Review of Meta-Heuristic Optimization based Artificial Neural Networks and its Applications

D. Devikanniga1,*, K. Vetrivel2, N. Badrinath3

1Department of Computer Science and Engineering, Rajas Engineering College, Tirunelveli, Tamil Nadu, India
2Department of Information Technology, N.M.S.S.V.N College, Madurai, Tamil Nadu, India
3Department of Computer Science and Engineering, Lords Institute of Engineering & Technology, Hyderabad, Telangana, India

*Corresponding author: e-mail: mail4kanniga@gmail.com

Abstract--There are several meta-heuristic optimization algorithms developed on inspiration from nature. Artificial neural network proves to be efficient among other machine learning techniques. The efficiency of classification and prediction is improved by optimizing artificial neural network using the meta-heuristic optimization algorithms. The review of some of these hybrid artificial neural networks that are applied for benchmark datasets and to specific real-time experiments for classification and prediction are discussed. Upcoming sections cover the current trending research topics dealing with optimized artificial neural network concepts and provide some interesting insights for researchers to use in their respective applications domains of interest.

Keywords: Artificial neural network; meta-heuristic optimization algorithms; classification; prediction.

1. Introduction

Many applications are developed to cater the day to day activities in various domains. This leads to an increased demand for problem-solving techniques. There are so many standard machine learning algorithms that are developed during previous decades. These algorithms are seen to be very efficient in the areas that include pattern recognition, prediction, decision making and many others. But still there is a gap between the domain specific applications and solving algorithms. Hence these algorithms are optimized to overcome these limitations. This helps to achieve faster convergence with minimum iterations, which in turn increases the efficiency of an algorithm.

There are several traditional optimization techniques like linear programming, non-linear programming and others, which are found to be efficient and also have some drawbacks. The optimization is the process of maximizing or minimizing an objective function or fitness function for some constraints to produce an objective value or fitness value and make the system work efficiently. In recent years, many nature-inspired meta-heuristic optimization algorithms (MHOA) were developed and successfully used for optimization of machine learning algorithms like artificial neural network (ANN), extreme learning machine, deep learning machine, support vector machine and so on. Some of the MHOAs use memory to keep track of the search process and find the optimal solution based on the previous solutions stored in the memory but others are found to be memory-less. The MHOAs are mostly categorized into swarm intelligence based, physics phenomenon based, evolutionary and others. Based on the search process, these MHOAs are broadly classified as single solution based and population based algorithms. The search process in the single solution based algorithm starts with one candidate solution (search agent or object) and improves over a specific number of iterations. But in the population based, the search process starts with a set of candidate solutions which gets improved over required iterations and finally the one with good fitness is selected as the optimal solution.1,2 The population based algorithms are mainly discussed in this survey. The population based algorithms are mainly discussed in this survey. The two MHOA key components are namely exploration and exploitation. The former is the diversification process of...
searching or exploring the entire search space for new better diverse solutions while the later is the intensification process of exploiting the information found in existing better solution in the local search region, that is processed on iterations based on some conditions to produce the optimal solution. The evolutionary operators such as selection, crossover and mutation influence the process of exploration and exploitation to provide good quality of the solutions.

The MHOAs are used with ANN for feature selection, finding optimal training parameters and other problems. This study focuses on the various MHOAs that are used to find the optimum solution for training ANN and their application for classification and prediction problems. The solution in this case is the set of synaptic weights and biases for training ANN. To increase the efficiency of the algorithm and to improve the classification accuracy of ANN, several optimization algorithms are fused with it to produce the optimal training parameters. Sometimes, a few algorithms which are considered to be efficient may produce poor results for certain applications. This may be due to the mismatch of the hybrid classifier to a specific application. To avoid this type of issues, there is a need to concentrate on selecting the suitable optimization algorithm and architecture for ANN, which should work well with the chosen application for obtaining the best performance. The hybrid ANNs classifiers are discussed in the alphabetical order under the two sections based on their applications, first with standard benchmark datasets relevant to classification and prediction taken from public repository such as University of California, Irvine Machine Repository and other is using the dataset from a specified real-time applications.

The rest of the paper is organized as follows; Section 2 explains the training of ANN using traditional optimization method, Section 3 gives the population encoding in MHOA, Section 4 discusses the general ANN training using MHOA, general algorithm of hybrid classifier is given in Section 5, application of hybrid ANN to benchmark datasets and specified purposes in Section 6 and Section 7 respectively and concluded in Section 8.

2. Traditional training of ANN

The ANN is a feed forward neural network (FFNN) structure which mostly has three layers namely input layer, hidden layer and output layer. The training of ANN is the continuous optimization which is the mapping of input into the output to find the optimal set of weights and biases within minimum number of iterations. The goal is to increase the classification accuracy and decrease the classification error. The ANN performance is purely based on the synaptic weights.\textsuperscript{3,4} The back-propagation network (BPN) is the traditional learning algorithm that is gradient-based and mostly used to train the ANN. While using BPN for training, the input signals are given in the forward direction and the errors are propagated in the reverse direction with no any loop in the network. The errors are the calculated as the difference between target output and network output. The BPN training causes the algorithm to get trapped in local minimum and results in slow convergence rates.\textsuperscript{5,6} In many cases the MHOAs are used together with BPN to overcome these limitations. The structure of ANN is given in Fig. 1.

![Fig. 1: Architecture of artificial neural network.](image)

3. Population encoding in MHOA
In every population based algorithm, the maximum population (NP) is initialized. The type of population depends on the optimization algorithm selected. Each member or search agent (P) of the population is considered as an F-dimensional vector and F calculated as, \( F = (N+M)+(M+R)+M+R \), where N, M and R are the number of nodes in input layer, hidden layer and output layer. The vector consists of the synaptic weights and biases associated with the connections of the network. \( P = \{ W_{1_{MN}}, B_{1_{M}}, W_{2_{MN}}, B_{2_{M}}, \ldots \} \), where \( W_{1_{MN}} \) and \( W_{2_{MN}} \) are weights between input hidden layer and with hidden to output layer. \( B_{1_{M}} \) and \( B_{2_{M}} \) are the biases of the hidden and output layer respectively.

The good optimization algorithm should maintain a proper trade-off between exploration and exploitation while maintaining its efficient search behavior to find “global” most optimal solution. This also helps the algorithm to avoid getting entangled in local optimum and premature convergence.

4. General training procedure using MHOA

The training of the ANN with any MHOA starts with the initialization of ANN with random weights and biases given by the MHOA used. The training input samples from the classification or prediction dataset are fed into the network. The ANN produces the output which is compared with the target output. This error value is considered as the fitness value using error functions such as mean square error (MSE), sum of squared error (SSE), root mean square error (RMSE) and others. The minimization of the objective function is performed to produce the minimum fitness value here. The MHOA is executed with the random values to produce the next set of weights and biases for the next iteration. Each candidate of the population is accessed by the objective function using the input variables. This process takes place till the stopping criteria are satisfied. Finally the solution with the best fitness value (minimum error) from all the iterations is considered as an optimal solution. This is used for the classification or prediction of any future unseen data. The general overall block diagram of classification or prediction process using optimized ANN is given in Fig. 2.

![Fig. 2: General overall block diagram of classification/prediction using optimized ANN.](image)

5. The general algorithm of MHOA optimized ANN

The algorithm that shows the general form of optimization of ANN using MHOA for classification and prediction is given below:

Begin

Initialize ANN with the random solution (training parameters) from meta-heuristic optimization algorithm;

Load the training data;

While (stopping criteria not reached)

For i = 1 to the length of training data
Compute the output from hidden layer;
Compute the actual network output;
Calculate the error;
End for
Calculate the fitness value using the objective function;
Update the weights and biases in the forward path;
Propagate the network error signal in the reverse direction;
The optimization algorithm calculates the best solution and train
ANN at each epoch till the network convergence takes place;
End while
Output the optimal set weights and biases after training;
Load the testing data;
Classify / predict the test data using the optimal parameters;
Evaluate both training and testing performance;
End

6. Application of optimized ANN classifier for the benchmark datasets

The hybrid classifiers are developed and applied to benchmark classification and prediction datasets
to prove their efficiency. Some of these applications in the literature are discussed in this section
along with a brief introduction to the MHOAs used.

A. Ant colony optimization based ANN

In ant colony optimization\(^7\) (ACO), the search agents are called as ants that mimic the foraging
behavior of the real ants. This comes under swarm intelligence. A colony of ants is maintained and
considered to be the population in this algorithm. Each ant searches food near its nest in a random
manner. It evaluates the quality and quantity of the food that is found. Based on this evaluation, it
deposits pheromone when it returns to its nest with the food. This pheromone trails provides indirect
communication with other ants to find the shortest path between its nest and food source. The value
associated with each path is taken in combination to evaluate the objective function. Subsequently,
the optimal value is found after required iterations with newly formed ant colonies. A different
objective function called square error percentage is used.\(^8\) An ACO algorithm is used\(^9\)
for feature subset selection which helps in removing the input attributes that produces noise. A
variant of ACO for continuous optimization is incorporated with classical gradient based methods to
train.

B. Artificial bee colony optimization based ANN

The artificial bee colony\(^10\) (ABC) algorithm is developed based on the intelligent foraging behavior
of the natural bee colony. The population of ABC consists of three types of artificial bees, namely
the employed bees which occurs half of the bee population that share the information of their food
source at the dancing area by a waggle dance, second is the onlooker bees that wait in the hive for
the food source information from employed bees and third one is the scout which search for the new
food source. Each employed bees are associated with one food source and it becomes scout bee once
its food source gets exhausted. The high quality food source selected by artificial bees represents the
good optimized solution to the network.
ABC is used to train ANN for many applications and a few are discussed.\textsuperscript{11,12} The bee colonies are used to minimize the number of connections on evolving the synaptic weights, its architecture and transfer function of each neuron. The MSE and classification error rate (CER) are used for testing the network. The dimensionality reduction is also achieved.\textsuperscript{12} An ABC based BPN method is used to train FFNN\textsuperscript{13} and performance is analyzed based on SSE, convergence speed and stability of optimum solution.

C. \textit{Bacterial foraging optimization algorithm based ANN}

The bacterial foraging optimization\textsuperscript{14} algorithm (BFOA) describes the chemotactic behavior of the \textit{Escherichia coli} bacteria that live in our intestines. The group of bacteria forms the population and each bacterium acts as the search agent. This foraging process is dictated by its control system using the steps such as chemotaxis, swarming, reproduction, elimination and disposal. The member of a bacterium which gives an optimal error during training is considered as optimum solution.\textsuperscript{15} The BFOA is applied for training FFNN and cascade FF ANNs.\textsuperscript{16} The position of the bacterium with highest fitness is considered as the set of optimum values and discussed in terms of accuracy.

D. \textit{Bat algorithm based ANN}

The bat inspired\textsuperscript{17} (BI) algorithm is based on echolocation behavior of micro bats while sensing distances. The colony of bats constitutes the population and each bat forms the candidate solution. BI algorithm is combined with back propagation algorithm to train the ANN.\textsuperscript{18} The position of the food source represents the solution. To improve the cooperation between subpopulations, the different cooperative strategies such as Ring Master-Slave and coevolving Master-Slave strategies are applied to the bat algorithm for better balance between exploration and exploitation. After this, bat algorithm is applied to optimize the ANN structure in terms of feature selection, number of hidden layers, number of nodes in the hidden layers, and weight of hidden layers. The chaotic map is used instead of random numbers.\textsuperscript{19}

E. \textit{Biogeography-based optimization - ANN}

The biogeography-based optimization\textsuperscript{20} (BBO) is an evolutionary algorithm based on the distribution of biological organisms geographically. Many habitats comprise the population for algorithm. The habitats are the search agents and these evolve on migration and mutation process that depends on habitat suitability index (HSI). The habitants in high HSI habitats emigrate to low HSI habitats and in turn, low HSI habitats have high immigration rate. The HSI of all habitats are improved by BBO algorithm. Each habitat is a MLP with own emigration, immigration and mutual rates. The number of habitants (the weights and biases) in a habitat is decided based on the problem type. The HSI is the fitness function which is calculated as the MSE of all training samples for all the habitats. Consequently, habitats are combined depending on emigration and immigration rates. With this, habitats are mutated based on mutation rate. The best habitat with good HSI is saved for the next generation.\textsuperscript{21}

F. \textit{Bird mating optimizer based ANN}

The bird mating optimizer\textsuperscript{22} (BMO) is inspired by the mating behavior of the birds. The population of the birds is called society which consists of four kinds of birds namely monogamous, polygynous, polyandrous and promiscuous. Each bird is the search agent that has the predefined number of genes. The birds take part at the mating process during the breeding season to brood with superior genes. The quality of each brood is calculated using the objective function or fitness function which
specifies the mating strategy. Based on the quality, the birds are sorted and divided into four categories. Then the mating process is carried out. The quality of the brood is compared with that of the bird and if better, the brood will replace the bird.

The BMO is used to find the weights for training ANN. Each bird is a solution to the problem. The number of genes in a bird is based on the problem variables and decides the d-dimension in which breeding happens. The quality of bird depends on its gene and if better than its brood, it causes brood to be abandoned from society. If otherwise, the contrary is followed. This is repeated for maximum generations. Finally the brood or bird with the good quality of genes is kept as the optimal solution of weights for training ANN.

G. Chemical reaction optimization based ANN

The chemical reaction optimization\textsuperscript{24} (CRO) is based on the microscopically viewed phenomenon of the chemical reactions that takes place between the molecules to form a product. A mole of molecules form the population. A molecule is considered as the search agent containing a group of atom with a characteristic structure based on the problem. A molecule is said to change based on two types of energies namely, potential energy (PE) and kinetic energy (KE). The translation between the PE and KE takes place due to type of collisions happened due to different elementary reactions such as on-wall ineffective collision, decomposition, inter-molecular ineffective collision and synthesis. The goal of algorithm is to locate the point with lowest PE while exploring different parts of PE surface while applying searching approaches, such as intensification and diversification. This is implemented using three separated stages as initialization, iteration and final stage. The structure with lowest PE is considered as the best solution in each run of the whole reaction.

CRO is used to optimize the network structure and weight adaptation.\textsuperscript{25} Each molecule is the solution representing network structure and the molecule’s current energy state. PE is concerned with fitness function value and KE represents the ability of the molecule from not struck into local minima. The final optimal molecule structure is considered as an optimal solution.

H. Cuckoo search via Levy flight based ANN

The cuckoo search\textsuperscript{26} (CS) optimization based on the brood parasitism, a breeding behavior of some cuckoo species and its searching strategy using levy flight characteristics. The asylum of cuckoos forms the population and each cuckoo is considered as the search agent. It is described using three rules as, (i) each cuckoo lays one egg in a randomly selected nest, (ii) the nest with high quality of eggs are kept for next generations, and (iii) the number of nests is kept fixed and the host bird identifies the cuckoo egg with the probability, \( P_a \in [0,1] \).

Once the host bird discovers the eggs, it may either throw the egg or abandon the nest. The number of eggs in a nest is taken based on the problem and its complexity. Each egg in a nest represents a solution and new solution is the cuckoo egg at that iteration. The CS algorithm is improved by varying the significant parameters namely, probability \( (P_a) \) and step size \( (\alpha) \), instead of keeping them as fixed values. The SSE is considered as the objective function used in this approach.\textsuperscript{27} CS algorithm is used to train BPN for faster convergence and local minima avoidance.\textsuperscript{28} Interestingly, CS algorithm is combined with Levenberg Marquardt algorithm and BPN to get even better performance.\textsuperscript{29} In these, each best nest represents a possible solution.

I. Firefly optimization based ANN

The firefly optimization algorithm\textsuperscript{30,31} (FA) is based on the characteristics of flashing light of the fireflies produced by bioluminescence process to attract the mating partners and the potential prey. The swarm of fireflies constitutes the population where each firefly is the search agent. The light intensity decreases as the distance between the fireflies increases. The flashing light is associated with the objective function. The three idealized rules describing the FA is given as, (i) all fireflies are attracted towards each other as they are considered to be unisex, (ii) attractiveness is proportional to
their brightness and the movement of one firefly towards other in the d-dimensional space based on this, and (iii) the brightness of a firefly is affected by the landscape of the objective function. The new solution is evaluated and the light intensity is updated at each time to find the attractiveness. The firefly that yields the best solution is considered to be the most optimal. The FA is used along with BPN to optimize its performance index. The SSE is used to calculate the performance index. The most attractive firefly is the one with good light intensity and considered to be optimal. The FA is used to train FFNN and evaluated with two transfer functions. MSE is used as the error function. The FA is used along with ANN to solve the time series classification problem and evaluated based on error rate.

J. Genetic algorithm based ANN

John Henry Holland and his team developed genetic algorithm (GA) based on the Charles Darwin’s “Theory on natural selection and genetics”. The simulation based on the survival of the fittest among the population over maximum generations is performed to solve the problem. A required number of individuals or chromosomes constitute the population. Each chromosome is the candidate solution which consists of a specific number of genes based on the problem and it is associated with a fitness value. The algorithm evolves over with three basic operations based on reproduction strategy such as selection, crossover and mutation. The parents are selected based on their fitness values for mating. After some fixed number of generations, the best chromosome or the offspring with good fitness is used as the optimal solution for solving the problem. The ANN is used in combination with GA (GA-ANN) in many works in the past. A few were done on classification task using GA-ANN. The GA is used to train a neural network with the best chromosome that has good fitness value and algorithm efficiency is proved on comparing its result with that of the MLP trained with traditional BPN algorithm. The GA is hybridized with BPN and LM as GABP and GALM and it is used for classification. The squared error percentage (SEP) and classification error percentage (CEP) are used to judge algorithm efficiency with that of ANN trained using traditional BPN and LM.

K. Gravitational search algorithm based ANN

The gravitational search algorithm (GSA) is another heuristic algorithm based on the Newton’s law of gravity and law of motion. The set of agents associated with masses constitute the entire population in GSA. The four parameters for a search agent are position, inertial mass, active gravitational mass and passive gravitational mass. The objects interact with each other by gravity force which causes global movement of all objects towards heavier mass objects based on its fitness values governed by the two laws. Heavier objects correspond to optimal solution. Two modifications were made in GSA for the faster convergence towards global optima. First, the random coefficient is replaced by adaptive weight and the same is multiplied with acceleration term. Second, a dynamic weight is introduced. The GSA is executed with the above modifications and the mass with good fitness is the optimal solution for ANN training process. The minimization of MSE is the objective function used.

L. Grey wolf optimizer based ANN

The grey wolf optimizer (GWO) is based on the leadership hierarchy and hunting behavior of grey wolves. The former is simulated using four types of grey wolves (search agents) that include alpha, beta, delta and omega. After the initialization of the wolf pack which constitutes the population, the fitness values are evaluated according to the objective function. Based on this, the first three best fitness valued wolves are considered in order as alpha, beta and delta, while the remaining of the wolf pack are kept as omega, whose positions are updated based on the first three in order to attack...
the prey. The three main steps in hunting the prey are searching, encircling and attacking the prey. Finally, the position of alpha is taken as the best optimal solution. The GWO is used to train the MLP with optimum parameters. The objective function is the average MSE to calculate the fitness of the wolves.

M. Harmony search based ANN

The Harmony search algorithm (HSA) mimics the artificial phenomenon of improvisation process of musical players' performances, which is searching for better harmony. A set of sounds played by joined instruments is kept as harmony memory (HM) during the practice (iteration). Aesthetic estimation (objective function) is determined from it based on the pitches of the instruments (variables). After each practice, if the New Harmony is found to be better than the minimum harmony in HM, minimum harmony is excluded, while including the New Harmony. This is done for the required number of practices till the fantastic harmony (global optimum) is achieved.

The HSA is improved to achieve better fine-tuning characteristics. The optimization to achieve improved HSA is converting static parameters such as pitch adjustment rate (PAR) and bandwidth (BW) to dynamic variables for each iteration (generation). This helps to achieve faster convergence towards best optimum. The best-to-worst (BtW) harmony ratio is introduced and the parameters such as PAR and BW are dynamically changed based on BtW of current HM. Additionally, this helps to determine the termination condition after attaining the expected quality. This modified improved HSA is used to train ANN. The harmony vector in the HM gives the solution (weights and biases). Minimization of SSE is the objective function used. The self-adaptive global best harmony search (SGHS) algorithm introduces new improvisation scheme and adaptive parameter tuning methods. The HM considering rate (HMCR) and PAR are chosen to be self-adaptive parameters. The BW is modified as the dynamic parameter. The best harmony vector from SGHS is used to train NN.

N. Invasive weed optimization based ANN

Invasive weed optimization (IWO) is the optimization algorithm inspired from colonization of invasive weeds. The process of weed colonization starts with the dispersal of finite number of seeds as population initialization. Every seed grows and reproduces to give seeds based on their fitness. Again these seeds are dispersed and grown into new plants (spatial dispersion). This process continues for the maximum number of plants. Only the plants with the lower fitness value survive, while the others are eliminated (competitive exclusion). Finally, the plant with the best fitness is considered to be the optimal solution. The IWO is modified on reducing the standard deviation of spatial dispersion for a weed when its objective function reaches minimum in that iteration. This makes seed dispersal within small neighborhood. The standard deviation is also varied based on its objective function. This makes a weed close proximity of optima on increasing its exploration and causes the filtering of the weeds with maximum objective function value. The objective function used in this pattern recognition process is CEP. The weed with best fitness value based on CEP is used to train the ANN.

O. Krill herd algorithm based ANN

The krill herd algorithm (KHA) is based on the krill herding behavior. The entire krill herd is considered as the population. The best solution is the position of a krill near the best food source and at high density herd which is the minimum of the objective function. The movement towards this position is given by three factors such as, induced motion, foraging motion and physical diffusion. Then the crossover and mutation operators are applied to enhance its performance. The position of each krill is updated after individual iteration. The final best position of a krill after maximum iterations is considered as the optimum solution. The global best solution based act as the set of parameters to train ANN. The fitness function is considered as minimum MSE and minimum SSE.
P. Moth-flame optimization algorithm based ANN

The moth-flame optimization\(^{53}\) (MFO) algorithm is based on the navigation property of the moths which forms the population. Each moth is considered to be the search agent. At night, the moths fly towards the moon (flame) in an angle on a straight line for long distances by a property called transverse orientation. They fly in a spiral motion towards artificial lights as the distance will be short in this cases when compared to that of moon. The moths are the search agents in this algorithm, and they constitute the whole population. The moths are candidate solutions and their positions determine the fitness values. Moth positions are updated when better solutions are found based on the flame with a better fitness value in each iteration. The moth at the best position is considered as the optimal solution on termination. The MFO algorithm is used to train the MLP with the best moth.\(^{54}\) The minimum average MSE is the objective function used to find the optimal solution.

Q. Multi-verse optimizer based ANN

The multi-verse optimizer\(^{55}\) (MVO) is inspired by the three cosmology concepts such as white hole, black hole and worm hole, based on the multi-verse theory in physics. According to this theory, many other universes exist and they also interact with each other. They collide with each other at white holes, while the black holes extract everything and this assists the exploration. The worm holes are the tunnels where an object from one universe travels to the other best one as it supports exploitation. The set of universes constitute the population and each universe is the candidate solution. The objects or variables move from one universe to other based on the inflation rate. At each time (iteration), the overall inflation rate is improved. The best universe is evaluated based on its objects status. The MVO is used to train the ANN with the evaluated best universe which lends parameters for its training.\(^{56}\)

R. Particle swarm optimization based ANN

The particle swarm optimization\(^{57}\) (PSO) is based on flocking behavior of the birds. The search agents are the flock of birds (also called particles) which makes the population of the algorithm. The flock of birds that search for food maintain the distance between them during their flight to avoid collision. The information about the food source is passed between the birds in the flock. Each bird remembers the best value (pbest) and that position (pbestx and pbesty). They adjust their X velocity (adjust with negative random weight if the movement is to the right of pbestx, else if it is to the left, then increment with random weight) and Y velocity (adjusted up and down for moving above and below pbesty) based on their positions during sudden movements and change of directions to maintain the flock movement. Each bird also remembers the global best position of one bird which found best value (gbest). Finally after maximum iterations designed, the flock of birds reaches the food source at the global best position on swirling movement. The bird with gbest is the referred as the best optimal solution. The ANN is also trained with the optimal parameters obtained from PSO.

The improved opposition based particle swarm optimization algorithm is used along with BPN to train the ANN for prediction.\(^{58}\) The opposition based learning\(^{59}\) is the scheme of learning to search in opposite manner or direction. The particle with worst fitness value is replaced with its opposite particles in the opposition based particle swarm optimization algorithm.\(^{58}\) An improved PSO method is the centripetal accelerated particle swarm optimization\(^{60}\) (CAPSO) which is developed on combining Newton’s laws of motion with PSO. This improves the learning and convergence speed of algorithm. The additional parameters are added to the program based on Newton’s laws of motion in mechanics. The CASPO is fused with ANN for diagnosis of the medical disease.\(^{61}\)
S. Social spider optimization based ANN

The social spider optimization\(^{62}\) (SSO) is the algorithm devised based on the set of complex cooperative behavior of the male and female spider (both are search agents here) and they constitute the population of colony. The female population is predominant compared to males. Their tasks depend on their gender and they include predation, mating, web design and social interaction. The entire search space is treated as a communal web. Each spider position in the search space represents a solution. The information regarding the trapped preys and mating details are transmitted through the web via vibrations of spiders. These vibrations depend on its weight and distance. Each spider acquires a weight depending on a fitness value. Based on this, the cooperative operators such as female cooperative operator, male cooperative operator and mating operator are executed. After mating, the offspring weight is compared with that of other spiders in the population. If offspring has better value, it is replaced with a worst spider, else the population remains unchanged. The best spider with the best fitness at the end of all iterations is considered to be optimal. The best spider from the SSO algorithm is chosen as the optimal parameters to train the ANN for classification.\(^{63}\) Mainly it is applied for Parkinson’s disease identification. The objective function used to find the fitness value of the spider is the minimum MSE function.

T. Tabu search based ANN

In the tabu search\(^{64}\) (TS), the solution adjacent to the current solution is constructed and is referred as neighborhood. The subset of moves in a neighborhood is classified as tabu or forbidden (already visited or past solution), which is maintained as tabu list. The overriding condition called as aspiration criteria is updated in each iteration, along with tabu list. The iteration continuous till stopping criteria is satisfied and the final best solution is retrieved. The extended TS and the preliminary TS is used for optimizing ANN.\(^{65}\) In extended TS is the improvement of TS with the modifications such as variable tabu list size, systematic exploration and exploitation for creating neighborhoods, and modified termination conditions. It is given that the optimal values were obtained within few function evaluations.

U. Wolf search algorithm based ANN

Wolf search algorithm\(^{66}\) (WSA) is based on the preying behavior of the wolf and its defense mechanism against its enemies. Each wolf is considered as a search agent and the pack of wolf constitutes the population. The algorithm follows three rules to find the best solution.

1. Each wolf senses its companion and the step distance with its fixed visual area defined by a radius.
2. The quality of the wolf’s current position is found using the objective function. The wolf’s decision regarding the movement is calculated.
3. The wolf moves beyond its visual range when its senses the enemy.

The WSA is used to produce the weight values for BPN to train ANN.\(^{67}\) Each wolf searches the best solution in multiple directions and unites with peer on a better location. The weights and biases are calculated and compared with best solution in the backward direction. This is repeated till the minimum MSE or the stopping condition is reached.

7. The application of ANN hybrids for specific real-world applications

The hybrid classifiers are modeled and developed particularly for specialized applications. These applications are discussed in this section.
A. **ACO based ANN**

The ACO\(^7\) algorithm used to train ANN for evaluating the performance of residential buildings based on the aspects namely, application performance, safety performance, durability, environmental performance and economic performance.\(^6\) 12 sets of sample data were given as an input to the network for this prediction process. The results for this prediction problem were said to be promising.

B. **ABC based ANN**

The ABC\(^10\) algorithm is used to train ANN (ABC-ANN) for S-system models.\(^6\) The biochemical profiles carry information related with the network structure and its system dynamics. It also helps in the retrieval of information such as identification of disease state patterns, drug toxicities etc. The S-systems are the non-linear dynamic systems which help to model complex biological networks. It is reported that the gene expressions are approximated using the ABC-ANN efficiently.

C. **BBO based ANN**

The BBO algorithm is used to optimize the FFNN with the training parameters and applies for the fruit classification application.\(^7\) For classification, 1653 chromatic fruit images of 18 categories were given as an input. The images undergo four-step pre-processing, feature extraction, principal component analysis and finally classified using BBO-FNN classifier. It is monitored that BBO-FNN gave a superior performance than the others. The next application of BBO trained MLP is for the identification of spam e-mail.\(^7\) Here BBO-MLP model acts as a filter that extracts the knowledge regarding the spam e-mails on training. With this knowledge acquired on training, the model identifies spam e-mails from ham e-mails with greatest accuracy relative to other classifiers.

D. **CRO based ANN**

The ANN model optimized with CRO algorithm (CROANN) is used to forecast the time series financial data (stock market indices) from non-linear system processes.\(^7\) Seven different historical stock market indices of 15 years data are used for training. The ACRNN accurately predicted the stock market data.

E. **CS based ANN**

The radial basis function (RBF) neural network is optimized with CS.\(^7\) This network model is used to predict another time-dependent application that is forecasting the level of flood. This prediction model used the real time data from Little Wabash River as input for training.

F. **Dragonfly algorithm based ANN**

The dragonfly algorithm\(^7\) (DA) is developed based on the two swarming behavior of dragonfly namely static and dynamic. The swarm of dragonflies forms the population in which each dragonfly is the search agent. The dragonflies are attracted towards food sources and distracted from enemies. Based on these behaviors, the dragonflies update their position with five main factors that includes separation, alignment, cohesion, food and enemy. The static swarm deals with first two factors and the remaining factors with dynamic swarm. The trade-off between exploration and exploitation is maintained by tuning on these factors on optimization. The best position of a dragonfly gives the optimal solution. The ANN trained with global best parameter (low MSE) derived from DA is used to forecast the prime fuel demands in India.\(^7\) This prediction model forecasts demands in prime fuels
of coal, lignite, crude oil and natural gas, till 2025. The historical data of the prime fuels between the year 1980 and 2012 are taken for prediction.

G. Firefly algorithm based ANN
The firefly algorithm is integrated with the ANN models like functional link ANN and radial basis function network. These hybrids are used for software cost prediction in spite of varied complexities. The accuracy is shown to have improved on using the datasets taken from the PROMISE software repository.

H. GA based ANN
The free-product such as oil spreads over the water table when the light hydrocarbon leaks and it is recovered. To proceed with this process, a complete set of input parameters are needed. This can be done by parameter estimation tools such as inverse modeling. The ANN optimized with GA (ANN-GA) is used to solve this problem. The real time experiments were conducted and the data obtained are used. It is said that the problems based on error minimization are solved by this ANN-GA efficiently. Another application of the ANN optimized GA is the prediction of the preliminary cost of the construction project. The historical data on the residential building project collected from the general contractors in Korea between the year 1997 and 2000 are used for training this prediction model. It is found that ANN-GA produced accurate results.

I. HSA based ANN
The quality of X20G13 type stainless steel depends upon its surface roughness. The ANN integrated with HSA is used to predict the surface roughness in face milling. Several experiments were conducted to derive the surface roughness value. The optimum cutting parameters which lead to the minimum surface roughness values were determined accurately by this prediction model. The armor stones protect the rubble mould breakwater from severe erosion caused by wave attacks. The stability number of these armor stones is predicted by HS optimized ANN. The input used in this prediction model is the experimental data of Van der Meer.

J. Imperialistic competitive algorithm based ANN
Imperialistic competitive algorithm (ICA) is based on the logic behind imperialistic competition among the empires. The population of this algorithm is the set of countries which is of two types, kept as imperialist and colony. The algorithm starts with initial population (countries). Based on the power of imperialist, the colonies are divided among them and form the empires. Bigger empires have more colonies and vice versa. The colonies are moved towards their relevant imperialist and if the cost (fitness) of a colony is found less than its imperialist, this is identified as best colony and its position is exchanged with its imperialist. The total cost of all empires are calculated, the colony of weaker empire is given to the stronger empire. Finally, the weaker empire or the empire with no colonies is collapsed. This algorithm is iterated until one empire exists. Now the convergence is attained and the algorithm terminates.

The ICA is used to optimize the ANN for the soil compaction indices prediction. The soil compaction is the penetration resistance and soil sinkage. The FFNN and cascade –forward network are used. The dataset is collected from the soil compaction experiments that are conducted out in a soil bin facility. The performance functions used are MSEREG (MSE added with mean of square of weights and biases) and coefficient of determination. Another application of ICA optimized ANN is applied for predicting the adsorption of sunset yellow dye by zinc-oxide nano particles – activated carbon in aqueous solution which helps in the treatment of wastewater. The MSE and the coefficient of determination are used as performance functions.
K. Monarch butterfly optimization based ANN

The monarch butterfly optimization (MBO) is based on the migration behavior of the monarch butterflies from one land to another during specific seasons. Swarm of monarch butterflies constitutes the population and each butterfly acts as search agent. The entire population is divided into two subpopulations. The migration operator and butterfly adjusting operator are the two operators used in the algorithm for position update of the monarch butterflies. This is carried out till the best solution is found. This MBO algorithm is hybridized with the ANN for the classification of the osteoporosis disease. It is also reported that this hybrid classifier performance is good.

L. PSO based ANN

The PSO algorithm is modified as the chaotic self-adaptive PSO algorithm (CSPSO) to achieve better performance, to accelerate the convergence speed and avoid premature convergence. In CSPSO, the self-adaptive inertia weight factor is introduced and the acceleration coefficients are adapted by chaotic sequences generated based on chaos theory. CSPSO is used along with BP algorithm for training the ANN and used for the prediction of gas solubility in polymers with varied pressure and temperature settings. It is proven that this prediction model have shown good prediction capability and accuracy.

M. Simulated annealing based ANN

Simulated annealing (SA) is based on the heating and cooling a material under controlled environment to attain a required crystalline structure. Like the other algorithms, it performs a random search and accepts better solution (decreased fitness value) or even the worst solution at first (at very high temperature). It improves over the iterations based on probability and finally the optimal solution is produced (on gradual cooling). SA is used long with ANN in the response surface methodology.

N. Teaching-learning based optimization based ANN

Teaching-learning based optimization algorithm (TLOA) is based on the teaching-learning process in which the results of the learners are based on the quality of teacher’s knowledge and capability of the learners. It is accessed in two phases namely teacher phase in which the learners learn from the teacher and the learning phase where the learners share their knowledge. A group of students constitute the population of this algorithm and each student is treated as a candidate solution. Every time the student with minimum fitness value (more knowledge) is chosen as teacher. This is carried out till all the students get equal knowledge. The two phases are alternatively performed till the stopping criterion is reached. The student with more knowledge is considered as the optimal solution. The TLOA is used with ANN to design the prediction model for determining energy consumption in Turkey. The objective function is the minimum of MSE. The relative error, mean absolute error and RMSE are used to evaluate the algorithm performance. It is reported that TLOA-ANN gives good estimation than others.

8. Conclusion

A good trade-off should be maintained to obtain the good quality solutions. Some MHOAs are found to be good in exploration and others in exploitation during the searching process. The exploration helps for global optimum while exploitation for local optimum. To maintain a good trade-off, the algorithms should be selected and combined to form hybrid optimization techniques. The algorithms chosen to form hybrid approaches should be in the category as one good with...
exploration rate and the other with good exploitation rate. Thus, details of some hybrid classifiers developed by fusing ANN with different MHOAs in the literatures are given in brief. The survey is done on their application to the benchmark datasets and other real-time experiments. Only a few were discussed and this will help researchers to get some light on applying MHOAs to optimize ANN. Further this provides ways to find suitable MHOA for a specific application. Many new MHOAs that are not used with ANN can be found and may be experimented. This shall give better results than the existing ones. Still many experiments are ongoing with the optimized ANN in different domains to determine solutions for day to day issues.

9. References

[1]. K. L. Du and M. N. S. Swamy, Search and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature, Springer, New York (2016) p. 434.

[2]. S. Koziel and A. Bekasiewicz, Multi-Objective Design of Antennas Using Surrogate Models, World Scientific, New Jersey (2016).

[3]. T. L. Fine, Feedforward Neural Network Methodology, Springer Verlag, New York (1999).

[4]. D. Graupe, Principles of Artificial Neural Networks, World Scientific Publishing, New Jersey (2007).

[5]. P. J Werbos, Backpropagation through time: what it does and how to do it, Proc. of IEEE, Vol. 78, (1990), pp. 1550-1560.

[6]. M. Gori and A. Tesi, On the problem of local minima in backpropagation, IEEE Trans. Pattern Anal. Mach. Intell., 14(1) (1992) 76-86, doi: 10.1109/34.107014.

[7]. M. Dorigo, V. Maniezzo and A. Colorni, Ant system: Optimization by a colony of cooperating agents, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 26(1) (1996) 29-41, doi: 10.1109/3477.484436.

[8]. K. Socha and C. Blum, An ant colony optimization algorithm for continuous optimization: application to feed-forward neural network training, Neural Computing and Applications, 16(3) (2007) 235-247, doi:10.1007/s00521-007-0084-z.

[9]. R. K. Sivagaminathan and S. Ramakrishnan, A hybrid approach for feature subset selection using neural networks and ant colony optimization, Expert Systems with Applications, 33(1) (2007) 49-60, doi: 10.1016/j.eswa.2006.04.010.

[10]. D. Karaboga and B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Glob. Optim. 39(3) (2007) 459-471, doi:10.1007/s10898-007-9149-x.

[11]. K. Dervis and O. Celal, Neural networks training by artificial bee colony algorithm on pattern classification, Neural Network World. Institute of Computer Science Czech Academy of Sciences, 19(3) (2009) 279-292.

[12]. B. A. Garro, H. Sossa and R. A. Vázquez, Artificial neural network synthesis by means of artificial bee colony (ABC) algorithm, IEEE Congress on Evolutionary Computation (CEC), New Orleans, LA, USA (2011) pp 331-338.

[13]. S. Nandy, P. P. Sarkar and A. Das, Training a feed-forward neural network with artificial bee colony based backpropagation method, Int. J. of Computer Science & Information Technology (IJCIST) 4(4) (2012) 33-46, doi : 10.5121/ijcist.2012.4404.

[14]. K. M. Passino, Biomimicry of bacterial foraging for distributed optimization and control, IEEE Control Systems Magazine 22(3) (2002) 52-67, doi:10.1109/MCS.2002.1004010.

[15]. I. A. A. AL-Hadi and S. Z. M. Hashim, Bacterial foraging optimization algorithm for neural network learning enhancement, Int. J. of Innovative Computing, 1(1) (2012) 8-14.
[16]. R. Kaur and B. Kaur, Artificial neural network learning enhancement using bacterial foraging optimization algorithm, *International Journal of Computer Applications* 102(10) (2014) 27-33, doi:10.5120/17852-8812.

[17]. X.-S. Yang, A new metaheuristic bat-inspired algorithm, in *Nature Inspired Cooperative Strategies for Optimization (NICOSSO 2010)* eds. J. R. Gonzalez, D. A. Pelta, C. Cruz, G. Terrazas and N. Krasnogor, Springer, Berlin, Heidelberg (2010) pp. 65-74.

[18]. N. M. Nawi, M. Z. Rehman and A. Khan, A new bat based back-propagation (BAT-BP) algorithm, in *Advances in Systems Science, Advances in Intelligent Systems and Computing*, eds. J. Swiatek, A. Grzech, P. Swiatek and J. Tomczak, Springer, Cham (2014), pp. 395-404.

[19]. N. S. Jaddi, S. Abdullah and A. R. Hamdan, Multi-population cooperative bat algorithm-based optimization of artificial neural network model, *J. Inform. Sci.* 294 (2014) 628-644, doi: 10.1016/j.ins.2014.08.050.

[20]. D. Simon, Biogeography-based optimization, *IEEE Trans. Evol. Comput.* 12 (2008) 702–713, doi: 10.1109/TEVC.2008.919004.

[21]. S. Mirjalili, S. M. Mirjalili and A. Lewis, Let a biogeography-based optimizer train your Multi-Layer Perceptron, *Information Sciences*, 269(2014) 188-209, doi: 10.1016/j.ins.2014.01.038.

[22]. A. Askarzadeh and A. Rezazadeh, A new heuristic optimization algorithm for modeling of proton exchange membrane fuel cell: bird mating optimizer, *Int. J. Energy Res.*, 37(2013) 1196-1204, doi: 10.1002/er.2915.

[23]. A. Askarzadeh and A. Rezazadeh, Artificial neural network training using a new efficient optimization algorithm, *Applied Soft Computing*, 13(2013) 1206-1213, doi:10.1016/j.asoc.2012.10.023.

[24]. A.Y.S. Lam and V.O.K. Li, Chemical-reaction-inspired metaheuristic for optimization, *IEEE Trans. Evol. Comput.*, 14(3) (2010) 381-399, doi: 10.1109/TEVC.2009.2033580.

[25]. J. J. Q. Yu, A. Y. S. Lam and V. O. K. Li, Evolutionary artificial neural network based on chemical reaction optimization, *Proc. IEEE Congress of Evolutionary Computation (CEC)*, (New Orleans, LA, USA, 2011) pp. 2083-2090.

[26]. X. S. Yang and S. Deb, Cuckoo search via Lévy flights, *Proc. World Congress on Nature & Biologically Inspired Computing*, Coimbatore, India (2009) pp. 210—214.

[27]. E. Valian, S. Mohanna and S. Tavakoli, Improved cuckoo search algorithm for feedforward neural network training, *Int. J. of Artificial Intelligence & Applications (IJAIA)*, 2 (2011), doi: 10.5121/ijaia.2011.2304.

[28]. N. M. Nawi, A. Khan, M. Z. Rehman, New back-propagation neural network optimized with cuckoo search algorithm, *Proc. Int. conf. Computational Science and Its Applications, ICCSA-2013*, Vol. 1, Ho Chi Minh City, Vietnam (2013) pp. 413-426.

[29]. N. M. Nawi, A. Khan, M. Z. Rehman, Data classification using metaheuristic Cuckoo Search technique for Levenberg Marquardt back propagation (CSLM) algorithm, AIP Conference Proceedings, Vol. 1660, 2015, Penang, Malaysia (2014).

[30]. X.S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, United Kingdom (2008).

[31]. X.-S. Yang, Firefly algorithms for multimodal optimization, in *Stochastic Algorithms: Foundations and Applications, SAGA 2009*, *Lecture Notes in Computer Sciences*, Vol.5792, eds. O. Watanabe and T. Zeugmann, Springer, Berlin, Heidelberg (2009) pp. 169-178.

[32]. S. Nandy, P. P. Sarkar and A. Das, Analysis of a nature inspired firefly algorithm based back-propagation neural network training, *Int. J. of Computer Applications* 43(2012) 8-16, doi: 10.5120/6401-8339.
[33]. I. Brajevic and M. Tuba, Training feed-forward neural networks using firefly algorithm, *Recent Advances in Knowledge Engineering and Systems Science: Proc. of the 12th Int. Conf. Artificial Intelligence, Knowledge Engineering and Data Bases (AIKED’13)*, (2013) 156-161.

[34]. M. Alweshah, Firefly Algorithm with Artificial Neural Network for Time Series Problems, *Research Journal of Applied Sciences, Engineering and Technology* 7(19) (2014) 3978-3982, doi: 10.19026/rjaset.7.757.

[35]. D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley: Reading, Canada (1989).

[36]. A.J.F. van Rooij; L.C. Jain and R.P. Johnson, Neural Network Training using Genetic Algorithms, Singapore: World Scientific (1996).

[37]. U. Seiffert, Multiple Layer Perceptron Training using Genetic Algorithms, *Proc. European Symposium on Artificial Neural Networks*, Bruges, Belgium (2001), pp. 159-164.

[38]. E. Alba and J. F. Chicano, Training Neural Networks with GA Hybrid Algorithms, *Genetic and Evolutionary Computation-GECCO 2004*, eds. K. Deb, Vol. 3102, Springer, Berlin, Heidelberg (2004) pp. 852-863.

[39]. E. Rashedi, H. Nezamabadi-pour and S. Saryazdi, GSA: A gravitational search algorithm, *Information Sciences*, 179 (2009) 2232-2248, doi:10.1016/j.ins.2009.03.004.

[40]. S. Saha, D. Chakraborty and O. Dutta, Guided convergence for training feed-forward neural network using novel gravitational search optimization, *Proc. 2014 International Conference on High Performance Computing and Applications (ICHPCA)*, Bhubaneswar, India (2014), pp. 1-6.

[41]. S. Mirjalili, S. M. Mirjalili and A. Lewis, Grey wolf optimizer, *Advances in Engineering Software*, 69(2014) pp. 46-61, doi: 10.1016/j.advengsoft.2013.12.007.

[42]. S. Mirjalili, How effective is the Grey Wolf optimizer in training multi-layer perceptrons, *Applied Intelligence*, 43(2015) pp. 150-161, doi: 10.1007/s10489-014-0645-7.

[43]. Z. W. Geem, J. H. Kim, and G. V. Loganathan, A new heuristic optimization algorithm: Harmony search, *Simulation*, 76(2001) pp. 60-68, doi: 10.1177/003754970107600201.

[44]. M. Mahdavi, M. Fesanghary, and E. Damangir, An Improved Harmony Search Algorithm for Solving Optimization Problems, *Applied Mathematics and Computation*, 188(2007) pp. 1567-1579, doi: 10.1016/j.amc.2006.11.033.

[45]. A. Kattan, R. Abdullah and R. A. Salam, Harmony search based supervised training of artificial neural networks, *Proc. 2010 Int.C. on Intelligent Systems, Modelling and Simulation*, Liverpool, United Kingdom (2010), pp. 105-110.

[46]. Q-K. Pan , P.N. Suganthan , M. F. Tasgetiren and J.J. Liang, A self-adaptive global best harmony search algorithm for continuous optimization problems, *Applied Mathematics and Computation*, 216(2010) pp. 830-848, doi: 10.1016/j.amc.2010.01.088.

[47]. S. Kulluk, L. Ozbakir and A. Baykasoglu, Training neural networks with harmony search algorithms for classification problems, *Engineering Applications of Artificial Intelligence*, 25(2012) pp. 11-19, doi: 10.1016/j.engappai.2011.07.006.

[48]. A. R. Mehrabian and C. Lucas, A novel numerical optimization algorithm inspired from weed colonization, *Ecological Informatics*, 1(2006) pp. 355-366, doi:10.1016/j.ecoinf.2006.07.003.

[49]. R. Giri, A. Chowdhury, A. Ghosh, S. Das, A. Abraham and Y. Snasel, A modified invasive weed optimization algorithm for training of feed-forward neural networks, *Proc. 2010 IEEE Int. Conf. on Systems Man and Cybernetics (SMC)*, Istanbul, Turkey (2010), pp. 3166-3173.

[50]. A. H. Gandomi and A. H. Alavi, Krill herd: A new bio-inspired optimization algorithm, *Communications in Nonlinear Science and Numerical Simulation* 17(2012) pp. 4831-4845, doi:10.1016/j.cnsns.2012.05.010.
[51]. N. S. Lari and M. S. Abadeh, Training artificial neural network by krill-herd algorithm, Proc. 2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference, Chongqing, China (2014), pp. 63-67.

[52]. P. A. Kowalski and S. Lukasik, Training neural networks with krill herd algorithm, Neural Processing Letters, 44(2016) pp. 5-17, doi: 10.1007/s11063-015-9463-0.

[53]. S. Mirjalili, Moth-Flame Optimization Algorithm: A novel nature-inspired heuristic paradigm, Knowledge-Based Systems, 89(2015) pp. 228-249, doi: 10.1016/j.knosys.2015.07.006.

[54]. W. Yamany, M. Fawzy, A. Tharwat and A. E. Hassanien, Moth-flame optimization for training Multi-Layer Perceptrons, Proc. 2015 11th International Computer Engineering Conference (ICENCO), Cairo, Egypt (2015), pp. 267-272.

[55]. S. Mirjalili, S. M. Mirjalili and A. Hatamlou, Multi-Verse Optimizer: a nature-inspired algorithm for global optimization, Neural Computing and Applications, 27(2016) pp. 495-513, doi: 10.1007/s00521-015-1870-7.

[56]. H. Faris, I. Aljarah and S. Mirjalili, Training feedforward neural networks using multi-verse optimizer for binary classification problems, Applied Intelligence, 45(2016) pp. 322-332, doi: 10.1007/s10489-016-0767-1.

[57]. J. Kennedy and R. Eberhart, Particle swarm optimization, in Proc. IEEE Int. Conf. on Neural Network, Perth, Australia (1995), pp. 1942–1948.

[58]. M. Yaghini, M. M. Khoshraftar and M. Fallahi, A hybrid algorithm for artificial neural network training, Engineering Applications of Artificial Intelligence, 26(2013) pp. 293-301, doi:10.1016/j.engappai.2012.01.023.

[59]. H. R. Tizhoosh, Opposition-based learning: a new scheme for machine intelligence, Proc. Int. Conf. Computational Intelligence Modeling Control and Automation, Vol. 1, Vienna, Austria (2005), pp. 695–701.

[60]. Z. Beheshti and S. M. H. Shamsuddin, CAPSO: centripetal accelerated particle swarm optimization, Information Sciences, 258(2014) pp. 54-79, doi: 10.1016/j.ins.2013.08.015.

[61]. Z. Beheshti, S. M. H. Shamsuddin, E. Beheshti and S. S. Yuhaniz, Enhancement of artificial neural network learning using centripetal accelerated particle swarm optimization for medical diseases diagnosis, Soft Computing 18(2014) pp.2253-2270, doi:10.1007/s00500-013-1198-0.

[62]. E. Cuevas, M. Cienfuegos, D. Zald’ivar, and M. P’erez-Cisnero, A swarm optimization algorithm inspired in the behavior of the social-spider, Expert Systems with Applications, 40(2013) pp. 6374–6384, doi: 10.1016/j.eswa.2013.05.041.

[63]. L. A. M. Pereira, D. Rodrigues, P. B. Ribeiro, J. P. Papa and S. A. T. Weber, Social-spider optimization-based artificial neural networks training and its applications for Parkinson's disease identification, Proc. 2014 IEEE 27th International Symposium on Computer-Based Medical Systems, New York, USA (2014), pp. 14-17.

[64]. F. Glover, Future paths for integer programming and links to artificial intelligence, Computers & Operations Research, 13(1986) pp. 533-549, doi: 10.1016/0305-0548(86)90048-1.

[65]. R. S. Sexton, B. Alidaeae, R. E. Dorsey and J. D. Johnson, Global optimization for artificial neural networks: A tabu search application, European Journal of Operational Research, 106(1998) pp. 570-584, doi:10.1016/S0377-2217(97)00292-0.

[66]. R. Tang, S. Fong, X. S. Yang and S. Deb, Wolf search algorithm with ephemeral memory, Proc. Seventh International Conference on Digital Information Management (ICDIM 2012), Macau (2012), pp. 165-172.
[67] N. M. Nawi, M. Z. Rehman and A. Khan, WS-BP: An efficient wolf search based backpropagation algorithm, *AIP Conference Proceedings*, 1660(2015), doi:10.1063/1.4915660.

[68] H. Shi and Wanqing Li, Artificial neural networks with ant colony optimization for assessing performance of residential buildings, *Proc. 2009 International Conference on Future BioMedical Information Engineering (FBIE)*, Sanya, China (2009) pp. 379-382.

[69] W-C. Yeh and T-J Hsieh, Artificial bee colony algorithm-neural networks for S-system models of biochemical networks approximation, *Neural Comput. & Applic*. 21(2012) pp.365-375, doi:10.1007/s00521-010-0435-z.

[70] Y. Zhang, P. Phillips, S. Wang, G. Ji, J. Yang and J. Wu, Fruit classification by biogeography-based optimization and feedforward neural network, *Expert Systems: The Journal of Knowledge Engineering*, 33(2016) pp. 239-253, doi: 10.1111/exsy.12146.

[71] A. Rodan, H. Faris and J. Alqatawna, Optimizing feedforward neural networks using biogeography based optimization for E-Mail spam identification, *Int. J. Communications, Network and System Sciences*, 9(2016) pp. 19-28, doi: 10.4236/ijcns.2016.91002.

[72] S.C. Nayak, B.B. Misra, H.S. Behera, Artificial chemical reaction optimization of neural networks for efficient prediction of stock market indices, *Ain Shams Engineering Journal*, 8(2017) pp. 371-390, doi: 10.1016/j.asej.2015.07.015.

[73] K. Chaowanawatee and A. Heednacram, Implementation of cuckoo search in RBF neural network for flood forecasting, *Proc. 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks*, Phuket, Thailand (2012), pp. 22-26.

[74] S. Mirjalili, Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems, *Neural Computing and Applications*, 27(2016) pp. 1053-1073, doi:10.1007/s00521-015-1920-1

[75] J. Kumaran and J. Sasikala, Dragonfly optimization based ANN model for forecasting India's primary fuels' demand, *International Journal of Computer Applications* 164(2017) pp. 18-22, doi:10.5120/ijca2017913636.

[76] A. Kaushik, D. K. Tayal, K. Yadav and A. Kaur, Integrating firefly algorithm in artificial neural network models for accurate software cost predictions, *J. of Software: Evolution and Process*, 28 (2016), doi: 10.1002/smr.1792.

[77] J. Morshed and J. J. Kaluarachchi, Parameter estimation using artificial neural network and genetic algorithm for free-product migration and recovery, *Water Resources Research*, 34(1998) pp. 1101-1113, doi:10.1029/98WR00006.

[78] G. H. Kim, D. S. Seo and K. I. Kang, Hybrid models of neural networks and genetic algorithms for predicting preliminary cost estimates, *Journal of Computing in Civil Engineering*, 19(2005) pp. 208-211, doi: 10.1061/(ASCE)0887-3801(2005)19:2(208).

[79] M. R. Razfar, R. F. Zinati and M. Haghshenas, Optimum surface roughness prediction in face milling by using neural network and harmony search algorithm, *The International Journal of Advanced Manufacturing Technology*, 52(2011) pp 487–495, doi: 10.1007/s00170-010-2757-5.

[80] A. Lee, Z. W. Geem and K-D. Suh, Determination of optimal initial weights of an artificial neural network by using the Harmony search algorithm: Application to breakwater armor stones, *Applied Sciences* 6(2016), doi: 10.3390/app6060164.

[81] E. Atashpaz-Gargari, C. Lucas, Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition, *IEEE Congress on Evolutionary Computation* 7(2007) pp. 4661-4667.

[82] H. Taghavifar, A. Mardani and L. Taghavifar, A hybridized artificial neural network and imperialist competitive algorithm optimization approach for prediction of soil compaction in
soil bin facility, Measurement Volume, 46(2013) pp. 2288-2299, doi: 10.1016/j.measurement.2013.04.077.

[83]. M. Ghaedi, A.M. Ghaedi, E. Negintaji, A. Ansari and F. Mohammadi, Artificial neural network - Imperialist competitive algorithm based optimization for removal of sunset yellow using Zn(OH)2 nanoparticles-activated carbon, Journal of Industrial and Engineering Chemistry, 20(2014) pp. 4332-4343, doi: 10.1016/j.jiec.2014.01.041.

[84]. G-G. Wang, S. Deb and Z-H. Cui, Monarch butterfly optimization, Neural Comput. Appl., 2015 pp. 1–20, doi:10.1007/s00521-015-1923-y.

[85]. D. Devikanniga and R. J. S. Raj, Classification of osteoporosis by artificial neural network based on monarch butterfly optimisation algorithm, Healthcare Technology Letters, 5(2018) pp. 70 – 75, doi:10.1049/htl.2017.0059.

[86]. M. Li, X. Huang, H. Liu, B. Liu and Y. Wu, Prediction of gas solubility in polymers by back propagation artificial neural network based on self-adaptive particle swarm optimization algorithm and chaos theory, Fluid Phase Equilibria, 356(2013), pp. 11-17, doi:10.1016/j.fluid.2013.07.017.

[87]. S. Kirkpatrick, C. D. Gerlatt Jr and M. P. Vecchi, Optimization by simulated annealing, Science, 220(1983) pp. 671–680, DOI: 10.1126/science.220.4598.671.

[88]. B. Abbasi and H. Mahlooji, Improving response surface methodology by using artificial neural network and simulated annealing, Expert Systems with Applications, 39(2012) pp. 3461-3468, doi:10.1016/j.eswa.2011.09.036.

[89]. R.V. Rao, V.J. Savsani and D.P. Vakharia, Teaching–Learning-Based Optimization: An optimization method for continuous non-linear large scale problems, Information Sciences, 183(2012) pp. 1-15, doi:10.1016/j.ins.2011.08.006.

[90]. E. Uzlu, M. Kankal, A. Akpinar and T. Dede, Estimates of energy consumption in Turkey using neural networks with the teaching-learning-based optimization algorithm, Energy, 75(2014) pp. 295-303, doi: 10.1016/j.energy.2014.07.078.

[91]. Z ZAIN, “High Speed And Lowpower Gdi Based Full Adder”, Journal of VLSI Circuits And Systems, 1 (01), 5-9,2019

[92]. Pb Agus Ristono*,"Design Of Reliable And Efficient Manchester Carry Chain Adder Based 8-Bit Alu For High Speed Applications",Journal Of VLSI Circuits And Systems, 1 (01), 1-4,2019