Volume 11, Issue 4, April – 2026 International Journal of Innovative Science and Research Technology
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

An Improved Whale Optimization Algorithm

Kwame Antwi Boasiako[1]

[1]Energy Consulting and Renewable Energy Electricity Company of Ghana Ltd. Accra, Ghana

Publication Date: 2026/08/19

Abstract: An improved algorithm that is used to alleviate the downsides of the whale optimization algorithm's (WOA) drawbacks which are slow convergence rate and falling into local optima, is proposed. There are five main variables introduced into this enhanced algorithm. These include the Cauchy density function, quasi-opposition-based learning, Euclidean distance, cubic chaotic mapping, and weight addition. The proposed algorithm was compared to the original WOA, two recent variants, and one other swarm-based intelligent algorithm, on twenty-three benchmark functions. The parameters used for comparison were Optimal value, Mean, Standard Deviation, and convergence rate. The improved WOA yields better optimal values with minimal iterations. The results validate that the Improved WOA outperforms the other four algorithms.

Keywords: *Whale Optimization Algorithm (WOA); Metaheuristic Optimization; Swarm Intelligence; Cubic Chaotic Mapping; Cauchy Density Function; Quasi-Opposition-Based Learning (QOBL); Euclidean Distance; Population Diversity; Convergence Rate; Global Optimization.*

How to Cite: Kwame Antwi Boasiako (2026) An Improved Whale Optimization Algorithm. *International Journal of Innovative Science and Research Technology*, 11(4), 5146-5156. https://doi.org/10.38124/ijisrt/26apr2221

I. INTRODUCTION

Metaheuristic optimization algorithms are high-level heuristic techniques [1] that can solve optimization problems without altering their structure since they are independent of optimization problem structure [2], [3]. Single solution-based and population-based techniques are the two primary groups of metaheuristics algorithms. Population-based algorithms, as opposed to single solution-based algorithms, can better explore the search space and exploit the global optimum [4]. According to the sources of inspiration, three broad categories can be applied to population-based algorithms: Swarm Intelligence Algorithms (SI), Evolutionary Algorithms (EAs), and Natural Phenomena (NP) [4]. Swarm-based algorithms among these population-based metaheuristics algorithms offer several advantages over NP and EAs. They typically have fewer operators than other population-based metaheuristics algorithms, making them simpler to implement [5].

A recently proposed algorithm known as the whale optimization algorithm is one of the swarm intelligent population-based algorithms. By imitating the hunting tactics of humpback whales, the whale optimization algorithm (WOA) is proposed [6]. When compared to many other swarm-intelligent population-based algorithms, the standard whale optimization algorithm has competitive benefits in handling complicated engineering issues and can produce solutions with higher precision [7]. Additionally, studies show that WOA predominates in conventional algorithms like PSO and GA [8].

The standard whale optimization algorithm, however, has some inherent flaws, including local optima stagnation, premature convergence, and an imbalance between exploration and exploitation [9], [10]. WOA has undergone a lot of modifications. The following are recent adjustments that have been made to improve WOA.

In [11], a modified WOA called OCDWOA used opposition-based learning, nonlinear parameter design, density peak clustering, and differential evolution to balance the ratio of the exploitation phase to the exploration phase, enhance the diversity of the population, and improve the ability to jump out of the local optimal solutions. WOAGWO [12], a hybridized form of WOA and GWO, was presented, with the GWO hunting mechanism included in the WOA exploitation phase. The purpose of GWO's hunting mechanism is to prevent whales from undesirable locations and to improve exploration efficiency. Levy flight trajectory-based WOA (LWOA) is introduced in [13]. The Levy flight trajectory helps to better the performance of escaping the local optimum and to increase population diversity against early convergence. The balanced variation of WOA (BWOA) proposed in [14] combined Levy flight and chaotic local search into WOA to improve the classic technique's inclusive exploratory and neighborhood-informed capabilities. A modified exploration strategy and quadratic interpolation are used in a quadratic interpolation-based WOA [15] to boost exploitation ability and search accuracy. RDWOA [16] was proposed as a more robust variant of WOA. To accelerate convergence, a random spare or random replacement technique is utilized. A double adaptive weight system is also employed to boost exploration ability early on and

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

exploitative ability later on. A hybrid PSOWOA was suggested by [17]. The standard WOA operator is used to investigate an uncertain environment, whereas the PSO operator is utilized during the exploitation phase. The experimental findings demonstrate the effectiveness of PSOWOA. [18] presented an altered WOA based on distinct searching routes and perceptual disturbance (PDWOA). Several spiral curves have been studied, and the findings show that the Archimedean spiral curve with equal pitch outperforms the others. The goal of WOA-BAT [19] was to improve WOA exploration and achieve a better solution during the exploitation phase. In comparison to conventional WOA and other algorithms, the WOA-BAT boosted the quality of findings.

Despite these improvements, the issues of premature convergence and local optimum trapping still need to be addressed comprehensively. Therefore, this study proposes an enhanced version of the WOA to overcome the aforementioned shortcomings of the WOA.

For improvement, the suggested technique employs five variables: Cauchy density function, weight inclusion, Quasi-opposition-based learning, Euclidean distance and cubic chaotic mapping is introduced to effectively improve the algorithm's global and local searchability, the population's quality and diversity enhance convergence rate, and improve population similarities. Thus, the 23 benchmark functions show that the proposed method is efficient, since only a few iterations are required to get the best solution.

The remainder of this paper is structured as follows. Section 2 describes the typical WOA algorithm. The proposed algorithm and its variables are described in Section 3. Section 4 displays the benchmark functions and test parameters used to evaluate the algorithms. Section 5 presents and analyzes the results. Finally, the conclusions are provided in Section 6.

II. GENERAL CONVENTIONAL WOA DESCRIPTION

To find prey, conventional WOA agents imitate three hunting behaviors of whales, namely encircling, attacking, and searching for prey.

- *Encircling Prey*

When hunting, whales can locate their prey and encircle them. Since the optimum value in this situation is unknown beforehand, an optimal agent or individual is required. As a result, the value that is closest to the optimum in the present population becomes the prey, and the rest of the search agents follow its lead to keep approaching it. The following are the mathematical expressions:

$$\vec{X}(t+1) = \overrightarrow{X_{best}}(t) - \vec{A}\cdot\vec{D} \tag{1}$$

$$\vec{D} = \left|C\cdot\overrightarrow{X_{best}}(t) - \vec{X}(t)\right| \tag{2}$$

$$\vec{A} = 2\vec{a}\cdot\vec{r} - \vec{a}$$
$$\vec{C} = 2\cdot\vec{r}$$
$$\vec{a} = 2 - \frac{2t}{T} \tag{3}$$

Where $\overrightarrow{X_{best}}$ is said to be the position vector of the best individual obtained so far; $\vec{X}$ is the position vector; t stands for the iteration number currently in effect; coefficient vectors are depicted by $\vec{A}$ and $\vec{C}$. $\vec{r}$ is a randomly generated number in the range [0,1]; $\vec{a}$ is the convergence factor, with a steady decline from trend from 2 to 0 with increasing iterations; T shows the maximum number of iterations.

- *Bubble-Net Attacking of Prey (Exploitation Phase)*

The search agents use the Shrinking Encircling Mechanism as well as the Spiral Updating Position to approach their target during the attack phase.

Shrinking Encircling Mechanism: As discussed earlier in the subsection for encircling prey, this is accomplished by linearly decrementing $\vec{a}$. Nevertheless, it can be observed that the range of $\vec{A}$ is also decreased by $\vec{a}$. Equations *1* and *2* define the shrinking encircling mechanism model mathematically.

Spiral Updating Position: When approaching prey, the whale moves in a spiral/helix pattern, which is modeled as:

$$\vec{D'} = \left|\overrightarrow{X_{best}}(t) - \vec{X}(t)\right| \tag{4}$$

$$\vec{X}(t+1) = \vec{D'}\cdot e^{bl}\cdot\cos(2\pi l) + \overrightarrow{X_{best}}(t) \tag{5}$$

Where $\vec{D'}$ is the distance vector between the current search agent and prey (optimal individual so far); b is a constant to determine the form of the logarithmic spiral; l is a random value in the range [–1,1].

Given these two strategies, it is predicted that there is a 50% probability that the whales will select one of them to update the whale position during optimization. The following equation models the link between spiral updating and shrinking encircling:

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X_{best}}(t) - \vec{A}\cdot\vec{D} & , \quad if\ p < 0.5 \\ \vec{D'}\cdot e^{bl}\cdot\cos(2\pi l) + \overrightarrow{X_{best}}(t), & \quad if\ p \geq 0.5 \end{cases} \tag{6}$$

Where p exhibits a random value in the range [0,1].

- *Prey Searching (Exploration Phase)*

Naturally, for the humpback whales to begin attacking, they must randomly choose the positions where they will be looking for prey, and they do this according to where each whale is located. To emulate search behavior, $|\vec{A}|$ with random values higher than one is used to induce the search agent to step away from a reference whale.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

Additionally, rather than updating with the optimal outcome thus far, the current solution will use a random solution from the current population.

It has the following mathematical expression:

$$\vec{D} = \left|C \cdot \overrightarrow{X_{rand}}(t) - \vec{X}(t)\right| \tag{7}$$

$$\vec{X}(t+1) = \overrightarrow{X_{rand}}(t) - \vec{A} \cdot \vec{D} \tag{8}$$

Where $\overrightarrow{X_{rand}}$ indicates a whale location in the current population chosen at random.

```
Initialize the whales' population Xi (i = 1, 2, ..., n)
Calculate the fitness of each search agent
X*=the best search agent
while (t < maximum number of iterations)
  for each search agent
    Update a, A, C, l, and p
      if1 (p < 0.5)
        if2 (|A| < 1)
          Update the position of the current search agent by the Eq. 2
        else if2 (|A| ≥ 1)
          Select a random search agent (X_rand)
          Update the position of the current search agent by the Eq. 8
        end if2
      else if1 (p ≥ 0.5)
        Update the position of the current search by the Eq. 5
      end if1
  end for
  Check if any search agent goes beyond the search space and amend it
  Calculate the fitness of each search agent
  Update X* if there is a better solution
  t=t+1
end while
```

Pseudocode 1 Conventional WOA

III. IMPROVED WOA

There are various drawbacks to conventional WOA. The computational limitations of the standard WOA formulae in its exploitation phase, together with how expensive it is to compute, cause it to quickly enter local optima and impede convergence rates. They are also undiversified because of how random it is in the initialization and exploration phase.

This section outlines five variables used to upgrade WOA: the Cauchy density function, quasi-opposition-based learning, Euclidean distance, cubic chaotic mapping, and the inclusion of a weight.

➢ *Cubic Chaotic Map Initialization Population.*

The basic WOA often randomizes the initial population, making it difficult to assure population diversity and, as a result, poor search results and efficiency. To improve the algorithm's performance during optimization, it is therefore required to increase the initial population's diversity. To overcome this issue, a chaotic map is utilized.

A deterministic nonlinear system can produce a certain form of random motion known as a chaos system or map. In addition to being regular, non-repeatable, and ergodic, it contains random characteristics. Due to these characteristics, the technique can escape the local optimal solution when solving function optimization problems, preserving population diversity and enhancing global searchabilities.

It is usually used to produce the population's initial position, which can effectively boost the population's diversity. Here, the cubic chaotic mapping [20] is used to initialize the whale population, whose definition as follows:

$$X_i = \rho X_{i-1}\left(1 - {X_{i-1}}^2\right), \qquad X_i \in (0,1) \tag{9}$$

Here, the sequence's chaotic number is denoted by X_i, where i stands for the iteration number and 'ρ' is a constant. From a number of assessed values on the cubic chaotic map in search of a better chaotic sequence, the value of ρ is found to be 2.59.

➢ *Cauchy Density Function*

The Cauchy density function is comparable to the Gauss density function; however, the density function of the Cauchy distribution has a higher two-wing probability feature and a vast distribution range than the randomness generated by that of the Gauss distribution, which substantially improves the global search efficiency of the algorithm and makes it simpler leave the local optimum [21].

The following is how the Cauchy distribution function is expressed:

$$F_t(x) = \frac{1}{2} + \frac{1}{\pi} tan^{-1}\left(\frac{x-a}{b}\right) \tag{10}$$

Where a is 0 and b is a proportional function with a positive value (= 1).

Both global and local searches employ the Cauchy operator to modify individual positions, which makes it simpler for the algorithm to exit the local optimal value and improves algorithm accuracy.

The accompanying changes have been made to the global search phase's location:

$$x = \vec{X}(t+1) \tag{11}$$

$$\overrightarrow{X_{new}}(t+1) = \frac{x}{Cauchy(x, a, b)} \tag{12}$$

The following changes have been made to the local search phase's location:

$$\overrightarrow{X_{new}}(t+1) = x \cdot Cauchy(x, a, b) \tag{13}$$

➢ *Quasi-Opposition Based Learning*

As the search range narrows in the WOA process, the population will continue to pursue the optimal solution, resulting in a decrease in population diversity. As a result, the quasi-opposition-based learning (QOBL) [22] technique was suggested to be applied to every iteration to continually update the individual position.

The QOBL equation incorporates the random generation of opposite values, which improves the algorithm's exploration/global capabilities.

It is utilized in this instance to compute the opposite solutions of agents to boost the probability of looking for global optimum solutions while also enhancing population quality and diversity during both the initialization and evolutionary phases. It aids in preventing the trapping of local optimum solutions.

$$c = \frac{lb + ub}{2} \tag{14}$$

$$\overrightarrow{X_{opp}}(t) = c + rand(0,1) \cdot (\vec{X}(t) - c) \tag{15}$$

➢ *Addition of Weight*

Traditional WOA, which relies on X_{rand}, is produced by a random search technique. The issue is that X_{rand}'s search strategy is very random; it may lead to population stacking and ignoring locations that ought to be explored more.

To overcome this problem, weight is added to X_{rand}. The weight value is obtained from the solution of using the required parameters for Levy flight calculation after a series of tested values. The weight and the new X_{rand} is given below:

$$W = 4.2426$$
$$\vec{X}_{r.new} = \overrightarrow{X_{rand}}(t) \cdot W \tag{16}$$

➢ *Euclidean Distance in Exploitation Phase*

In WOA, the formula for computing the gap between the leader and the whale for both exploration and exploitation is the absolute of their difference. Using that distance estimate for exploratory purposes seems to be the best option.

However, it appears that employing that same distance computation for exploitative purposes is not the greatest option. This is due to the fact that it restricts the depth to which exploitation may occur. As a result, the distance between the leader and the whale should be estimated using their shortest distance. This is represented by the Euclidean distance.

This will improve the convergence rate, as well as the local search capabilities and agent similarities/clustering.

The new distance between the leader and the whale is given below:

$$\overrightarrow{D'_{new}} = \left\| \overrightarrow{X_{best}}(t)^2 - \vec{X}(t)^2 \right\| \tag{17}$$

```
Initialize the whales' population using Eqn. 9
Generate 'opposite' population using Eqn. 15
Calculate the fitness of each search agent
X*=the best search agent
while (t < maximum number of iterations)
  for each search agent
    Update a, A, C, l, and p
      if1 (p< 0.5)
        if2 (|A|< 1)
          Calculate the position of the current search agent by the Eqn. 2
        else if2 (|A| >1)
          Select a random search agent (X_rand) using Eqn. 16
          Calculate the position of the current search agent by the Eqn. 8
        end if2
        Update the position of the current search agent by the Eqn. 12
      else if1 (p≥ 0.5)
        Calculate the position of the current search agent by the Eqn. 5 and Eqn. 17
        Update the position of the current search by the Eqn. 13
      end if1
  end for
  Generate 'opposite' positions for each search agent using Eqn. 15
  Check if any search agent goes beyond the search space and amend it
  Calculate the fitness of each search agent
  Update X* if there is a better solution
  t=t+1
end while
return X*
```

Pseudocode 2 Improved WOA

IV. TEST PARAMETER SETTINGS AND BENCHMARK FUNCTIONS

23 benchmark functions, numbered F1 through F23, were used in this section to assess the effectiveness of the suggested approach. These functions, the range of variation, the dimension or design variables, and the optimal value fmin are all displayed in Table 1. Unimodal functions with a single global optimal solution are F1 through F7. Multimodal functions F8–F13 have a substantial number of local optimal solutions. Multimodal functions having well-defined dimensions that cannot be changed in their mathematical formulations are known as fixed-dimension multimodal functions (F14–F23). In statistical comparison analysis, the mean and corresponding standard deviation are the metrics utilized.

The standard WOA, together with its two most recent variations, the OCDWOA [11] and WOA-BAT [19], as well as GWO [23], are compared with the suggested Improved WOA. The parameter b for all algorithms is set to 1. The parameter $\vec{a}$ for WOA, WOA-BAT, Improved WOA, and GWO decreases linearly from 2 to 0. For OCDWOA, the crossover probability CR is set to 0.9, the nonlinear parameter $\vec{a}$ declines exponentially from 2 to 0, and the zoom factor F is randomly selected (0.2, 0.8). The population size for each method is 30, and the maximum number of iterations is 1000. The algorithms are run 10 times for each benchmark function on an Intel(R) Core(TM) i5-6198DU CPU @ 2.30GHz 2.40 GHz and 8 GB. The software to run this test is MATLAB 2021a.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

Table 1 Description of Benchmark Functions

Function	Dimension	Range	f_{min}
F1	30	[-100,100]	0
F2	30	[-10,10]	0
F3	30	[-100,100]	0
F4	30	[-100,100]	0
F5	30	[-30,30]	0
F6	30	[-100,100]	0
F7	30	[-1.28,1.281	0
F8	30	[-500,500]	-419.9829xDim
F9	30	[-5.12,5.12]	0
F10	30	[-32,32]	0
F11	30	[-600,600]	0
F12	30	[-50,50]	0
F13	30	[-50,50]	0
F14	2	[-65,65]	1
FIS	4	[-5,5]	0.003
F16	2	[-5,5]	-1.0316
F17	2	[-5,5]	0.398
F18	2	[-2,2]	3
F19	3	[1,3]	-3.86
F20	6	[0,1]	-3.32
F21	4	[0,10]	-10.1532
F22	4	[0,10]	-10.4028
F23	4	[0,10]	-10.5363

V. RESULTS AND DISCUSSIONS

- *Statistical Results*

The statistics for the benchmark function results are shown in Table 2. Each comparison algorithm's best value attained (BV), mean/average (AVG), and standard deviation (SD) are listed.

Improved WOA produced the most effective values for BV, mean, and SD in F1, F2, F3, F4, and F7 unimodal functions. Although it had the poorest AVG and SD in F5, it had the second-best BV. It obtained the poorest BV and AVG statistics in F6. It did, however, have the finest SD.

For multimodal functions, the Improved WOA is the absolutely best in F10, and F11. With F9, all algorithms achieved efficient values except for GWO which had the worst values for AVG and SD. For F8, only GWO had the worst values for BV, but the Improved WOA was second best in terms of AVG and worst in SD. Improved WOA was second best in terms of BV, and in terms of AVG and SD, it was the worst in F12. Moreover, in F13, Improved WOA was also second best in BV, fourth best in AVG, and worst in SD.

All five algorithms produced effective BV when fixed-multimodal functions were taken into consideration. Additionally, in terms of AVG and SD, Improved WOA showed better performance except in F16 and F17. In F16, Improved WOA’s AVG was best, alongside the other algorithms. However, Improved WOA’s SD was the worst in F16. In F17, Improved WOA was worst in terms of AVG and SD.

- *Convergence Analysis*

Figure 1-3 depicts the convergence curves of all five techniques, with the x-axis labeled Iteration and y-axis labeled Best score obtained so far.

On F1, F2, F3, F4, F6, F7, F9, F10, F11, and F18, it can be seen that the Improved WOA had the best convergence rate. The convergence curves were second best in F12, F13, F14, F15, F19, F21, F22, and F23. Though it placed second, for these functions, it had the best starting points/values on the curves. The Improved WOA ranked third in F8 along with OCDWOA. Also, in F20, Improved WOA ranked third best. Finally, the Improved WOA displayed poor convergence on F5 and F17, though in F17, it had the best initial/starting point.

Table 2 Optimization Outcomes for the Benchmark Function Comparison

FUNCTION	ALGORITHM	OPTIMAL VALUE	MEAN	STD. DEVIATION
F1	WOA	4.3504e-163	4.8356e-155	5.8541e-155
	Improved WOA	0	0	0
	GWO	2.0794e-61	5.7375e-59	8.2418e-59
	OCDWOA	1.9992e-171	8.085e-156	1.8077e-155
	WOABAT	4.1347e-07	7.4069e-07	4.2001e-07
F2	WOA	2.841e-112	4.2058e-101	1.3296e-100

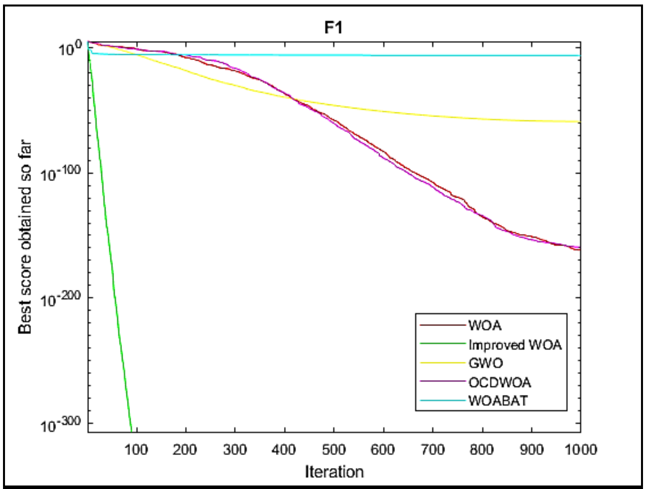
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

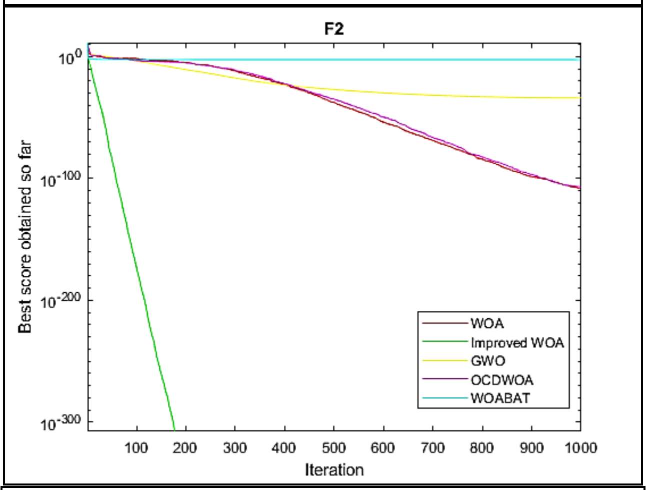
	Improved WOA	0	0	0
	GWO	1.5404e-35	1.1729e-34	1.2212e-34
	OCDWOA	4.2138e-113	5.8863e-105	1.7074e-104
	WOABAT	0.0018935	0.0070357	0.0021389
F3	WOA	1937.5	19556	15103
	Improved WOA	0	0	0
	GWO	1.3848e-18	1.6608e-14	5.1219e-14
	OCDWOA	3925.2	19183	11530
	WOABAT	6.4746e-06	7.9394e-06	1.3009e-06
F4	WOA	0.25664	36.56	28.864
	Improved WOA	0	0	0
	GWO	5.544e-16	1.9988e-14	3.7988e-14
	OCDWOA	0.0041386	37.855	28.924
	WOABAT	0.00084985	0.00097704	5.7122e-05
F5	WOA	26.406	27.298	0.58723
	Improved WOA	27.785	28.617	0.31661
	GWO	25.164	26.7	0.71807
	OCDWOA	26.245	27.145	0.47808
	WOABAT	3.6419e-05	4.9842	10.75
F6	WOA	0.0078927	0.032339	0.020221
	Improved WOA	2.4001e-05	1.1235	1.8055
	GWO	0.24889	0.68022	0.4043
	OCDWOA	0.0089105	0.19106	0.24159
	WOABAT	5.9016e-07	1.1307e-06	4.3036e-07
F7	WOA	1.7688e-05	0.0016111	0.0018073
	Improved WOA	6.835e-08	7.7691e-05	0.00017237
	GWO	0.00021479	0.00087487	0.00049291
	OCDWOA	5.3918e-06	0.0016288	0.0023195
	WOABAT	1.8071e-05	0.0014985	0.0028327
F8	WOA	-12569	-11184	1588.7
	Improved WOA	-12569	-11235	2200.2
	GWO	-7132.6	-5938	806.71
	OCDWOA	-12569	-11110	1642.3
	WOABAT	-12569	-12356	852.39
F9	WOA	0	0	0
	Improved WOA	0	0	0
	GWO	0	0.78337	3.0717
	OCDWOA	0	0	0
	WOABAT	5.892e-05	5.373	11.584
F10	WOA	8.8818e-16	4.0856e-15	2.4074e-15
	Improved WOA	8.8818e-16	8.8818e-16	0
	GWO	7.9936e-15	1.5667e-14	2.5234e-15
	OCDWOA	8.8818e-16	3.6593e-15	2.0666e-15
	WOABAT	0.00022017	0.00075616	0.00020295
F11	WOA	0	0.0039168	0.020398
	Improved WOA	0	0	0
	GWO	0	0.0026324	0.0063972
	OCDWOA	0	0.0049412	0.020277
	WOABAT	9.0156e-09	4.8371e-08	2.7949e-08
F12	WOA	0.00077012	0.0083736	0.0088227
	Improved WOA	7.5731e-09	0.15903	0.28845
	GWO	2.0491e-06	0.037743	0.018915
	OCDWOA	0.0007559	0.0066784	0.006251
	WOABAT	2.2543e-09	9.2448e-09	5.1426e-09
F13	WOA	0.012806	0.20975	0.14104
	Improved WOA	1.645e-07	0.32416	0.6089
	GWO	0.099701	0.49888	0.20142
	OCDWOA	0.022577	0.25786	0.23064

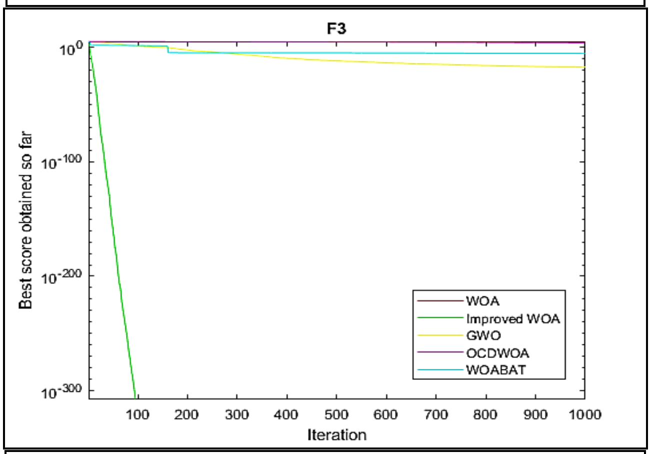
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

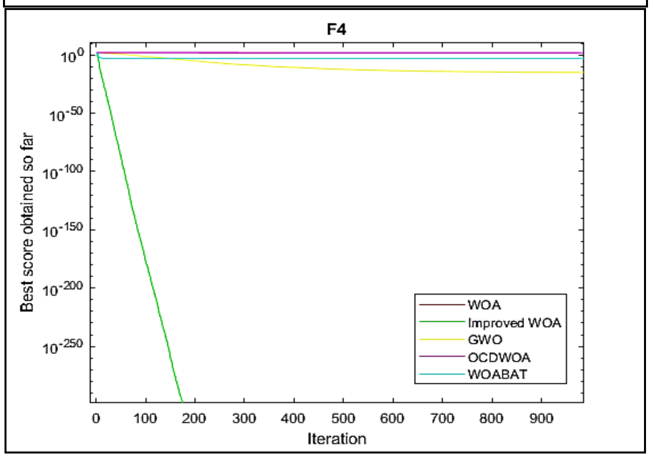
	WOABAT	3.3692e-08	1.4953e-07	7.9883e-08
F14	WOA	0.998	1.9745	3.088
	Improved WOA	0.998	1.122	0.25784
	GWO	0.998	5.8884	4.7011
	OCDWOA	0.998	2.0867	1.5744
	WOABAT	0.998	1.5885	1.8674
F15	WOA	0.00030786	0.0007029	0.00039039
	Improved WOA	0.00030754	0.0003206	2.9679e-05
	GWO	0.00030749	0.0039371	0.010022
	OCDWOA	0.00030796	0.00064285	0.00039401
	WOABAT	0.00030749	0.00033857	0.00017029
F16	WOA	-1.0316	-1.0316	3.5154e-11
	Improved WOA	-1.0316	-1.0316	3.3233e-05
	GWO	-1.0316	-1.0316	8.7969e-09
	OCDWOA	-1.0316	-1.0316	3.7285e-11
	WOABAT	-1.0316	-1.0316	5.0054e-16
F17	WOA	0.39789	0.39789	8.1641e-07
	Improved WOA	0.39789	0.3979	1.0214e-05
	GWO	0.39789	0.39789	8.2274e-07
	OCDWOA	0.39789	0.39789	2.2505e-06
	WOABAT	0.39789	0.39789	2.1787e-14
F18	WOA	3	3	2.0906e-05
	Improved WOA	3	3	2.9439e-06
	GWO	3	3	1.4797e-05
	OCDWOA	3	3	6.7768e-05
	WOABAT	3	7.86	10.478
F19	WOA	-3.8628	-3.859	0.0071731
	Improved WOA	-3.8628	-3.8626	0.0001642
	GWO	-3.8628	-3.8622	0.0017743
	OCDWOA	-3.8628	-3.8606	0.0025648
	WOABAT	-3.8628	-3.8625	0.0015601
F20	WOA	-3.322	-3.2637	0.082563
	Improved WOA	-3.3205	-3.3048	0.022168
	GWO	-3.322	-3.2562	0.06502
	OCDWOA	-3.322	-3.2511	0.090343
	WOABAT	-3.322	-3.2744	0.076253
F21	WOA	-10.153	-9.0764	2.1788
	Improved WOA	-10.153	-10.153	1.3699e-05
	GWO	-10.153	-9.5955	1.7156
	OCDWOA	-10.153	-9.2642	2.0221
	WOABAT	-10.153	-8.632	2.3473
F22	WOA	-10.403	-9.0662	2.7157
	Improved WOA	-10.403	-10.403	4.3047e-05
	GWO	-10.403	-10.403	0.00025957
	OCDWOA	-10.403	-8.6926	2.6405
	WOABAT	-10.403	-9.3465	2.1344
F23	WOA	-10.536	-7.8886	2.9121
	Improved WOA	-10.536	-10.536	4.2242e-05
	GWO	-10.536	-10.536	0.00019841
	OCDWOA	-10.536	-8.6247	2.943
	WOABAT	-10.536	-9.5677	2.0886

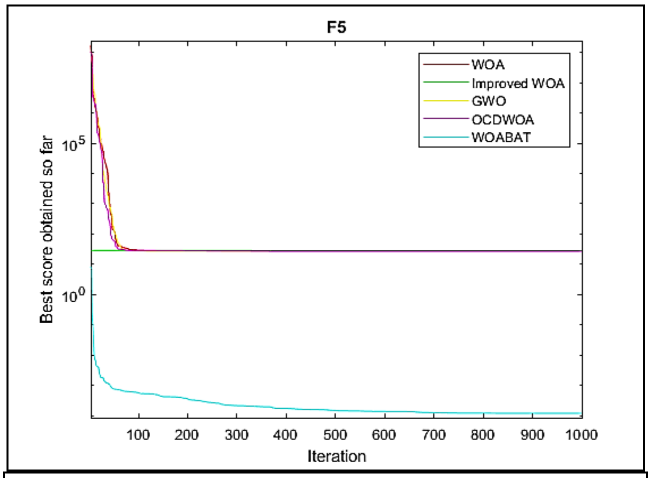
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

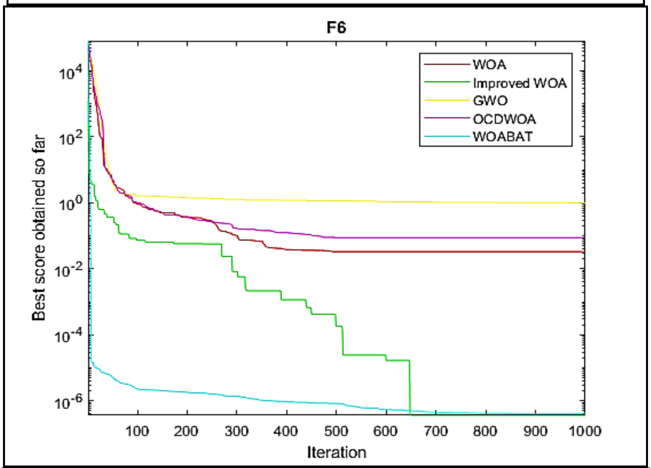


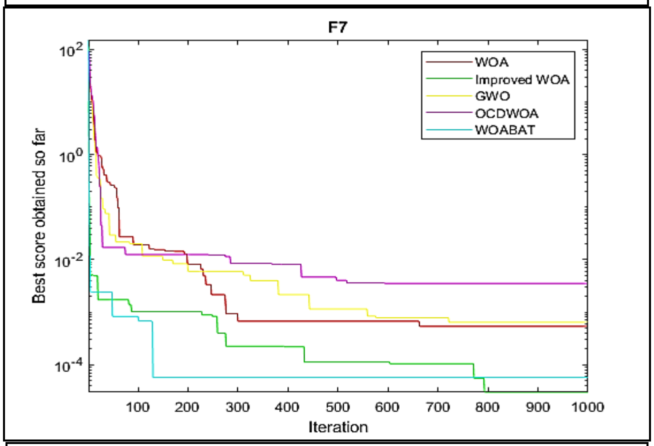


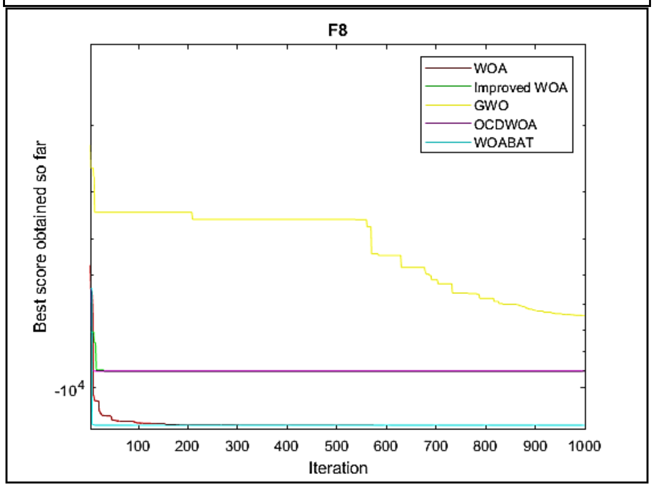




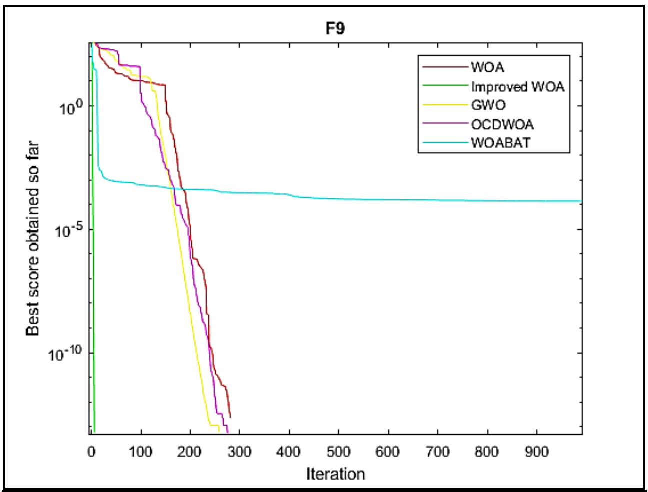


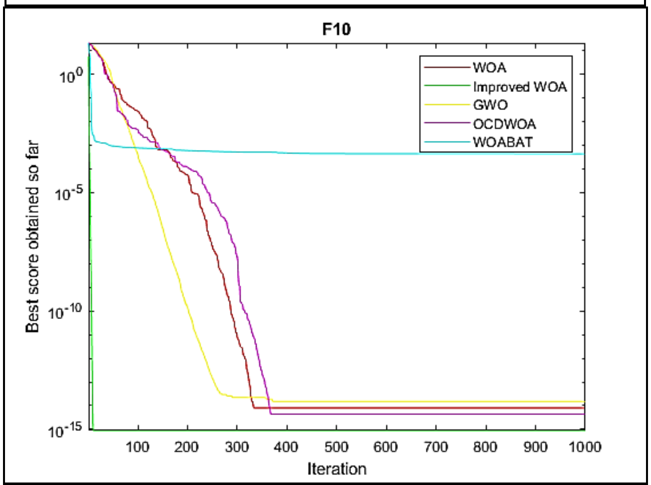


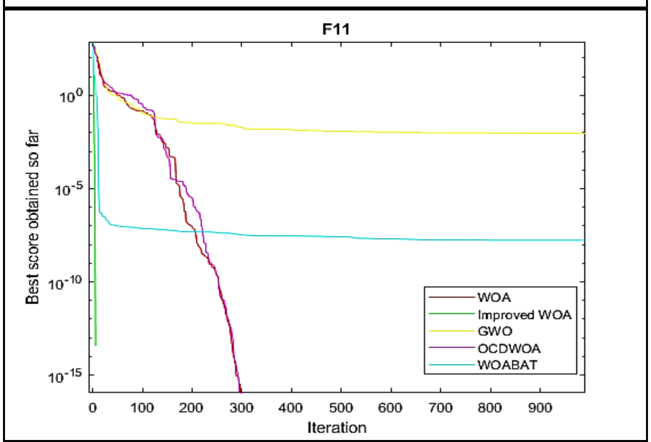


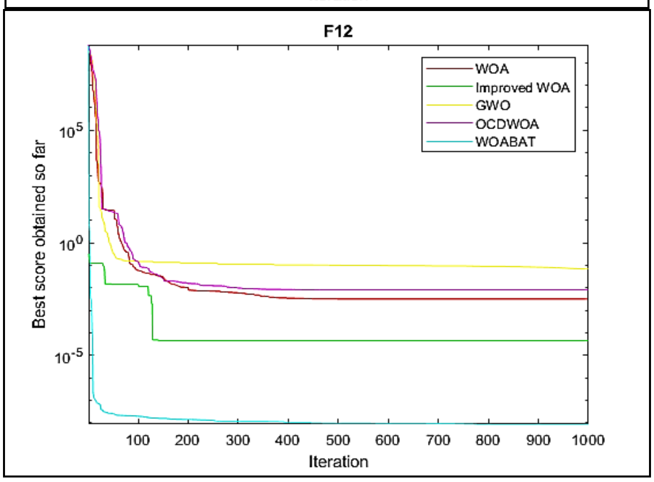


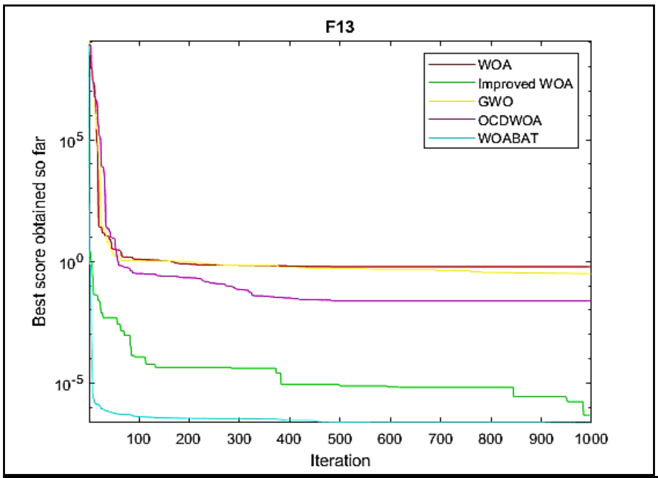
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

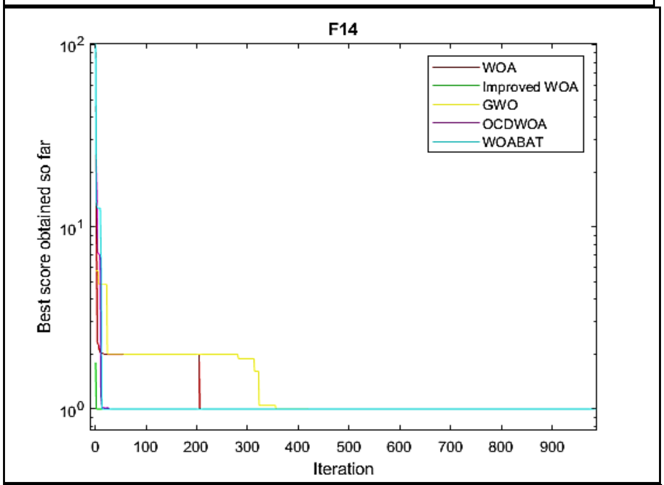


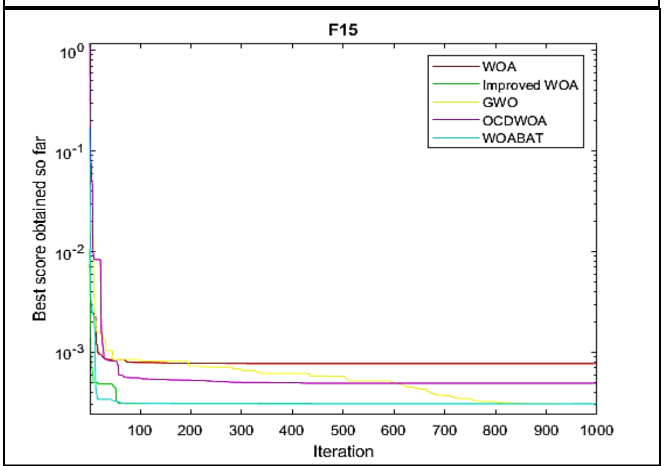


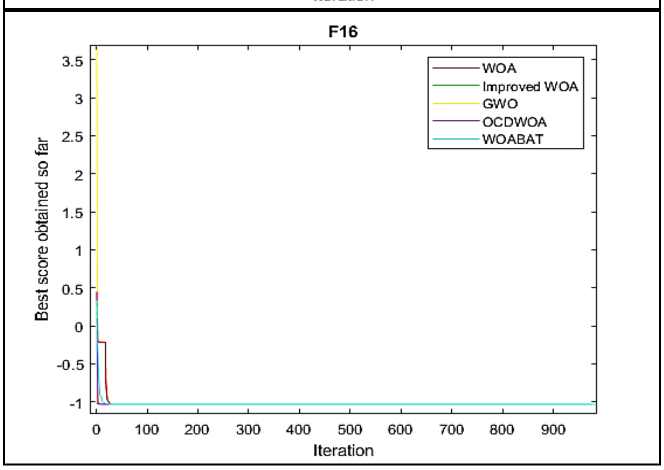




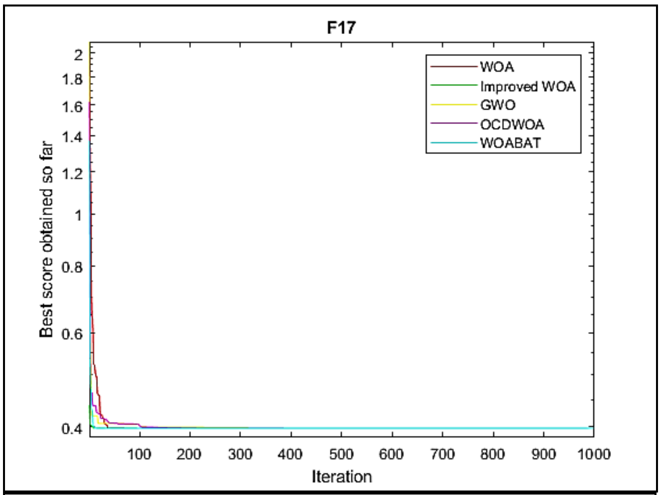


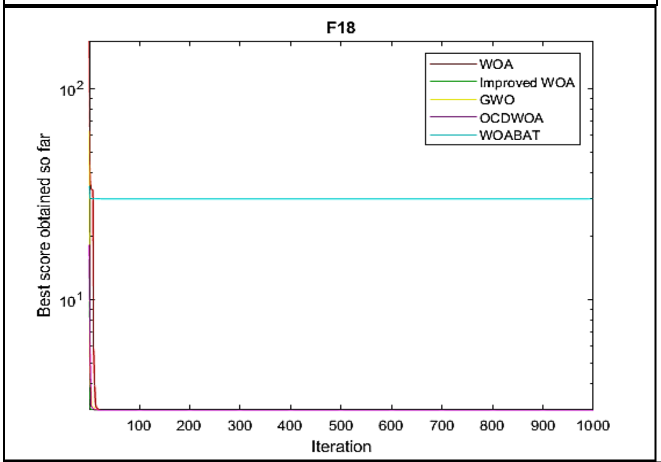


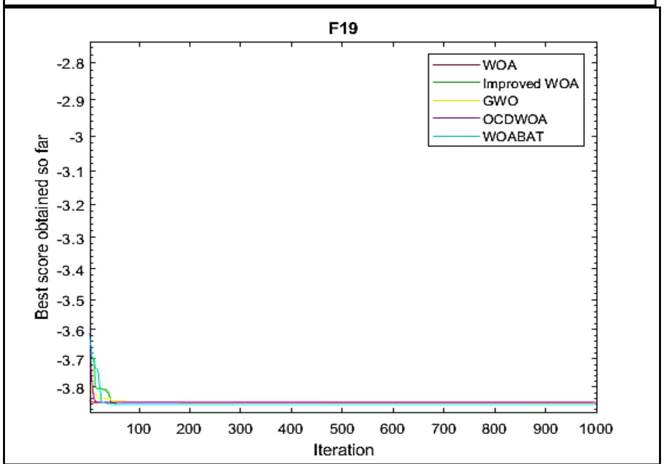


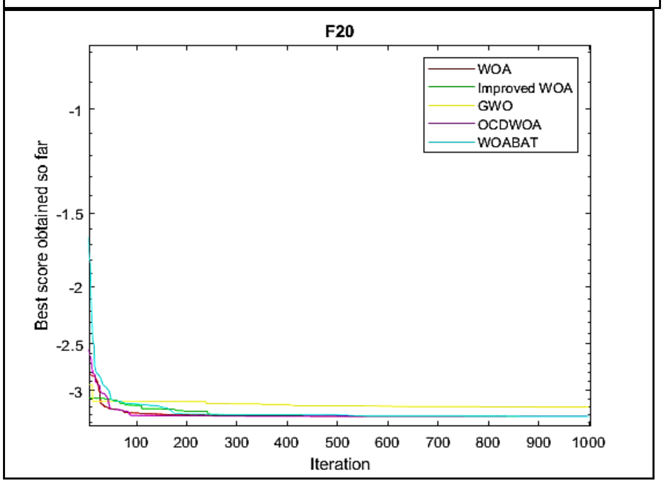


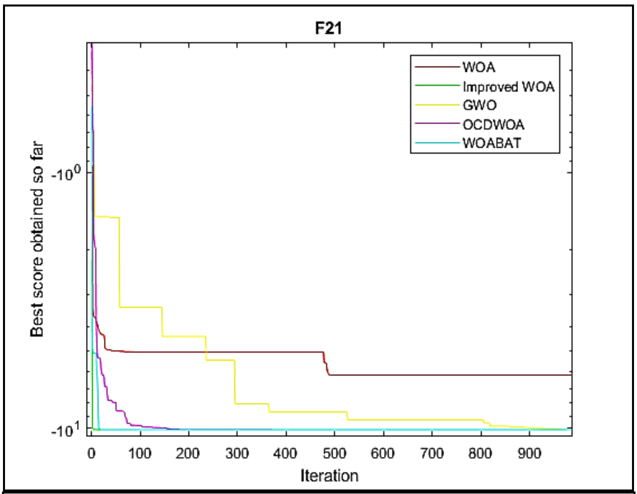
ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

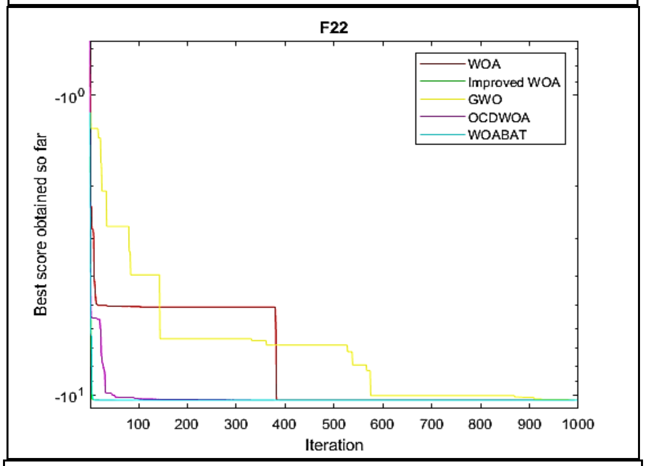












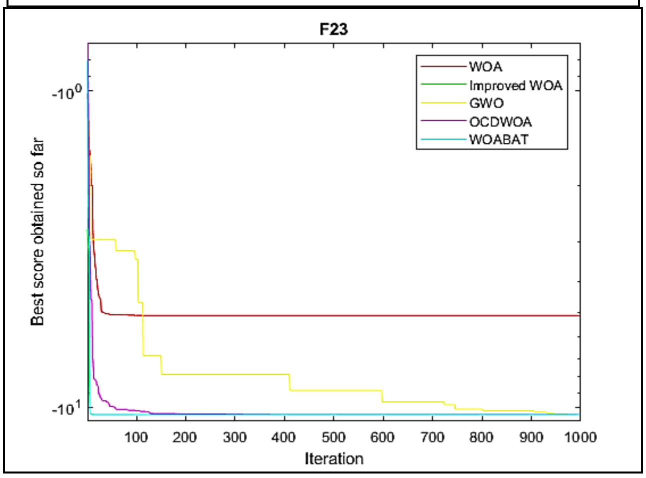


Fig 1 Comparison of F1 to F23 Convergence Curves

VI. CONCLUSION

A whale optimization algorithm that is improved has been suggested. This algorithm uses five variables to enhance the basic WOA. Out of the twenty-three benchmark functions used for testing, the Improved WOA showed superior performance in 16 functions in terms of BV, AVG, and SD. Better performance of the Improved WOA is also displayed in twenty-one functions considering the convergence rate compared to both the basic and other variants of WOA.

ISSN No: -2456-2165 https://doi.org/10.38124/ijisrt/26apr2221

REFERENCES

[1]. M. O. Okwu and L. K. Tartibu, *Metaheuristic Optimization : Nature-Inspired Algorithms Swarm and Computational Intelligence , Theory and Applications*. 2021.

[2]. P. D. Mohammad Kiani-Moghaddam, Mojtaba Shivaie, Weinsier, *Modern Music-Inspired Optimization Algorithms for Electric Power Systems Modeling, Analysis and Practice*. 2019.

[3]. J. A. Parejo, A. Ruiz-Cortés, S. Lozano, and P. Fernandez, "Metaheuristic optimization frameworks: A survey and benchmarking," *Soft comput*, vol. 16, no. 3, pp. 527–561, 2012, doi: 10.1007/s00500-011-0754-8.

[4]. I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Inf Sci (N Y)*, vol. 237, no. February, pp. 82–117, 2013, doi: 10.1016/j.ins.2013.02.041.

[5]. H. Zhang, J. Li, M. Hong, and Y. Man, "Artificial intelligence algorithm-based multi-objective optimization model of flexible flow shop smart scheduling," *Applications of Artificial Intelligence in Process Systems Engineering*, pp. 447–472, Jan. 2021, doi: 10.1016/B978-0-12-821092-5.00008-5.

[6]. S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software*, vol. 95, pp. 51–67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[7]. G. Chen, X. Zhao, Y. Xiang, X. Zeng, and H. Long, "Research on Multi-objective Active Power Optimization Simulation of Novel Improved Whale Optimization Algorithm," *IAENG International Journal of Applied Mathematics*, vol. 51, no. 3, pp. 1–20, 2021.

[8]. W. Yankai, W. Shilong, L. Dong, S. Chunfeng, and Y. Bo, "An improved multi-objective whale optimization algorithm for the hybrid flow shop scheduling problem considering device dynamic reconfiguration processes," *Expert Syst Appl*, vol. 174, no. April 2020, p. 114793, 2021, doi: 10.1016/j.eswa.2021.114793.

[9]. M. H. Nadimi-Shahraki, S. Taghian, S. Mirjalili, L. Abualigah, M. A. Elaziz, and D. Oliva, "Ewoa-opf: Effective whale optimization algorithm to solve optimal power flow problem," *Electronics (Switzerland)*, vol. 10, no. 23, 2021, doi: 10.3390/electronics10232975.

[10]. S. Chakraborty, A. K. Saha, R. Chakraborty, and M. Saha, "An enhanced whale optimization algorithm for large scale optimization problems," *Knowl Based Syst*, vol. 233, p. 107543, 2021, doi: 10.1016/j.knosys.2021.107543.

[11]. X. Liang, S. Xu, Y. Liu, and L. Sun, "A Modified Whale Optimization Algorithm and Its Application in Seismic Inversion Problem," vol. 2022, 2022.

[12]. H. Mohammed and T. Rashid, "A novel hybrid GWO with WOA for global numerical optimization and solving pressure vessel design," *Neural Comput Appl*, vol. 32, no. 18, pp. 14701–14718, 2020, doi: 10.1007/s00521-020-04823-9.

[13]. Y. Ling, Y. Zhou, and Q. Luo, "Lévy Flight Trajectory-Based Whale Optimization Algorithm for Global Optimization," *IEEE Access*, vol. 5, pp. 6168–6186, 2017, doi: 10.1109/ACCESS.2017.2695498.

[14]. H. Chen, Y. Xu, M. Wang, and X. Zhao, "A balanced whale optimization algorithm for constrained engineering design problems," *Appl Math Model*, vol. 71, pp. 45–59, 2019, doi: 10.1016/j.apm.2019.02.004.

[15]. Y. Sun, T. Yang, and Z. Liu, "A whale optimization algorithm based on quadratic interpolation for high-dimensional global optimization problems," *Applied Soft Computing Journal*, vol. 85, p. 105744, 2019, doi: 10.1016/j.asoc.2019.105744.

[16]. H. Chen, C. Yang, A. A. Heidari, and X. Zhao, "An efficient double adaptive random spare reinforced whale optimization algorithm," *Expert Syst Appl*, vol. 154, p. 113018, 2020, doi: 10.1016/j.eswa.2019.113018.

[17]. I. N. Trivedi, P. Jangir, A. Kumar, N. Jangir, and R. Totlani, "A novel hybrid PSO–WOA algorithm for global numerical functions optimization," *Advances in Intelligent Systems and Computing*, vol. 554, pp. 53–60, 2018, doi: 10.1007/978-981-10-3773-3_6.

[18]. W. Z. Sun, J. S. Wang, and X. Wei, "An improved whale optimization algorithm based on different searching paths and perceptual disturbance," *Symmetry (Basel)*, vol. 10, no. 6, pp. 1–31, 2018, doi: 10.3390/sym10060210.

[19]. P. Nandagopal, "Hybrid WOA-BAT Optimization for Performance Enhancement of Coexisting WBANs," 2021.

[20]. M. Vallabhaneni, B. Maddula, S. Jayan, and R. Subramani, "Chaotic Hooke-Jeeves Algorithm using Cubic map with MATLAB code," *2020 IEEE International Conference for Innovation in Technology, INOCON 2020*, pp. 2020–2023, 2020, doi: 10.1109/INOCON50539.2020.9298244.

[21]. H. Chen, M. Wang, and X. Zhao, "A multi-strategy enhanced sine cosine algorithm for global optimization and constrained practical engineering problems," *Appl Math Comput*, vol. 369, p. 124872, Mar. 2020, doi: 10.1016/J.AMC.2019.124872.

[22]. B. Mandal and P. K. Roy, "Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization," *International Journal of Electrical Power & Energy Systems*, vol. 53, no. 1, pp. 123–134, Dec. 2013, doi: 10.1016/J.IJEPES.2013.04.011.

[23]. M. O. Okwu and L. K. Tartibu, "Grey Wolf Optimizer," *Studies in Computational Intelligence*, vol. 927, pp. 43–52, 2021, doi: 10.1007/978-3-030-61111-8_5.