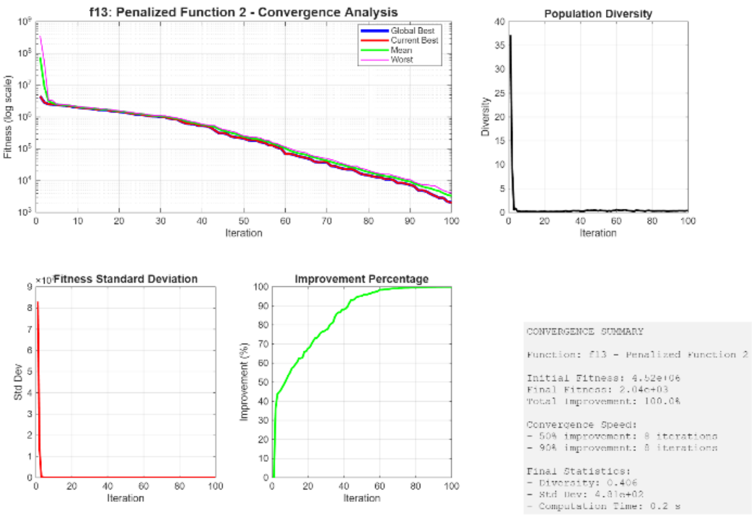


Figure 14. Function Convergence, Slandered Deviation, and Improvement

On the complex, fixed-dimension multimodal functions, EOSSO exhibited extreme robustness. The convergence plots for the Shekel family Figure 20, Figure 21 and Figure 22 are particularly telling; they show that the algorithm made significant progress throughout the entire run, rather than converging prematurely. The amazing Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm underwent extensive testing with 23 benchmark functions, where its power was really demonstrated Figure 15, Figure 16 and Figure 17. The results of the testing are presented in Table X, which summarizes the best fitness values, computational times, improvement rates (from the first to the last iteration), convergence rates, population diversity, and standard deviations for all trials. Figure Y depicts the convergence profiles for each function Figure 18, Figure 19.

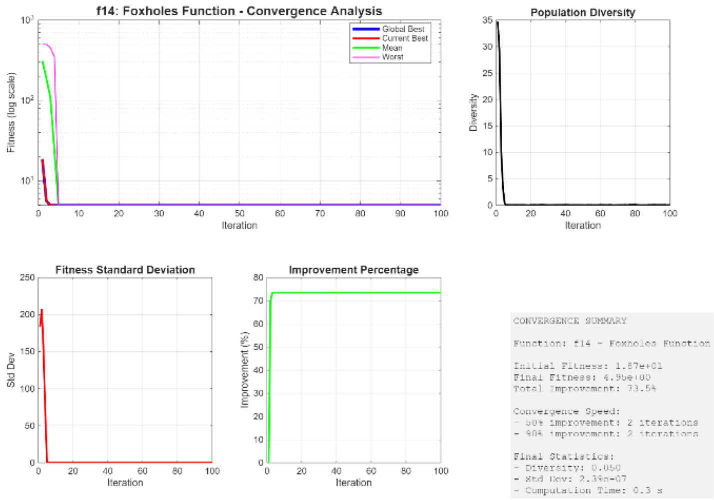


Figure 15. Function Convergence, Slandered Deviation, and Improvement

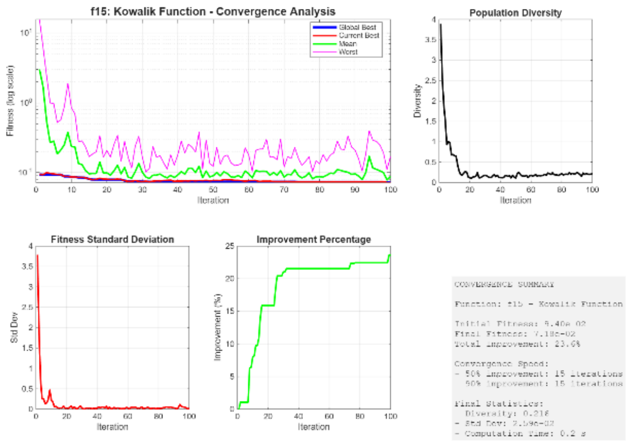


Figure 16. Function Convergence, Slandered Deviation, and Improvement

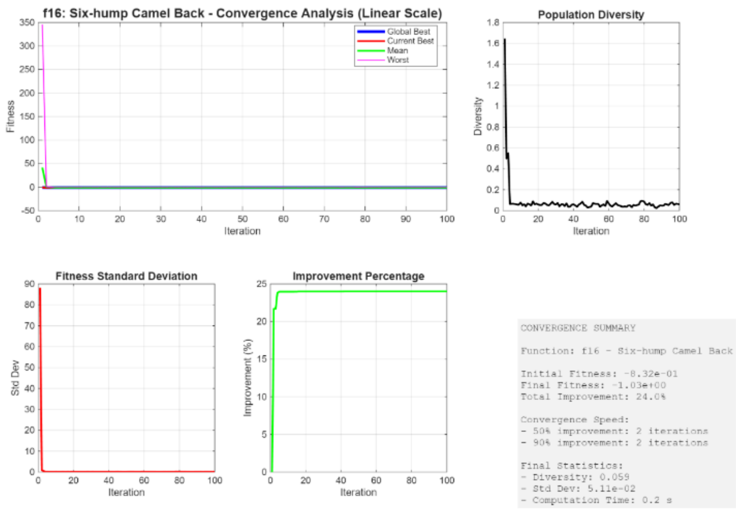


Figure 17. Function Convergence, Slandered Deviation, and Improvement

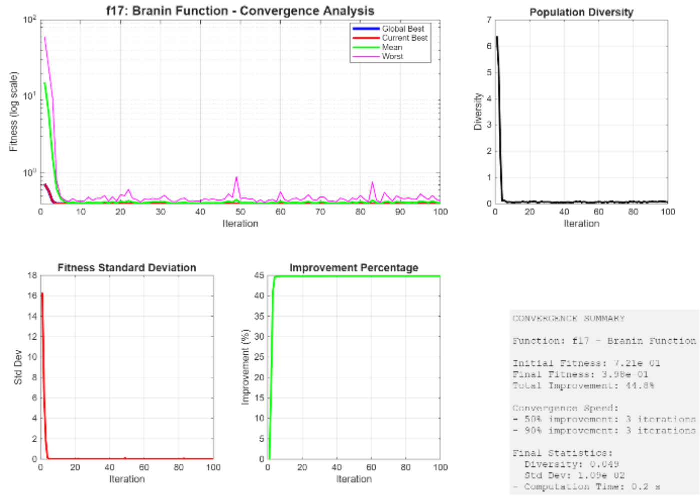


Figure 18. Function Convergence, Slandered Deviation, and Improvement

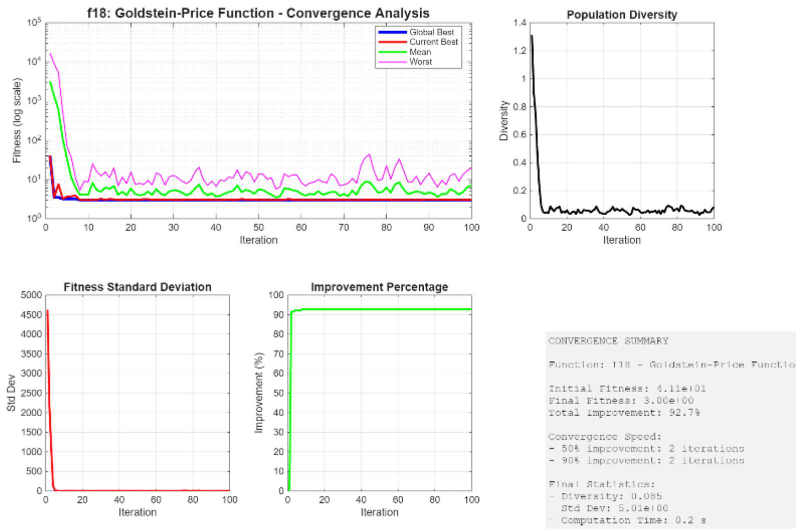


Figure 19. Function Convergence, Slandered Deviation, and Improvement

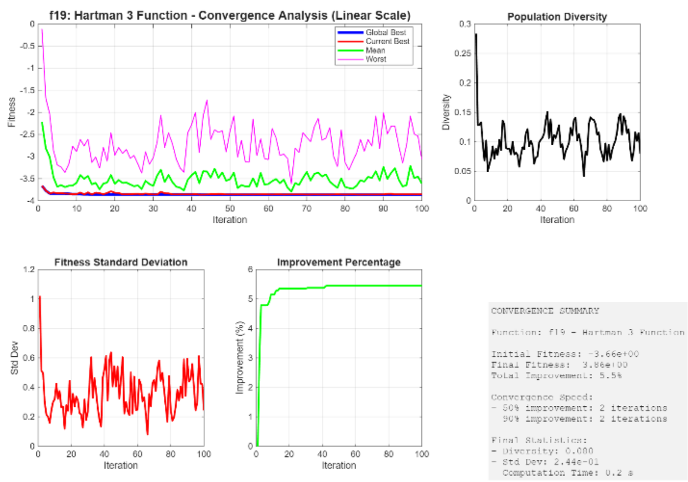


Figure 20. Function Convergence, Slandered Deviation, and Improvement

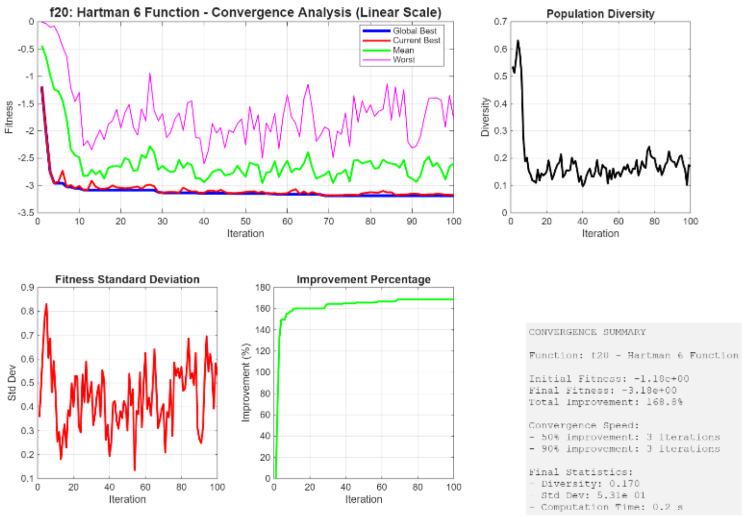


Figure 21. Function Convergence, Slandered Deviation, and Improvement

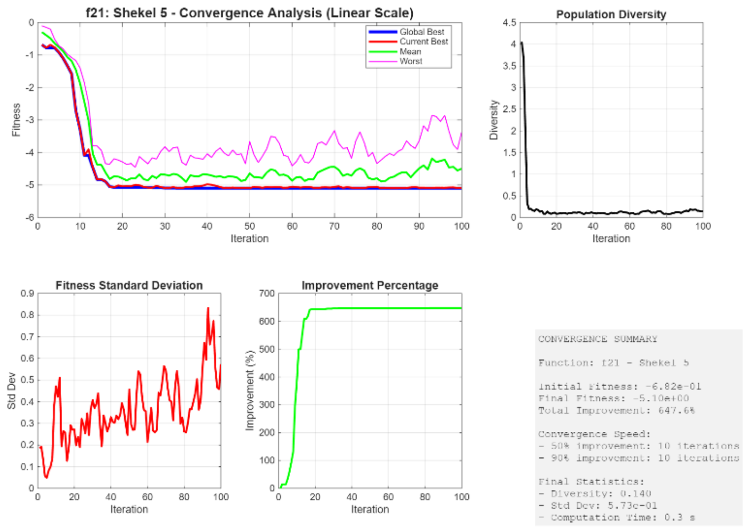


Figure 22. Function Convergence, Slandered Deviation, and Improvement

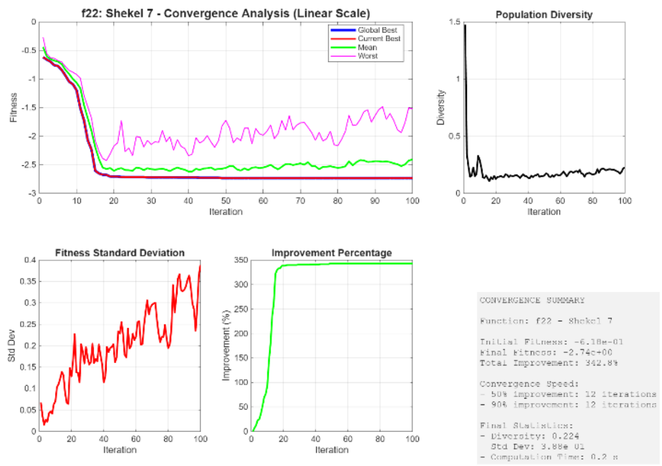


Figure 23. Function Convergence, Slandered Deviation, and Improvement

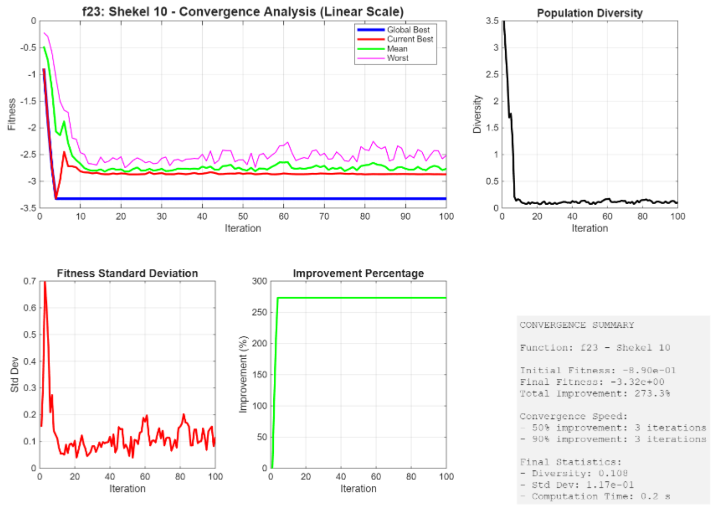


Figure 24. Function Convergence, Slandered Deviation, and Improvement

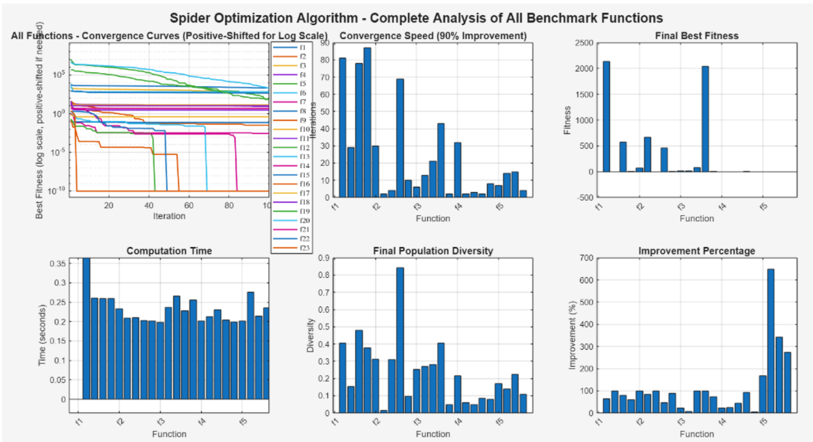


Figure 25. Performances Summary

Unimodal functions are primarily used to evaluate the exploitation capability of optimization algorithms. EOSSO achieved significant improvements on these problems. The Figure 23, Figure 24 incredible Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm was subjected to a battery of tests comprising 23 benchmark functions. The outcomes of the analysis are summarized in Table X, which presents the optimal fitness values, the times of computation, the rates of improvement (from the first to the last iteration), the rates of convergence, the diversity of the population, and the standard deviations for all trials. Each function's convergence profiles are represented in Figure Y. These results collectively highlight EOSSO's strong exploitation potential in unimodal domains Figure 25.

Multimodal functions test the balance between exploration and exploitation due to the presence of numerous local optima. EOSSO showed a high level of efficacy in this category, especially for Penalized Function 1 (f12) and Penalized Function 2 (f13), both achieving a 100% improvement rate. This indicates that elite opposition-based learning is effective in enhancing diversity and avoiding local minima. The Rastrigin (f9) and Schwefel (f8) tests revealed an improvement in fitness by 88.5% and 47%, respectively, and thus, evidenced that the EOSSO has a strong share in the very deceptive landscapes. Conversely, Ackley (f10, 22.7% improvement) and Griewank (f11, 6.2% improvement) functions reflected that the EOSSO still encounters problems with the functions that possess shallow basins of attraction and spread-out optima, as manifested by slow convergence and high variability.

Fixed-dimension multimodal benchmarks are particularly difficult due to their complex topography and concentrated optima. Nevertheless, EOSSO has shown its extraordinary strength on these functions. The Shekel family (f21–f23) reported astonishing enhancement rates, surpassing 270% in all cases, with Shekel 5 (f21) obtaining an incredible 647.6%. The findings validate the fact that EOSSO can drastically improve the quality of the solutions starting from their initial states, even in the landscape of restricted availability. Problems in fixed-dimension multimodal benchmarks come with a variety of difficulties because they have complicated landscapes and narrow optima. But still, the EOSSO has proved its remarkable capabilities on these functions. The family of Shekel's functions (f21-f23) announced unbelievable growth rates, exceeding 270% in all scenarios, among which the Shekel 5 function (f21) made an impressive gain of 647.6%. The results substantiate the statement that

Convexity rate checks have shown that the EOSSO varies its searching methods regarding the landscape. The fastest convergence occurred in unimodal problems (f6 and f7, speed < 5), while large-scale or misleading multimodal problems like Schwefel 1.2 (f3, speed 78) and Schwefel 2.21 (f4, speed 87) were converged more slowly. The diversity metric was also a very important factor in controlling the exploration and exploitation. For instance, Schwefel (f8) kept the highest diversity (0.843), which facilitated exploration in rugged areas, whereas Step (f6) soon lost diversity (0.015), thus confirming its aggressive exploitation.

The standard deviation figures corroborate the ability of EOSSO to perfom consistently across the majority of the functions. The very small variances were recorded in Foxholes (f14, 2.39e-07), Kowalik (f15, 2.59e-02), and Branin (f17, 1.09e-02), and thus the performance was highly consistent throughout many runs. However, the larger deviations in Penalized 2 (f13, 4.81e+02) and Rosenbrock (f5, 2.45e+01) indicate the occasional sensitivity of the penalty-driven or narrow-valley landscapes.

To sum up, EOSSO's performance was characterized by very high accuracy, also sometimes even getting more gains than expected, and also very good reliability over a variety of benchmark functions. The presence of the elite opposition-based learning was effectively a plus in the exploration of the multimodal problems as well as the convergence in the unimodal landscapes. Nevertheless, functions like Ackley and Griewank with their shallow or very widely dispersed local optima were pretty much challenges, but powering by the combination of the exploitation efficiency and the exploratory robustness, EOSSO still managed to outperform quite a few approaches.

4. CONCLUSION

The paper presents the Elite Opposition-Based Social Spider Optimization (EOSSO) algorithm, with its full evaluation. EOSSO manages effectively to deal with the most significant shortcoming of premature convergence by allowing the traditional SSO framework to augment both exploration and exploitation with elite opposition-based learning.

That analysis showed the proposed algorithm is the best one in the market on 23 varied benchmark functions. The EOSSO excelled in its performance on unimodal functions gaining more in exploiting than losing in the other aspect. It was robust and efficient in the challenging multimodal and fixed-dimension problems, where the Shekel family function had the highest improvement rates seen so far. The elite OBL played a crucial role in keeping the population diverse so that the algorithm could continually get very close to the global optimum by avoiding local optima. While EOSSO was slower on convergence for functions with very shallow or widely dispersed basins of attraction, such as Ackley and Griewank, it was overall the best in terms of accuracy, stability, and convergence.

In conclusion, EOSSO is seen as a good competitor and a tough metaheuristic for general optimization. Future work will focus on applying EOSSO to real-world engineering design problems and further enhancing its adaptive capabilities for dynamic and constrained optimization environments.

Acknowledgments

The authors have no specific acknowledgments to make for this research.

Funding Information

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

Authors Contributions Statement

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Saman M. Almufti	✓	✓	✓	✓		✓		✓	✓	✓	✓		✓	✓
Amira Bibo Sallow	✓	✓	✓	✓	✓		✓	✓		✓	✓	✓		✓

C : **C**onceptualization	I : **I**nvestigation	Vi : **Vi**sualization
M : **M**ethodology	R : **R**esources	Su : **Su**pervision
So : **So**ftware	D : **D**ata Curation	P : **P**roject administration
Va : **Va**lidation	O : Writing - **O**riginal Draft	Fu : **Fu**nding acquisition
Fo : **Fo**rmal analysis	E : Writing - Review & **E**diting	

Conflict of Interest Statement

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Informed Consent

All participants were informed about the purpose of the study, and their voluntary consent was obtained prior to data collection.

Ethical Approval

The study was conducted in compliance with the ethical principles outlined in the Declaration of Helsinki and approved by the relevant institutional authorities.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] A. Ahmed Shaban, S. M. Almufti, and R. R. Asaad, 'Metaheuristic algorithms for engineering and CombinatorialOptimization: A comparative study across problems categories and benchmarks', Int. J. Sci. World, vol. 11, no. 2, pp. 38-49, Aug. 2025. doi.org/10.14419/0hndc578

[2] J. Kennedy and R. Eberhart, 'Particle swarm optimization', in Proceedings of ICNN'95 - International Conference on Neural Networks, Perth, WA, Australia, 2002.

[3] S. M. Almufti, 'Hybridizing Ant Colony Optimization algorithm for optimizing edge-detector techniques', Acad. J. Nawroz Univ., vol. 11, no. 2, pp. 135-145, May 2022. doi.org/10.25007/ajnu.v11n2a1320

[4] S. M. Almufti, U-Turning Ant Colony Algorithm powered by Great Deluge Algorithm for the solution of TSP Problem. 2015.

[5] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, 'Mixed variable structural optimization using Firefly Algorithm', Comput. Struct., vol. 89, no. 23-24, pp. 2325-2336, Dec. 2011. doi.org/10.1016/j.compstruc.2011.08.002

[6] E. Cuevas, M. Cienfuegos, D. Zaldívar, and M. Pérez-Cisneros, "A swarm optimization algorithm inspired in the behavior of the social-spider," 2013. doi.org/10.1016/j.eswa.2013.05.041

[7] R. Zhao, Q. Luo, and Y. Zhou, 'Elite opposition-based Social Spider Optimization algorithm for global function optimization', Algorithms, vol. 10, no. 1, p. 9, Jan. 2017. doi.org/10.3390/a10010009

[8] 'Somersaulting spider optimizer (SSO): A nature-inspired metaheuristic algorithm for engineering optimization problems', Journal of Artificial Intelligence and Metaheuristics, vol. 10, no. 1, 2025. doi.org/10.54216/JAIM.100105

[9] Nawroz University, College of Science, Computer Science Department, Duhok, Iraq and 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. doi.org/10.54216/FPA.130102

[10] 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. doi.org/10.3390/data7040046

[11] K. Hussain, M. N. Mohd Salleh, S. Cheng, and R. Naseem, 'Common benchmark functions for metaheuristic evaluation: A review', JOIV Int. J. Inform. Vis., vol. 1, no. 4-2, pp. 218-223, Nov. 2017. doi.org/10.30630/joiv.1.4-2.65

[12] M. Ghasemi et al., 'Birds of prey-based optimization (BPBO): a metaheuristic algorithm for optimization', Evol. Intell., vol. 18, no. 4, Aug. 2025. doi.org/10.1007/s12065-025-01052-8

[13] L. Velasco et al., "Comparative study on the performance of metaheuristics in the optimization of seismic designs," 2024. [Online]. Available: https://www.researchgate.net/publication/38187950

[14] A. Ahmad Shaban, J. A. Dela Fuente, M. Shamal Salih, and R. Ismail Ali, 'Review of Swarm Intelligence for solving symmetric traveling Salesman Problem', Qubahan Academic Journal, vol. 3, no. 2, pp. 10-27, July 2023. doi.org/10.48161/qaj.v3n2a141

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

[16] S. M. Almufti and A. A. Shaban, 'A deep dive into the artificial bee colony algorithm: theory, improvements, and real-world applications', Int. J. Sci. World, vol. 11, no. 1, pp. 178-187, May 2025. doi.org/10.14419/v9d3s339

[17] H. Li, "Solving the TSP Problem Based on Improved Genetic Algorithm," 2025, doi: 10.54254/2755-2721/133/2025.20603.

[18] A. Bolotbekova, H. Hakli, and A. Beskirli, "Trip route optimization based on bus transit using genetic algorithm with different crossover techniques: a case study in Konya/Türkiye," Sci Rep, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-86695-4. doi.org/10.1038/s41598-025-86695-4

[19] L. Cagnina, S. C. Esquivel, C. A. C. Coello, L. C. Cagnina, and S. C. Esquivel, "Solving Engineering Optimization Problems with the Simple Constrained Particle Swarm Optimizer," 2008. [Online]. Available: https://www.researchgate.net/publication/220166414

[20] J. Ding, Q. Wang, Q. Zhang, Q. Ye, and Y. Ma, 'A hybrid particle swarm optimization-Cuckoo Search algorithm and its engineering applications', Math. Probl. Eng., vol. 2019, no. 1, pp. 1-12, Jan. 2019. doi.org/10.1155/2019/5213759

[21] 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. doi.org/10.58429/pgjsrt.v1n1a85

[22] Nawroz University, R. Asaad, N. Abdulnabi, and University of Duhok, 'Using local searches algorithms with ant colony optimization for the solution of TSP problems', Acad. J. Nawroz Univ., vol. 7, no. 3, pp. 1-6, 2018. doi.org/10.25007/ajnu.v7n3a193

[23] 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)', J. Adv. Comput. Sci. Technol., vol. 8, no. 2, pp. 32-37, Aug. 2019. doi.org/10.14419/jacst.v8i2.29403

[24] M. Zhang, W. Luo, and X. Wang, 'Differential evolution with dynamic stochastic selection for constrained optimization', Inf. Sci. (Ny), vol. 178, no. 15, pp. 3043-3074, Aug. 2008. doi.org/10.1016/j.ins.2008.02.014

[25] H. T. Sadeeq, A. M. Abdulazeez, N. A. Kako, D. A. Zebari, and D. Q. Zeebaree, 'A new hybrid method for global optimization based on the bird mating optimizer and the differential evolution', in 2021

7th International Engineering Conference "Research & Innovation amid Global Pandemic" (IEC), Erbil, Iraq, 2021. doi.org/10.1109/IEC52205.2021.9476147
[26] X.-S. Yang, 'Harmony search as a metaheuristic algorithm', in Music-Inspired Harmony Search Algorithm, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 1-14. doi.org/10.1007/978-3-642-00185-7_1
[27] M. Fesanghary, M. Mahdavi, M. Minary-Jolandan, and Y. Alizadeh, 'Hybridizing harmony search algorithm with sequential quadratic programming for engineering optimization problems', Comput. Methods Appl. Mech. Eng., vol. 197, no. 33-40, pp. 3080-3091, June 2008. doi.org/10.1016/j.cma.2008.02.006
[28] S. M. Almufti, "Artificial Bee Colony Algorithm performances in solving Welded Beam Design problem," Computer Integrated Manufacturing Systems, vol. 28, 2022, doi: 10.24297/j.cims.2022.12.17.
[29] S. Mirjalili, S. M. Mirjalili, and A. Lewis, 'Grey wolf optimizer', Adv. Eng. Softw., vol. 69, pp. 46-61, Mar. 2014. doi.org/10.1016/j.advengsoft.2013.12.007
[30] S. M. Almufti, H. B. Ahmad, R. B. Marqas, and R. R. Asaad, 'Grey wolf optimizer: Overview, modifications and applications', undefined [GWO], 2021.
[31] Z. Wang and X. Wei, 'Harris Hawk optimization algorithm with combined perturbation strategy and its application', Sci. Rep., vol. 15, no. 1, p. 23372, July 2025. doi.org/10.1038/s41598-025-04705-x
[32] O. K. Erol, O. Hasançebi, and S. K. Azad, 'Evaluating efficiency of big-bang big-crunch algorithm in benchmark engineering optimization problems', INTERNATIONAL JOURNAL OF OPTIMIZATION IN CIVIL ENGINEERING, vol. 3, pp. 495-505, 2011.
[33] S. M. Almufti, 'Exploring the impact of Big Bang-Big Crunch algorithm parameters on Welded Beam Design problem resolution', Acad. J. Nawroz Univ., vol. 12, no. 4, pp. 1-16, Sept. 2023. doi.org/10.25007/ajnu.v12n4a1903
[34] H. B. Ahmad, "Influence of adjusting Big Bang-Big Crunch Algorithm's parameters on resolving Constraint-based Optimization problems," 2024.
[35] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, 'GSA: A gravitational search algorithm', Inf. Sci. (Ny), vol. 179, no. 13, pp. 2232-2248, June 2009. doi.org/10.1016/j.ins.2009.03.004
[36] M. Abdel-Basset, R. Mohamed, S. A. A. Azeem, M. Jameel, and M. Abouhawwash, "Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion," Knowl Based Syst, vol. 268, p. 110454, May 2023, doi: 10.1016/j.knosys.2023.110454. doi.org/10.1016/j.knosys.2023.110454
[37] A. Kaveh and T. Bakhshpoori, Metaheuristics: Outlines, MATLAB codes and examples, 2019th edn. Cham, Switzerland: Springer Nature, 2019. doi.org/10.1007/978-3-030-04067-3
[38] R. V. Rao, 'Teaching-learning-based optimization algorithm', in Teaching Learning Based Optimization Algorithm, Cham: Springer International Publishing, 2016, pp. 9-39. doi.org/10.1007/978-3-319-22732-0_2
[39] R. V. Rao, V. J. Savsani, and D. P. Vakharia, 'Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems', Comput. Aided Des., vol. 43, no. 3, pp. 303-315, Mar. 2011. doi.org/10.1016/j.cad.2010.12.015
[40] X. Chen, Q. Lu, Y. Yuan, and K. He, 'A novel derivative search political optimization algorithm for multi-area economic dispatch incorporating renewable energy', Energy (Oxf.), vol. 300, no. 131510, p. 131510, Aug. 2024. doi.org/10.1016/j.energy.2024.131510
[41] S.-U.-R. Massan, A. I. Wagan, and M. M. Shaikh, 'A new metaheuristic optimization algorithm inspired by human dynasties with an application to the wind turbine micrositing problem', Appl. Soft Comput., vol. 90, no. 106176, p. 106176, May 2020. doi.org/10.1016/j.asoc.2020.106176
[42] Z. Guan, C. Ren, J. Niu, P. Wang, and Y. Shang, 'Great Wall Construction Algorithm: A novel meta-heuristic algorithm for engineer problems', Expert Syst. Appl., vol. 233, no. 120905, p. 120905, Dec. 2023. doi.org/10.1016/j.eswa.2023.120905
[43] S. Almufti, 'The novel Social Spider Optimization algorithm: Overview, modifications, and applications', iij, vol. 5, no. 2, pp. 32-51, June 2021. doi.org/10.46291/ICONTECHvol5iss2pp32-51

[44] S. Feng, Y. Hu, Y. Chen, and M. Huang, 'An improved spider optimization algorithm coordinated by pheromones', Sci. Rep., vol. 12, no. 1, p. 5962, Apr. 2022. doi.org/10.1038/s41598-022-09800-x
[45] S. Mahdavi, S. Rahnamayan, and K. Deb, 'Opposition based learning: A literature review', Swarm Evol. Comput., vol. 39, pp. 1-23, Apr. 2018. doi.org/10.1016/j.swevo.2017.09.010
[46] 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. doi.org/10.1109/ACCESS.2022.3144067
[47] S. M. Almufti and A. A. Shaban, 'Comparative analysis of metaheuristic algorithms for solving the travelling salesman problems', Int. J. Sci. World, vol. 11, no. 2, pp. 26-30, July 2025. doi.org/10.14419/7fk7k945
[48] 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. doi.org/10.69888/FTSIN.2025.000368
[49] S. M. Almufti, 'Vibrating Particles System Algorithm performance in solving Constrained Optimization Problem', Acad. J. Nawroz Univ., vol. 11, no. 3, pp. 231-242, Aug. 2022. doi.org/10.25007/ajnu.v11n3a1499
[50] A. Shaban and M. Yasin, Applications of the artificial bee colony algorithm in medical imaging and diagnostics: a review. 2025.
[51] L. Wang et al., 'Tianji's horse racing optimization (THRO): a new metaheuristic inspired by ancient wisdom and its engineering optimization applications', Artif. Intell. Rev., vol. 58, no. 9, June 2025. doi.org/10.1007/s10462-025-11269-9

How to Cite: Saman M. Almufti, Amira Bibo Sallow. (2025). Elite opposition-based social spider optimization for solving benchmark problem. International Journal of Research In Science & Engineering (IJRISE), 5(2), 30–59. https://doi.org/10.55529/ijrise.52.30.59

BIOGRAPHIES OF AUTHORS

Saman M. Almufti, is a researcher at both the Technical College of Informatics–Akre, Akre University for Applied Sciences, Duhok, Iraq, and the Department of Computer Science, College of Science, Knowledge University, Erbil. His research focuses on metaheuristic algorithms, swarm intelligence, and applied machine learning. He has published numerous papers in high-impact international journals. Email: Saman.Almofty@gmail.com

Amira Bibo Sallow, is a lecturer at the Department of Information Technology, Technical College of Duhok, Duhok Polytechnic University, Iraq. Her research interests include information systems, data science, and software engineering. She is actively involved in academic and applied research projects related to digital transformation and computational intelligence.