Soft Computing (2021) 25:10275–10286
https://doi.org/10.1007/s00500-021-05983-z

OPTIMIZATION

A modified whale optimization algorithm to overcome delayed convergence in artificial neural networks

Rashmi Kushwah[1] · Manika Kaushik[1] · Kashish Chugh[1]

Accepted: 15 June 2021 / Published online: 1 July 2021

Abstract

Artificial neural network (ANN) is modeled to predict and classify problems. However, in the training phase of ANNs discovering faultless values of the weights of a network is extremely troublesome. Traditional weight updating methods often get stuck into local optima and converge to optimal solutions very slowly. Therefore, to overcome these drawbacks a modified version of a nature-based algorithm which merges meta-heuristics with weight-updating technique of ANN has been used in this paper. Whale optimization algorithm (WOA) is a well-established, efficient and competitive algorithm inspired by the hunting mechanism of the whales including their behavior in finding and attacking their prey with their bubble-net feeding technique. In WOA, the next location of the search individuals or whales is modified depending on some probability. Due to the high exploration rate of WOA, there is a disproportion between exploration and exploitation in the WOA and it also converges to the solution slowly. Thus, to establish an equilibrium between exploration and exploitation a new variant of WOA called modified whale optimization algorithm (MWOA) is proposed to overcome the problem of delayed convergence. In MWOA, roulette wheel selection is combined with WOA to enhance the convergence speed of WOA. MWOA is tested on 11 benchmark functions, and the outcomes are compared with WOA. The results prove that MWOA has gained success in overcoming the problem of the slow convergence of WOA. Also, the results show that the proposed MWOA technique, when applied to ANN, can overcome the problems of traditional techniques and has improved the results.

Keywords Artificial neural network · Whale optimization algorithm · Meta-heuristic · Roulette wheel selection

1 Introduction

Artificial neural networks (ANNs) are processing frameworks which imitate the working of the human nervous system. Initially, ANNs were used to solve problems like a human cerebrum. However, after some time, thoughts moved to perform tasks in every field rather than just science. ANNs have widely been utilized in performing some day-to-day chores such as video games, machine interpretation and even in exercises (Schmidhuber 2015; Chatterjee et al. 2017).

The network adapts and adjusts itself by taking into account test perceptions; this process is known as learning. Learning includes modifying the loads and discretionary limits of the system to improve the precision of the outcome. Learning is finished, while looking at extra perceptions does not conveniently decrease the blunder rate. Significantly, in the wake of learning, the blunder rate ordinarily does not arrive at 0. On the off chance that in the wake of learning, the mistake rate is excessively high, and the system should normally should be updated. Essentially this is finished by characterizing a cost work that is assessed intermittently during learning. For whatever length of time that its yield keeps on declining, learning proceeds. The expense is much of the time characterized as a measurement whose worth must be approximated. The yields are real numbers, so when the mistake is low, the contrast between the yield (very likely a feline) and the right answer (feline) is little. Learning endeavors to diminish the aggregate of the distinctions over the observations. Most learning models can be seen as a clear use of the streamlining hypothesis and factual estimation.

Whale optimization algorithm (WOA) is a nature-based algorithm introduced in 2016 which imitates the hunting

✉ Rashmi Kushwah
rashmi.31130@gmail.com

Manika Kaushik
manika.kaushik1430@gmail.com

Kashish Chugh
chughkashish12@gmail.com

[1] Jaypee Institute of Information Technology, Noida, India

behavior of whales in locating and hunting down the preys. This strategy is famously known as bubble-net hunting strategy Mirjalili and Lewis (2016). The WOA algorithm consists of two parts: exploration (searching for prey) and exploitation (attacking prey). It is the prime requirement of every meta-heuristic method to establish an equilibrium among the exploration phase and the exploitation one.

Recently, the meta-heuristic optimization algorithms are becoming more and more popular in many research fields and engineering applications. The meta-heuristic optimization algorithms can bypass local optima, and it relies on rather simple concepts and is easy to implement Lee and Lu (2020). Nature-inspired meta-heuristic optimization algorithms are divided into four major categories such as evolutionary algorithms, physics-based algorithms, swarm-based algorithms and human-based algorithms Mirjalili and Lewis (2016), Lee and Lu (2020). The first group of nature-inspired methods includes evolutionary algorithms which are inspired by the laws of natural evolution. It starts with a randomly generated population and allows the population to be optimized over the course of generations. The most popular techniques in evolutionary algorithms are genetic algorithms (GAs) (Maulik and Bandyopadhyay 2000; Kushwah et al. 2019), evolution strategy (ES) Tinkle et al. (1970), genetic programming (GP) Weimer et al. (2009), biogeography-based optimizer (BBO) Simon (2008), population-based incremental learning (PBIL) Baluja (1994) and differential evolution (DE) (Storn and Price 1997; Wu et al. 2018). The second group of nature-inspired methods includes physics-based algorithms which emulate the physical rules in the universe. The most popular algorithms in this category are gravitational search algorithm (GSA) Rashedi et al. (2009), Big Bang–Big Crunch (BBBC) Erol and Eksin (2006) algorithm, charged system search (CSS) Kaveh and Talatahari (2010) algorithm and central force optimization (CFO) Formato (2007) algorithm. The third group is swarm-based algorithms which include ant colony optimization (ACO) Dorigo and Stutzle (2019), particle swarm optimization (PSO) Sharif et al. (2020), artificial bee colony (ABC) Xue et al. (2018) and cuckoo search (CS) Mareli and Twala (2018). The last category is based on human-based algorithms which are inspired by human behaviors. The most popular techniques in this category are group search optimizer (GSO) Abualigah (2020) and teaching–learning-based optimization (TLBO) Li et al. (2019) .

The solutions of highly nonlinear problems generally call for sophisticated optimization algorithms, and traditional algorithms may brawl to deal with such types of problems. Theoretical analyses of nature-inspired and swarm intelligence algorithms lack a unified mathematical framework to get in-depth understanding of their stability, rates of convergence and robustness. Also, it is not clear that in all nature-inspired algorithms what is the best values of algorithm-dependent parameters and how to tune those parameters to achieve the best result Yang (2020). Meta-optimization is a time-consuming process; therefore, a multi-fidelity approach based on meta-optimization method is proposed in Li et al. (2020) to enhance the parameter optimization. Multi-fidelity technique can significantly speed up meta-optimization systems and has the possibility to generalize to various nature-inspired optimizations. In Mohammadi-Balani et al. (2021) a golden eagle optimizer (GEO) algorithm is proposed to solve the global optimization problems. The GEO mimics the hunting procedure of golden eagles to improve the fitness and find the optimal solutions. The major finding of this work is that the GEO is proposed for solving multi-objective problems and applied on constrained engineering design problems. Yan et al. (2021) discussed that the basic WOA has the drawback of search stagnation, low calculation accuracy and slow convergence speed. To avoid premature convergence, an enhanced WOA is considered for global optimization. The ranking-based mutation operator is considered in the enhanced WOA so that the algorithm converges to optimal solutions and prevents the algorithm from falling into local optima.

The major issue in existing whale optimization algorithm is its slow convergence speed; due to this problem, it becomes very difficult to apply it in certain area problems especially when the problems contain multiple local optima. The limitation of traditional weight update techniques is the dependency on initial parameters, sticking into local optima and converges to optimal solutions very slowly. The contribution of the paper is to propose a new WOA variant which merges meta-heuristics with weight-updating technique of ANN to establish equilibrium between the exploring and exploiting phases in the search expanse of WOA. The paper presents an enhanced version of WOA, i.e., modified whale optimization algorithm (MWOA) which is based on the roulette wheel selection (RWS). The WOA algorithm can be optimized by overcoming its slow convergence speed problem using RWS which selects a fitness proportionate whale rather than a randomly selected whale to update the position of the current whale. The algorithm proposed is tested on various functions and compared with naive WOA. The result shows that MWOA has gained success in overcoming the problem of the slow convergence of WOA. In comparison to traditional weight update technique, MWOA reached a better solution. Also, the fast convergence of mean square error (MSE) proves that the proposed method works faster than the traditional one. The efficiency of the MWOA algorithm developed in this work is evaluated using 11 benchmark functions for the performance criterion out of which MWOA shows better results on 7 benchmark functions for the performance criterion average fitness, while for 2 benchmark functions it performs equally well. Optimization results demonstrate that MWOA is very competitive and converge faster as compared to the existing WOA. The hybrid method proposed, i.e., modified whale optimization algorithm (MWOA), has a unique blend of quality of WOA and roulette wheel.

1.1 Motivation and Objective

The traditional whale optimization algorithm (WOA) mimics the social behavior of humpback whales and inspired by the bubble-net hunting strategy. Whale is the biggest mammals in the world and it can grow up to 30 m long. Whales are considered as the fancy creatures which have seven different species such as killer, Minke, Sei, humpback, right, finback and blue. The whales are considered as predators, and they are highly intelligent animals with emotion. Whales take the breath from the surface of oceans and they have special cells in their brains similar to human called as spindle cells. The spindle cells in whales are double than human which makes them smarter and these cells are also responsible to think, learn, judge and communicate. Whales are mostly observed in groups. The size of adult humpback whale is huge almost as size of a school bus. It is observed that hunting is performed by producing distinctive bubbles along a circle which utilize tag sensors. The humpback whales have special hunting mechanism which is called bubble-net feeding method and it can only be seen in humpback whales.

In this paper whale optimization algorithm (MWOA) is proposed to overcome the problem of delayed convergence and optimization is performed by mathematical formulation of spiral bubble-net feeding maneuver. The traditional WOA mimics the hunting behavior of humpback whales and uses spiral to simulate bubble-net attacking mechanism of humpback whales. The proposed technique merges meta-heuristics with weight-updating technique of ANN to establish equilibrium between the exploring and exploiting phases and overcome the problem of delayed convergence.

2 Related work

Over the past few decades, ANNs have been explored and specialists have sought their application in classifying, estimating, recognizing and determining patterns (Schmidhuber 2015; Braik et al. 2008). The proficiency of ANNs is profoundly influenced by their learning procedure. Backpropagation calculation and its variations are considered as standard instances of inclination-based strategies and generally mainstream among analysts Kim and Jung (2015). There are three fundamental drawbacks in these existing strategies: the weights might get caught in neighborhood minima, gentle intermingling and a huge reliance on the underlying variables (Faris et al. (2016); Rakitianskaia and Engelbrecht (2012); Mirjalili (2015)). As solid options in contrast to gradient-dependent techniques, heuristic pursuit calculations have been proven in earlier writings to enhancing neural systems. When compared to gradient approaches, meta-heuristics proved their efficacy in keeping away from neighborhood minima Crepinsek et al. (2013).

Meta-heuristic algorithms are gaining popularity these days because they have maximum flexibility and are multifaceted Yang (2020). They contributed significantly to a number of optimization problems. There are two classes of meta-heuristics: the population-based class which consists of a defined number of solutions for optimization and the single solution class which starts with a single solution and then it is enhanced along with course of iterations Pandey et al. (2017). The population-based branch of meta-heuristics is further divided into swarm intelligence and evolutionary algorithms. These algorithms are necessarily based on some natural habits and behaviors. The evolutionary theory is the main root of the expansion of all evolutionary algorithms (EAs). EAs are quickly developing algorithms that use the same approaches which creatures naturally use to select.

Swarm intelligence (SI) is the class of meta-heuristic which depends on the actions, habits and intelligence of swarms Kennedy (2006). SI algorithms mimic the behavior of the natural groups like shoal of fishes, provinces of insects, bird swarms, development of bacteria and groups of animals. In SI algorithms, the approach of the swarm members towards the food sources depends on their intelligence and demeanor. In EA, it is of utmost importance that the best candidate survives as each individual follows the best candidate. If the best individual deviates from the right path, then every individual deviates from the right path and it becomes impossible to reach the best solution. In SI there are minute chances that the optimal solution deviates because in SI, selections are done based on a cumulative understanding of all the individuals of a population. Some of the algorithms that depend on SI are the particle swarm optimization (PSO) Sharif et al. (2020), based on bird and fish intelligence Kennedy (2011), and the ant colony optimization (ACO) Mirjalili (2019), which imitates the way ants make use of their community understanding in locating the minimal route between their home and source of food. Grey wolf optimizer (GWO) Mirjalili et al. (2014) is dependent on the hunting strategy and decisions made by grey wolves also; this algorithm makes use of their social rank system in its equations.

Mirjalili and Lewis (2016) introduced WOA which is based on the hunting strategy of whales. This strategy is sometimes known as the bubble-net hunting strategy. The most important characteristics of nature-inspired algorithms are finding the best candidate among all the swarms (exploration phase) and then adapting to the environment (exploitation phase). The exploration phase can be considered as spotting the prey in the whole search expansion carefully, while in the exploitation phase, instead of searching in the whole area of search, the prey is looked for in the most promising region, i.e., most likely to contain local optima. In WOA, the exploration phase is that in which whales spot their prey in the whole search region and the exploitation phase is when they encircle their prey Kaur and Arora (2018). It has been observed that there is an immense exploration in WOA due to which it reaches the global solution very slowly.

Sarkar et al. (2012) proposed a hybrid learning system which is a decision tree and genetic algorithm (DTGA). The DTGA received less classification error and comparatively less sensitive to missing data than its base learner C4.5. The DTGA has less time complexity than other GA-based approaches and handles the interpretability problem when compared with most of the learning classifier system. Data sampling is used in DTGA to overcome the imbalance problem of data set. In data sampling an equal proportion of class distribution is maintained to select overall informative rule and balanced features for classification. DTGA attains smaller standard deviation and achieves better average accuracy than other competent learners. Ghasemiyeh et al. (2017) presented a hybrid model to predict prices on stock exchange using artificial neural network models and meta-heuristic algorithms. The scheme uses cuckoo search, GA and particle swarm optimization. The model shows that swarm optimization is a dominant meta-heuristic approach to predict stock price as it find out the best fitness value in the neural networks (NNs). The model extracts event-knowledge from stock exchange history and input them into NNs. To build a structural system, it is important to sort out a large amount of data while producing information to support decision making with intelligence features. The scheme can be efficiently used to predict prices on stock exchange with a better convergence at less error rates.

Sana et al. (2019) proposed a mathematical model formulated as a multi-objective optimization problem considering ergonomic constraints and solved using genetic algorithm. The model allows the generation of diversified results and avoids the loss of convergence. The algorithm generates a set of quality of responses and shows better results when considering ergonomic risks. Takami et al. (2016) proposed teaching–learning-based optimization (TLBO) algorithm in supply chain management. The algorithm observes the effect of product portfolio management that considers both the perfect and imperfect quality products. The aim of the algorithm is to optimize the product portfolio, production rate and supply rate simultaneously. Haseli et al. (2020) presented base criterion method (BCM) framework to solve multi-criteria decision-making problems. In BCM, a base criterion is selected by the decision maker and then pairwise comparisons are acquired between base criterion and other criteria. Thereafter, a max-min problem is framed and solved to determine the weight of the criteria. The BCM is also used to find the lost pairwise comparisons to retain the pairwise comparisons matrix stable. The BCM has good consistency ratio and better ability to analyze missing pairwise comparisons. Birjandi et al. (2019) presented a method based on intuitionistic fuzzy theory to select contractor in bidding using multi-criteria group decision-making technique. The method uses reciprocal preference relation to complete information and credibility function to select the best contractor. A case study is also presented to select the contractor in a power plant project. Jamali et al. (2018) proposed a stochastic inventory control problem using discrete Markov-modulated demand. A simulation-based optimization technique is used to approximate good quality solutions of this problem. The model is suitable for the practical applications such as supply chain management. Ospina-Mateus et al. (2021) suggested a model for prediction of severity of traffic accidents by motorcyclists. The method develops the strategies to enhance road safety using genetic algorithm and simulated annealing technique. Data mining and machine learning techniques are used to inspect accidents by motorcyclists. The model is helpful to improve road safety.

Sanel et al. (2020) defined several similarity and distance measures for octahedron sets to apply the concept of octahedron sets to multi-criteria group decision making (MCDM). The MCDM method is presented using five steps: formation of ideal octahedron set decision matrix, construction of octahedron set decision matrix, determination of attribute weight, calculation of weighted similarity measure and ranking of alternatives. In this work, the results and consequences could be useful from the point of view of octahedron set theory. Lee et al. (2020) considered an octahedron set, internal octahedron sets and external octahedron sets and defined various properties of type i-order, type i-intersection and type i-union. Some properties of image and preimage of an octahedron set have also been discussed in their work. Senel (2016) defined a soft bitopological Hausdorff space (SBT Hausdorff space) and some of its new concept such as SBT point, SBT continuous function and SBT homeomorphism. The relationship is investigated between SBT space and SBT subspace. Senel (2018) presented the concept of soft ditopology relates to the soft topology and discussed various properties of soft interior and soft closure in soft ditopological subspace. The relationship between the soft topology and soft ditopology has been discussed. Also soft ditopological subspace and its related properties are considered in their work.

Abualigah (2020) has conducted survey of group search optimizer (GSO) algorithm. GSO is a nature-inspired optimization algorithm inspired by animal searching behavior and solves different optimization problems. The survey considers multiple applications of the GSO algorithms such as benchmark functions, classification, machine learning applications, networking and other problems. Several engineering application such as scheduling, control of power systems, economic load dispatch problem and optimal design problem are discussed. Multiple variants of group search optimizer are also discussed such as basic group searching optimizer, discrete group searching optimizer, modifications of group searching optimizer, hybridizations of group searching optimizer, chaotic group searching optimizer and multi-objective group searching optimizer.

Abedinpourshotorban et al. (2016) proposed electromagnetic field optimization (EFO) which is a physics-inspired meta-heuristic optimization algorithm inspired by the behavior of electromagnets. The EFO is a population-based algorithm which takes advantage of a nature-inspired ratio and the population is divided into three fields which is positive, negative and neutral. The nature-inspired ratio helps particles to converge quickly and effectively in EFO. Chou and pham Chou and Pham (2017) proposed a nature-inspired meta-heuristic optimization algorithm to enhance prediction accuracy. The algorithm discussed the predictive artificial intelligence models such as artificial neural networks, classification and regression trees, Chi-squared automatic interaction detector, generalized linear model and ensemble model. The algorithm provides an effective tool for making fast and correct predictions to solve practical problems.

Yazdani and Jolai (2016) proposed a lion optimization algorithm (LOA) that mimics special lifestyle of lions and their cooperation characteristics. The algorithm considers various characteristics of lions such as hunting, roaming, mating, defense and migration. In LOA, the encircling mechanism has an advantage that it offers an opportunity for solutions to escape from local optima. The results provide fast convergence and global optima as compared to other meta-heuristics algorithms. Faramarzi et al. (2020) proposed a nature-inspired meta-heuristic method called marine predators algorithm (MPA). The performance of MPA is evaluated on 29 test functions and 58 mathematical benchmark functions. The MPA is compared with three classes of existing optimization techniques. The optimization process of MPA is divided into three main phases based on different velocity ratio such as high velocity ratio, unit velocity ratio and low velocity ratio. Dhiman (2019) proposed a hybrid bio-inspired meta-heuristic optimization technique which mimics the huddling and swarm behaviors of emperor penguin optimizer. The efficiency of algorithm is assessed using three parameters such as scalability analysis, convergence analysis and sensitivity analysis. The huddling behavior of emperor penguin optimizer is defined using huddling boundary, temperature around the huddle and distance between each penguin.

All the WOA methods discussed above are known to be suffering from risk converging at a slow rate which is due to speeding the exploration phase in whales in some cases. Therefore, a hybrid version of WOA has been proposed in this paper to maintain a balance between the exploring and exploiting phases in the search expanse of WOA and to overcome the problem of slow convergence. This MWOA is merged with the ANN learning process that eases the process of weight update in ANN.

3 Proposed work

3.1 Artificial neural network and its working

ANNs have an uncommon structure of regulated neural systems. ANNs comprise parts called neurons. Neurons are appropriated over various layers and every layer is wholly associated with the forward layer. The primary layer is the input layer, and it inserts the information factors to organize. The endmost layer is the yield (output) layer. All the layers between the information layer and the yield layer are called hidden layers Basheer and Hajmeer (2000). The simplest kind of ANN is multilayer perceptron (MLP). In MLP, neurons are interconnected in a single direction. Each and every connection has some weight which ranges between [−1,1].

In the process of finding the output using a neural network, numerous steps are followed. Initially, the weighted average of the inputs from the input layer and their biases using Eq. 1 are determined, where In_i are the values of inputs having N inputs and w_{ij} are the weights of the connection. B_j are the biases.

$$\mathrm{Sum_j} = \sum_{i=1}^{N} \mathrm{w_{ij} In_i} + \mathrm{B_j} \tag{1}$$

To revive the outputs of different layers of ANN, activation functions are used. There are numerous activation functions which can be applied to MLP such as sigmoid, logarithmic, signum, ReLu, etc. The ReLu function is used in this work to stimulate the output of the hidden layer using Eq. 2.

$$g(z) = \begin{cases} z & \mathrm{z} > 0 \\ 0 & \mathrm{z} <= 0 \end{cases} \tag{2}$$

Output of each node k of the final layer O_k is calculated using Eq. 3

$$O_k = \sum_{i=1}^{m} W_{kj} p_i + B_k \tag{3}$$

W_{kj} is the weight of the connection between kth layer nodes and output layer nodes, while p_i is the input from previous layer to kth layer.

3.2 Whale optimization algorithm

WOA uses following three main steps in their hunting strategy:

1. Encircling prey mechanism
2. Bubble net attacking mechanism.
3. Exploration phase (Search for prey)

3.2.1 Encircling prey mechanism

Here the lead whale dives into the water first and then starts creating bubbles in a spiral path around the prey to trap them, behavior which is later followed by other whales. For updating positions, whales use Eqs. (4) and (5):

$$\overrightarrow{\mathrm{Dis}} = |\overrightarrow{\mathrm{R}}.\overrightarrow{\mathrm{Y}^*}(t) - \overrightarrow{\mathrm{Y}}(t)| \tag{4}$$

$$\overrightarrow{\mathrm{Y}}(t+1) = \overrightarrow{\mathrm{Y}^*}(t) - \overrightarrow{\mathrm{A}}.\overrightarrow{D} \tag{5}$$

Where t is indicating the current iteration and . indicates element-by-element multiplication. $\overrightarrow{\mathrm{R}}$ and $\overrightarrow{\mathrm{A}}$ are coefficient vectors. $\overrightarrow{\mathrm{Y}}$ shows the position vector and Y^* represents the position vector of the best solution obtained so far. In each iteration if there is a better solution then Y^* should be updated.

3.2.2 Bubble net attacking mechanism

This step is further divided into two parts:

(i) Shrinking encircling mechanism:

In this step, the value a is decreased using Eq. 6. A is estimated arbitrarily in the interval [− a, a], and in every cycle, we decrease the value a in the range 2 to 0. If we set A arbitrarily in the range [− 1,1], new spots of different whales are allocated in the area between the current spot and the best agent's spot. c is a random vector in [0, 1] and a is linearly decreasing from 2 to 0 over the course of iterations. The step of shrinking encircling mechanism is represented in Eqs. 6 and 7.

$$\overrightarrow{\mathrm{A}} = 2.\overrightarrow{\mathrm{a}}.\overrightarrow{\mathrm{c}} - \overrightarrow{\mathrm{a}} \tag{6}$$

$$\overrightarrow{\mathrm{R}} = 2.\overrightarrow{\mathrm{c}} \tag{7}$$

(ii) Spiral update mechanism:

In this step, the distance between the location of the whale (X, Y) and location of the prey (X*, Y*) is initially calculated. Then, a spiral equation is created between the whale's and prey's position which imitates the humpback whale's helix-shaped movements using Eqs.8 and 9.

$$\overrightarrow{\mathrm{Y}}(t+1) = \overrightarrow{\mathrm{Dis}'}.e^{fk}.\cos(2\pi k) + \overrightarrow{\mathrm{Y}^*}(t) \tag{8}$$

$$\overrightarrow{\mathrm{Dis}'} = |\overrightarrow{\mathrm{Y}^*}(t) - \overrightarrow{\mathrm{Y}}(t)| \tag{9}$$

where $\overrightarrow{\mathrm{Dis}'}$ represents the distance of the ith whale to the prey, constant f represents spiral movement of whales, k is an arbitrary value ranging from [−1,1] and . denotes element-by-element multiplication.

The humpback whales swim along a spiral-shaped path and around the prey within a shrinking circle simultaneously. To model this scenario, it is assumed that there is a probability of 50% to select between either the spiral model or the shrinking encircling mechanism to update the position of whales during optimization. The mathematical model to represent this behavior is shown in Eq.10:

$$\overrightarrow{\mathrm{Y}}(t+1) = \begin{cases} \overrightarrow{\mathrm{Dis}'}.e^{fk}.\cos(2\pi k) + \overrightarrow{\mathrm{Y}^*}(t) & \mathrm{q} >= 0.5 \\ \overrightarrow{\mathrm{Y}^*(\mathrm{t})} - \overrightarrow{\mathrm{A}}.\overrightarrow{\mathrm{Dis}} & \mathrm{q} < 0.5 \end{cases} \tag{10}$$

where q is any random number in [0,1].

3.2.3 Exploration phase (Search for prey)

For searching the prey, the variations are done on vector $\overrightarrow{\mathrm{A}}$. Whales randomly search their prey by looking at and updating positions depending on other whales. Therefore, $\overrightarrow{\mathrm{A}}$ vector outside the range [− 1,1] is also used which has random values so that whales do not stick into local optima. When compared to the exploitation phase, the position of each individual is updated according to a random whale instead of the best individual. To put enhance exploration, $|\overrightarrow{\mathrm{A}}| > 1$ is used, which allows WOA to focus on a global level rather than sticking to the local one. Mathematical model is represented in Eqs.11 and 12 as follows:

$$\overrightarrow{\mathrm{Dis}} = |\overrightarrow{\mathrm{R}}.\overrightarrow{Y_{\mathrm{rand}}} - \overrightarrow{\mathrm{Y}}| \tag{11}$$

$$\overrightarrow{\mathrm{Y}}(t+1) = \overrightarrow{\mathrm{Y}_{\mathrm{rand}}} - \overrightarrow{\mathrm{A}}.\overrightarrow{\mathrm{Dis}} \tag{12}$$

where Y_{rand} is a random whale chosen from the current population.

3.3 Roulette wheel selection (RWS)

RWS is also known as fitness proportionate selectionShukla et al. (2015). Among all the possible mutations, RWS acts as an administrator and chooses the potential one (Pencheva et al. 2009; Lipowski and Lipowska 2012). Fitness is allotted to all possible chromosomes or solutions. The probability of selection of each individual solution depends on its fitness level. Division in different parts of the wheel depends on the fitness values and each individual is given the portions according to their fitness. This algorithm of selection is used in casino machines. Each value of fitness is fractioned by the sum of all selections, consequently normalizing them to 1. Then, just like a roulette wheel a random selection is made.

Let fit_i be the fitness of individual i in the population, its selection probability is calculated as:

$$p_i = \frac{fit_i}{\sum_{j=i}^{N} fit_j} \tag{13}$$

where N is the number of individuals belonging to the population.

Lower fitness solutions are more likely to be eliminated because of their smaller portion in the wheel, while the solutions which have higher fitness have less probability to get eliminated. But, their selection is not guaranteed as none of the candidates has a probability equal to 1. Likewise, a weaker solution may also get selected because even though they have few chances of selection, it is not zero. Therefore, it is possible that a weaker solution may survive. Similarly, there is a probability for weaker solutions to get selected in the selection process. This can be actually helpful in some situations because there is a possibility that weak solutions might have some useful information or features which might be useful in the process. A search candidate is chosen independently and it might be considered similar to playing N games on the roulette wheel. Different algorithms for selection are also prevailing these days, such as tournament selection and stochastic universal sampling. These selection techniques are very easy to implement and also have less stochastic noise and are fast. These are implemented using the cumulative distribution function (CDF) of the probabilities relative to the fitness of individuals. To pick a candidate, any number is selected ranging in [0,1) and then by mapping it to the CDF we can pinpoint the individual. It can be related to the ball in the casino wheel falling in any portion of the wheel which has the width proportional to its probability. By applying the binary search over the CDF, we can quickly find the inverse of the universe random number which is included in the bin.

3.4 Modified whale optimization algorithm (MWOA)

The main characteristics of all the algorithms which are dependent or inspired by nature or natural actions and behaviors are electing the best and fittest individual amongst them and then making changes according to that best candidate. These attributes are referred to as exploration and exploitation. Maintaining a balance between both the exploration and exploitation phases is important to find the optimum solution in optimum time. In WOA, the combination of explorations and exploitation helps to reach a solution which has optimum value. WOA takes a very long time to reach the optimum solution because of its diverse exploration phase and much less exploitation. Therefore, the WOA method has the risk of slow convergence due to excess of exploration in whales for some cases (Kaur and Arora 2018; Mafarja et al. 2018; Ling et al. 2017).

Algorithm 1: Modified Whale Optimization Algorithm

```
Produce the population Y_i (i = 1, 2, 3, ..., n) of MWOA
arbitrarily;
Assign initial value to a, A, and R ;
Use the fitness function and find the elite individual, i.e., Y*;
citer = 1 ;
while citer <= TerminationCond do
    for every whale (individual);
    if |A| < 1 then
        Modify the locality of whale (individual) employing Eq. 5
        and Eq. 7 ;
    else
        Select the agent using Roulette Wheel Selection Yrow;
        Update the position of the current agent by Eq. 14 and Eq.
        15;
        end for;
        Revise a, A, and R;
        Determine the fitness of every whale (individual);
        Modify Y* if there is a better solution.;
        citer=citer+1;
    end
end
```

Algorithm 2: ANN weight updation using MWOA

```
Produce the population Y_i (i = 1, 2, 3, ..., n) of MWOA each of
length l;
Assign initial value to a, A, and R ;
citer=1;
while citer <= TerminationCond do
    for each whale vector;
    Allocate the values of whale vector Y_i to ANN's weights and
    biases;
    Calculate the output of network by training on the dataset.;
    Calculate the fitness of the whale vector using MSE Eq. 17;
    Update Y*, i.e., the best performing vector;
    end for;
    if |A| < 1 then
        Modify the locality of whale (individual) employing Eq. 5
        and Eq. 7;
    else
        Select the agent using Roulette Wheel Selection Yrow;
        Update the position of the current agent by Eq. 14 and Eq.
        15;
        Revise a, A, and R;
        Modify Y* if there is a better solution.;
        citer=citer+1;
    end
end
return Y*;
```

In this paper, a modified WOA has been proposed using the roulette wheel selection technique to overcome the problem of slow convergence. There is a possibility that the solution which has the higher probability (more fitness) will be selected, i.e., $Yrow$ using Eq. 13, and then the positions of the rest of the whales are updated according to $Yrow$ instead of $Yrand$. This prevents whales from moving in any random direction and converges to the solution fastly. The equation for updating the positions of the whales using RWS is shown in Eqs. 14 and 15.

$$\overrightarrow{\text{Dis}} = |\overrightarrow{\text{R}}.\overrightarrow{Y_{\text{row}}} - \overrightarrow{\text{Y}}| \tag{14}$$

$$\overrightarrow{\text{Y}}(t+1) = \overrightarrow{\text{Y}_{\text{row}}} - \overrightarrow{\text{A}}.\overrightarrow{\text{Dis}} \tag{15}$$

Working of the MWOA is explained in Algorithm 1.

Now, to apply MWOA for ANN weight update, each whale is considered as a one-dimensional vector. Each whale vector comprises the weights and biases between the input and the hidden layer and between the hidden layer and the output layer Aljarah et al. (2018).

Eq.16 presents the method to calculate length of the whale vector.

$$l = x * y + (2 * y) + 1 \tag{16}$$

where x represents the number of input layer nodes, y is the total number of hidden layer neurons while l is the length of each whale vector. Whale vector length l indicates that each and every node present in ANN network. To calculate the value l, we need to count number of nodes inside our network, number of input layer nodes and total number of hidden layer neurons.

The fitness function which is used to find out the fitness of whale vectors is the mean square error (MSE). MSE is the variation of the value obtained by training a' from the required value a. It is shown in Eq.17

$$MSE = \frac{1}{n}\sum(a - a')^2 \tag{17}$$

The algorithm of using MWOA for ANN weight updation is shown in Algorithm 2.

4 Experimental results

The efficiency of the MWOA method has been evaluated on a desktop with a 2.66 GHz core i5 processor and 8 GB RAM using MATLAB 2016a. Both methods have been run 1000 times, and the number of search agents is 30. For every benchmark function, the WOA has run for 30 times.

To verify the results, the MWOA algorithm is compared with standard WOA Mirjalili and Lewis (2016). The results are displayed in Table 1 which shows average fitness. From the outcomes given in Table 1, it can be obtained that the MWOA shows better results on 8 benchmark functions for the performance criterion average fitness, while for 2 benchmark functions it performs equally well.

Existing WOA Mirjalili and Lewis (2016) slowly converges the solution; therefore, for the sake that the algorithm will take less time to converge, we have used MWOA and the verification is done using convergence curves as shown in Fig. 1 and Fig. 2. In convergence curves, the X- axis shows the number of iterations and the Y- axis shows the fitness. By the keen observation of convergence curves, it can be shown that MWOA can overcome the slow convergence problem for the multimodal function in Fig. 1, while for the unimodal function in Fig. 2, both WOA and MWOA perform equally well. To prove the efficacy of the proposed method, 11 benchmark functions have been considered as mentioned in Table 2 (Simon 2008; Pandey and Rajpoot 2019).

Table 1 Fitness comparison of WOA and MWOA

Fn	WOA	MWOA
Fn1.	**5.71e-128**	1.04e-111
Fn2.	9.76e-64	**4.05e-64**
Fn3.	2.67e1	2.87e1
Fn4.	6.00e-3	**1.15e-5**
Fn5.	− 1.76e3	− 1.90e3
Fn6.	3.52e-1	**3.07e-2**
Fn7.	3.05e-10	**3.29e-15**
Fn8.	5.22e-80	**1.85e-100**
Fn9.	4.77e-2	**2.38e-3**
Fn10.	1.80e-1	**8.80 e-24**
Fn11.	3.52e-1	**3.07e-3**

Bold indicate the better performance between WOA and MWOA on 11 benchmark functions for the performance criterion average fitness in Table 1

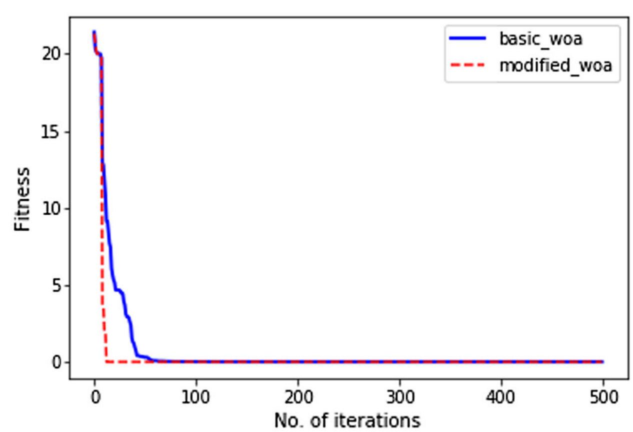

Fig. 1 Convergence plot for the proposed approach and existing approach Mirjalili and Lewis (2016) on Griewank function (multimodal)

Table 2 Benchmark functions

Sr. No.	Function Name	Equation	Optimal-value and Category
1.	Sphere	$F(X) = \sum_{i=1}^{d} x_i^2$	Unimodal
2.	Schwefel3	$F(X) = \sum_{i=1}^{d} \lvert x_i \rvert + \prod_{i=1}^{d} \lvert x_i \rvert$	Unimodal
3.	Quartic	$F(X) = \sum_{i=1}^{d} i x_i^4$	Unimodal
4.	Rastrigin	$F(X) = 10d + \sum_{i=1}^{d} (x_i^2 - 10cos(2\pi x_i))$	Multi-modal
5.	Ackley	$F(X) = \max_i (\lvert x_i \rvert : i \epsilon \{1, \cdots, d\})$	Multi-modal
6.	Griewank	$F(X) = 1 + \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos(\frac{x_i}{\sqrt{i}})$	Multi-modal
7.	Penalty1	$F(X) = 10\sin^2(\pi x_1) + \sum_{i=1}^{d-1} (x_i - 1)^2 [1 + 10\sin^2(\pi x_{(i+1)})] + (x_d - 1)^2 + \sum_{i=1}^{d} u_i$	Multi-modal
8.	Penalty2	$F(X) = \sum_{i=1}^{d} u_i + 0.1 \left\{10\sin^2(3\pi x_1) + \sum_{i=1}^{d-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_{i+1})] + (x_d - 1)^2 [1 + \sin^2(2\pi x_d)]\right\}$	Multi-modal
9.	Alpine	$F(X) = \sum_{i=1}^{d} \lvert x_i \sin(x_i) + 0.1 x_i \rvert$	Multi-modal
10.	Brown	$F(X) = \sum_{i=1}^{d-1} (x_i^2)^{(x_{i+1}^2+1)} + (x_{i+1}^2)^{(x_i^2+1)}$	Multi-modal
11.	Powell's Second Singular	$F(X) = \sum_{i=1}^{d-2} (x_{i-1} + 10x_i)^2 + 5(x_{i+1} - x_{i+2})^2 + (x_i - 2x_{i+1})^4 + 10(x_{i-1} - x_{i+2})^4$	Multi-modal

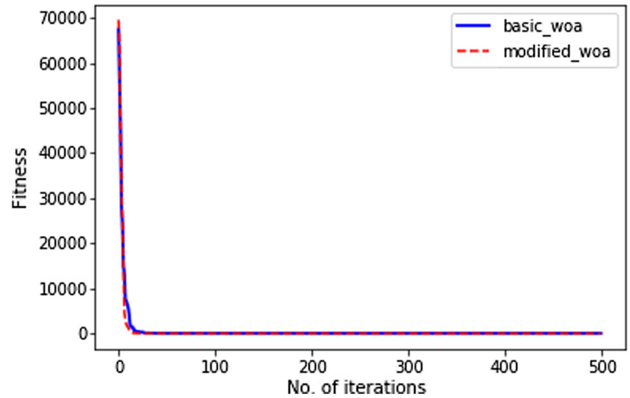

Fig. 2 Convergence plot for the proposed approach and existing approach Mirjalili and Lewis (2016) on Sphere function (unimodal)

To assess the performance of MWOA on ANN weight update, two datasets have been used. Dataset 1 is churn modeling in which the number of inputs is 11 and the number of hidden layer neurons is varied. Dataset 2 is the MNSIT dataset, in which the number of inputs is 24*24 images encoded in the form of digits and the number of hidden layer neurons is also varied. Back-propagation has been applied to both the datasets and compared with MWOA weight update technique. The results of using the back propagation technique on dataset 1 with different and varying parameters are given in Table 3, while the results of using back-propagation on dataset 2 are given in Table 4.

The network model of dataset 1 is shown in Fig 3 when MWOA has been applied to ANN for weight update using the proposed algorithm with varying number of hidden layer neurons. The results produced are given in Table 5. Similarly, the network model of dataset 2 is shown in Fig 4 using the proposed algorithm when a variable number of hidden layer neurons are present in the network dataset 2 and the results produced are given in Table 6.

Mean square error (MSE) is the variation of the value obtained by training from the required value. The proposed algorithm needs to reach the best solution easily. Therefore, the convergence curve of MSE is plotted in which the X-axis shows the number of iterations while Y-axis shows the value of MSE. Figs. 5 and 6 show the convergence of MSE on both datasets.

5 Conclusion and future work

To strengthen the convergence rate and to achieve better performance, a new variant of WOA, namely the modified whale optimization algorithm (MWOA) has been proposed. It can be seen that RWS has helped in selecting only the best agents rather than selecting random candidates which helped

Table 3 Accuracy measure by tuning parameters on dataset 1

S. No.	No. of epochs	No. of hidden layer neurons	Learning Rate	Test/Train Ratio	Accuracy (%)
1.	5	10	0.9	40:60	72.65
2.	7	7	0.4	30:70	70.25
3.	10	15	0.01	20:80	80.56
4.	10	12	0.02	10:90	80.23
5.	15	8	0.1	35:65	78.45

Table 4 Accuracy measure by tuning parameters on dataset 2

S.no.	No. of epochs	No. of hidden layer neurons	Learning rate	Test/train ratio	Accuracy (%)
1.	5	15	0.9	40:60	88.977
2.	7	18	0.4	30:70	92.66
3.	10	17	0.01	20:80	89.55
4.	10	12	0.02	10:90	91.39
5.	15	16	0.1	35:65	89.33

Table 5 MWOA on ANN with dataset 1

Sno.	No. of hidden layer neurons	Accuracy (%)
1.	10	84.85
2.	8	82.26
3.	7	79.25
4.	1	84.87
5.	12	84.91

to converge to the solution faster. The limitation of traditional weight update techniques is the dependency on initial parameters and sticking into local optima which has been solved using MWOA for updating weights of ANN. In comparison to traditional weight update technique, MWOA reached a better solution. Also, the fast convergence of MSE proves that the proposed method works faster than the traditional one. Although MWOA has proved its efficacy in updating the weights of ANN, there are some scopes of improvement. The number of hidden layers is increased as it leads to incremented learning of ANN. The MWOA weight updating technique, which has been used for classification purposes, can also be used for prediction and other considerable problems. Also hybridization with evolutionary search schemes may be the subject of future studies.

Conflict of interest The authors declare that they have no conflict of interest. This article does not contain any studies with human participants or animals performed by any of the authors.

Declarations

Conflict of interest The authors declare that they have no conflict of interest regarding the publication of this paper.

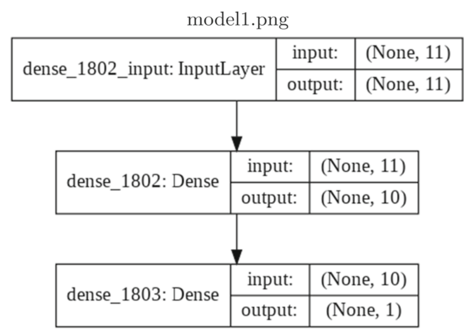

Fig. 3 Network model of ANN using MWOA on dataset 1

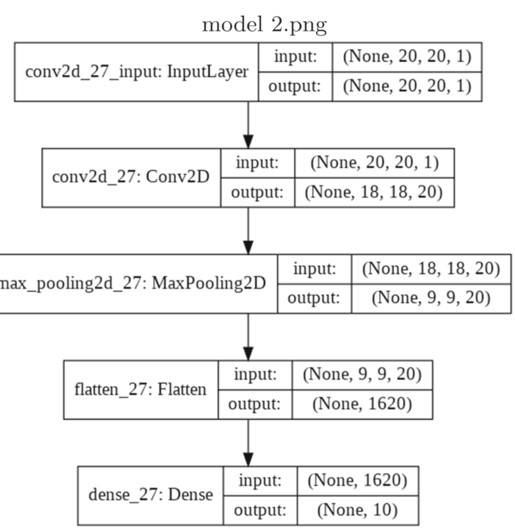

Fig. 4 Network Model of ANN using MWOA on dataset 2

Table 6 MWOA on ANN with dataset 2

S. no.	No. of hidden layer neurons	Accuracy (%)
1.	15	95.77
2.	10	96.19
3.	18	96.75
4.	14	94.76
5.	12	92.42

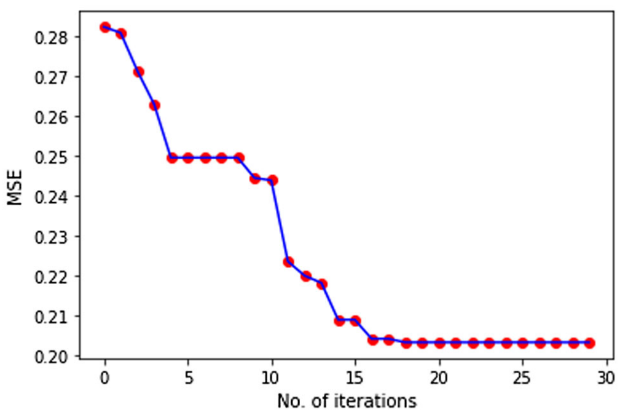

Fig. 5 MSE convergence curve on dataset 1

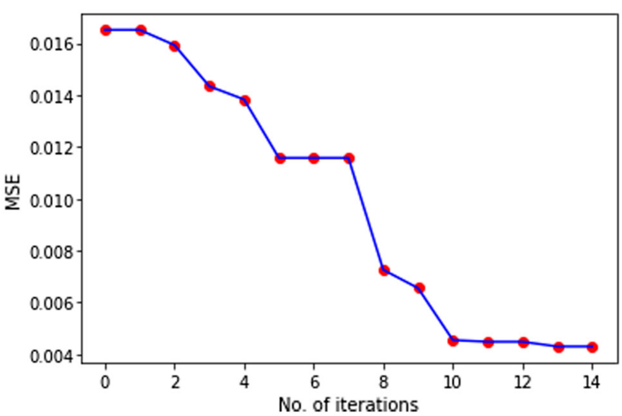

Fig. 6 MSE convergence curve on dataset 2

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

References

Abedinpourshotorban H, Shamsuddin SM, Beheshti Z, Jawawi DN (2016) Electromagnetic field optimization: a physics-inspired metaheuristic optimization algorithm. Swarm Evol Comput 26:8–22

Abualigah L (2020) Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. Neural Comput Appl 33:1–24

Aljarah I, Faris H, Mirjalili S (2018) Optimizing connection weights in neural networks using the whale optimization algorithm. Soft Comput 22(1):1–15

Baluja S (1994) Population-based incremental learning: a method for integrating genetic search based function optimization and competitive learning. School of Computer Science, Carnegie Mellon University, 1994

Basheer IA, Hajmeer M (2000) Artificial neural networks: fundamentals, computing, design, and application. J Microbiol Methods 43(1):3–31

Birjandi AK, Akhyani F, Sheikh R, Sana SS (2019) Evaluation and selecting the contractor in bidding with incomplete information using MCGDM method. Soft Comput 23(20):10569–10585

Braik M, Sheta A, Arieqat, A (2008). A comparison between GAs and PSO in training ANN to model the TE chemical process reactor. In Proceedings of the AISB symposium on swarm intelligence algorithms and applications (pp. 24–30)

Chatterjee S, Sarkar S, Hore S, Dey N, Ashour AS, Balas VE (2017) Particle swarm optimization trained neural network for structural failure prediction of multistoried RC buildings. Neural Comput Appl 28(8):2005–2016

Chou JS, Pham AD (2017) Nature-inspired metaheuristic optimization in least squares support vector regression for obtaining bridge scour information. Inf Sci 399:64–80

Crepinsek M, Liu SH, Mernik M (2013) Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput Surv (CSUR) 45(3):1–33

Dhiman G (2019) ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems. Eng Comput 37:1–31

Dorigo M, Stutzle T (2019) Ant colony optimization: overview and recent advances. Handbook of metaheuristics, pp. 311–351

Erol OK, Eksin I (2006) A new optimization method: big bang-big crunch. Adv Eng Softw 37(2):106–111

Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH (2020) Marine Predators Algorithm: a nature-inspired metaheuristic. Expert Syst Appl 152:113377

Faris H, Aljarah I, Mirjalili S (2016) Training feedforward neural networks using multi-verse optimizer for binary classification problems. Appl Intell 45(2):322–332

Formato RA (2007) Central force optimization: a new metaheuristic with applications in applied electromagnetics. Prog Electromagn Res 77:425–491

Ghasemiyeh R, Moghdani R, Sana SS (2017) A hybrid artificial neural network with metaheuristic algorithms for predicting stock price. Cybern Syst 48(4):365–392

Haseli G, Sheikh R, Sana SS (2020) Base-criterion on multi-criteria decision-making method and its applications. Int J Manag Sci Eng Manag 15(2):79–88

Jamali G, Sana SS, Moghdani R (2018) Hybrid improved cuckoo search algorithm and genetic algorithm for solving Markov-modulated demand. RAIRO-Operations Res 52(2):473–497

Kaur G, Arora S (2018) Chaotic whale optimization algorithm. J Comput Design Eng 5(3):275–284

Kaveh A, Talatahari S (2010) A novel heuristic optimization method: charged system search. Acta Mechanica 213(3):267–289

Kennedy J (2011) Encyclopedia of machine learning. Particle Swarm Optimization (pp. 760–766)

Kennedy J (2006) Swarm intelligence. Handbook of nature-inspired and innovative computing. Springer, Berlin, pp 187–219

Kim JS, Jung S (2015) Implementation of the RBF neural chip with the back-propagation algorithm for on-line learning. Appl Soft Comput 29:233–244

Kushwah R, Tapaswi S, Kumar A (2019) A detailed study on Internet connectivity schemes for mobile ad hoc network. Wireless Personal Commun 104(4):1433–1471

Lee KC, Lu PT (2020) Application of Whale Optimization Algorithm to Inverse Scattering of an Imperfect Conductor with Corners. Int J Antennas Propagation 2020:1–9

Lee JG, Senel G, Lim PK, Kim J, Hur K (2020) Octahedron sets. Ann Fuzzy Math Inform 19(3):211–238

Li S, Gong W, Yan X, Hu C, Bai D, Wang L, Gao L (2019) Parameter extraction of photovoltaic models using an improved teaching-learning-based optimization. Energy Convers Manag 186:293–305

Li H, Huang Z, Liu X, Zeng C, Zou P (2020) Multi-fidelity meta-optimization for nature inspired optimization algorithms. Appl Soft Comput 96:106619

Ling Y, Zhou Y, Luo Q (2017) Levy flight trajectory-based whale optimization algorithm for global optimization. IEEE Access 5:6168–6186

Lipowski A, Lipowska D (2012) Roulette-wheel selection via stochastic acceptance. Phys A Stat Mech Appl 391(6):2193–2196

Mafarja M, Aljarah I, Heidari AA, Faris H, Fournier-Viger P, Li X, Mirjalili S (2018) Binary dragonfly optimization for feature selection using time-varying transfer functions. Knowledge-Based Syst 161:185–204

Mareli M, Twala B (2018) An adaptive Cuckoo search algorithm for optimisation. Appl Comput Inf 14(2):107–115

Maulik U, Bandyopadhyay S (2000) Genetic algorithm-based clustering technique. Pattern Recognit 33(9):1455–1465

Mirjalili S (2019) Ant Colony Optimisation. In: Evolutionary Algorithms and Neural Networks, vol 780. pp 33–42. https://doi.org/10.1007/978-3-319-93025-1_3

Mirjalili S (2015) How effective is the Grey Wolf optimizer in training multi-layer perceptrons. Appl Intell 43(1):150–161

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

Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Advances in engineering software 69:46–61

Mohammadi-Balani A, Nayeri MD, Azar A, Taghizadeh-Yazdi M (2021) Golden eagle optimizer: a nature-inspired metaheuristic algorithm. Comput Ind Eng 152:107050

Ospina-Mateus H, Jimenez LAQ, Lopez-Valdes FJ, Garcia SB, Barrero LH, Sana SS (2021) Extraction of decision rules using genetic algorithms and simulated annealing for prediction of severity of traffic accidents by motorcyclists. J Ambient Intell Humanized comput 1–22. https://doi.org/10.1007/s12652-020-02759-5

Pandey AC, Rajpoot DS, Saraswat M (2017). Hybrid step size based cuckoo search. In 2017 Tenth International Conference on Contemporary Computing (IC3) (pp. 1–6). IEEE

Pandey AC, Rajpoot DS (2019) Spam review detection using spiral cuckoo search clustering method. Evol Intell 12(2):147–164

Pencheva T, Atanassov K, Shannon A (2009) Modelling of a roulette wheel selection operator in genetic algorithms using generalized nets. Int J Bioautomation 13(4):257–264

Rakitianskaia AS, Engelbrecht AP (2012) Training feedforward neural networks with dynamic particle swarm optimisation. Swarm Intell 6(3):233–270

Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Inf Sci 179(13):2232–2248

Sana SS, Ospina-Mateus H, Arrieta FG, Chedid JA (2019) Application of genetic algorithm to job scheduling under ergonomic constraints in manufacturing industry. J Ambient Intell Humanized Comput 10(5):2063–2090

Sanel G, Lee JG, Hur K (2020) Distance and similarity measures for octahedron sets and their application to MCGDM problems. Mathematics 8(10):1690

Sarkar BK, Sana SS, Chaudhuri K (2012) A genetic algorithm-based rule extraction system. Appl Soft Comput 12(1):238–254

Schmidhuber J (2015) Deep learning in neural networks: an overview. Neural Networks 61:85–117

Senel G (2016) A new approach to Hausdorff space theory via the soft sets. Math Problems Eng 2016:1–6. https://doi.org/10.1155/2016/2196743

Senel G (2018) The relation between soft topological space and soft ditopological space. Commun Faculty Sci Univ Ankara-series A1 Math Stat 67(2):209–219

Sharif M, Amin J, Raza M, Yasmin M, Satapathy SC (2020) An integrated design of particle swarm optimization (PSO) with fusion of features for detection of brain tumor. Pattern Recogn Letts 129:150–157

Shukla A, Pandey H M, Mehrotra D (2015) Comparative review of selection techniques in genetic algorithm. In 2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE) (pp. 515–519). IEEE

Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12(6):702–713

Storn R, Price K (1997) Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359

Takami MA, Sheikh R, Sana SS (2016) Product portfolio optimisation using teaching-learning-based optimisation algorithm: a new approach in supply chain management. Int J Syst Sci Oper Logistics 3(4):236–246

Tinkle DW, Wilbur HM, Tilley SG (1970) Evolutionary strategies in lizard reproduction. Evolution 24(1):55–74

Weimer W, Nguyen T, Le Goues C, Forrest S (2009). Automatically finding patches using genetic programming. In 31st International Conference on Software Engineering (pp. 364–374) IEEE

Wu G, Shen X, Li H, Chen H, Lin A, Suganthan PN (2018) Ensemble of differential evolution variants. Inf Sci 423:172–186

Xue Y, Jiang J, Zhao B, Ma T (2018) A self-adaptive artificial bee colony algorithm based on global best for global optimization. Soft Comput 22(9):2935–2952

Yan Z, Zhang J, Zeng J, Tang J (2021) Nature-inspired approach: an enhanced whale optimization algorithm for global optimization. Math Comput Simul 185:17–46

Yang XS (2020) Nature-inspired optimization algorithms: challenges and open problems. J Comput Sci 46:10114

Yang XS (2020) Nature-inspired optimization algorithms. Academic Press, Cambridge

Yazdani M, Jolai F (2016) Lion optimization algorithm (LOA): a nature-inspired metaheuristic algorithm. J Comput Design Eng 3(1):24–36