Abstract—Optimization is an important tool in making decisions and in analysing physical systems. In mathematical terms, an optimization problem is the problem of finding the best solution from among the set of all feasible solutions. The paper discusses the Whale Optimization Algorithm (WOA), and its applications in different fields. The algorithm is tested using MATLAB because of its unique and powerful features. The benchmark functions used in WOA algorithm are grouped as: unimodal (F1-F7), multimodal (F8-F13), and fixed-dimension multimodal (F14-F23). Out of these benchmark functions, we show the experimental results for F7, F11, and F19 for different number of iterations. The search space and objective space for the selected function are drawn, and finally, the best solution as well as the best optimal value of the objective function found by WOA is presented. The algorithmic results demonstrate that the WOA performs better than the state-of-the-art meta-heuristic and conventional algorithms.

Keywords—Optimization, optimal value, objective function, optimization problems, meta-heuristic optimization algorithms, Whale Optimization Algorithm, Implementation, MATLAB.

I. INTRODUCTION

Optimization is fundamental at every stage in our daily lives: a desire to improve the situation or be the best in almost every field. In engineering, for instance, we wish to create the most ideal results with the available resources. In the increasingly competitive world, we cannot simply be satisfied with “just satisfactory” solutions’ performance; rather we have to be looking to design the best systems. Designing new products in any field: aerospace, agriculture, automobile, biomedical, chemical, electrical, etc., we must use design tools which provide the desired results in a timely and economical fashion [7], [14], [15], [29], [34].

Optimization is central to any problem involving decision making regardless of whether in engineering, technology, software development, or in economics. The task of decision making involves selecting between various alternatives. The measure of goodness of the alternatives is described by an objective function. Optimization theory and methods deal with selecting the best alternative in the sense of the given objective function.

The area of optimization has received enormous attention in recent years, primarily because of the rapid progress in computer technology, including the development and availability of user-friendly software, high-speed and parallel processors, and artificial neural networks [22].

The optimization process involves creating a suitable model. The modeling is the procedure of identifying and expressing the objective, the variables, and the constraints of the problem, mathematically. An objective is a quantitative measure of the performance of the system to be minimized or maximized. The variables are the components of the system whose values are to be found. A constraint is a condition of an optimization problem that the solution must satisfy [15], [32], [34].

Mathematically, the optimization problem looks as follows.

\[
\text{minimize } f(x) \\
\text{subject to } x \in \Omega
\]

The function \( f: \mathbb{R}^n \to \mathbb{R} \) that we wish to minimize is a real-valued function, and is called the objective function, or cost function. The vector \( x \) is an \( n \)-vector of independent variables, that is \( x = [x_1, x_2, \ldots, x_n]^T \in \mathbb{R}^n \). The variables \( x_1, x_2, \ldots, x_n \) are often called as decision variables. The set \( \Omega \) is a subset of \( \mathbb{R}^n \), called as the constraint set or feasible set.

The optimization problems are of various types [3], [16], [22], [32]:

- Single-objective optimization algorithms
- Particle Swarm Optimization
- Optimization of problems with constraints
- Optimization of problems with binary and/or discrete variables
- Optimization of problems with multiple objectives
- Optimization of problems with uncertainties

In recent times, meta-heuristic optimization algorithms are becoming more and more popular in engineering applications. Typical features of meta-heuristic optimization algorithms are: (i) they rely on rather simple concepts and are easy to implement; (ii) they do not require gradient information; (iii) they can bypass local optima; (iv) they can be utilized in a wide range of problems covering different disciplines. Nature-inspired meta-heuristic algorithms solve optimization problems by mimicking biological or physical phenomena.

The meta-heuristic optimization algorithms can be classified into the following categories [16]:

- Evolutionary algorithms
- Physics-based algorithms
- Swarm-based algorithms
- Human-based algorithms

S. Adhirai is with the Department of Computer Science and Engineering, PDM University, Bahadurgarh (Delhi NCR), Haryana, India, PIN – 124507 (e-mail: adhiraiadhirai@hotmail.com).

Paramjit Singh is with the Department of Computer Science and Engineering, PDM University, Bahadurgarh (Delhi NCR), Haryana, India, PIN – 124507.

R. P. Mahapatra is with the Department of Computer Science and Engineering, SRM University, Modinagar, Ghaziabad (Delhi NCR), U.P., India, PIN – 201204.
The WOA is a nature-inspired meta-heuristic optimization algorithm proposed by Mirjalili and Lewis in 2016 [28]. It mimics the social behaviour of humpback whales. WOA has got so much attention that it has been tested with 29 mathematical optimisation problems and six structural design problems.

There are many versions of WOAs; in this paper, we will describe the basic WOA. Section II describes the basic WOA, Section III lists out some of the important applications of WOA Algorithm, Section IV presents the experimental results, and finally Section V concludes the paper.

II. BASIC WOA

Whales are considered as the biggest mammals in the world, they never sleep, and they are the highly intelligent animals with emotion [9], [11], [33]. The most interesting thing about the humpback whales, a species of whales, is their special hunting method. This foraging behaviour is called bubble-net feeding method [12], [28]. Humpback whales prefer to hunt school of krill or small fishes close to the surface. This foraging is done by creating distinctive bubbles along a circle or ‘9’-shaped path as shown in Fig. 1. The bubble-net feeding is a unique behaviour that can only be observed in humpback whales [28]. Here, spiral bubble-net feeding maneuver is mathematically modeled in order to perform optimization. Their favorite preys are krill and small fish herds. Fig. 1 shows this mammal.

Let us now consider the three typical behaviours of Humpback whales and model them mathematically.

A. Encircling Prey

The Humpback whales can recognize the location of prey and encircle them [28]. The WOA algorithm assumes that the current best candidate solution is the target prey or is close to the optimum. After the best search agent is defined, the other search agents will hence try to update their positions towards the best search agent. This behaviour is represented by:

\[ \vec{D} = |\vec{C} \times \vec{X}(t) - \vec{X}(t)| \]  

\[ \vec{X}(t+1) = \vec{X}(t) - \vec{A} \cdot \vec{D} \]  

where \( t \) indicates the current iteration, \( \vec{A} \) and \( \vec{C} \) are th coefficient vectors, \( \vec{X}^* \) is the position vector of the best solution obtained so far, \( \vec{X} \) is the position vector, | | is the absolute value, and is an element-by-element multiplication. It is worth mentioning here that \( \vec{X}^* \) should be updated in each iteration if there is a better solution.

The vectors \( \vec{A} \) and \( \vec{C} \) are calculated as:

\[ \vec{A} = 2 \vec{a} \cdot \vec{r} - \vec{a} \]  

\[ \vec{C} = 2 \vec{r} \]  

where \( \vec{a} \) is linearly decreased from 2 to 0 over the course of iterations (in both exploration and exploitation phases) and \( \vec{r} \) is a random vector in \([0,1]\).

As mentioned in the previous section, the humpback whales also attack the prey with the bubble-net strategy. This method is mathematically formulated as follows.

B. Bubble-Net Attacking Method (Exploitation Phase)

The bubble-net behaviour of humpback whales is mathematically modeled using following the two approaches:

(i) Shrinkring encircling mechanism: Here, the value of \( \vec{a} \) is decreased in (3). Also, the variation range of \( \vec{A} \) is also decreased by \( \vec{a} \). The new position of a search agent can be defined anywhere in between the original position of the agent and the position of the current best agent by setting random values for \( \vec{A} \) in \([-1,1]\) [Mirjalili and Lewis [28, pp. 53-54].

(ii) Spiral updating position: Here, firstly, the distance between the whale and the prey located at \((X, Y)\) and \((X^*, Y^*)\), respectively, is calculated. A spiral equation is then created between the position of whale and prey as follows (Mirjalili and Lewis [28, p. 54]):

\[ \vec{X}(t+1) = \vec{D} \cdot e^{b \cdot (2\pi l)} + \vec{X}^*(t) \]  

where \( \vec{D} = |\vec{X}^*(t) - \vec{X}(t)| \) and indicates the distance of the \( i \) th whale to the prey (best solution obtained so far), \( b \) is a constant for defining the shape of the logarithmic spiral, and \( l \) is a random number in \([-1,1]\).

Note that humpback whales swim around the prey within a shrinking circle and along a spiral-shaped path simultaneously. To model this simultaneous behaviour, it is assumed that there is a probability of 50% to choose between either the shrinking encircling mechanism or the spiral model to update the position of whales during optimization. The mathematical model is as follows:

\[ \vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - \frac{\vec{A}}{\vec{D}^2} \cdot e^{b \cdot (2\pi l)} + \vec{X}^*(t) & \text{if } p < 0.5 \\ \frac{\vec{D}^2}{\vec{D}^2} \cdot e^{b \cdot (2\pi l)} + \vec{X}^*(t) & \text{if } p \geq 0.5 \end{cases} \]

where \( p \) is a random number in \([0,1]\).

C. Search for Prey (Exploration Phase)

In this case, \( \vec{A} \) is used with the random values greater than 1
or less than −1 to force search agent to move far away from a reference whale [28]. This mechanism and $|\dot{A}| > 1$ emphasize exploration and allow the WOA algorithm to perform a global search. The mathematical model is as follows:

$$\overline{\dot{D}} = |\overline{\dot{c}}|\overline{X_{\text{rand}}} - \overline{\dot{x}}$$ (7)

$$\overline{x}(t+1) = \overline{X_{\text{rand}}} - \overline{A}\overline{D}$$ (8)

where $\overline{X_{\text{rand}}}$ is a random position vector (a random whale) chosen from the current population.

The pseudocode of the basic WOA algorithm is given below in Fig. 2.

### III. APPLICATIONS OF WOA

The WOA has been applied in almost all fields including engineering, medicine, control systems, natural language processing, economics, neural networks, construction engineering, image processing, etc. Some of the applications of WOA ranging over these different fields are listed below.

- Arabic Handwritten Characters [26]
- Breast Cancer Diagnosis [27]
- Combined Emission Constrained Economic Dispatch with Valve Point Effect Loading Problem [4]
- Economic Dispatch Problem [30]
- Emission Constraint Environment Dispatch Problem [31]
- Feature Selection [35]
- Global MPP Tracking of Photovoltaic System [5]
- Historic Handwritten Manuscript Binarisation [10]
- Liver Segmentation in MRI Images [21]
- Multilevel Thresholding Image Segmentation [8]
- Neural Network Training [1]
- Optimal Power Flow Problem [2]
- Optimal Siting of Capacitors in Radial Distribution Network [23]
- Skeletal Structures [17]
- Unit Commitment Problem Solution [18]
IV. EXPERIMENTATION AND RESULTS

MATLAB (abbreviated for MATrix LABoratory), developed by MathWorks, is a fourth-generation high-level programming language and interactive environment for numerical computation, visualization and programming. MATLAB provides powerful facilities for algebraic equations, algorithms, bioinformatics, calculus and differential equations, communications systems, computational biology, computational finance, control systems, creation of user interfaces, curve fitting, data acquisition and analysis, database access and reporting, differential equations, econometrics, fuzzy logic, image processing and computer vision, interfacing with programs written in other languages like C, C++, FORTRAN, Java, and Python, linear algebra, machine learning, matrix manipulation, neural networks, non-linear functions, numerical integration, optimization, parallel computing, plotting of data and functions, signal processing, statistics, symbolic mathematics, text analytics, trading toolbox, etc. [19], [20], [25]. MATLAB is widely used in all areas of applied mathematics, in education and research at universities, and in the industry. It is because of this unique and powerful phenomenon that the researchers are shifting to MATLAB for implementation of algorithms.

The main function main.m implementing the WOA algorithm coded in MATLAB R2013a is shown in Fig. 3 below.

The benchmark functions used in WOA algorithm are grouped as: unimodal (F1-F7), multimodal (F8-F13), and fixed-dimension multimodal (F14-F23). Out of these benchmark functions, we show the experimental results for some typical functions, namely for F7, F11, and F19.

For each benchmark function, we initialize the population size (i.e., number of search agents) to be 30 and maximum number of iterations to be (a) 400, (b) 500, and (c) 600. For each of these iterations, we load the details of the selected function by calling the function Load_Function_Details() to get the values of variables lb, ub, dim, and ObjectiveFunction. We then call the function WOA() with input parameters numberOfSearchAgents, Max_No_Of_Iterations, lb, ub, dim, ObjectiveFunction; and to get the output parameters as Best_score, Best_pos, and WOA_cg_curve. We then draw search space, and objective space for the selected function. We finally record the best solution as well as the best optimal value of the objective function found by WOA. The results are summarized in Figs. 4 and 5.

It can be observed from Figs. 5 and 6 that for iterations >= 500, the best optimal values of the objective function under consideration are consistent, whereas for iterations less than 500, the best optimal values of the objective function are not consistent.
<table>
<thead>
<tr>
<th>Trial No.</th>
<th>Function</th>
<th>No. of Iterations</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>F7</td>
<td>400</td>
<td>The best solution found by WOA is: 0.098581 0.033542 0.015444 0.033314 -0.015459 -0.018481 0.024695 -0.037702 0.031124 0.052429 0.028131 0.058106 0.10342 -0.10469 0.1014 0.025109 -0.12406 0.038962 0.038851 0.045022 0.032566 0.018397 -0.04467 0.0035048 0.082337 0.0032771 0.0013812 0.02281 0.0018845 -0.020522, and, the best optimal value of the objective function obtained by the algorithm is: 0.010911.</td>
</tr>
<tr>
<td>2</td>
<td>F7</td>
<td>500</td>
<td>The best solution found by WOA is: 0.0072913 0.043482 0.14471 -0.009688 0.0041403 -0.0020246 0.042303 0.018979 0.0075254 0.0030061 0.0070325 -0.0091676 0.045242 -0.051089 0.066826 0.049396 0.023803 0.044236 -0.0065359 -0.014048 -0.037641 0.031342 0.018399 0.053036 -0.080668 0.0093768 0.070078 -0.0065712 -0.020195, and, the best optimal value of the objective function obtained by the algorithm is: 0.0041892.</td>
</tr>
<tr>
<td>3</td>
<td>F7</td>
<td>600</td>
<td>The best solution found by WOA is: -0.031856 -0.019695 -0.099673 -0.043843 -0.011524 -0.046187 0.10471 -0.019369 0.057222 -0.025775 -0.00015645 0.01627 0.024126 0.025239 -0.046473 -0.042651 0.062806 -0.016215 0.043074 -0.03064 -0.05983 0.0058947 -0.00057045 -0.049203 0.007543 0.0080329 -0.010613 0.02612, and, the best optimal value of the objective function obtained by the algorithm is: 0.0036435.</td>
</tr>
<tr>
<td>4</td>
<td>F11</td>
<td>400</td>
<td>The best solution found by WOA is: 3.4383e-09 -2.4135e-09 3.6406e-09 6.2847e-09 -6.9836e-09 8.6736e-09 8.9532e-11 -1.3898e-08 1.152e-09 -2.8516e-08 0.01627 0.024126 0.025239 -0.046473 -0.042651 0.062806 -0.016215 0.043074 -0.03064 -0.05983 0.0058947 -0.00057045 -0.049203 0.007543 0.0080329 -0.010613 0.02612, and, the best optimal value of the objective function obtained by the algorithm is: 0.0036435.</td>
</tr>
<tr>
<td>7</td>
<td>F19</td>
<td>400</td>
<td>The best solution found by WOA is: 0.046991 0.55603 0.85275 and, the best optimal value of the objective function obtained by the algorithm is: -3.86.</td>
</tr>
<tr>
<td>8</td>
<td>F19</td>
<td>500</td>
<td>The best solution found by WOA is: 0.048637 0.55571 0.85259 and, the best optimal value of the objective function obtained by the algorithm is: -3.8602.</td>
</tr>
<tr>
<td>9</td>
<td>F19</td>
<td>600</td>
<td>The best solution found by WOA is: 0.10027 0.55571 0.85259 and, the best optimal value of the objective function obtained by the algorithm is: -3.8627.</td>
</tr>
</tbody>
</table>

We now present the 2-D Representation for F7, F11, F19 along with their Parameter Space, and Objective Space, in turn.

(i) Unimodal Benchmark Function F7

The 2-D Representation of Unimodal Benchmark Function F7 (number of iterations = 400) along with its Parameter Space and Objective Space is shown in Figs. 6 (a) and (b).

(ii) Multimodal Benchmark Function F11

The 2-D Representation of Multimodal Benchmark Function F11 (number of iterations = 500) along with its Parameter Space and Objective Space is shown in Figs. 7 (a) and (b).
The best solution found by WOA is:

\[ 3.4383 \times 10^{-9} \quad 2.4135 \times 10^{-9} \quad 3.6406 \times 10^{-9} \quad 6.2847 \times 10^{-9} \]
\[ -6.9836 \times 10^{-9} \quad 8.6736 \times 10^{-9} \quad 8.9532 \times 10^{-8} \quad 1.1388 \times 10^{-7} \]
\[ -2.3408 \times 10^{-7} \quad 1.6535 \times 10^{-7} \quad 1.1805 \times 10^{-7} \quad -2.6863 \times 10^{-7} \]
\[ 8.7489 \times 10^{-7} \quad 6.2810 \times 10^{-7} \quad 3.8511 \times 10^{-8} \quad -2.5484 \times 10^{-8} \]
\[ 4.3204 \times 10^{-8} \quad -2.2658 \times 10^{-8} \quad -3.0648 \times 10^{-8} \quad 9.5932 \times 10^{-9} \]
\[ 2.0312 \times 10^{-7} \quad 1.0984 \times 10^{-7} \quad -2.6022 \times 10^{-7} \quad -1.3715 \times 10^{-7} \]
\[ -3.9802 \times 10^{-7} \quad 1.4866 \times 10^{-7} \quad 8.3143 \times 10^{-7} \quad 1.8603 \times 10^{-7} \]

and, the best optimal value of the objective function obtained by the algorithm is: 0.

The best solution found by WOA is:

\[ 0.10027 \quad 0.55571 \quad 0.85259 \]

and, the best optimal value of the objective function obtained by the algorithm is: -3.8627.
Optimization has become an integral part of our daily lives. WOA is a novel nature-inspired meta-heuristic optimization algorithm. WOA being an optimization algorithm, it has been applied across almost all fields: chemistry, economics, electrical engineering, electronics engineering, mechanical engineering, image processing, information technology, medical sciences, natural language processing, neural networks, physics, and so on. WOA has been tested with 29 mathematical optimization problems and six structural design problems. The experimental results establish that the WOA is a powerful search algorithm. Optimization results prove that the WOA algorithm is very competitive as compared to the state-of-art meta-heuristic algorithms as well as conventional methods. The algorithm has been implemented in MATLAB because of the unique and powerful features of MATLAB which are not available in any other programming language. It was empirically observed that the best optimal values of the objective function under consideration are not consistent if the number of iterations is taken to be less than 500.

**REFERENCES**


[29] Touma, Haider J. "Study of The Economic Dispatch Problem on IEEE


