Levy Flight Dream Optimization Algorithm: A Modified Version of Dream Optimization Algorithm

[1]Aliyu Sabo, [2]Abel E. Airoboman, [3]Kazeem Lawal, [4]Dahiru Dauda, [5]Kenneth E. Okedu, [6]Usman Isah Yusuf, [7]Ilhami Colak

[1,2,3&4]Department of Electrical/Electronic Engineering, Nigerian Defence Academy, Kaduna.
[5]School of Information Technology and Engineering, Melbourne Institute of Technology, Melbourne, Victoria, Australia.
[6]Department of Computer Engineering, Ahmadu Bello University, Zaria-Nigeria
[7]Department of Electrical and Electronic Engineering, Istinye University, Istanbul, Turkey

ABSTRACT
This manuscript elucidates a novel Modified Dream Optimization Algorithm (mDOA). The foundational framework of the Dream Optimization Algorithm (DOA) is informed by the cognitive phenomena associated with human dreaming. These cognitive mechanisms (memory retention, forgetfulness, supplementation strategies, and dream sharing processes) are systematically encoded as an optimization agent designed to tackle global optimization dilemmas. The DOA is afflicted by the challenge of an imbalance between exploration and exploitation, exhibiting a higher propensity for exploration than for exploitation, which results in an elevated likelihood of becoming ensnared in local optima. The enhancement of mDOA was achieved through the integration of a Levy flight variant to boost the exploitation phase. The efficacy of mDOA is evaluated against six prominent metaheuristics utilizing ten benchmark test functions (Schwefel, Ackley, Michalewicz, Griewank, Pathologic, Rastrigrin, Rosenbrock, Schaffer, Sphere, and Bohachevsky1), it demonstrated 85% enhancement in its convergence towards global optima. From the simulation results obtained, it shows that the mDOA succeeded in attaining the optimal global solution in 7 out of 10 cases, constituting 70.0% of the benchmark functions. Conversely, the 0ther algorithms used achieved 3 out of 10 cases, representing 30.0% of the benchmark functions. These shows an improvement in the mDOA.

INTRODUCTION

The use of nature-inspire optimization methodologies in recent years has demonstrated efficiency in addressing different types of optimization issues with considerable performance [1]. Optimization comprises a systematic approach for initiating solutions to challenges that are confined by specific limitations through the most effective utilization of available resources. The same outcome is produced by decisive search algorithms if the inceptive conditions remain the same. Even when the initial conditions remain stable, stochastic algorithms generate definite solutions every time they run because there is irregular in their search procedure [2]. Universal optimization algorithms, analytical intelligence, and contemporary soft computing paradigms extensively depend on metaheuristic approaches that are inspired by natural phenomena. Optimization algorithms come in two varieties: deterministic and stochastic. [3].

Past studies as classified Stochastic algorithms as either heuristic or metaheuristic. The heuristic methodology is limited to a single type of optimization problem, making them problem dependent. The metaheuristic-based search algorithm is problem-independent

universal problem solver algorithm that is used to tackle a variety of optimization problems [4]. The metaheuristic search algorithms combine exploitation (intensification) and exploration (diversification). To find the optimal local solutions inside the search space, the algorithm is directed by the exploration process. The exploitation procedure directs the algorithm to search among the generated local optimal for the global optimum solution. In finding a balance between exploration and exploitation, the metaheuristic search algorithm can connect with the global optimum result [5].

Metaheuristic search algorithms draw inspiration from biological systems. Numerous optimization challenges have been solved using these nature-inspired metaheuristic search algorithms [6]. Particles swarm are an illustration of a metaheuristic search algorithm influence by nature. Swarms of birds and fish shoal served as inspiration for swarm optimization [7]. However, classical PSO still has some weaknesses, such as poor local search that may lead to traps in local minimum affecting the convergence performance that results in uncertainties in the outcomes obtained[8].

The firefly algorithm simulates flashing behavior of fireflies [9], while the cat swarm optimization algorithm imitates the hunting and stalking manners of cats towards their target [10], The echolocation behavior of microbats is mimicked by bat algorithms [11], while the foraging behavior of ant colonies is mimicked by ant colony optimization algorithms [12]. The Dream Optimization Algorithm (DOA) was motivated by desire [13], which shares many properties with the optimization process in metaheuristic algorithms, including partial memory retention, forgetting, and logical self-organization.

To stabilize exploration and exploitation, DOA incorporate a fundamental memory method, a forgetting and supplementing technique, and a dream-sharing strategy to enhance the amplitude to escape local optima. Exploration and exploitation phases make up the optimization process, which produces good optimization outcomes. According to the literature review, DOA offers potential benefits over other metaheuristic algorithms including Particle Swarm Optimization (PSO) [14], Grey Wolf Optimization (GWO) [15], Sparrow Search Algorithm (SSA) [16], Differentiated Creative Search (DCS) [17], Great Wall Construction Algorithm (GWCA) [18], and Whale Optimization Algorithm (WOA) [19], among others. Strong convergence, progress, stability, adaptability, resilience, and reliability to initial control parameter values are some of these benefits. The dream optimization method still has an imbalance issue despite these many benefits, allying exploration and exploitation because of the continual impact of control settings, optimization hyperspace, and incomplete knowledge.

The paper employed levy flying process to produce an improved Dream Optimization Algorithm. The capacity of the mDOA algorithm was estimated to be using ten (10) standard test functions, the outcome of the results was contrast to those of the traditional DOA, White Shark Optimization (WSO), Seagull Optimization Algorithm (SOA), Sunflower Optimization (SFO), Golden Jackal Optimization (GJO) and African Vultures Optimization Algorithm (AVOA). Comparison demonstrated the dominance of the mDOA algorithm over the other algorithm. The contribution of this paper is the modification of DOA using levy flight variant.

The report's remaining sections are assembled as follows: Section 2 introduces the DOA algorithm and its levy flight. In Section 3, the proposed mDOA result is shown. In Section 4, the effectiveness of mDOA is evaluated and compared to the traditional DOA and others algorithms approach.

RESEARCH DESIGN METHOD

Concept of DOA Algorithm

Optimization algorithm assumptions

When integrating the attributes of human aspirations with the principles of optimization algorithms, we delineate the subsequent four postulations:

1. Fitness values can be used to assess the quality of dreams.

Corresponding author: Kazeem Lawal
✉ dawooddahiru@gmail.com
 Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

2. The basis of pre-existing memories is intimately linked to the beginning of a dream.
3. People add logically self-organized information to partially forgotten memories.
4. Memory capacities are rather unpredictable and vary between people or groups.

The dream optimization method is developed based on these fundamental presumptions. These four underlying presumptions are reflected in the procedural architecture, exploration phase, and various approaches used during the algorithm's development phase.

Initialization phase

In order to initiate the algorithm's optimisation process, mDOA first creates a random sample inside the search space called the initial sample. The initial sample can be acqured using equation 1:

$$X_i = X_l + rad \times (X_u - X_l), \quad i = 1,2, \dots\dots\dots N \tag{1}$$

where N is the number of individuals, i.e., the sample size. X_i is the ith individual in the sample; X_u and X_l for the lower and upper boundaries of the search space, accordingly. The resulting sample can be shown as follows: rad is a Dim-dimensional vector, where each proportion is a random number between 0 and 1.

$$X = \begin{bmatrix} X_1 \\ X_2 \\ X_N \end{bmatrix} = \begin{bmatrix} X_{11} & \cdots & X_{1,Dim} \\ \vdots & \ddots & \vdots \\ X_{N,1} & \cdots & X_{N,Dim} \end{bmatrix} \tag{2}$$

Where X_{ij}, denotes the location of the ith individual in the jth dimension, and Dim denotes the dimensionality of the optimization issue.

Exploration Phase

The formulas and particular update mechanism are as follows:

Memory strategy

First, group q members can reset their formation by recalling the formation position of their group's best member before dreaming, according to the foundational memory technique. position to that of the best member of the group:

$$X_i^{t+1} = X_{bestq}^t \tag{2}$$

Where X_i^{t+1} denotes the ith person at iteration $t+1$ and X_{bestq}^t denotes the best individual in group q at iteration t.

Forgetting and supplementation strategy

The forgetting and supplementing approach combines local and global search capabilities. This strategy, which is like the memory technique, enables people to self-organize and forget the position information in the forgetting dimensions. The following is the revised formula:

$$X_{ij}^{t+1} = X_{bestqj}^t + \left(X_{ij} + rand \times (X_{uj} - X_{lj})\right) \times \frac{1}{2} \times \left(\cos\left(\pi \times \frac{t+T_{max}-T_d}{T_{max}}\right) + 1\right), \quad j = K_1, K_2 \dots\dots\dots K_{k_q} \tag{3}$$

Where X_{ij}^{t+1} 1; tthe position of the ith individual in the jth dimension at iteration $t+1$: the X_{bestqj}^t indicate a position of the best location in group q in the jth dimension at iteration t; X_{lj} and X_{uj} represent the minimum and maximum bounds of the search space in the jth dimension, respectively; $rand$ is a random number between 0 and 1;t is the current iteration number, T_{max} is the maximum number of iterations, and T_d is the maximum number of iterations during the exploration phase.

Dream-sharing strategy

In mDOA, the dream-sharing technique improves the capacity to break out of local optima. People can randomly gather position information from others in the forgetting dimensions using this method, which functions in agreement with the forgetting and augmenting methods and follows the memory strategy. The following is the update formula:

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

$$X_{ij}^{t+1} = \begin{cases} X_{mj,}^{t+1} & m \le i \\ X_{mj,}^{t} & i < m \le N \end{cases} \quad j = K_1, K_2, \dots \dots \dots K_{k_q} \tag{4}$$

Where X_{ij}^{t+1}, is the location of the ith person in the jth dimension at iteration t+1; m is a arbitrarily chosen natural number from the range [1,N] for each dimension update.

Exploitation Phase

The Levy Flight (LF) was used to enhance the exploitation phase; grouping is no longer done throughout the stage of development (iteration count from T_d to T_{max}). The population shows the best dream from the past iterations of the entire population or the best individual from the previous iterations before each dreaming session. Each person's whereabouts in the forgetting dimensions is then updated. The number of forgetting dimensions k_r is the same for every member of the population. The locations in the k_r forgetting dimensions—designated as $K_1, K_2, \dots \dots \dots K_{k_r}$ are updated once they are randomly chosen from the D dimensions.

The improving method is like Equation (2) and (3), with the update formula as follows:

a. Memory strategy

$$X_i^{t+1} = X_{best}^{t} \tag{5}$$

where X_i^{t+1} is the ith individual at iteration $t+1$,and X_{best}^{t} indicate the best person of the sample at a given iteration t.

Forgetting and supplementation strategy

$$X_{ij}^{t+1} = X_{bestj}^{t} + \left(X_{ij} + rand \times \left(X_{uj} - X_{lj}\right)\right) \times \frac{1}{2} \times \left(\cos\left(\pi \times \frac{t}{T_{max}}\right) + 1\right), \quad j = K_1, K_2 \dots \dots \dots K_{k_r} \tag{6}$$

Where X_{ij}^{t+1} is the position of the ith individual in the jth dimension at iteration $t+1$, X_{bestj}^{t} denotes the position of the best person of the entire population in the jth dimension at iteration t; X_{lj} and X_{uj} are the minimum and maximum bounds of the search space in the jth

dimension, respectively; $rand$ is a arbitra number between 0 and 1; t is the current iteration number, and T_{max} is the maximum number of iterations for the algorithm.

In a similar vein, Eq. (5) demonstrates that in dimensions other than $K_1, K_2 \dots \dots \dots K_{k_r}$ individuals are able to preserve the exact placements in these dimensions while dreaming by recalling the position information of the best position in the population during prior iterations.Equation (6) demonstrates that in dimensions $K_1, K_2 \dots \dots \dots K_{k_r}$, people forget the location of the population's finest person during the prior.

Levy Flight Algorithms

A Levy flight algorithm represents a specific variant of random walk distinguished by step lengths that adhere to a Levy distribution, which is characterized by a power-law tail, resulting in infrequent long jumps interspersed with shorter movements. This distinctive pattern is employed in numerous optimization algorithms, especially within the realm of mathematics, to augment both exploration and exploitation capabilities. The functionalities of the Levy flight are operationalized as

$$Le'vy(n,d) = \frac{u}{|v|^{1/\beta}}, \quad u \sim N(0, \sigma^2), \quad v \sim N(0,1) \tag{7}$$

Where $\beta = 1.5$ and σ is computed as:

$$\left(\frac{\Gamma(1+\beta).\sin(\pi\beta/2)}{\Gamma\left(\frac{1+\beta}{2}\right).\beta.2^{\frac{\beta-1}{2}}}\right)^{1/\beta} \tag{8}$$

In the foundational dream optimization algorithm, the methodology demonstrates a robust capacity for exploitation; however, it is susceptible to becoming entrenched in local optima. To mitigate this challenge, the paper incorporates a Lévy flight strategy into the exploitation phase to improve the convergence rate. Below is table 1 and 2 showing the pseudocode for both the standard Dream Optimization Algorithm and the modified version, denoted as DOA-LF.

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Table 1 : Standard DOA Pseudo-code

Algorithm 1 Pseudo-code of DOA

Input : Population size (N_{max}) the lower limit of variables (X_l), the upper limit of variables (X_u), size of problem (D_{im}), the current number of iteration (t), the number of iteration as a demaracation (T_d), the maximum number of iteration (T_{max}), forgetting dimensions of each group and of exploitation $(k_1, k_2, k_3, k_4\ k_5, k_r)$

Output: the best solution X_{best} and the minimum fitness $Fitness_{best}$

1. Generate an initial population $X\ of\ N$ individuals using Eqs. (2)
2. Check the bounds of the solution
3. Evaluate the fitness of the soluti
4. Detect the best solution X_{best} and
5. Define the current iteration $t = 1$
6. **while** $t < T_d\ do$
7. Update the best solution X_{best} a
8. **for** $q = 1 : 5\ do$
9. Update the best solution X_{best} a
10. Update k_q using Eq. (10)
11. Update X_i^{i+1} using Eq. (4)
12. $\left(K_1, K_2, \dots \dots K_{k_q}\right) =$ $rad\ perm\left(k_q, N\right)$
13. $for\ i = {(((q-1)}/{(5 \times N)}} + 1) : \left({}^{q}/{5 \times N}\right)\ do$
14. $if\ rad < u\ then$
15. Update $x_{i,j}$ using Eq. (5)
16. Check the bound of $x_{i,j}$
17. **else**
18. Update $x_{i,j}$ using Eq. (6)
19. **end if**
20. **end for**
21. **end for**
22. Update the current number of ite $t + 1$
23. **end while**
24. while $t < T_d\ and\ t < T_{max} do$
25. Update k_r using Eq. (11)
26. Update X_i^{t+1} using Eq. (7)

Algorithm 1 Pseudo-code of DOA

27. $\left(K_1, K_2, \dots \dots K_{k_r}\right) =$ $rad\ perm(k_r, N)$
28. **for** $i = 1 : N\ do$
29. Update $x_{i,j}$ using Eq. (8)
30. Check the bound of $x_{i,j}$
31. **end for**
32. Update the current number of ite $t + 1$
33. **end while**

Table 2: DOA with Levy Flight (DAO-LF) Pseudo-code

Algorithm 1 Pseudo-code of DOA with Levy Flight Enhancement

1. 1nput : population size, iterations T, bounds lb, ub, dimension D, objective function fobj
2. Initialise:
3. $x \leftarrow$ random poputation in $[lb, ub]$
4. $fbestd[m] \leftarrow \infty, sbestd[m] \leftarrow empty, for\ m = 1 \dots 5$
5. $fbest[m] \leftarrow \infty, sbestd[m] \leftarrow empty, for\ m = 1, \dots \dots ..5$
6. $fbest_{history} \leftarrow zeros(T)$
7. $for\ i = 1 \rightarrow [0.9T]do$
8. $for\ m = 1 \rightarrow 5\ do$
9. $k \leftarrow$ $random\ integer\ in\ \left[[D(8m)], \left[\frac{j}{3}\right.\right.$
10. for each solution j in subgroup m do
11. $fit \leftarrow fobj\ (x_j)$
12. if $fit < fbestd[m] then$
13. $fbestd[m] \leftarrow fit, sbestd[m] \leftarrow x_j$
14. end if
15. end for
16. for each solution j in subgroup m do
17. $x_j \leftarrow sbestd[m]$
18. Choose k random dimensions in
19. if rand < 0.9 then
20. $step \leftarrow$ Levy Flight (1.5D)
21. for each $h \in in$ do

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Algorithm 1 Pseudo-code of DOA with Levy Flight Enhancement

22. $x_j[h] \leftarrow x_j[h]+0.01 \, step[h].(ub[h] - lb[h]). \dfrac{\cos\left(i+\frac{T}{\frac{10\pi}{T}}\right)+1}{2}$
23. Apply boundary chech
24. end for
25. else
26. for each $h \in in$ do
27. $x_j[h] \leftarrow$ random element from populatin
28. $end \, for$
29. end if
30. end for
31. if $fbestd[m] < fbest$ then
32. $fbest \leftarrow fbestd[m], sbest \leftarrow sbestd[m]$
33. $end \, if$
34. end for
35. $fbest\_history[i] \leftarrow fbest$
36. end for
37. $w_{levy} \leftarrow 0.5$
    …Weight fot Levy steps
38. for $i = [0.9T] + 1 \rightarrow T$ do
    ….Exploitation phase
39. for $p = 1 \rightarrow pop$ do
40. $fit \leftarrow fobj(x_p)$
41. if $fit < fbest$ then
42. $fbest \leftarrow fit, sbest \leftarrow x_p$
43. $end \, if$
44. end for
45. for $j = 1 \rightarrow pop$ do
46. $k \leftarrow random \, integer \, in \, [2. \max\left(2\left[\frac{D}{3}\right]\right.$
47. $x_j \leftarrow sbest$
48. Choose k random dimensions in
49. for each $h \in in$ do
50. if rand $< w_{levy}$ then
51. $step \leftarrow$ Levy Flight (1.5,D)
52. $x_j[h] \leftarrow x_j[h]+0.01 \, step[h].(ub[h] - lb[h]). \dfrac{\cos(i\pi+T)+1}{2}$
53. $else$

Algorithm 1 Pseudo-code of DOA with Levy Flight Enhancement

54. $x_j[h] \leftarrow x_j[h]+0.01 \, rand.(ub[h] - lb[h]). \dfrac{\cos(i\pi+T)+1}{2}$
55. end if
56. Apply boundary check   1
57. end for
58. end for
59. $fbest\_history[i] \leftarrow fbest$

Code comparison to other levy flight developed optimization algorithms

This paper applied the Levy Flight at the exploitation phase. The modification is as follows:

a. Modify the forgetting and supplementation strategy equation using levy flight, from equation 6 of the standard DOA using equation 8 for specific variant of random walk at exploitation phase to enhance the convergence speed.

b. evaluate performance against existing algorithms with parameters such as convergences speed, efficiency, and robustness.

c. The Levy step size is uniquely scaled by a cosine-wave term, $\frac{\cos(,)+1}{2}$ which creates a smooth, oscillatory, and non-linear decay of step length over iterations, differing from standard static or linearly decreasing scales.

d. The number of dimensions (k) to perturb using Levy flights is dynamically and differently calculated per subgroup during exploration, making the search effort adaptive to the subgroup's role and problem dimension.

e. The use of Levy flight differs fundamentally between phases: in exploration, it's applied around subgroup-best solutions (sbestd[m]) and competes with random replacement, while in exploitation, it's

Corresponding author: Kazeem Lawal
✉ dawooddahiru@gmail.com
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

applied more directly around the global best (sbest) with deterministic weighting

Evaluation Method

The capacity of any improved or developed optimization algorithm is evaluated using benchmark functions, which are applied mathematical functions. As algorithms are created to address real-world engineering optimization issues, this is seen to be crucial. The set of ten standard test functions given in Table 3 were carefully selected to effectively measure the capacity of developed mDOA algorithms. The mDOAs performance measures of interest are listed as follows.

Execution speed: a gauge of how fast the mDOA algorithm variation can find the best convergence. The precision of the solution, which quantifies how closely each mDOA's results resemble the ideal; The convergence rate: This is utilized to ascertain when the corresponding mDOAs find their answer.

Table 3: Ten-Dimensional Standard Functions [20]

Fn	Names	Description
F$_1$	Shere function [21]	$f(x) = \sum_{i=1}^{d} x_i^2$
F$_2$	Schwefel function [22]	$f(x) = \sum_{i=1}^{30} \left(x_i^2 + x_{i+1}^2 \right)^{0.5} \left\{ \left[\sin 50 \left(x_i^2 + x_{i+1}^2 \right)^{0.1} \right]^2 \right\}$
F$_3$	Rosenbrock Function [23]	$f(x) = \sum_{i=1}^{n-1} ((x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2)$
F$_4$	Rastrigin Function [24]	$f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)]$
F$_5$	Griewank Function [25]	$f(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + $
F$_6$	Ackley function [26]	$f(X) = -2o \exp\left[-\frac{1}{5} \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right] - $

Fn	Names	Description
F$_7$	Michalewicz function [27]	$f(x) = -\sum_{i=1}^{d} \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi} \right)$
F$_8$		
F$_9$	Bohachevsky1 function [28]	$f(x_1, x_2) = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x$
F$_{10}$	Schaffer function [29]	$f(x) = \sum_{i=1}^{30} \left(x_i^2 + x_{i+1}^2 \right)^{0.5} \left\{ \left[\sin 50 \left(x_i^2 + x_{i+1}^2 \right)^{0.1} \right]^2 \right\}$
	Pathologic function [30]	$f(x) = \sum_{n=1}^{n} (0,5$ $+ \frac{\sin\left(\sqrt{100x_i^2 + x_{i\neq1}^2} \right)^2 - 0}{(1 + 0.001(x_i^2 - 2x_i x_{i\neq1} + x}$

RESULTS ANALYSIS

Convergence Comparison of mDOA

The mDOA has shown a better convergence in comparison with some of the algorithms used as shown in figures below. The mDOA shows vibrances in convergence with other algorithms used such as standard DOA, WSO, SOA, SFO, GJO and AVOA on standard functions. For the unimodal and multimodal standard function, the standard deviation, mean, and best values attained by mDOA and other algorithms are shown in Tables 4 and 5. Furthermore, Figures 1 and 2 show how best mDOA and other algorithms perform in comparison with unimodal and multimodal standard functions.
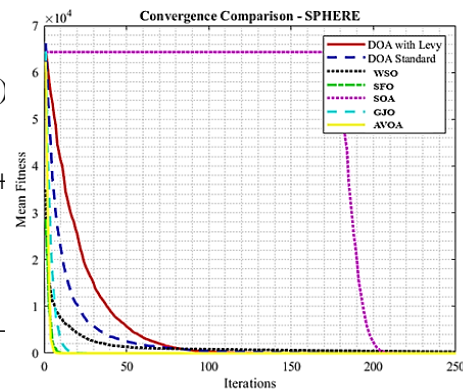
Convergence Comparison - SPHERE

Corresponding author: Kazeem Lawal
✉ dawooddahiru@gmail.com
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.
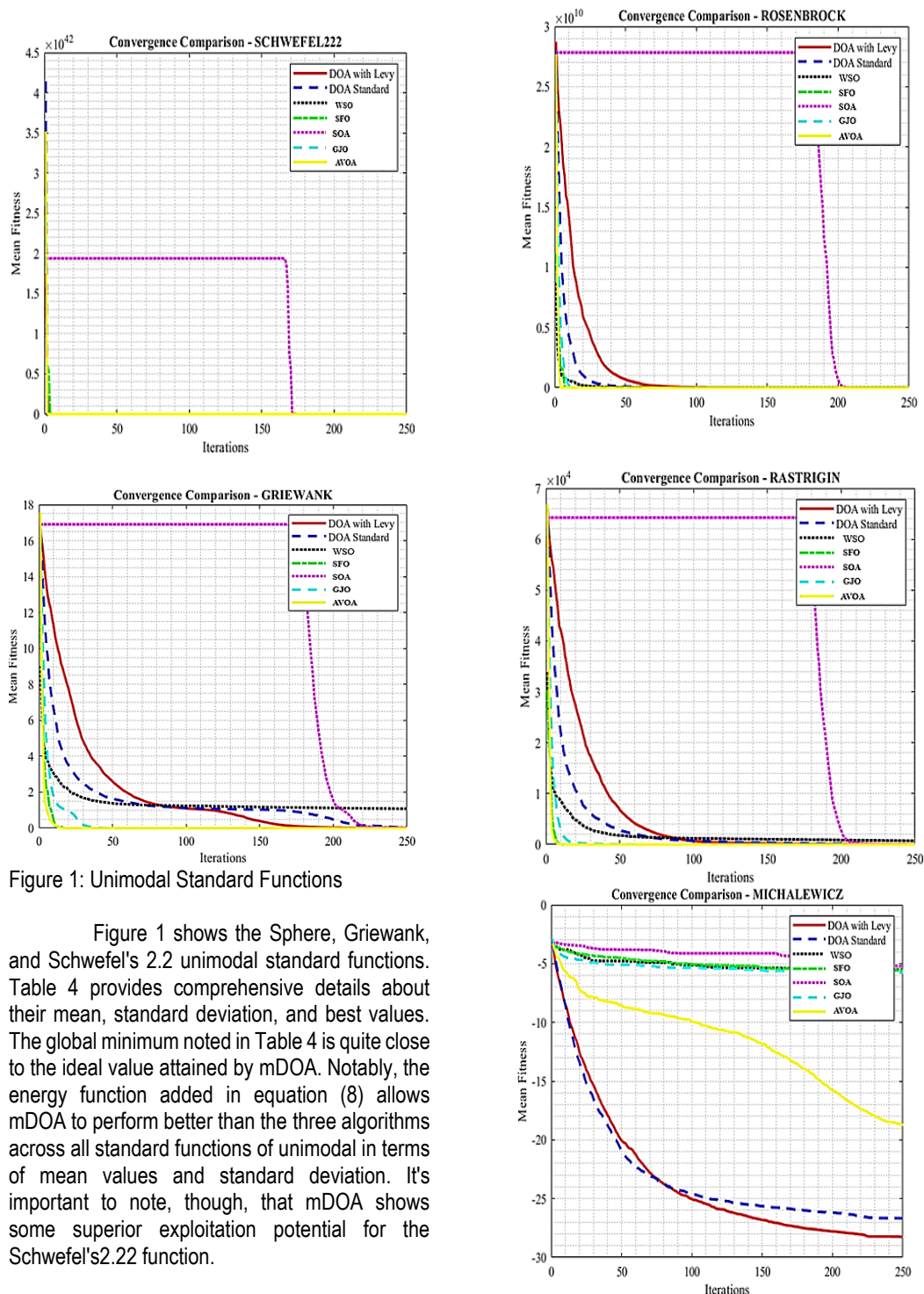
Figure 1: Unimodal Standard Functions

Figure 1 shows the Sphere, Griewank, and Schwefel's 2.2 unimodal standard functions. Table 4 provides comprehensive details about their mean, standard deviation, and best values. The global minimum noted in Table 4 is quite close to the ideal value attained by mDOA. Notably, the energy function added in equation (8) allows mDOA to perform better than the three algorithms across all standard functions of unimodal in terms of mean values and standard deviation. It's important to note, though, that mDOA shows some superior exploitation potential for the Schwefel's2.22 function.

Corresponding author: Kazeem Lawal

✉ *dawooddahiru@gmail.com*

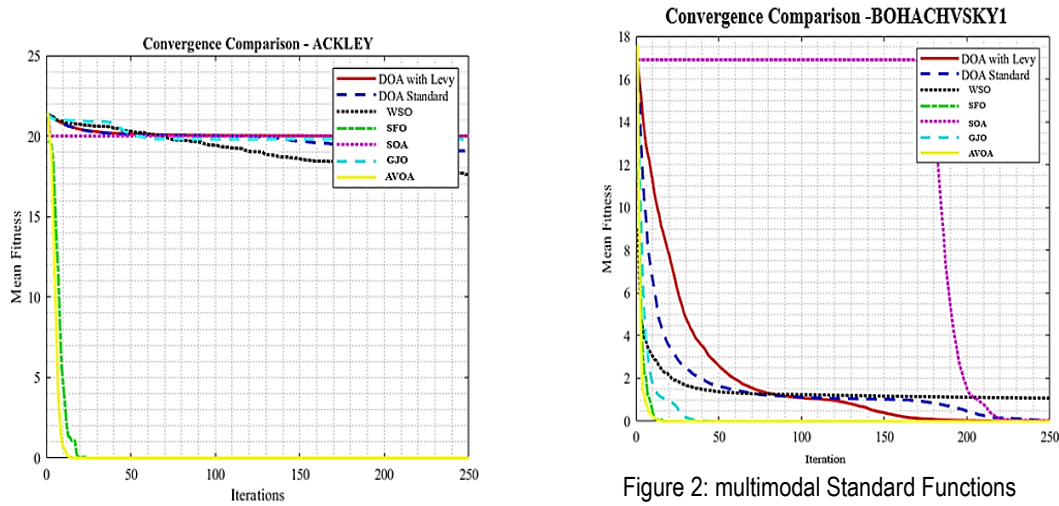*Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.*
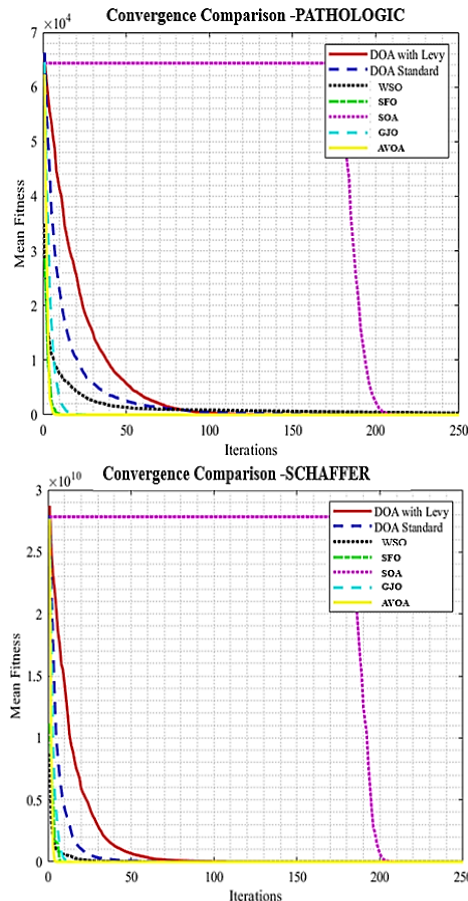
233

Figure 2: multimodal Standard Functions

The multimodal standard functions depicted in Figure 2 include Rosenbrock, Rastrigin, Michalewicz, Rosenbrock, Bohachevsky1, Schaffer, and Pathologic functions. The statistical analysis regarding their standard deviation, mean, and best values can be found in Table 5. The optimal value achieved by mDOA, as indicated in Table 5, closely approximates the global minimum highlighted, except for the Ackley, Michalewicz and Rosenbrock function. The mDOA has better exploitative ability compared to the DOA and other five algorithms in comparison. It exhibits better exploitation and exploration abilities for the Step function.

Table 4: Unimodal Standard Functions

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	Global minimum function value
Sphere	Best	**4.0265e-03**		4.0266e-03	1.6706e-01	4.0266e-03	1.6706e-01	1.6706e-01	0
	Mean	2.8503e-02	1.6706e-01			2.8503e-02	2.7535e-01	2.7535e-01	
	Std		2.7535e-01	2.8503e-02	2.7535e-01	2.3109e-02	7.8468e-02	7.8468e-02	
	Dev	2.3109e-02	7.8468e-02	2.3109e-02	7.8468e-02				
Schwefel2.2	Best		1.2339e+0	8.8827e-02	1.2339e+0	8.8727e-02		1.2339e+0	0
	Mean	**8.8727e-02**	0		0	1.4009e+0	1.2339e+0	0	
	Std	1.4009e+0	1.6050e+0	1.4009e+0	1.6050e+0	1	0	1.6050e+0	
	Dev	1	0	1	0	2.1106e+0	1.6050e+0	0	
		2.1106e+0	2.2127e-01	2.1106e+0	2.2127e-01	1	0	2.2127e-01	
		1		1			2.2127e-01		
Griewank	Best	**4.2316e-04**	1.4267e-02	4.2326e-04	1.4267e-02	4.2326e-04	1.4267e-02	1.4267e-02	0
	Mean	4.6401e-03	3.0762e-02	4.6401e-03	3.0762e-02	4.6401e-03	3.0762e-02	3.0762e-02	
	Std	6.4389e-03	1.3988e-02		1.3988e-02	6.4389e-03	1.3988e-02	1.3988e-02	
	Dev			6.4389e-03					

Table 5: Multimodal Standard Functions

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	Global minimum function value
Michalewicz	Best	-2.8797e+0 1	-	-2.8797e+0 1	**-2.7537e+0 1**	-2.8797e+0 1	**-2.7537e+0 1**	**-2.7537e+0 1**	−8.22015
	Mean		**-2.7537e+0 1**						
	Std	-2.8282e+0 1	-2.6672e+0 1	-2.8282e+0 1	-2.6672e+0 1	-2.8282e+0 1	-2.6672e+0 1	-2.6672e+0 1	
	Dev	3.5005e-01	5.8910e-01	3.5005e-01	5.8910e-01	3.5005e-01	5.8910e-01	5.8910e-01	
Rosenbrock	Best	2.7636e+0 2	**1.5119e+0 2**	2.7636e+0 2	**1.5119e+0 2**	2.7636e+0 2	1.5119e+0 2	**1.5119e+0 2**	0
	Mean		3.9598e+0 2						
		2.8630e+0 4		2.8630e+0 4	3.9598e+0 2	2.8630e+0 4	3.9598e+0 2	3.9598e+0 2	
	Std	8.5402e+0 4	2.2181e+0 2	8.5402e+0 4	2.2181e+0 2	8.5402e+0 4	2.2181e+0 2	2.2181e+0 2	
	Dev								
Rastrigin	Best	**2.3966e+0 1**	2.7677e+0 1						0
	Mean	3.7232e+0 1	4.2162e+0 1	2.3966e+0 1	2.7677e+0 1	2.3966e+0 1	2.7677e+0 1	2.7677e+0 1	
	Std	8.6681e+0 0	8.1255e+0 0	3.7232e+0 1	4.2162e+0 1	3.7232e+0 1	4.2162e+0 1	4.2162e+0 1	
	Dev			8.6681e+0 0	8.1255e+0 0	8.6681e+0 0	8.1255e+0 0	8.1255e+0 0	
Ackley	Best	1.9998e+0 1	**1.6679e+0 0**	1.9998e+0 1	**1.6679e+0 0**	1.9998e+0 1	1.6679e+0 0	1.6679e+0 0	0
	Mean	2.0000e+0 1	1.9077e+0 1	2.0000e+0 1	1.9077e+0 1	2.0000e+0 1	1.9077e+0 1	1.9077e+0 1	
	Std	8.2108e-04	4.0976e+0 0	8.2108e-04	4.0976e+0 0	8.2108e-04	4.0976e+0 0	4.0976e+0 0	
	Dev								
Bohachevsky 1	Best	**4.2316e-04**	1.4268e-02	4.2326e-04	1.4277e-02	4.2326e-04	1.4277e-02	1.4287e-02	0
	Mean	4.6401e-03	3.0762e-02	4.6411e-03	3.0762e-02	4.6401e-03	3.0762e-02	3.0762e-02	
	Std	6.4389e-03	1.3988e-02	6.4389e-03	1.3988e-02	6.4389e-03	1.3988e-02	1.3988e-02	
	Dev								
Schaffer	Best	**1.4636e+0 2**	1.6119e+0 2	2.7636e+0 2	1.6119e+0 2	2.7636e+0 2	1.6119e+0 2	1.6119e+0 2	

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	Global minimum function value
	Mean	2.8630e+04	3.9598e+02	2.8630e+04	3.9598e+02	2.8630e+04	3.9598e+02	3.9598e+02	
	Std Dev	8.5402e+04	2.2181e+02	8.5402e+04	2.2181e+02	8.5402e+04	2.2181e+02	2.2181e+02	
Pathologic	Best	**4.0065e-03**	1.6706e-01	4.0266e-03	1.6206e-01	4.0266e-03	1.6206e-01	1.6206e-01	0
	Mean	2.8503e-02	2.7535e-01	2.8503e-02	2.7535e-01	2.8503e-02	2.7535e-01	2.7535e-01	
	Std Dev	2.3109e-02	7.8468e-02	2.3109e-02	7.8468e-02	2.3109e-02	7.8468e-02	7.8468e-02	

Statistical value analysis on the stability of mDOA

The mDOA statistical value on common benchmark functions is shown in this subsection. For the unimodal and multimodal benchmark function, the P values attained by mDOA, DOA, WSO, SFO, SOA, GJO and AVOA are shown in Tables 6 and 7, along with comparisons with mDOA and other algorithms. Furthermore, Figures 3 and 4 show how mDOA stability performs best in comparison with ten classes of the benchmark functions.
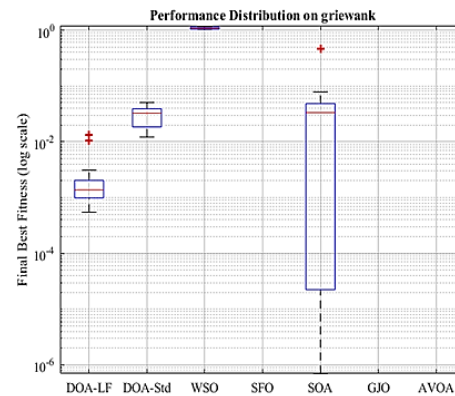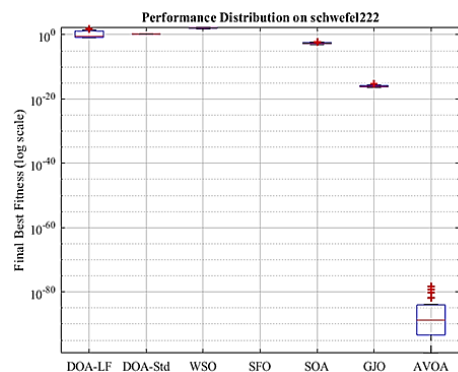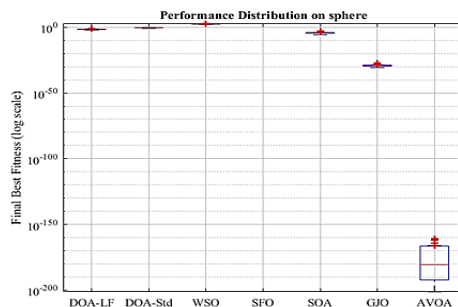

Figure 3: Statistical graph of unimodal Standard Functions

Figure 3 above for Sphere, Schwefel's 2.2, and Griewank unimodal standard show that the box plot of the DOA-Levy method has almost negligible spread and clusters tightly around very low Final Fitness levels, indicating far more stability than other algorithm approaches. DOA-Standard, SOA, AVOA on the other hand, shows less stability because of its significant variability and considerable variation in Final Fitness among runs. While Table 6 provides comprehensive details about their P value. The P value noted in Table 6 is quite close to the ideal value attained by mDOA which shows that it is more stable compared to other algorithms with only exception in Greiwalk benchmark function. It's important to note that mDOA shows some superior exploitation potential for the Sphere and Schwefel's2.22 function.

Corresponding author: *Kazeem Lawal*
✉ dawooddahiru@gmail.com

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.
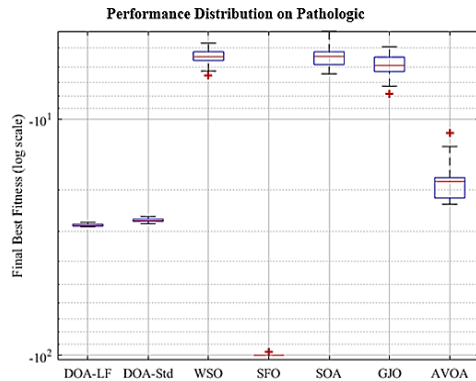
237
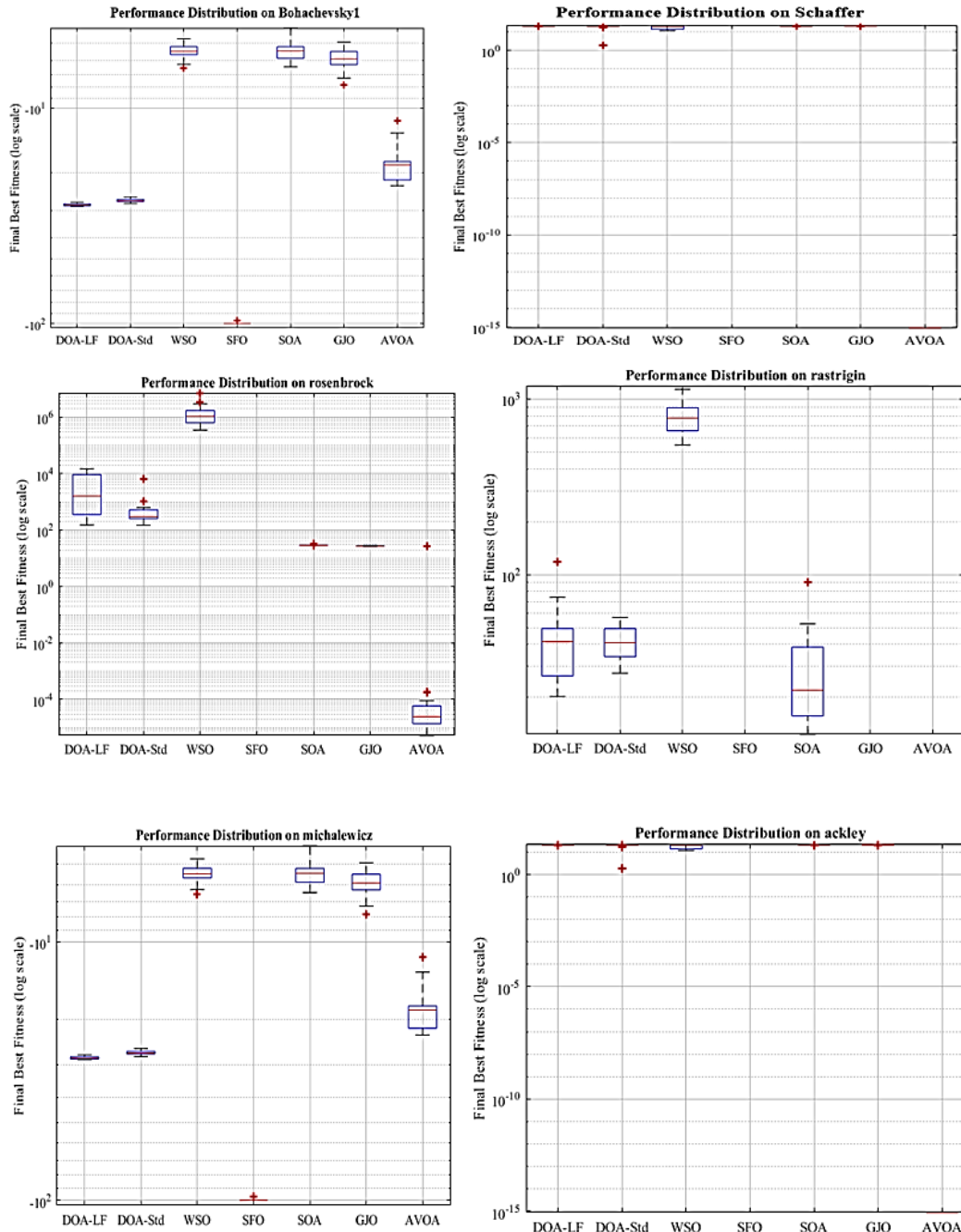
Figure 4: Statistical graph of multimodal Standard Functions

The multimodal Standard functions distribution performance box stability depicted in Figure 4 include Rosenbrock, Rastrigin, Michalewicz, Bohachevsky1, Schaffer, Rosenbrock and Pathologic functions. The statistical analysis regarding their P values can be found in Table 7. The extremely small range of Final Fitness values around a mean close to 20 indicates that the DOA-Levy approach exhibits more stability. In contrast, WSO, SOA, GJO and AVOA exhibits far less stability across runs, with a much wider range in Final Fitness, from close to zero to roughly 20. Meanwhile, mDOA performed more stable in the whole run with the exception in two benchmark functions (Rosenbrock and Rastrign) which account for 71.4% performance output. A statistically significant difference in stability between the two approaches is confirmed by the modest p-value as highlighted except for the Rastrigin and Rosenbrock function. The mDOA has better exploitative stability compared to standard DOA, WSO, SFO, SOA, GJO and AVOA. Also has better exploitation and exploration stability for the Step function.

Table 6: Unimodal statistical P-Value

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	p-value(h=1) - wilcoxon
Sphere	Mean	2.0725e-03	1.4602e-02	-1.9061e-01	0.0000e+0	-1.5316e-05	-1.8600e-16	3.9121e-82	8.441e-01
	Std Dev	3.2269e-02	1.1079e-01	3.7233e+00	0.0000e+0	1.7405e-03	1.3188e-15	1.7441e-81	
Schwefel2.2	Mean	3.9101e-01	2.6542e-02	3.2559e-02	0.0000e+0	-4.2666e-06	2.2424e-18	-1.6916e-82	4.3570e-01
	Std Dev	1.7349e+00	1.2444e-01	8.0073e+00	0.0000e+0	1.1247e-04	9.6828e-18	7.5899e-82	
Griewank	Mean	3.4328e-02	-1.4398e-01	-7.2771e-01	0.0000e+0	-4.2484e-02	-3.8484e-09	6.2540e-10	2.1913e-01
	Std Dev	1.1382e-01	3.2691e-01	2.8051e+00	0.0000e+0	1.3603e-01	3.2240e-08	9.0111e-09	

Table 7: Multimodal statistical P-Value

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	p-value(h=1)- wilcoxon
Michalewicz	Mean	-2.4555e+01	-3.0967e+00	-3.6259e+00	-5.6465e+00	-5.3113e+00	-1.2084e+01	2.1775e-01	3.4814e-01
	Std Dev	4.9638e+01	6.2553e+01	6.0753e+01	5.6244e-01	6.7024e+01	5.8175e+01	4.4956e+01	

Corresponding author: Kazeem Lawal
✉ dawooddahiru@gmail.com
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Function (Fn)		DOA-LF	DOA	WSO	SFO	SOA	GJO	AVOA	p-value(h=1)-wilcoxon
Rosenbrock	Mean	2.6145e+01	3.0302e+01	1.1194e+00	2.8995e+01	8.0618e-04	1.1320e-03	1.0000e+00	4.0861e-01
	Std Dev	3.5467e+01	3.3805e+01	1.4562e-01	8.1831e-03	9.1972e-03	3.7325e-03	2.1592e-04	
Rastrigin	Mean	1.0024e-01	-1.8504e-01	5.6895e-01	0.0000e+00	1.0909e-01	-7.5355e-11	-2.8443e-11	5.0987e-01
	Std Dev	8.4322e-01		4.1988e+00	0.0000e+00	6.9642e-01	7.3357e-10	2.1750e-10	
			1.0059e+00						
Ackley	Mean	-5.5484e+00	6.0965e+00	3.7080e+00	8.8818e-16	1.0000e+01	6.2901e+00	-6.9508e-17	6.0777e-01
	Std Dev	5.9868e+01		6.6258e-01	0.0000e+00	1.0208e+02	6.1768e+01	1.5215e-16	
			3.4515e+01						
Bohachevsky1	Mean	-2.4565e+01	-3.0977e+00	-3.8251e+00	-5.8466e+00	-5.6223e+00	-1.2090e+01	2.4765e-01	3.8224e-01
	Std Dev	4.9538e+01	6.3553e+01	6.0955e+01	5.7233e-01	6.8022e+01	5.8385e+01	4.4987e+01	
Schaffer	Mean	-5.8414e+00	6.0261e+00	3.9000e+00	8.7828e-16	1.0020e+01	6.9801e+00	-6.9801e-17	6.0700e-01
	Std Dev	5.7858e+01		6.7211e-01	0.1000e+00	1.0601e+02	6.1378e+01	1.7115e-16	
			3.8514e+01						
Pathologic	Mean	-2.6551e+01	-3.0064e+00	-3.9250e+00	-5.6666e+00	-5.1111e+00	-1.2224e+01	2.1885e-01	3.0014e-01
	Std Dev	4.8438e+01	6.3513e+01	6.0023e+01	5.5444e-01	6.5025e+01	5.7115e+01	4.4900e+01	

COMPUTATIONAL COMPLEXITY

The computational complexity of proposed mDOA algorithm is $k \leftarrow random\ integer\ in\ \left[2. \max\left(2\left[\frac{D}{3}\right]\right)\right]$

Whereas, the space complexity of all algorithms represents the maximum quantity of allocated space at any given moment, which is evaluated during the initialization phase. In this study, the space complexity for all methodologies is examined as $i = [0.9T] + 1 \rightarrow T$

However, the mean execution duration of the proposed mDOA algorithm in comparison to other algorithms is delineated in Table 8. It is evident that mDOA exhibits a reduced temporal requirement relative to alternative methodologies, measured in seconds. Consequently, one may deduce that the computational efficiency of the proposed enhanced algorithm significantly surpasses that of its competing counterparts.

Table 8: Average running time of inproved mDOA and competitive approaches

Algorithms	Average time (in seconds)
Modified Dream Optimization Algorithm (mDOA)	1.5245
Dream Optimization Algorithm (DOA)	1.8000
White Shark Optimization (WSO)	
Seagull Optimization Algorithm (SOA)	1.7636
Sunflower Optimization (SFO)	1.6246

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

Algorithms	Average time (in seconds)
Golden Jackal Optimization (GJO)	
African Vultures Optimization	2.7891
2.7333	
Algorithm (AVOA)	2.8766

CONCLUSION

Using levy flight, this research introduces an enhance version of the Dream Optimisation Algorithm.The levy flight inherent in the position update phase was used to develop modified DOA algorithm called the mDOA algorithm. ten standard optimization functions were used to assess the improved mDOA's performance. Multimodal and unimodal standard optimization functions (Ackley, Griewangk, Michalwicz, Rastrigin Rosenbrock, Shwefel, Bohachevsky1 Schaffer, Pathologic and Sphere) comprise these functions. According to the simulation results, mDOA outperformed and attaining the optimal global solution in 7 out of 10 cases, constituting 70.0% of the benchmark functions. it demonstrated 85% enhancement in its convergence towards global optima. Except for the Ackley and Rosenbrock function, where other algorithms has outperformed mDOA, this provides a greater capacity to escape local minima than the normal DOA, WSO. AVOA, GJO , SOA and SFO. overall performance on the benchmark test suites the mDOA, showing better convergence, stability, significance, and reliability, over most of the algorithms compared with in this research.

REFERENCES

[1] Yang, X.-S. "Nature-inspired metaheuristic algorithms: Success and new challenges". 2012. p1211.6658.

[2] Yılmaz, S., & Küçüksille, E. U. "A new modification approach on bat algorithm for solving optimization problems". Applied Soft Computing, 2015 p28, 259-275.

[3] Yang, X.-S. "Firefly algorithm, stochastic test functions and design optimization". 2010a. International Journal of Bio-Inspired Computation, 2(2), 78-84.

[4] Shehab, M., Khader, A. T., & Al-Betar, M. A. "A survey on applications and variants of the cuckoo search algorithm". 2017. Applied Soft Computing.

[5] Civicioglu, P., & Besdok, E. "A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms". Artificial intelligence review, 1-32. 2013.

[6] Yang, X.-S., & Deb, S. "Cuckoo search via Lévy flights. Paper presented at the Nature & Biologically Inspired Computing", 2009. NaBIC 2019.

[7] Eberhart, R. C., & Kennedy, J. "A new optimizer using particle swarm theory". 1995.

[8] V.K. Pathak, A.K. Singh, Optimization of morphological process parameters in contactless laser scanning system using modified particle swarm algorithm, Measurement (2017),

[9] Yang, X.-S. "Firefly algorithms for multimodal optimization. Paper presented at the International Symposium on Stochastic Algorithms". 2019.

[10] Chu, S.-C., Tsai, P.-W., & Pan, J.-S. "Cat swarm optimization". Paper presented at the Pacific Rim International Conference on Artificial Intelligence. 2016.

[11] Yang, X.-S. "A new metaheuristic algorithm cooperative Nature strategy bat-inspired inspired for optimization" (NICSO 2010) (pp. 65-74): Springer. 2010b.

[12] Dorigo, M., & Thomas, S. "Ant Colony Optimization". Cambridge, vol. 9, Dec. 20012: MIT Press. 2012.

[13] Lang, Y.; Gao, Y. Dream Optimization Algorithm (DOA): A novel metaheuristic optimization algorithm inspired by human dreams and its applications to real-world engineering problems. Comput. Methods Appl. Mech. Eng. 436, 117718, 2025.

[14] Dongshu Wang, Dapei Tan, Lei Liu, Particle swarm optimization algorithm: an

Corresponding author: Kazeem Lawal
✉ *dawooddahiru@gmail.com*
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

---

overview, Soft Comput. 22 (2018) 387–408

[15] Seyedali Mirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.

[16] Jiankai Xue, Bo Shen, A novel swarm intelligence optimization approach: sparrow search algorithm, Syst. Sci. Control Eng. 8 (1) (2020) 22–34

[17] Poomin Duankhan, Khamron Sunat, Sirapat Chiewchanwattana, Patchara Nasa-ngium, The Differentiated Creative search (DCS): Leveraging Differentiated knowledge-acquisition and Creative realism to address complex optimization problems, Expert Syst. Appl. (2024) 123734.

[18] Ziyu Guan, Changjiang Ren, Jingtai Niu, Peixi Wang, Yizi Shang, Great Wall Construction Algorithm: A novel meta-heuristic algorithm for engineer problems, Expert Syst. Appl. 233 (2023) 120905

[19] Seyedali Mirjalili, Andrew Lewis, The whale optimization algorithm, Adv. Eng. Softw. 95 (2016) 51–67.

[20] Salawudeen, A. T., Mu'azu, M. B., Yusuf, A., & Adedokun, A. E. "A Novel Smell Agent Optimization (SAO)": An extensive CEC study and engineering application. *Knowledge-Based Systems, 232*, 107486. 2021.

[2] Wu, Y., Gao, X. Z., & Zenger, K. 1"Knowledge-based Artificial Fish-Swarm algorithm". Paper presented at the 18th IFAC World Congress, Milano. 2011.

[22] Tang, K., Yáo, X., Suganthan, P. N., MacNish, C., Chen, Y.-P., Chen, C.-M., & Yang, Z. "Benchmark functions for the CEC'2008 special session and competition on large scale global optimization". Nature Inspired Computation and Applications Laboratory, USTC, China. 2007.

[23] Momin, J., & Yang, X. S. "A literature survey of benchmark functions for global optimization problems". International Journal of Mathematical Modelling and Numerical Optimization, 4(2), 150-194. 2013.

[24] Wang, Y., Ma, J., & Wang, Y. Application of ant colony algorithm in path planning of the data center room robot. Paper presented at the AIP Conference Proceedings. 2017.

[25] Hansen, N. "Compilation of results on the 2005 CEC benchmark function set". Online, May. 2006.

[26] Salawudeen, A. T. "Development of an Improved Cultural Artificial Fish Swarm Algorithm with Crossover". (Master of Science Thesis), Ahmadu Bello University Zaria, Nigeria., Kubani. (25) 2015.

[27] Jamil, M., & Yang, X.-S. "A literature survey of benchmark functions for global optimization problems. International Journal of Mathematical Modelling and Numerical Optimizations, 4(2), 150-194. 2013.

[28] Yang, X.-S. (2010). A new metaheuristic bat-inspired algorithm Nature inspired cooperative strategies for optimization. Nature Inspired Cooperative Strategies for Optimization 2010 Springer. 65-7

[29] Li, X., Tang, K., Omidvar, M. N., Yang, Z., Qin, K., & China, H. Benchmark functions for the CEC 2013 special session and competition on large-scale global optimization. 2013.

[30] Rahnamayan, S., Tizhoosh, H. R., & Salama, M. M. A novel population initialization method for accelerating evolutionary algorithms. Computers & Mathematics with Applications, 53(10), 1605-1614. 2007.

Corresponding author: Kazeem Lawal
✉ dawooddahiru@gmail.com
Department of Electrical and Electronics Engineering, Faculty of Engineering, Nigerian Defence Academy, Kaduna.

241