IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 24, 2020, accepted July 7, 2020, date of publication July 15, 2020, date of current version July 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009533

An Efficient Intrusion Detection Model Based on Hybridization of Artificial Bee Colony and Dragonfly Algorithms for Training Multilayer Perceptrons

WAHEED ALI H. M. GHANEM[1,2,3,4], AMAN JANTAN[5], SANAA ABDULJABBAR AHMED GHALEB[3,6], AND ABDULLAH B. NASSER[7]

[1]Faculty of Engineering, University of Aden, Aden, Yemen
[2]Faculty of Education-Saber, University of Aden, Aden, Yemen
[3]Faculty of Education, University of Aden, Aden, Yemen
[4]Faculty of Ocean Engineering Technology and Informatics, Universiti Malaysia Terengganu, Kuala Terengganu 21030, Malaysia
[5]School of Computer Science, Universiti Sains Malaysia, Gelugor 11800, Malaysia
[6]Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Kuala Terengganu 22200, Malaysia
[7]Faculty of Computer Systems and Software Engineering, Universiti Malaysia Pahang, Kuantan 26300, Malaysia

Corresponding author: Waheed Ali H. M. Ghanem (waheed.ghanem@gmail.com)

This work was supported in part by the Universiti Sains Malaysia under RUI Grant under Grant 1001/PKOMP/8014017.

ABSTRACT One of the most persistent challenges concerning network security is to build a model capable of detecting intrusions in network systems. The issue has been extensively addressed in uncountable researches and using various techniques, of which a commonly used technique is that based on detecting intrusions in contrast to normal network traffic and the classification of network packets as either normal or abnormal. However, the problem of improving the accuracy and efficiency of classification models remains open and yet to be resolved. This study proposes a new binary classification model for intrusion detection, based on hybridization of Artificial Bee Colony algorithm (ABC) and Dragonfly algorithm (DA) for training an artificial neural network (ANN) in order to increase the classification accuracy rate for malicious and non-malicious traffic in networks. At first the model selects the suitable biases and weights utilizing a hybrid (ABC) and (DA). Next, the neural network is retrained using these ideal values in order for the intrusion detection model to be able to recognize new attacks. Ten other metaheuristic algorithms were adapted to train the neural network and their performances were compared with that of the proposed model. In addition, four types of intrusion detection evaluation datasets were applied to evaluate the proposed model in comparison to the others. The results of our experiments have demonstrated a significant improvement in inefficient network intrusion detection over other classification methods.

INDEX TERMS Intrusion detection system (IDS), multilayer perceptron (MLP), metaheuristic algorithm (MA), artificial bee colony algorithm (ABC), dragonfly algorithm (DA).

I. INTRODUCTION

As it stands the problem of cyber-attacks on networking systems is undeniably quite prevalent and expanding with time which make it imperative to establish intrusion detection systems (IDS)s in any business and non-business environments a like nowadays. In 1980, Anderson [1] introduced IDSs, which were later improved by Denning [2]. There have been constant improvements of IDSs in the form of hardware or software since. The key function of the intrusion detection system (IDS) is to identify and respond to intrusive and harmful actions faced by the system resources. This is performed by monitoring and evaluating the activities in the network [3]. The IDSs are classified into anomaly-based and misuse-based [4], following the detection method. The IDS searches for the attack fingerprint within a huge database that contains the entire attack signatures to discover the misuse. Contrarily, the IDS observes the variation in the system behavior to detect abnormalities. The hybrid approach combines the application of the aforementioned approaches.

The associate editor coordinating the review of this manuscript and approving it for publication was Firooz B. Saghezchi.

Generally, the hybrid or enhanced technique is superior compared to individual or original methods [5].

Lately, data mining and machine learning techniques have become important for improving the quality of the performance of IDSs. They expedite the procedure of classifying the types of attacks and improve the efficiency of the IDS. The key function of the data mining technique is to gather simulating data from huge data repositories and transform it into explicit and meaningful information [6]. Clustering, classification, information preprocessing and recognizing patterns are the most common techniques applied in data mining. Classification is an important method, where it is applied to precisely analyze the intended category for individual samples of the data [7]. Classification involves discovery of the hidden data patterns. This is a common task in data mining and machine learning techniques [8], [9].

A number of approaches are available to identify abnormalities using data mining. Prominent examples of these approaches are the artificial neural network, radial basis function [10], [11], multi-layer perceptron [12], recurrent neural network [13], fuzzy neural network, evolving fuzzy neural network [14], self-organizing map [15], [16], convolutional neural network [17], support vector machines [18]–[20] and SVM with modified versions [21], [22].

There have been many studies on biology and natural phenomena such as swarm intelligence, which describe the behavior of animals and insects [23]. This behavior encompasses actions intended for methodological assessment of discovering food reserves, forming nests and shifting the nests from their origin, as well as other activities. This facilitates the enhancement of the IDS performance. With its improved ability to impute cause of the attack, it is feasible to distinguish malicious and un-malicious conduct as well as identify complex disputes [24].

The artificial neural network (ANN) is considered as one of the widely used machine learning techniques, Also, the ANN is known as the most significant part of artificial intelligence, which is divided into supervised learning neural networks and unsupervised learning neural networks. Essentially, the supervised network requires an individual's guidance, while the unsupervised network does not require guidance [25]. The following parameters determine the performance of its action [26]: the design of the system, the training algorithm; attributes applied in the training. The aforementioned parameters form the optimum pattern as a challenging subject [27]. Besides, a training algorithm will only reach a local minimum if any of these parameters are selected appropriately and precisely. In light of this, a previous study reported several techniques, which are formulated on the heuristic algorithms for acquiring ANNs models [28].

The ANN possesses several features that enable it to solve a number of disputes including pattern classification, regression as well as forecasting. Moreover, the ANN has remarkable attributes as following: capacity to study using instances, flexibility to simplify and ability to resolve issues including, categorizing patterns, estimating functions and optimization [29], [30].

One of the most used ANN models is the well-known Multi-Layer Perceptron (MLP). The majority of applications utilize a feed-forward type of NNs that implies the use of the typical back-propagation (BP) learning method. This type of training is commonly completed by applying the back-propagation gradient-descent technique [31]. As the algorithm depends on the gradient, several problems might arise after applying this algorithm. One of the most notable problems is the chance to get stuck in a local minimum as a result of the quite low speed of convergence [31].

Moreover, the BP algorithm has to be able to determine a few essential learning parameters such as the rate of learning, the momentum, and the prearranged structure. The BP algorithm has a prefixed NNs structure, which means it trains only its weights in its structure. As a result, there is no solution as for how to design a nearly perfect NNs arrangement for an application [32]. Global optimum search methods are capable of avoiding local minima and are usually utilized to regulate weights of MLPs, such as artificial bee colony (ABC) [26], [32], particle swarm optimization (PSO) [33], [34], evolutionary algorithms (EA), simulated annealing (SA) and ant colony optimization (ACO), which means they can also be used in order to dispose of the problems in standard back-propagation algorithm.

The MLP is considered as one of the widely used machine learning techniques to enhance the detection rate of the intrusion detection systems, the training process of MLPs for pattern classification problems consists of two tasks, the first one is the selection of appropriate architecture for the problem, and the second is the adjustment of the connection weights of the network, this study focuses on dealing with the fixed structure of the MLP, in which numerous difficult hands-on issues have been well elucidated. Nonetheless, the general architecture of MLP still has limitations in terms of the local optima and low convergence speed issues [35], [36]. As such, the three major drawbacks of an intrusion detection system based on Multi-layer perceptron neural networks (MLPNN) are as follows:

- The error function of MLPNN is a multimodal function that is frequently trapped into local minima.
- This type of MLPNN-based IDS demonstrates a slow convergence speed.
- The occurrence of over-fitting, which usually creates an overly complex model.

To overcome the shortcomings attributed to BP training algorithm and prevent the fall into the trap of local minima, the current study proposes the new hybrid algorithm (HAD) algorithm to train MLP. At the same time, the new approach using the HAD might provide an effective and suitable alternative solution for MLP training and the problem of global as well as local optima in a multimodal search space. Our previous work evidenced that the HAD algorithm could guarantee finding a global optimum solution [37], while the

BP algorithm could only guarantee finding the initial point at the end of the slope of the search space (local optimum).

Therefore, based on the above-mentioned limitations and weaknesses, this work proposed the new HADMLP-IDS model, which is aimed to solve the problem of training MLP and is evaluated against four IDS datasets. Two of the datasets are new datasets, namely UNSW-NB15 and ISCX2012, and two are conventional datasets, namely, KDD Cup 99 and NSL-KDD. The new HADMLP-IDS technique improves the intrusion detection rate as well as reduces the false alarm rate. The study demonstrates the measures applied to solve the dataset-related problems and steps taken to enhance the detection of intrusions. The main contributions in this research are summarized as follows:

A new HAD algorithm is proposed to optimize the MLP neural network. Moreover, it is aimed to achieve its effectiveness by addressing the shortcomings of MLP in the field of network intrusion detection.

The performance, reliability, and validity of the new approach in detecting a new attack were assessed by using two new datasets (ISCX2012 and UNSW-NB15), which were then compared with the (KDD Cup 99 and NSL-KDD) datasets. The new proposed model was compared with other related works of evolutionary and swarm intelligence algorithms. This was conducted by using the four of IDS datasets. The proposed new HADMLP-IDS model has several advantages as following: a high accuracy rate in detecting intrusions on the network; possibility of detecting an unknown intrusion; and reduction in the false alarm rate.

The remainder of this paper is structured as follows: Section 2 reviews the related work. Section 3 describes the methodology of the study, then it outlines an overview of the HADMLP-IDS framework, HAD, the neural network, and discusses how HAD can be deployed to train the ANN. Section 4 presents the experimental setup. Section 5 discusses the results of the experiments we have carried out. And finally section 6 summarizes the conclusions of the research.

II. RELATED WORK

This section briefly introduces the relevant work to ANNs used in IDSs, and then, we proceed to related work for MLP. Stochastic population-based search methods of computational swarm intelligence (CSI) can be used to train neural networks; they offer an alternative to trajectory driven methods. The combination of stochastic population-based search methods and the artificial neural network learning process is known as Stochastic Global Optimization (SGO). Stochastic global optimizations are usually inspired from biological or physical processes such as Ant Colony Optimization (ACO), Artificial Bee Colony (ABC), Particle Swarm Optimization (PSO), or Genetic Algorithms (GA). Many stochastic population-based search methods showed improvements in accuracy and efficiency computation, in comparison with the trajectory-driven methods such as Back Propagation (BP) and Levenberg Marquardt (LM) algorithms. Using such computational swarm intelligence methods for neural network training, many problems associated with BP can be overridden. However, Computational swarm intelligence methods require many function evaluations for convergence and might be slow [38].

This study introduces the use of a certain hybrid HAD algorithm method for the training of neural networks in the application of intrusion detection. To put this use in perspective and highlight the position of the introduced method in the literature, the next paragraphs summarize related works of applying swarm intelligence techniques to train neural networks for the purpose of intrusion detection.

Many researchers have investigated the deployment of ANNs for IDSs. Several ANN approaches have been used in enhanced IDS environments. In article [39], Tian, Lihao, and Jieqing, they developed a detection model based on training BP neural network using Artificial Fish Swarming Algorithm (AFSA). The algorithm optimizes the weights of BP neural network, shortens the sample training time and improves the classification accuracy.

Additionally, Shi *et al.* [40] have developed an approach where the algorithm of particle swarm optimization (PSO) is applied to improve back propagation (BP) neural network, and principal components analysis (PCA) method is used to deal with the original dataset. After optimization of BP neural network, it is employed into the intrusion detection system.

Sheikhan and Jadidi [41] also have proposed an approach for MLP-based IDS that can be used for intrusion detection on an offline mode. This study uses a Multilayer Perceptron (MLP) neural classifier to distinguish benign and malicious traffic in a flow-based NIDS. A modified gravitational search algorithm, as a modern heuristic technique, is employed to optimize the interconnection weights of the neural anomaly detector.

Tian and Liu [42], have presented an IDS using ANNs. The ANN is also trained by a particle swarm optimization (PSO) algorithm to identify attacks and unknown attacks. The ANN trained by PSO showed higher accuracy and faster convergence speed.

Authors Wang *et al.* [43], have proposed IDS based on ANN trained by GA. Their model, which was encoded in a binary system using a network audit dataset, has demonstrated a very high detection rate.

Xu *et al.* [44], have equally introduced a hybrid classifier composed of Kernel Principal Component Analysis (KPCA), RBFNN and PSO. KPCA is used to reduce the dimensions of the original sample data. RBF network is the core classifier of data and PSO algorithm is used to optimize the parameters of the RBF neural network.

In [45], the authors present a light-weight framework, which is called deep-full-range (DFR), for the detection of novel attacks, which uses deep learning for encrypted traffic classification, and intrusion detection.

Vinayakumar *et al.* [46] proposed IDS based on a distributed deep learning model with DNNs for handling and analyzing data in real-time. They collected host-based and network-based features in real-time and used the proposed

DNN model for detecting attacks and intrusions. They implemented several experiments in order to compare the DNN model with other classical ML methods. The experiments were conducted on publicly available datasets including the NSL-KDD and UNSW-NB15. The results showed that the DNN outperformed other models for the binary classification setup. Using the NSL-KDD, a DNN with 5 layers yielded a detection accuracy of 78.9% for binary categorization. On the UNSW-NB15, a DNN with 5 layers got a detection accuracy of 76.1%.

In [47], a deep learning algorithm for intrusion detection in networks was implemented and evaluated. The proposed model is trained on the NSL-KDD dataset and the deep neural network presented a much better model fitting and better accuracy on the test set with a 0.793 accuracy, with just the 6 features out of the 41 features, the deep learning model gives an accuracy of 0.759 on the test set with unseen intrusions.

Azizjon *et al.* [48] proposed IDS based on a deep learning approach using a one-dimensional Convolutional Neural Network (1DCNN), the 1D-CNN was used for supervised learning on time-series data by serializing TCP/IP packets in a predetermined time range as an invasion Internet traffic model for the IDS, where normal and abnormal network traffics are categorized and labeled for supervised learning in the 1D-CNN. The experiments were conducted on the publicly available dataset UNSW-NB15. The results showed that the 1D-CNN outperformed other models for the binary classification. Using the UNSW-NB15, a 1D-CNN yielded a detection accuracy of 0.9091.

Alazzam *et al.* [49] proposed a new wrapper feature selection algorithm for IDS using Pigeon Inspired Optimizer (PIO). The proposed PIO feature selection is designed to select the most important features needed to build a robust IDS, while ensuring a high detection rate with reduced false alarms.

Zhang *et al.* [50] proposed a unified model combining Multiscale Convolutional Neural Network with Long Short-Term Memory (MSCNN-LSTM). They attempted to use the Multiscale Convolutional Neural Network (MSCNN) to analyze the spatial features of the data stream, and then use Long Short-Term Memory (LSTM) network to process the temporal features. Finally, the model employs the spatial-temporal features to perform the classification. The experiments were conducted on the publicly available dataset UNSW-NB15.

Monshizadeh *et al.* [51] proposed a hybrid anomaly detection model, which is a platform that filters network traffic and identifies malicious activities on the network. The platform uses a combination of linear and learning algorithms combined with a protocol analyzer. The linear algorithms filter and extract distinctive attributes and features of the cyber-attacks while the learning algorithms use these attributes and features to identify new types of cyber-attacks.

Yang *et al.* [52] designed two deep learning approaches for better feature learning that was employed for the detection of malware behind encrypted TLS streams. They used an autoencoder to generate features and a Convolutional Neural Network to apply these features to train the classifier.

In [53] the authors employed the kernel principal component analysis for dimensionality reduction and feature extraction and combined the differential evolution (DE) algorithm and gravitational search algorithm (GSA) to optimize the parameters of HKELM. Then, a novel intrusion detection approach, KPCA-DEGSA-HKELM, was obtained.

In [54] the authors proposed Dendron, which is used to generate detection rules in order to classify the attacks using decision trees and genetic algorithms to develop accurate detection and linguistically interpretable rules.

Wang *et al.* [55] proposed the equality constrained-optimization-based ELM (C-ELM), which is a modified version of ELM by integration with the features of least squares support vector machines and then applied C-ELM to network intrusion detection. An adaptively incremental learning strategy is proposed to derive the optimal number of hidden neurons. The optimization criteria and a method of adaptively increasing hidden neurons with dual research were developed.

In [56], the researchers proposed new IDS approach that employed the multivariate control chart based on the fast minimum covariance determinant (MCD) algorithm to improve the capabilities of the proposed control chart to quickly and accurately detect the outliers, and kernel density estimation (KDE) to adaptively follow the network traffic data pattern, to reduce the occurrence of false alarms.

Mazini *et al.* [57] proposed a new hybrid method for an anomaly network-based IDS (A-NIDS) using artificial bee colony (ABC) and AdaBoost algorithm to gain a high detection rate and low false-positive rate. ABC algorithm is used for feature selection and AdaBoost is used to evaluate and classify the features.

In [58], Rababah and Srivastava proposed a new meta-classifier detection model for an anomaly network-based IDS which used different machine learning algorithms. First, they applied information gain attribute evaluation method to reduce the number of features and then used the combined multiple classification models via a meta-classifier for the stacking scheme in phase 2 by using Decision Tree and Random Forest.

Shone *et al.* [59] proposed a new type of autoencoder, namely non-symmetric deep autoencoder (NDAE), and utilized a deep learning classification model constructed using stacked NDAEs. At the end of the stacked NDAEs, the authors attached the random forest algorithm that undertakes the classification task based on the features learned from the NDAEs.

Further to the aforementioned work, we propose an IDS model built using the most promising HAD algorithm to train the MLP in solving the problems encountered by the MLPs training algorithm. Our proposed model is able to detect attacks and false alarm rates in ISCX 2012 and UNSW-NB15 datasets with higher accuracy. These datasets contain a new emerging attack compared with the KDD99 and NSL-KDD

datasets. Our experiments have shown that our new approach performs better than all the other techniques in the literature.

III. METHODOLOGY OF THE STUDY

The methodology of this study based on the design and implementation of a new intrusion detection system model, which is trained by hybridization of artificial bee colony algorithm and dragonfly algorithm, HAD. The objective of the study is to design a completely new model that achieves best global convergence whilst exhibits a strong robustness using HAD algorithm for training the MLP.

A. DESIGN A HYBRID ALGORITHM OF ABC AND DA

A good metaheuristic algorithm includes a balanced combination of exploiting prior knowledge that has been gathered at some point in the search process with exploring new areas in the search space, which can yield more optimal results. The Dragonfly Algorithm (DA) is a novel optimizer that tries to inspire the social life of dragonfly insect in nature [60]. It seems to have a good ability to explore and exploit the search space effectively by employing several factors affecting the exploitation versus exploration balance as shown in equation (1) and (2). However, the position update expression in equation (3) uses Levy Flight, which results in large moves leading to poor ability to exploit the local search space and pushing the algorithm apart from the global optimum.

$$\Delta X_{t+1} = (sS_i + aA_i + cC_i + fF_i + eE_i) + \omega \Delta X_t \quad (1)$$

$$x_{t+1} = X_t + \Delta X_{t+1} \quad (2)$$

$$X_{t+1} = X_t + levy\,(d) \times X_t \quad (3)$$

$$levy\,(x) = 0.01 \times (r_1 \times \sigma) \Big/ |r_2|^{(1/\beta)} \quad (4)$$

$$\sigma = \left(\Gamma\,(1+\beta) \times \sin\left(\pi\beta/2\right)/\Gamma\left(1+\beta/2\right) \times \beta \times (2)^{\beta-1/2}\right)^{1/\beta} \quad (5)$$

In equation (1), s is the separation weight, $S_i$isthe separation of the ith individual, a is the alignment weight, A_i is the alignment of the ith individual, c is the cohesion weight, C_i is the cohesion of the ith individual, f is the food factor, F_i is the food source of the i^{th} individual, e is the enemy factor, E_i is the enemy position of the ith individual, w is the inertia weight and finally t refers to the iteration number. Where X is the current position of the individual, and t is current iteration and d is the dimension of the position vectors. Where r_1 and r_2 are two random numbers in [0, 1], β is a constant (*equal to 1.5 in this work*).

On the other hand, the ABC has better ability in finding local optima through the two phases of *employee* and *onlooker*, which are considered local search operators. The onlooker bees move straight to one of the better nectar source areas of the employed bees. ABC is mostly based on selecting the solutions that improve the local search. The main difference between the phases of employed bees and onlooker bee is that the latter is based on the probability of a solution to have high fitness value in order to select it. Furthermore, the global search is implemented in the ABC algorithm by the *scout* phase, which leads to reducing the convergence speed during the search process.

After careful reflection on the previous algorithms in order to avoid their drawbacks, the result is a novel hybrid algorithm for optimization named after the ABC and DA, as (HAD). This algorithm is listed in ***Algorithm 1***. Three main phases comprises HAD: the *DA* phase, the *onlooker bee* phase, and the *modified scout bee* phase.

Algorithm 1 Main HAD Algorithm

Input: *Objective function $f(x)$ and constraints*
Initialization
1. Parameters initialization: MaxGen (Maximum Number of Generation); FN (Number of Food Sources); limit; Separation weight; Alignment weight; Cohesion weight; Food attraction weight; Enemy distraction weight.
2. Population initialization: The dragonflies' population x_i ($i = 1, 2\ldots$ FN), Step vectors Δx_i ($i = 1, 2\ldots$ FN);
3. Set Prob = 0.1 and generation iter = 0;
Iterations
4. While iter ≤ MaxGen do
5. for i = 1, 2,..., FN do
6. if rand ≤ Prob then
*7. **Dragonfly Bee Phase ();***
8. else
*9. **Onlooker Bee Phase ();***
10. end if
*11. **Modified Scout Bee Phase ();***
12. end for
13. iter = iter + 1;
14. end while
The final stage
Output: *The best solution;*

The contribution of the new hybrid algorithm is based on two improvements; first, to modify the *scout bee* phase in the ABC algorithm in order to improve the search diversity and counterbalance the shortfall of ABC algorithm in global search efficacy. The modified version of the phase is shown in ***Algorithm 4***. The second improvement is to integrate the dragonfly operator from DA into ABC as a replacement for the first phase of the standard ABC (*employee bee* phase). The improved operator is named as *dragonfly-bee* phase, and is shown in ***Algorithm 2***. The unmodified *onlooker bee* phase is listed in ***Algorithm 3***.

The pseudo-code of the modified *scout bee* phase is shown in ***Algorithm 4***. The role of this phase in the proposed algorithm is similar to its role in the original ABC algorithm. After finishing both the *dragonfly-bee* phase and the *onlooker bee* phase, HAD will check to see if there is any exhausted source to be abandoned. Special counters are used to decide whether the source is to be deserted. These counters are incremented in the previous phases whenever a *dragonfly-bee* cannot bring

Algorithm 2 Dragonfly-Bee Phase

Calculate the objective values of all dragonflies
Update the food source and enemy
Update w, s, a, c, f, and e
Calculate S, A, C, F, and E
Update neighboring radius
if a dragonfly has at least one neighboring dragonfly
    Update velocity vector using equation (1)
    Update position vector using equation (2)
else
    Update position vector using equation (3)
end if
Evaluate the fitness value of the candidate solution;
Apply a greedy selection process to select the best one;
If solution does not improve, triali = triali + 1, otherwise triali = 0;
Check and correct the new positions based on the boundaries of variables.

Algorithm 3 Onlooker Bee phase

Input: *A dragonfly position xi*
1. Select high fitness values from all x_i
2. Calculate the probability values p for the selected x_i *using* $P_i = (\frac{fit_i}{\sum_{i=1}^{FN} fit_i})$
3. for each of dragonfly bee do
4. *if rand(0, 1)* $\leq P$ *then*
5. *Update a new produced solution xi by using* $fit_i = 1 + abc\left(fit_i\right) or(\frac{1}{1} + f_i(t))$;
6. $x_{ij} = x_{ij} + \emptyset_{ij}\left(x_{ij} - x_{kj}\right)$;
7. *Apply a greedy selection process to select the best solution*
8. *end if*
9. End for
Output: *the new position xi*

better new solutions. If the counter value is greater than the parameter *limit*, then the food source is replaced with a new source, and the corresponding *dragonfly-bee* becomes a new scout bee. Assuming the abandoned source is x_i, the modified scout bee then generates a new food source to replace x_i, using the equation:

$$x_i^{t+1} = 0.5 \times rand \times (x_i^t - -BestSolution) \quad (6)$$

In HAD, it is supposed that only one source can be exhausted in each cycle, and only one *dragonfly-bee* can be a scout. If more than one counter exceeds the limit value, then one of the maximum ones might be chosen programmatically. The scout bee phase in the original ABC generates the solution randomly, which provides diversity in the search. However, this would reduce the convergence rate through the iterations. The modified scout bee phase of HAD maintains the diversity of the search by choosing a new food source at random, therefore maintaining good exploration, but also contributes to the exploitation by considering the best solution so far and generating the new solution based on both the current and best solutions as in Equation (6).

Algorithm 4 Modified Scout Bee Phase

Input: *A dragonfly position* x_i
1. Update a new produced solution x_i *using Equation (6)*
2. Apply a greedy selection process to select the best solution.
Output: *the new position* x_i

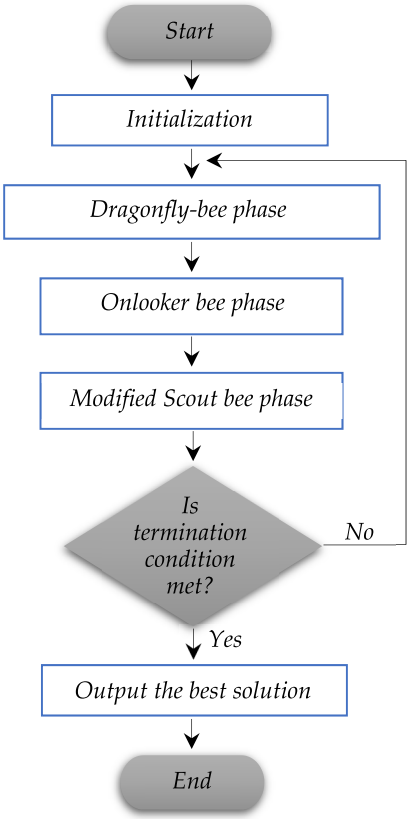

FIGURE 1. The flowchart of the HAD algorithm.

The proposed HAD algorithm is illustrated in Figure 1. This algorithm includes four phases: initialization, *dragonfly-bee* phase, *onlooker bee* phase, and *modified scout bee* phase, where the onlooker phase is the same phase inherited from the standard ABC algorithm. Thus, the new HAD algorithm is essentially an integration of the effective two global search phases (*dragonfly-bee* and *modified scout bee*) and local search phase (*onlooker bee*) for effective global optimization.

An initialization phase is used to define all the parameters including (control, ABC, and DA parameters) and assign them suitable values. The HAD algorithm adopts all parameters from the original ABC and DA algorithms and adds one new control parameter: probability parameter Prob, which is used in the HAD algorithm in order to balance the application of the dragonfly-bee phase and onlooker bee phase and

balance between exploration and exploitation. This probability parameter is set to 0.1 in this work.

B. PERCEPTRON NEURAL NETWORKS FEEDFORWARD

The most popular form of ANN models is the feedforward neural networks, which can perceive and estimate computational models using their multi parallel layered architecture. Each layer contains a set of nodes (neurons - to act as processing nodes - distributed across a series of fully connected stacked layers. The special class of neural networks is MLP.

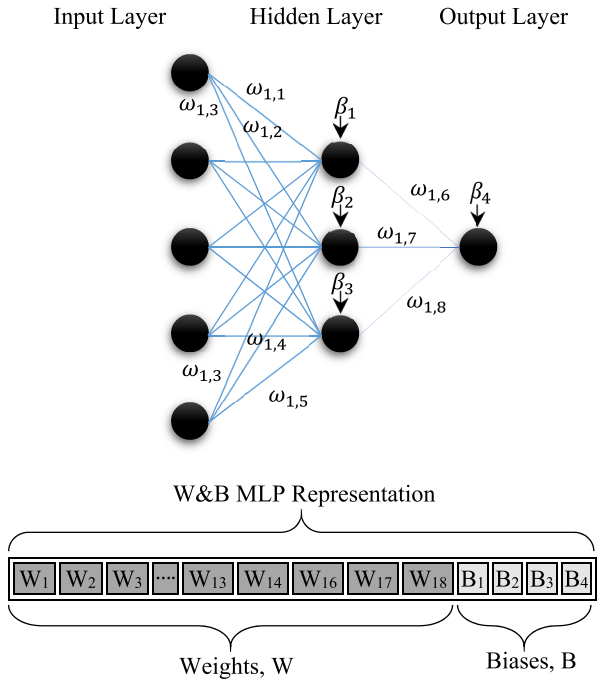

FIGURE 2. Solution representation of HADMLP-IDS model.

In MLP, neurons must be organized in unidirectional mode. MLP data transmission occurs between three classes of parallel layers: input, hidden layers, and output layers. Figure 2 shows a neural network with a single hidden layer. The connections between layers should be distinguished by some weights that are a range $[-1, 1]$. All the neurons of the MLP carry out two functions: summation and activation. The outcome of inputs, weights, and bias are summed using the summation function in Equation (7).

$$S_i = \sum_{i=1}^{n} \omega_{ij} x_i + \beta_j \tag{7}$$

where n represents the number of inputs, x_i represents the input variable i, β_i represents a bias term, and ω_{ij} represents the connection weight. An activation function should be instigated using the output of Equation (7). There are several formulations of activation functions that can be utilized in MLP. The most common one from past works is the sigmoid function [61]–[63], which is illustrated in Equation (8).

$$f_i(x) = 1/1 + e^{-s_i} \tag{8}$$

Thus, the concluding output of the neuron i can be obtained by Equation (9):

$$y_i = f_i(\sum_{i=1}^{n} \omega_{ij} x_i + \beta_j) \tag{9}$$

Once the architecture of MLP is designed, the learning step is implemented to tune and update the weights of the network. These weights are rationalized to evaluation the outcomes and minimize the error of the outputs. Learning (training) procedure of the MLP is a challenging task that can represent the capability of the MLP for tackling various categories of optimizing problems.

C. HAD FOR TRAINING MLPs

This section describes in details the proposed intrusion detection model, HAD based MLP trainer (HADMLP_IDS). As mentioned before, HAD is used to train the MLP network. Thus, two critical points must be addressed before we start: how to encode the solution in HAD optimizer, and how to present the fitness function. All the solutions of the proposed model are encoded as one-dimensional vectors of randomly generated real values range [0, 1]. Figure 2 depicts the way of representing the encoding strategy of HAD in the HADMLP-IDS model. Figure 2 shows the encoded vector with connection weights and bias series of the solutions, which in turn corresponds to the weights and biases of the trained MLP model. The total number of weights and biases in the target network determines the length of these vectors. A similar encoding strategy is used for HADMLP-IDS. The next objective is to carefully consider the choice of the fitness function. In order to acquire the fitness of the HAD algorithm, they must be sent to the MLP network as the connection weights.

The MLP can evaluate these vectors according to the training data set. Finally, the neural networks will gain the fitness values of the corresponding solutions. In this study, the mean squared error (MSE) is utilized as the fitness function in the HADMLP-IDS trainer for evaluating the fineness of the model. The aim is to reduce the MSE value as much as possible. For training samples, the MSE scale can be obtained using the actual and expected solution variance from the generated solutions (MLPs). The MSE calculated by Equation (10).

$$MSE = 1/n \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{10}$$

where y_i represents the actual value, $\hat{y}_i$ represents the predicted one, and n represents the total number of instances.

IV. THE EXPERIMENTAL FRAMEWORK

The implementation and evaluation of the proposed model was conducted using a laptop loaded with Core i5 2.4 GHz CPU and 8 GB RAM, and MATLAB R2014a running on a Windows 7. To evaluate the performance of HADMLP-IDS model, four of experiments were performed. A different dataset was used in each experiment for the offline evaluation of the IDSs, namely KDD Cup 99, NSL-KDD, ISCX2012,

and UNSW-NB15, against eleven of the metaheuristic algorithms which are have been adapted with ANNs by the same method that's used to adapting the framework of the current proposal.

A. IDS DATASETS

Unlike the datasets for general classification problems, the evaluation of the final neural network model for the special purpose of intrusion detection dictates the use of special benchmark datasets for this specific application. This section briefly explains four available datasets for testing IDSs.

1) KDD CUP 99 DATASET

The most popular and widely used dataset regarding the detection of intruders and anomalies is the KDD Cup 1999 Dataset, which was created and developed in 1999 by Lee and Stolfo [64]. This dataset was built on the information obtained from MIT Lincoln Laboratory, under Defense Advanced Research Projects Agency (DARPA ITO) and Air Force Research Laboratory (AFRL/SNHS) sponsorship. It is made out of a set of records that can be approximated at about 5 million. It represents TCP/IP packet connections, where each packet connection contains 41 attributes (features) out of which 38 are numeric and 3 are symbolic.

The KDD Cup '99 Dataset has 23 attacks, which have been categorized into four types of assault data: Denial of service (DOS), probing (PROBE), and User to Root (U2R) and Remote to Local (R2L). The set of KDD Cup '99 attributes is divided into three parts: basic, content and traffic attributes, respectively. Basic attributes contain all attributes obtained from the packet headers, whereas the content attributes include those extracted from packets payload in order to find dubious behavior in the payload section. The traffic attributes are also classified into two types: the ''same host'' and ''same service''. Each of them is aiding to determine whether the connection is with the same host or the same service respectively [65], [66].

In this research, four subsets of the KDD Cup '99 dataset have been used, which were created and randomized by [67], and are used by many researchers [68]–[71]. Every single data subset houses approximately 4000 records, of which nearly half of the data (50 to 55%) belong to the normal category and the leftovers are mere attacks. Dataset 1 is used for training, while datasets 2, 3, and 4 are utilized for testing. The classes of all the datasets, number of records and the percentage of occurrence of the feature classes are tabulated in Table 1.

TABLE 1. Distribution statistics of the kdd cup 99 training and testing datasets.

Type	Dataset 1		Dataset 2		Dataset 3		Dataset 4	
	Actual	%	Actual	%	Actual	%	Actual	%
Dos	1000	25%	1203	30%	1050	26%	903	23%
Probe	563	14%	400	10%	491	12%	475	12%
R2L	122	3%	55	1%	30	1%	62	2%
U2R	15	0%	45	1%	30	1%	10	0%
Normal	2300	58%	2300	57%	2400	60%	2550	64%
Total	4000	100%	4003	100%	4001	100%	4000	100%

2) NSL-KDD DATASET

NSL-KDD was proposed to resolve many of the inherent problems of the KDD'99 dataset. It has a reasonable size, which makes it affordable to apply the full set in one pass; hence, evaluation results of different research work will be consistent and comparable [72], [73]. The NSL-KDD dataset has also the following benefits over the original KDD dataset:

- No redundant records in the training set, so classifiers will not be biased towards more frequent records.
- No duplicate records in the testing set; thus the performance of the learners will not be influenced by the methods which have better detection rates on frequent records.
- Each level of difficulty group would have a number of records that is inversely proportional to the percentage of records in the original KDD dataset. Therefore, it caters for more accurate evaluation of different learning techniques, simply due to the diversity in range of classification rates of distinct machine learning.

This dataset is formed from the different parts of the original KDD Cup 99 dataset, without the redundancies and duplications. In addition, the problem of having an unbalanced distribution in each class, either in the training set or the testing set, was solved, to improve the accuracy of the IDS evaluation. The NSL-KDD dataset includes 41 attributes, which are labeled normal connections or attack types. The NSL-KDD dataset is divided into training and testing sets, and it has four attack classes: DoS, U2R, R2L, and probe [74], [75]. This dataset is available in (*http://nsl.cs.unb.ca/NSL-KDD/*). Table 2 shows the distribution of the records in the NSL-KDD dataset for the training and testing sets.

TABLE 2. Distribution statistics of the NSL-KDD training and testing datasets.

	Train NSL-KDD		Test NSL-KDD	
	Actual	%	Actual	%
Attack	11743	46.61%	12829	56.90%
Normal	13449	53.38%	9714	43.09%
Total	25192	100%	22543	100%

3) ISCX 2012 DATASET

In order to overcome the limitations of the KDD cup 1999 dataset, the ISCX 2012 IDS intrusion evaluation dataset at Information Security Center of excellence (ISCX) is further used to test and evaluate the performance of the proposed approach for intrusion detection.

The entire ISCX labeled dataset comprises nearly 1512000 packets with 20 features and covers seven days of network activity (i.e. normal and intrusion). The ISCX 2012 dataset is available in the packet capture form. Features

are extracted from the packet format by using ***tcptrace*** utility (http://www.tcptrace.org) and applying the following command:

$$tcptrace\ csv - l\ filename1.7z > filename1.csv.$$

The author of the dataset decided to select incoming packets for a particular host and particular days as presented in Table 3. The training data contains 54344 normal traces and 27171 attack traces while the testing data contains 16992 normal traces and 13583 additional attack traces [76].

TABLE 3. Distribution statistics of the ISCX 2012 training and testing datasets.

Date	Train ISCX 2012		Test ISCX 2012	
	Normal	Attack	Normal	Attack
11^{th}	0	0	0	0
12^{th}	2775	1388	1388	690
13^{th}	27144	13572	3393	6786
14^{th}	5028	2514	2514	1257
15^{th}	12459	6229	6229	3115
16^{th}	0	0	0	0
17^{th}	6938	3468	3468	1735
Total	54344	27171	16992	13583
	81515		30575	

4) UNSW-NB15 DATASET

This dataset is a hybrid of modern synthesized attack activities and normal traffic (available at http://www.cybersecurity.unsw.adfa.edu.au/ADFA%20NB15%20Data sets). The UNSW-NB15 dataset was created in 2015 by the researchers Nour and Jill using IXIA PerfectStorm tool in the Cyber Range Lab of the Australian Centre for Cyber Security. Similar to the KDD'99 and NSL-KDD datasets, UNSW-NB15 dataset has more than forty features. However, It is important to mention that the first two datasets share only slight common features with the UNSW-NB15 dataset, and the rest of the features are different, which makes it harder to compare them [77].

The UNSW-NB15 dataset includes nine different modern attack types (compared to 23 attack types in KDD'99 and NSL-KDD datasets) and wide varieties of real normal activities as well as 44 features in addition to the class label, consisting total of 2,540,044 records. The features of UNSW-NB15 are classified into six groups: Basic Features (BF), Content Features (CF), Flow Features (FF), Time Features (TF), Additional Generated Features (AGF) and class features. The Additional Generated Features are further classified into two sub-groups, namely the Connection Features and the General Purpose Features.

The UNSW-NB15 dataset has been divided into two subsets, the first represents the training dataset and contains 175,341 records (56000 Attacks and 119341 Normal). The second dataset contains 82,332 records (45332 Attacks and 37000 Normal) and represents the testing dataset. Both the training and testing datasets have 45 features. The distribution of these datasets us shown again in Table 4 after aggregating the attack types into one class. It is important to note that the first feature (the *id* attribute) was not mentioned in the full UNSW-NB15 dataset features and also the features *scrip*, *sport*, *dstip*, *stime* and *ltime* are missing in the training and testing dataset [78], [79].

TABLE 4. Distribution statistics of the UNSW-NB15 training and testing datasets.

	Train UNSW NB15		Test UNSW NB15	
	Actual	%	Actual	%
Attack	119341	68.06%	45332	55.06%
Normal	56000	31.94%	37000	44.94%
Total	175341	100%	82332	100%

V. RESULTS AND EVALUATION OF HAD IN TRAINING MLP NEURAL NETWORKS

The design elements explained in the previous section are used to implement multilayer-perceptron training algorithms: HADMLP-IDS, named after the hybrid metaheuristics developed in the previous section. The MLP prefix highlights the fact that the algorithm is being used to train an MLP - for the purpose of intrusion detection. This section is devoted to evaluating this algorithm against a number of standard IDS datasets. Before listing our findings, we would explain the common setup in which all experimental evaluations were conducted.

Firstly, each evaluation experiment compares 11 algorithms, including the proposed one, against four different IDS benchmark datasets: KDD CUP 1999, NSL-KDD, UNSW-NB15, and ISCX2012. These datasets range from old to the very recent, and have been introduced in section (4.1). For the ISCX2012 dataset, there is a set of five experiments, as the dataset comprises subsets of traffic on five different days. The compared algorithms include the following list: ABC, ACO, ALO, CS, DE, EHO, GSA, HAD, MFO, SCA, and WOA. Each of these algorithms is applied for training an MLP and trained as well as tested using the above datasets.

Secondly, the results of each experiment are presented in three forms: a table that lists the numerical values of the performance indicators for each algorithm; a plot that visually represents the convergence performance of each algorithm; a set of confusion matrices for HADMLP-IDS algorithm and some of the other algorithms. Each algorithm was run for a maximum of 100 iterations, and the results are calculated based on 100runs.

The aforementioned performance indicators include the accuracy ACC, detection rate DR, false alarm rate FAR, sensitivity, specificity, and precision. The FAR, DR, and ACC are calculated based on certain types of instances: true positives TP, false positives FP, true negatives TN, and false negatives FN. These four main criteria were collected from the confusion matrix. The confusion matrix summarizes the classification results. Table 5 shows the confusion matrix for binary classification.

The definitions of these types are given in Table 6, while the definitions of all performance indicators are given in Equations (11-16).

TABLE 5. The confusion matrix for binary classification.

		ACTUAL		TOTAL
		Normal	Attacks	
PREDICTED	Normal	**TN**	**FN**	TN + FN
	Attacks	**FP**	**TP**	FP + TP
TOTAL		TN + FP	FN + TP	

TABLE 6. Definitions of measurement types used to calculate performance indicators.

Type of measurement	Definition
True positive (TP)	Indicates the amount of attack data detected is actually attack data.
True negative (TN)	Indicates the amount of normal data detected is actually normal data.
False positive (FP)	Represents the normal data that is detected as attack data.
False negative (FN)	Represents the attack data that is detected as normal data.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}$$

$$DR = \frac{TP}{TP + FN} \tag{12}$$

$$FAR = \frac{FP}{FP + TN} \tag{13}$$

$$Specificity = \frac{TN}{TN + FP} \tag{14}$$

$$Sensitivity = \frac{Tp}{TP + FN} \tag{15}$$

$$Precision = \frac{TP}{TP + FP} \tag{16}$$

Thirdly, most of the benchmarking datasets contain data of different ranges; hence, there is a need to normalize feature values so that they can be effectively applied for training MLPs. The min-max normalization method is shown in Equation (17) below, where x is mapped from the interval [a, b] to [c, d].

$$x' = \frac{(x - a) \times (d - c)}{(b - a)} + c \tag{17}$$

Finally, one of the most important factors that influence the outcome of a neural network is the network structure in terms of the number of nodes in the hidden layer(s). For all experiments in this chapter, the formula shown in Equation (18) is used to determine the number of nodes in the hidden layer of the trained MLP. N is the number of attributes in the datasets (number of input nodes) and H is the number of hidden nodes.

$$H = 2 * N + 1 \tag{18}$$

As mentioned earlier, the results of evaluating the proposed algorithms comprise of eight sets: one for the evaluation against the KDD CUP 99 dataset, one for NSL-KDD dataset, five sets for ISCX2012 datasets (one subset per day for network traffic over five days), and a the last set is for the UNSW NB15 dataset. Each set's result is laid out in the following section and comprises of a table, a plot, and a confusion matrix.

A. THE KDD CUP 1999 RESULTS

Tables 7-10 show the detailed performance measurements of every compared algorithm when trying to detect anomalies via an MLP trained by our algorithm. Tables 7-10 illustrate the results of four main experiments, where each experiment executes three sub tests (dataset 1 for training and dataset 2, 3, and 4 for the purpose of testing) that were conducted using the KDD Cup 99 dataset.

The results of the proposed algorithm are shaded in grey. The results in Tables 7-10 are calculated based on the definitions in Table 6 and Equations (11-16). The TP, TN, FN and FP measurements are averaged over 100 iterations the remaining columns are derived from these basic measurements. The most important indicators are the classification accuracy, the detection rate, and the false alarm rate. The last three columns in the table highlight the rank of each algorithm according to these three indicators; the smaller the better. It is obvious from the results that the new proposed algorithm is of the top-performing MLP trainers.

The classification process is mainly affected by the data set, so different subsets of data will lead to different results, especially if the data set includes new attacks. Accordingly, the proposed approach was tested with a different classification of the data set to obtain a reliable evaluation.

Table 7 presents the results of the three first experiments, which were carried out using dataset 1 for training and datasets 2, 3 and 4 for testing. Intrusion detection model (HADMLP-IDS) performance was evaluated based on three criteria: Accuracy (ACC), Detection Rate (DR), and False Warning Rate (FAR). The experiment results have clearly demonstrated the superiority of the HADMLP-IDS model against the other models in terms of accuracy (97.2% and 94.4% respectively), HADMLP-IDS was also ranked the fourth and third with respect to the detection rate at around (99.6% and 98.7% respectively), and it is ranked the first and third best with respect to false alarm rate of (0.060 and 0.120 respectively). On the other hand, HADMLP-IDS model did not outperform the other models when tested on dataset 4 in terms of accuracy, detection rate, and false alarm rate (72.7%, 65.9%, and 0.223 respectively).

Table 8 lists the results of the second experiment using dataset 2 for training and the datasets 1, 3 and 4 as the testing datasets to evaluate the proposed model. The results for dataset 1 and 3 indicated that the HADMLP-IDS ranked the top with respect to accuracy, scoring 84.7% and 95.2% respectively. Whilst using dataset 4 it ranked the second best after WOAMLP-IDS, with a score of 88.2%. With dataset

TABLE 7. Performance measurements of 11 algorithms used to train an MLP to detect anomalies in the KDD CUP 99 dataset, by training across dataset 1 and testing across data sets 2, 3, and 4.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
Testing Dataset: 2														
1	ABCMLP-IDS	2228	1224	72	479	0.97	0.72	0.82	86.2%	96.9%	0.281	6	10	7
2	ACOMLP-IDS	2299	1117	1	586	1.00	0.66	0.80	85.3%	100.0%	0.344	7	1	10
3	ALOMLP-IDS	2288	1378	12	325	0.99	0.81	0.88	91.6%	99.5%	0.191	4	5	6
4	CSMLP-IDS	2295	1421	5	282	1.00	0.83	0.89	92.8%	99.8%	0.166	3	3	5
5	DEMLP-IDS	2241	1125	59	578	0.97	0.66	0.79	84.1%	97.4%	0.339	9	7	9
6	EHOMLP-IDS	2234	1431	66	272	0.97	0.84	0.89	91.6%	97.1%	0.160	5	9	4
7	GSAMLP-IDS	1784	1530	516	173	0.78	0.90	0.91	82.8%	77.6%	0.102	11	11	2
8	HADMLP-IDS	2290	1601	10	102	1.00	0.94	0.96	97.2%	99.6%	0.060	1	4	1
9	MFOMLP-IDS	2299	1455	1	248	1.00	0.85	0.90	93.8%	100.0%	0.146	2	1	3
10	SCAMLP-IDS	2267	1127	33	576	0.99	0.66	0.80	84.8%	98.6%	0.338	8	6	8
11	WOAMLP-IDS	2241	1114	59	589	0.97	0.65	0.79	83.8%	97.4%	0.346	10	7	11
Testing Dataset: 3														
1	ABCMLP-IDS	2361	362	40	1238	0.98	0.23	0.66	68.1%	98.3%	0.774	10	4	11
2	ACOMLP-IDS	2307	1141	94	459	0.96	0.71	0.83	86.2%	96.1%	0.287	4	5	7
3	ALOMLP-IDS	2265	1165	136	435	0.94	0.73	0.84	85.7%	94.3%	0.272	6	6	6
4	CSMLP-IDS	2382	1236	19	364	0.99	0.77	0.87	90.4%	99.2%	0.228	2	2	4
5	DEMLP-IDS	107	3469	396	29	0.21	0.99	0.79	89.4%	21.3%	0.008	3	11	1
6	EHOMLP-IDS	1315	834	1086	766	0.55	0.52	0.63	53.7%	54.8%	0.479	11	10	8
7	GSAMLP-IDS	2039	1271	362	329	0.85	0.79	0.86	82.7%	84.9%	0.206	7	9	3
8	HADMLP-IDS	2370	1408	31	192	0.99	0.88	0.93	94.4%	98.7%	0.120	1	3	2
9	MFOMLP-IDS	2242	1205	159	395	0.93	0.75	0.85	86.2%	93.4%	0.247	5	7	5
10	SCAMLP-IDS	2401	549	0	1051	1.00	0.34	0.70	73.7%	100.0%	0.657	9	1	10
11	WOAMLP-IDS	2219	739	182	861	0.92	0.46	0.72	73.9%	92.4%	0.538	8	8	9
Testing Dataset: 4														
1	ABCMLP-IDS	1604	1742	96	558	0.94	0.76	0.74	83.7%	94.4%	0.243	3	7	4
2	ACOMLP-IDS	1690	1443	10	857	0.99	0.63	0.66	78.3%	99.4%	0.373	5	2	6
3	ALOMLP-IDS	1610	1375	90	925	0.95	0.60	0.64	74.6%	94.7%	0.402	6	4	7
4	CSMLP-IDS	1672	856	28	1444	0.98	0.37	0.54	63.2%	98.4%	0.628	9	3	10
5	DEMLP-IDS	1609	2038	91	262	0.95	0.89	0.86	91.2%	94.6%	0.114	1	5	1
6	EHOMLP-IDS	987	1250	713	1050	0.58	0.54	0.48	55.9%	58.1%	0.457	11	11	8
7	GSAMLP-IDS	1571	1847	129	453	0.92	0.80	0.78	85.5%	92.4%	0.197	2	9	2
8	HADMLP-IDS	1121	1786	579	514	0.66	0.78	0.69	72.7%	65.9%	0.223	7	10	3
9	MFOMLP-IDS	1577	1676	123	624	0.93	0.73	0.72	81.3%	92.8%	0.271	4	8	5
10	SCAMLP-IDS	1700	1076	0	1224	1.00	0.47	0.58	69.4%	100.0%	0.532	8	1	9
11	WOAMLP-IDS	1609	686	91	1614	0.95	0.30	0.50	57.4%	94.6%	0.702	10	5	11

3 and 4 HADMLP-IDS ranked the first with respect to the detection rate with scores of 93.8% and 93.5% respectively. However with dataset 1 it came as the third after DEMLP-IDS and CSMLP-IDS with a score of 84.5%. The results also revealed that HADMLP-IDS on dataset 1, 3, and 4 ranked the first, third and eighth respectively with respect to false alarm rate with scores of 0.152, 0.027, and 0.157 respectively.

Table 9 shows the results of the third experiment, which utilized dataset 3 for training and the other data sets for testing purposes. Using dataset 1, CSMLP-IDS has clearly triumphal other models in terms of detection rate, followed by HADMLP-IDS, where it has the first rank with scores 87.4% and 87.0%, respectively. Nonetheless, it ranked fifth and ninth in terms of accuracy and FAR, scoring 72.0% and 0.544 respectively. The performance of the model on data set 2 indicated that it had best performance with respect to the three performance measures (ACC, DR, and FAR) with values of (98.4%, 98.7%, and 0.019 respectively). Additionally, with dataset 4 HADMLP-IDC ranked third best with respect to accuracy (88.2%), the ninth in detection rate (86.8%) and the second with respect to false alarm rate (0.107).

Table 10 presents the results of dataset 4 and dataset 1, 2, and 3 that were used for training and testing, respectively. Our findings have substantiated the superiority of HADMLP-IDS against all other algorithms in terms of accuracy, where the HADMLP-IDS had the ranked top with 82.6%, 95.5%, and 94.9% respectively. As far as detection rate is concerned, HADMLP-IDS had the second, fifth, and third ranks, scoring 82.3%, 95.6%, and 95.8% respectively. Moreover, with respect to false alarm rate, HADMLP-IDS approach had the best performance compared to the other ten models, where HADMLP-IDS achieved fourth, third and first ranks using the three datasets (1, 2, and 3) with FAR scores of 0.168, 0.047, and 0.064, respectively.

TABLE 8. Performance measurements of 11 algorithms used to train an MLP to detect anomalies in the KDD CUP 99 dataset, by training across dataset 2 and testing across data sets 1, 3, and 4.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
Testing Dataset: 1														
1	ABCMLP-IDS	1754	920	796	530	0.69	0.63	0.77	66.9%	68.8%	0.366	10	10	10
2	ACOMLP-IDS	1499	1168	1051	282	0.59	0.81	0.84	66.7%	58.8%	0.194	11	11	5
3	ALOMLP-IDS	1836	1185	714	265	0.72	0.82	0.87	75.5%	72.0%	0.183	7	9	4
4	CSMLP-IDS	2181	1135	369	315	0.86	0.78	0.87	82.9%	85.5%	0.217	3	2	6
5	DEMLP-IDS	2222	1132	328	318	0.87	0.78	0.87	83.9%	87.1%	0.219	2	1	7
6	EHOMLP-IDS	2076	964	474	486	0.81	0.66	0.81	76.0%	81.4%	0.335	6	5	8
7	GSAMLP-IDS	1910	1218	640	232	0.75	0.84	0.89	78.2%	74.9%	0.160	5	7	3
8	HADMLP-IDS	2156	1230	394	220	0.85	0.85	0.91	84.7%	84.5%	0.152	1	3	1
9	MFOMLP-IDS	2013	1227	537	223	0.79	0.85	0.90	81.0%	78.9%	0.154	4	6	2
10	SCAMLP-IDS	2079	931	471	519	0.82	0.64	0.80	75.3%	81.5%	0.358	8	4	9
11	WOAMLP-IDS	1897	810	653	640	0.74	0.56	0.75	67.7%	74.4%	0.441	9	8	11
Testing Dataset: 3														
1	ABCMLP-IDS	1961	1428	440	172	0.82	0.89	0.92	84.7%	81.7%	0.108	8	7	8
2	ACOMLP-IDS	1873	1473	528	127	0.78	0.92	0.94	83.6%	78.0%	0.079	9	10	6
3	ALOMLP-IDS	1966	1556	435	44	0.82	0.97	0.98	88.0%	81.9%	0.028	4	6	4
4	CSMLP-IDS	2216	1441	185	159	0.92	0.90	0.93	91.4%	92.3%	0.099	2	2	7
5	DEMLP-IDS	2005	1397	396	203	0.84	0.87	0.91	85.0%	83.5%	0.127	7	5	9
6	EHOMLP-IDS	1905	1546	496	54	0.79	0.97	0.97	86.3%	79.3%	0.034	6	9	5
7	GSAMLP-IDS	2039	1018	362	582	0.85	0.64	0.78	76.4%	84.9%	0.364	11	4	11
8	HADMLP-IDS	2253	1557	148	43	0.94	0.97	0.98	95.2%	93.8%	0.027	1	1	3
9	MFOMLP-IDS	1632	1597	769	3	0.68	1.00	1.00	80.7%	68.0%	0.002	10	11	1
10	SCAMLP-IDS	1919	1595	482	5	0.80	1.00	1.00	87.8%	79.9%	0.003	5	8	2
11	WOAMLP-IDS	2206	1344	195	256	0.92	0.84	0.90	88.7%	91.9%	0.160	3	3	10
Testing Dataset: 4														
1	ABCMLP-IDS	1447	1978	253	322	0.85	0.86	0.82	85.6%	85.1%	0.140	8	9	6
2	ACOMLP-IDS	1398	2084	302	216	0.82	0.91	0.87	87.1%	82.2%	0.094	7	10	3
3	ALOMLP-IDS	1517	2005	183	295	0.89	0.87	0.84	88.1%	89.2%	0.128	3	6	5
4	CSMLP-IDS	1570	1950	130	350	0.92	0.85	0.82	88.0%	92.4%	0.152	4	4	7
5	DEMLP-IDS	1573	1806	127	494	0.93	0.79	0.76	84.5%	92.5%	0.215	9	3	9
6	EHOMLP-IDS	1387	2129	313	171	0.82	0.93	0.89	87.9%	81.6%	0.074	6	11	2
7	GSAMLP-IDS	1586	1587	114	713	0.93	0.69	0.69	79.3%	93.3%	0.310	11	2	11
8	HADMLP-IDS	1590	1939	110	361	0.94	0.84	0.81	88.2%	93.5%	0.157	2	1	8
9	MFOMLP-IDS	1491	2028	209	272	0.88	0.88	0.85	88.0%	87.7%	0.118	5	7	4
10	SCAMLP-IDS	1536	1662	164	638	0.90	0.72	0.71	80.0%	90.4%	0.277	10	5	10
11	WOAMLP-IDS	1456	2131	244	169	0.86	0.93	0.90	89.7%	85.6%	0.073	1	8	1

As illustrated in Table 11, HADMLP_IDS has particularly shown superior performance over the other models in terms of the average classification accuracy, detection rate, and false alarm rate for the four datasets. It ranked the first with respect to accuracy with 88.7% and 0.141 for a false alarm rate. While the HADMLP_IDS was ranked second with respect to the detection rate at a score of 90.2%. The superiority was evaluated using our proposed approach, which achieved a distinctive gain in terms of the three performance measures we have previously stated - ACC, FAR, and DR. After all, the main criteria for measuring the efficacy of our algorithm over the other standard algorithms would be by comparing its ability to train the neural network and reduce the error rate.

Figure 3 shows the convergence plots for all models. The below plot is for one of the experiments carried out on the KDD Cup 99 data set. The main purpose of the below illustration is to ascertain each model's ability to evade a local minimum and its speed of convergence. Figure 4 illustrates the confusion matrices for the proposed model in comparison to other models. It should be noted that their choice was random and the goal was to demonstrate the performing trainers against the KDD CUP 99 dataset.

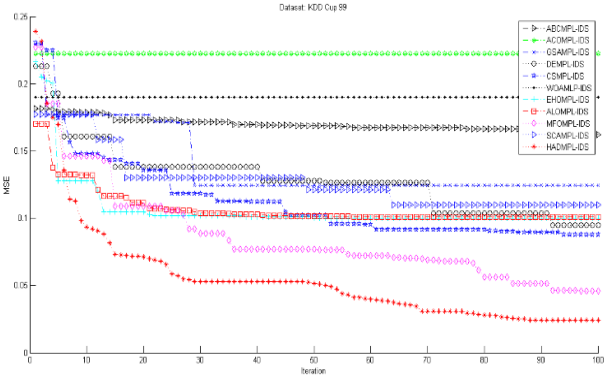

FIGURE 3. Convergence curves of all models based on averages of MSE for the benchmark classification KDD Cup 99 datasets.

TABLE 9. Performance measurements of 11 algorithms used to train an MLP to detect anomalies in the KDD CUP 99 dataset, by training across dataset 3 and testing across data sets 1, 2, and 4.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
Testing Dataset: 1														
1	ABCMLP-IDS	1733	1197	817	253	0.68	0.83	0.87	73.3%	68.0%	0.174	3	8	1
2	ACOMLP-IDS	1596	1059	954	391	0.63	0.73	0.80	66.4%	62.6%	0.270	8	9	5
3	ALOMLP-IDS	2027	982	523	468	0.79	0.68	0.81	75.2%	79.5%	0.323	1	5	6
4	CSMLP-IDS	2229	504	321	946	0.87	0.35	0.70	68.3%	87.4%	0.652	7	1	11
5	DEMLP-IDS	2066	791	484	659	0.81	0.55	0.76	71.4%	81.0%	0.454	6	3	8
6	EHOMLP-IDS	1394	1193	1156	257	0.55	0.82	0.84	64.7%	54.7%	0.177	10	11	2
7	GSAMLP-IDS	1902	1065	648	385	0.75	0.73	0.83	74.2%	74.6%	0.266	2	7	4
8	HADMLP-IDS	2219	661	331	789	0.87	0.46	0.74	72.0%	87.0%	0.544	5	2	9
9	MFOMLP-IDS	2029	869	521	581	0.80	0.60	0.78	72.5%	79.6%	0.401	4	4	7
10	SCAMLP-IDS	2010	565	540	885	0.79	0.39	0.69	64.4%	78.8%	0.610	11	6	10
11	WOAMLP-IDS	1563	1072	987	378	0.61	0.74	0.81	65.9%	61.3%	0.261	9	10	3
Testing Dataset: 2														
1	ABCMLP-IDS	2112	1592	188	111	0.92	0.93	0.95	92.5%	91.8%	0.065	7	11	3
2	ACOMLP-IDS	2204	1556	96	147	0.96	0.91	0.94	93.9%	95.8%	0.086	3	7	5
3	ALOMLP-IDS	2218	805	82	898	0.96	0.47	0.71	75.5%	96.4%	0.527	11	6	11
4	CSMLP-IDS	2176	1564	124	139	0.95	0.92	0.94	93.4%	94.6%	0.082	6	8	4
5	DEMLP-IDS	2241	1512	59	191	0.97	0.89	0.92	93.8%	97.4%	0.112	5	3	7
6	EHOMLP-IDS	2224	1637	76	66	0.97	0.96	0.97	96.5%	96.7%	0.039	2	5	2
7	GSAMLP-IDS	2251	1007	49	696	0.98	0.59	0.76	81.4%	97.9%	0.409	9	2	10
8	HADMLP-IDS	2269	1670	31	33	0.99	0.98	0.99	98.4%	98.7%	0.019	1	1	1
9	MFOMLP-IDS	2231	1523	69	180	0.97	0.89	0.93	93.8%	97.0%	0.106	4	4	6
10	SCAMLP-IDS	2137	1441	163	262	0.93	0.85	0.89	89.4%	92.9%	0.154	8	9	8
11	WOAMLP-IDS	2121	1028	179	675	0.92	0.60	0.76	78.7%	92.2%	0.396	10	10	9
Testing Dataset: 4														
1	ABCMLP-IDS	1447	1976	253	324	0.85	0.86	0.82	85.6%	85.1%	0.141	8	10	5
2	ACOMLP-IDS	1408	2111	292	189	0.83	0.92	0.88	88.0%	82.8%	0.082	5	11	1
3	ALOMLP-IDS	1616	1242	84	1058	0.95	0.54	0.60	71.5%	95.1%	0.460	11	3	11
4	CSMLP-IDS	1570	1950	130	350	0.92	0.85	0.82	88.0%	92.4%	0.152	4	6	6
5	DEMLP-IDS	1662	1841	38	459	0.98	0.80	0.78	87.6%	97.8%	0.200	6	1	8
6	EHOMLP-IDS	1534	2038	166	262	0.90	0.89	0.85	89.3%	90.2%	0.114	2	7	3
7	GSAMLP-IDS	1586	1587	114	713	0.93	0.69	0.69	79.3%	93.3%	0.310	9	4	9
8	HADMLP-IDS	1476	2053	224	247	0.87	0.89	0.86	88.2%	86.8%	0.107	3	9	2
9	MFOMLP-IDS	1631	1980	69	320	0.96	0.86	0.84	90.3%	95.9%	0.139	1	2	4
10	SCAMLP-IDS	1480	1949	220	351	0.87	0.85	0.81	85.7%	87.1%	0.153	7	8	7
11	WOAMLP-IDS	1585	1488	115	812	0.93	0.65	0.66	76.8%	93.2%	0.353	10	5	10

B. THE NSL-KDD RESULTS

This section introduces the comparative effectiveness of 11 meta-heuristics that ran against the NSL-KDD intrusion detection benchmark dataset. It demonstrates the performance measurements and their visual representation. A sample of four confusion matrices was included in our model, in addition to other best performing models. The HADMLP-IDS model is the best performing model using this dataset. The findings listed in Table 12 undoubtedly substantiates that with respect to ACC and FAR, HADMLP-IDS ranked the top scoring 91.7% and 0.108, respectively. HADMLP_IDS ranked the best third when it came to detection rate at 93.6%.

The HADMLP-IDS is the top-performing model in this dataset. Table 12 evidence that in regard to ACC and FAR the HADMLP-IDS model was ranked first across the two measurements at scores of 91.7% and 0.108, respectively, while the HADMLP_IDS model was ranked the third with respect to detection rate at a score of 93.6%. The EHOMLP-IDS was almost similar to HADMLP_IDS with regard to the accuracy of 87.9% and a false alarm rate of 0.117, followed by ALOMLP-IDS with a detection rate of 95.5%. The EHOMLP-IDS model ranked the second with respect to ACC as well as FAR, and the eighth with respect to DR. Additionally, the ALOMLP-IDS model ranked 6th, 2nd and 10th with respect to ACC, DR, and FAR respectively.

On the other hand, two models performed poorly in the NSL-KDD dataset: ABCMLP-IDS with an inferior ACC of 73.8% preceded by ACOMLP-IDS with an inferior ACC of 81.0%, GSAMLP-IDS with an inferior FAR of 0.310 preceded by ALOMLP-IDS with an inferior FAR of 0.291 and ABCMLP-IDS algorithm with an inferior DR of 70.2% preceded by ACOMLP-IDS model with an inferior DR of 83.6%. Figure 5 demonstrates the convergence plots of all models using the NSL-KDD dataset. Core evidence of this figure is

TABLE 10. Performance measurements of 11 algorithms used to train an MLP to detect anomalies in the KDD CUP 99 dataset, by training across dataset 4 and testing across data sets 1, 2, and 3.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	ACC	DR	FAR	AR	DR	FR
Testing Dataset: 1														
1	ABCMLP-IDS	1496	655	1054	795	0.59	0.45	0.65	53.8%	58.7%	0.548	11	10	10
2	ACOMLP-IDS	1558	1259	992	191	0.61	0.87	0.89	70.4%	61.1%	0.132	6	8	1
3	ALOMLP-IDS	1315	1238	1235	212	0.52	0.85	0.86	63.8%	51.6%	0.146	9	11	2
4	CSMLP-IDS	2036	1061	514	389	0.80	0.73	0.84	77.4%	79.8%	0.268	2	4	6
5	DEMLP-IDS	1717	1208	833	242	0.67	0.83	0.88	73.1%	67.3%	0.167	5	7	3
6	EHOMLP-IDS	1534	915	1016	535	0.60	0.63	0.74	61.2%	60.2%	0.369	10	9	8
7	GSAMLP-IDS	2251	537	299	913	0.88	0.37	0.71	69.7%	88.3%	0.630	7	1	11
8	HADMLP-IDS	2098	1206	452	244	0.82	0.83	0.90	82.6%	82.3%	0.168	1	2	4
9	MFOMLP-IDS	2048	922	502	528	0.80	0.64	0.80	74.3%	80.3%	0.364	3	3	7
10	SCAMLP-IDS	1856	1070	694	380	0.73	0.74	0.83	73.2%	72.8%	0.262	4	6	5
11	WOAMLP-IDS	1942	833	608	617	0.76	0.57	0.76	69.4%	76.2%	0.426	8	5	9
Testing Dataset: 2														
1	ABCMLP-IDS	2250	965	50	738	0.98	0.57	0.75	80.3%	97.8%	0.433	11	3	11
2	ACOMLP-IDS	2094	1586	206	117	0.91	0.93	0.95	91.9%	91.0%	0.069	5	9	5
3	ALOMLP-IDS	1971	1626	329	77	0.86	0.95	0.96	89.9%	85.7%	0.045	7	11	2
4	CSMLP-IDS	2176	1564	124	139	0.95	0.92	0.94	93.4%	94.6%	0.082	4	6	6
5	DEMLP-IDS	2137	1640	163	63	0.93	0.96	0.97	94.4%	92.9%	0.037	2	7	1
6	EHOMLP-IDS	2073	1595	227	108	0.90	0.94	0.95	91.6%	90.1%	0.063	6	10	4
7	GSAMLP-IDS	2251	1007	49	696	0.98	0.59	0.76	81.4%	97.9%	0.409	10	2	10
8	HADMLP-IDS	2198	1623	102	80	0.96	0.95	0.96	95.5%	95.6%	0.047	1	5	3
9	MFOMLP-IDS	2256	1492	44	211	0.98	0.88	0.91	93.6%	98.1%	0.124	3	1	7
10	SCAMLP-IDS	2137	1441	163	262	0.93	0.85	0.89	89.4%	92.9%	0.154	8	7	8
11	WOAMLP-IDS	2201	1195	99	508	0.96	0.70	0.81	84.8%	95.7%	0.298	9	4	9
Testing Dataset: 3														
1	ABCMLP-IDS	1961	1428	440	172	0.82	0.89	0.92	84.7%	81.7%	0.108	8	8	5
2	ACOMLP-IDS	1912	1352	489	248	0.80	0.85	0.89	81.6%	79.6%	0.155	11	10	7
3	ALOMLP-IDS	2265	1431	136	169	0.94	0.89	0.93	92.4%	94.3%	0.106	3	5	4
4	CSMLP-IDS	2362	1275	39	325	0.98	0.80	0.88	90.9%	98.4%	0.203	5	1	11
5	DEMLP-IDS	2271	1325	130	275	0.95	0.83	0.89	89.9%	94.6%	0.172	6	4	10
6	EHOMLP-IDS	1848	1451	553	149	0.77	0.91	0.93	82.5%	77.0%	0.093	10	11	3
7	GSAMLP-IDS	2167	1337	234	263	0.90	0.84	0.89	87.6%	90.3%	0.164	7	7	9
8	HADMLP-IDS	2300	1498	101	102	0.96	0.94	0.96	94.9%	95.8%	0.064	1	3	1
9	MFOMLP-IDS	2305	1350	96	250	0.96	0.84	0.90	91.4%	96.0%	0.156	4	2	8
10	SCAMLP-IDS	1952	1374	449	226	0.81	0.86	0.90	83.1%	81.3%	0.141	9	9	6
11	WOAMLP-IDS	2204	1494	197	106	0.92	0.93	0.95	92.4%	91.8%	0.066	2	6	2

to know the models' ability to avoid a local minimum and the speed of convergence rate for each model. Figure 6 illustrates the confusion matrices for the new proposed model in addition to other models. It should be noted that their choice was random and the goal was to demonstrate the performing trainers against the NSL-KDD dataset.

C. THE UNSW-NB15 RESULTS

In this section, the proposed model is evaluated using the most recent UNSW-NB 15 standard data set for intrusion detection. So that the evaluation is fair and equitable we have used the same 11 meta-heuristics algorithms applied in previous experiments to run against the UNSW-NB 15 in order to demonstrate the performance measurements and their visual representation.

The HADMLP-IDS has proven to be the best performing model with this data set, which has also been substantiated by all previous results. Audited the consideration of the last three columns of Table 13 showing the ranks per ACC and FAR, HADMLP-IDS model was ranked top at ACC and FAR, scoring 94.4% and 0.049 respectively. Whereas, HADMLP-IDS was ranked second best in DR at 95.72%. HADMLP-IDS model was followed by WOAMLP-IDS at accuracy of 92.5%, detection rate of 92.0% and a false alarm rate of 0.070. The WOAMLP-IDS was ranked second with respect to ACC, fourth with respect to DR, and third with respect to FAR. ALOMLP-IDS model closely followed with FAR

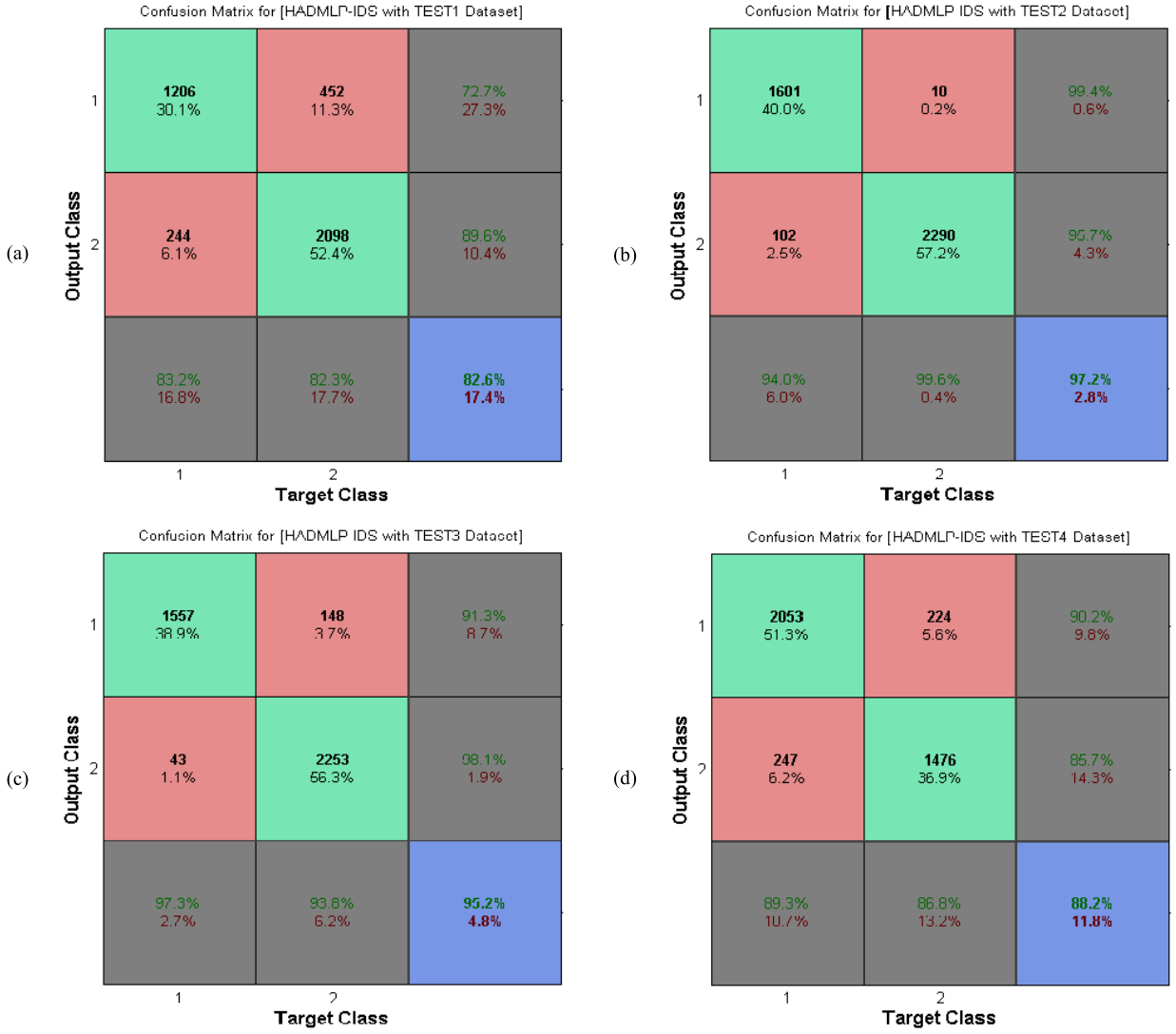

FIGURE 4. The confusion matrices against the KDD Cup 99 dataset.

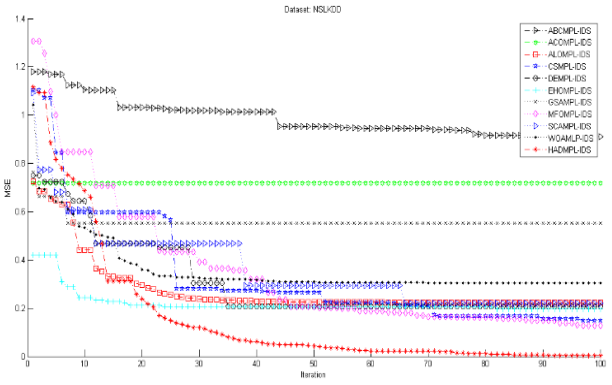

FIGURE 5. Convergence curves of all models based on averages of MSE for the benchmark classification NSL-KDD datasets.

TABLE 11. The average of the evaluation variables (ACC, FAR, and DR) for the four datasets.

No.	Alg.	ACC	DR	FAR	AR	DR	FR
1	ABCMLP-IDS	78.8%	84.0%	0.282	9	8	8
2	ACOMLP-IDS	81.6%	82.3%	0.180	5	10	3
3	ALOMLP-IDS	81.0%	86.2%	0.234	6	7	6
4	CSMLP-IDS	85.0%	92.9%	0.244	4	1	7
5	DEMLP-IDS	85.7%	84.0%	0.180	2	9	2
6	EHOMLP-IDS	78.1%	76.8%	0.199	10	11	5
7	GSAMLP-IDS	79.9%	87.5%	0.294	7	5	9
8	HADMLP-IDS	88.7%	90.2%	0.141	1	2	1
9	MFOMLP-IDS	85.6%	89.0%	0.186	3	3	4
10	SCAMLP-IDS	79.7%	88.0%	0.303	8	4	10
11	WOAMLP-IDS	77.4%	87.2%	0.338	11	6	11

of 0.059. It was ranked fifth in ACC, eighth in DR and second with respect to FAR. While the DEMLP-IDS model was ranked first with respect to DR with a score of 94.0%, but seventh with respect to ACC, and tenth with respect to FAR. On the other hand, the ACOMLP-IDS performed

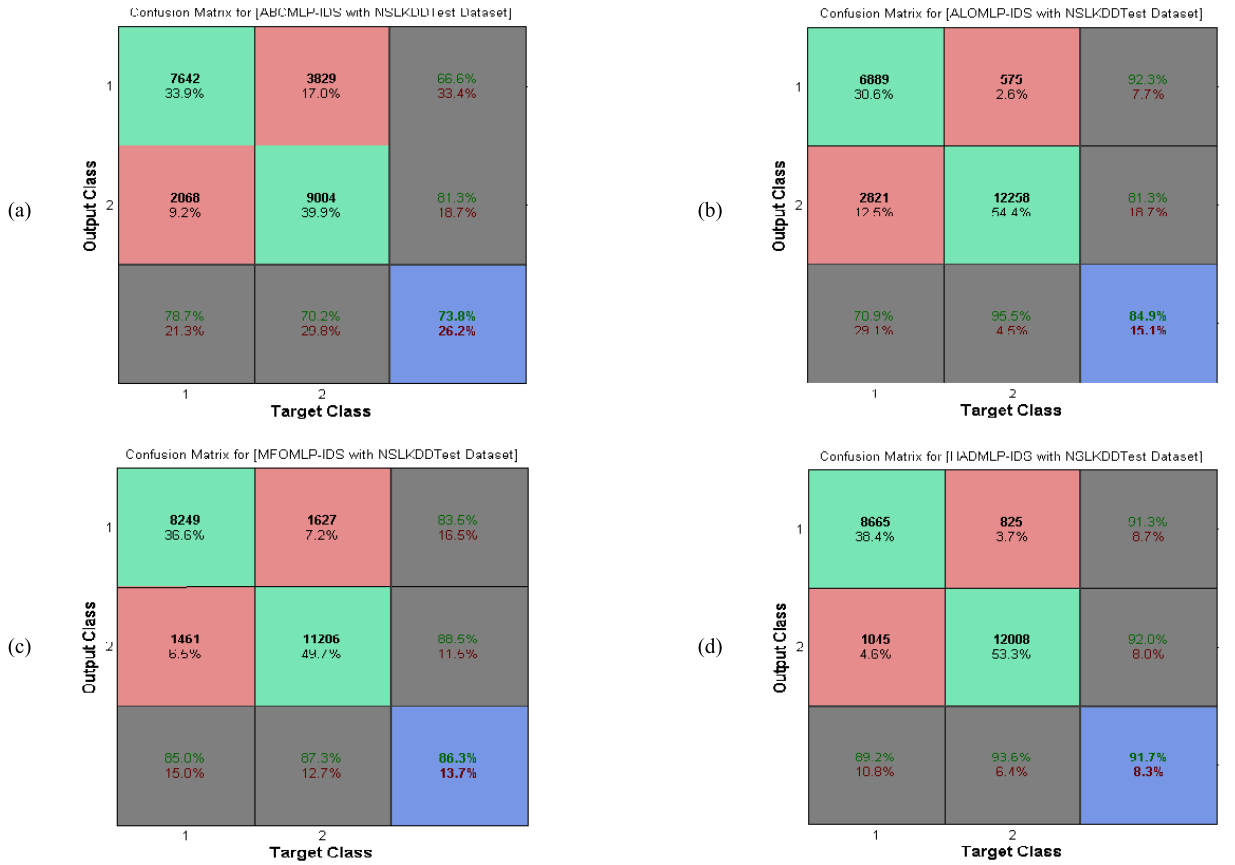

FIGURE 6. The confusion matrices against the NSL-KDD Dataset.

relatively poorly with UNSW-NB15 dataset in comparison to the other algorithms with an accuracy of 83.8%, a detection rate of 77.3%. The worst models in this experiment in terms of false alarm rates were DEMLP_IDS and EHOMLP_IDS scoring 0.203 and 0.235. Figure 7 demonstrates the convergence plots of all the models, for the UNSW-NB15 datasets, the main reason behind it is to assess each model's ability to avoid a local minimum, as well as illustrating the speed of convergence rate for each model. Figure 8 illustrates the confusion matrices for the new proposed model in addition to other models. It should be noted that their choice was random and the goal was to demonstrate the performing trainers against the UNSW-NB15 dataset.

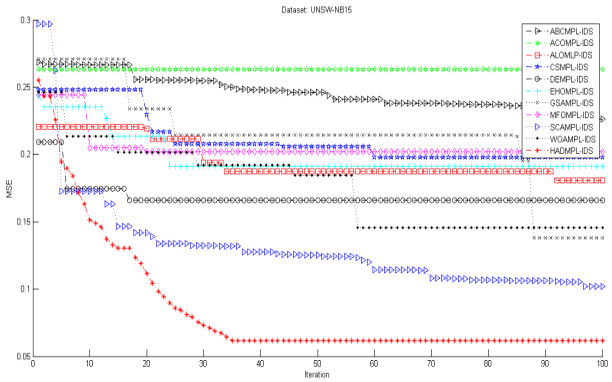

FIGURE 7. Convergence curves of all models based on averages of MSE for the benchmark classification UNSW-NB15 datasets.

D. THE ISCX 2012 RESULTS

Similar to the previous set of results, this section introduces the numerical performance measurements and their visual representation for the 11 metaheuristics algorithms when run against the ISCX2012 intrusion detection benchmark dataset. As before, sample confusion matrices are also given for the proposed models in addition to another best performing algorithm, while the confusion matrices to the remaining 11 algorithms are given in tables (14 – 18).

However, the ISCX2012 dataset is different from the other dataset because it was divided due to the large size into a number of subsets, each of which corresponds to the collected traffic in a single day. Five days were used in the experimental evaluation of the tested metaheuristics: 12, 13, 14, 15, and 17. The respective subsets are named ISCX2012-12, ISCX2012-13, ISCX2012-14, ISCX2012-15, and ISCX2012-17. Consequently, the results of this section include five sets for each of ISCX2012 subsets.

Tables 14-18 lists the detailed performance measurements of the evaluated 11 models, one table per day. The score of

TABLE 12. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the NSL-KDD dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	ACC	DR	FAR	AR	DR	FR
1	ABCMLP-IDS	9004	7642	3829	2068	0.70	0.79	0.81	73.8%	70.2%	0.213	11	11	5
2	ACOMLP-IDS	10727	7531	2106	2179	0.84	0.78	0.83	81.0%	83.6%	0.224	10	10	8
3	ALOMLP-IDS	12258	6889	575	2821	0.96	0.71	0.81	84.9%	95.5%	0.291	6	2	10
4	CSMLP-IDS	11726	7845	1107	1865	0.91	0.81	0.86	86.8%	91.4%	0.192	3	5	4
5	DEMLP-IDS	11486	7640	1347	2070	0.90	0.79	0.85	84.8%	89.5%	0.213	7	7	6
6	EHOMLP-IDS	11228	8576	1605	1134	0.87	0.88	0.91	87.9%	87.5%	0.117	2	8	2
7	GSAMLP-IDS	12332	6697	501	3013	0.96	0.69	0.80	84.4%	96.1%	0.310	8	1	11
8	HADMLP-IDS	12008	8665	825	1045	0.94	0.89	0.92	91.7%	93.6%	0.108	1	3	1
9	MFOMLP-IDS	11206	8249	1627	1461	0.87	0.85	0.88	86.3%	87.3%	0.151	4	9	3
10	SCAMLP-IDS	11633	7593	1200	2117	0.91	0.78	0.85	85.3%	90.7%	0.218	5	6	7
11	WOAMLP-IDS	11744	6913	1089	2797	0.92	0.71	0.81	82.8%	91.5%	0.288	9	4	9

TABLE 13. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the UNSW-NB15 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	ACC	DR	FAR	AR	DR	FR
1	ABCMLP-IDS	37782	32530	7550	4470	0.83	0.88	0.89	85.4%	83.3%	0.121	9	9	8
2	ACOMLP-IDS	35039	33958	10293	3042	0.77	0.92	0.92	83.8%	77.3%	0.082	11	11	4
3	ALOMLP-IDS	37206	36683	6126	2317	0.86	0.94	0.94	89.8%	85.9%	0.059	5	8	2
4	CSMLP-IDS	41433	31901	3899	5099	0.91	0.86	0.89	89.1%	91.4%	0.138	6	5	9
5	DEMLP-IDS	42629	29493	2703	7507	0.94	0.80	0.85	87.6%	94.0%	0.203	7	1	10
6	EHOMLP-IDS	41276	28319	4056	8681	0.91	0.77	0.83	84.5%	91.1%	0.235	10	7	11
7	GSAMLP-IDS	37689	33527	7643	3473	0.83	0.91	0.92	86.5%	83.1%	0.094	8	10	5
8	HADMLP-IDS	43436	34243	2896	1757	0.94	0.95	0.96	94.4%	93.7%	0.049	1	2	1
9	MFOMLP-IDS	41359	32849	3973	4151	0.91	0.89	0.91	90.1%	91.2%	0.112	4	6	6
10	SCAMLP-IDS	41933	32782	3399	4218	0.93	0.89	0.91	90.8%	92.5%	0.114	3	3	7
11	WOAMLP-IDS	36191	39985	3141	3015	0.92	0.93	0.92	92.5%	92.0%	0.070	2	4	3

the new proposed model is shaded in gray, and the last three columns show the rank of each algorithm with respect to the three main performance indicators: ACC, DR, and FAR. The results of this dataset are quite unique compared to the results against all other datasets (KDD CUP 99, NSL-KDD and UNSW-NB15).

On the one hand, several models perform outstandingly in most of the days. For example, accuracies of 100% and false alarm rates of zero can be found on several rows of the 12th, 14th, 15th, and 17th day. On the other hand, unlike the case in the other datasets, HAD-MLP outperforms other models on some days. The superior performance of these two algorithms is remarkable on this dataset particularly.

For the 12th day (Table 14), six models have surprisingly achieved the same perfect score of 100% detection rate, but the only HADMLP-IDS model that has achieved the perfect score of 100% accuracy as well as zero false alarms.

This result is unusual and seems particular for this set of data. In terms of accuracy, DEMLP-IDS did the best at a score of 99.9%, followed by CSMLP-IDS accuracy of 99.9% and then GSAMLP-IDS at a score of 99.5%. But the ALOMLP-IDS and ACOMLP-IDS record the worst scores of 65.2% and 65.6%, respectively. In terms of false alarm rate, ABCMLP-IDS and MFOMLP-IDS did the worst at a score of 0.096 and 0.024, respectively.

The results for the second set of data on the 13th day are less impressive (Table 15). In terms of accuracy, HADMLP-IDS did the best at a score of 88.5%, followed by ABCMLP-IDS accuracy of 88.5% and then ALOMLP-IDS at a score of 88.5%. GSAMLP-IDS and EHOMLP-IDS models are ranked the eleventh and tenth at an accuracy of 29.7% and 30.4%, respectively. In terms of detection rate, HADMLP-IDS is ranked the fourth 88.8%, whereas WOAMLP-IDS in ranked the first 91.6%, followed by ABCMLP-IDS 90.6%. HADMLP-IDS model has ranked the seventh at a false alarm rate of 0.120. whilst, MFOMLP-IDS and DEMLP-IDS are ranked the first and second at FAR of 0.006 and 0.012, respectively, except for HADMLP-IDS, none of the two algorithms scored well on the ACC and DR. Overall, combining the three performance indicators (assuming they have equal importance), ABCMLP-IDS performed the best, followed by HADMLP-IDS then ALOMLP-IDS with respect to the 13th day ISCX2012 dataset.

On the dataset of the 14th day (Table 16), the proposed model is ranked at the top with superior performance across the three main performance indicators: ACC, DR, and FAR.

TABLE 14. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the ISCX2012-12 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	ACC	DR	FAR	AR	DR	FR
1	ABCMLP-IDS	694	1255	0	133	1.00	0.90	0.84	93.6%	100.0%	0.096	7	1	11
2	ACOMLP-IDS	0	1366	694	22	0.00	0.98	0.00	65.6%	0.0%	0.016	10	10	8
3	ALOMLP-IDS	0	1358	694	30	0.00	0.98	0.00	65.2%	0.0%	0.022	11	10	9
4	CSMLP-IDS	694	1385	0	3	1.00	1.00	1.00	99.9%	100.0%	0.002	3	1	4
5	DEMLP-IDS	694	1386	0	2	1.00	1.00	1.00	99.9%	100.0%	0.001	2	1	3
6	EHOMLP-IDS	94	1388	600	0	0.14	1.00	1.00	71.2%	13.5%	0.000	9	9	1
7	GSAMLP-IDS	694	1377	0	11	1.00	0.99	0.98	99.5%	100.0%	0.008	4	1	6
8	HADMLP-IDS	694	1388	0	0	1.00	1.00	1.00	100.0%	100.0%	0.000	1	1	1
9	MFOMLP-IDS	694	1355	0	33	1.00	0.98	0.95	98.4%	100.0%	0.024	5	1	10
10	SCAMLP-IDS	627	1368	67	20	0.90	0.99	0.97	95.8%	90.3%	0.014	6	7	7
11	WOAMLP-IDS	273	1378	421	10	0.39	0.99	0.96	79.3%	39.3%	0.007	8	8	5

TABLE 15. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the ISCX2012-13 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	ACC	DR	FAR	AR	DR	FR
1	ABCMLP-IDS	6150	2861	636	532	0.91	0.84	0.92	88.5%	90.6%	0.157	2	2	9
2	ACOMLP-IDS	3723	3060	3063	333	0.55	0.90	0.92	66.6%	54.9%	0.098	8	9	4
3	ALOMLP-IDS	6034	2975	752	418	0.89	0.88	0.94	88.5%	88.9%	0.123	3	3	8
4	CSMLP-IDS	5606	3271	1180	122	0.83	0.96	0.98	87.2%	82.6%	0.036	5	5	3
5	DEMLP-IDS	5303	3353	1483	40	0.78	0.99	0.99	85.0%	78.1%	0.012	6	6	2
6	EHOMLP-IDS	45	3046	6741	347	0.01	0.90	0.11	30.4%	0.7%	0.102	10	10	5
7	GSAMLP-IDS	4	3020	6782	373	0.00	0.89	0.01	29.7%	0.1%	0.110	11	11	6
8	HADMLP-IDS	6028	2985	758	408	0.89	0.88	0.94	88.5%	88.8%	0.120	1	4	7
9	MFOMLP-IDS	3906	3371	2880	22	0.58	0.99	0.99	71.5%	57.6%	0.006	7	7	1
10	SCAMLP-IDS	3774	2102	3012	1291	0.56	0.62	0.75	57.7%	55.6%	0.380	9	8	11
11	WOAMLP-IDS	6216	2789	570	604	0.92	0.82	0.91	88.5%	91.6%	0.178	4	1	10

TABLE 16. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the ISCX2012-14 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
1	ABCMLP-IDS	1131	2496	126	18	0.90	0.99	0.98	96.2%	90.0%	0.007	6	7	6
2	ACOMLP-IDS	173	2420	1084	94	0.14	0.96	0.65	68.8%	13.8%	0.037	11	9	10
3	ALOMLP-IDS	127	2510	1130	4	0.10	1.00	0.97	69.9%	10.1%	0.002	10	11	4
4	CSMLP-IDS	1253	2477	4	37	1.00	0.99	0.97	98.9%	99.7%	0.015	2	3	7
5	DEMLP-IDS	137	2514	1120	0	0.11	1.00	1.00	70.3%	10.9%	0.000	9	10	1
6	EHOMLP-IDS	1057	2501	200	13	0.84	0.99	0.99	94.4%	84.1%	0.005	7	8	5
7	GSAMLP-IDS	1186	2512	71	2	0.94	1.00	1.00	98.1%	94.4%	0.001	5	6	3
8	HADMLP-IDS	1257	2514	0	0	1.00	1.00	1.00	100.0%	100.0%	0.000	1	1	1
9	MFOMLP-IDS	1227	2293	30	221	0.98	0.91	0.85	93.3%	97.6%	0.088	8	5	11
10	SCAMLP-IDS	1252	2449	5	65	1.00	0.97	0.95	98.1%	99.6%	0.026	4	4	9
11	WOAMLP-IDS	1254	2476	3	38	1.00	0.98	0.97	98.9%	99.8%	0.015	2	2	8

HADMLP-IDS is the best performing model here with a maximum score of 100% accuracy, 100% detection rate, and zero false alarm rate. WOAMLP-IDS followed with an accuracy of 98.9%, a detection rate of 99.8% and a false alarm rate of 0.015.

The results of the 15th day (Table 17) for the proposed model are very similar to the 14th day, the HADMLP-IDS records the best performing model with a maximum score of 100% accuracy, 100% detection rate, and zero false alarm rate. Except that EHOMLPIDS and CSMLPIDS model shares with HADMLPIDS the best performance at a false alarm rate of around zero. Also, the DEMLP-IDS shares with HADMLPIDS the best performance at a detection rate of 100%.

Last, the results of the 17th day (Table 18) is quite different from the previous datasets and deviate from their pattern.

TABLE 17. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the ISCX2012-15 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
1	ABCMLP-IDS	1564	5551	1551	678	0.50	0.89	0.70	76.1%	50.2%	0.109	11	11	7
2	ACOMLP-IDS	2959	5465	156	764	0.95	0.88	0.79	90.2%	95.0%	0.123	8	8	9
3	ALOMLP-IDS	3113	6023	2	206	1.00	0.97	0.94	97.8%	99.9%	0.033	2	3	4
4	CSMLP-IDS	2256	6228	859	1	0.72	1.00	1.00	90.8%	72.4%	0.000	7	10	3
5	DEMLP-IDS	3115	5214	0	1015	1.00	0.84	0.75	89.1%	100.0%	0.163	9	1	11
6	EHOMLP-IDS	2814	6229	301	0	0.90	1.00	1.00	96.8%	90.3%	0.000	4	9	1
7	GSAMLP-IDS	3113	5564	2	665	1.00	0.89	0.82	92.9%	99.9%	0.107	5	3	6
8	HADMLP-IDS	3115	6229	0	0	1.00	1.00	1.00	100.0%	100.0%	0.000	1	1	1
9	MFOMLP-IDS	2963	5327	152	902	0.95	0.86	0.77	88.7%	95.1%	0.145	10	7	10
10	SCAMLP-IDS	3010	5519	105	710	0.97	0.89	0.81	91.3%	96.6%	0.114	6	6	8
11	WOAMLP-IDS	3109	5939	6	290	1.00	0.95	0.91	96.8%	99.8%	0.047	3	5	5

TABLE 18. The measurements of performance of 11 algorithms used to train the MLP to detect anomalies in the ISCX2012-17 dataset.

No.	Alg.	TP	TN	FN	FP	Sen.	Spe.	Pre.	Acc	Dr	Far	AR	DR	FR
1	ABCMLP-IDS	0	3238	1735	230	0.00	0.93	0.00	62.2%	0.0%	0.066	11	10	11
2	ACOMLP-IDS	14	3306	1721	162	0.01	0.95	0.08	63.8%	0.8%	0.047	10	8	9
3	ALOMLP-IDS	1735	3401	0	67	1.00	0.98	0.96	98.7%	100.0%	0.019	2	1	7
4	CSMLP-IDS	12	3468	1723	0	0.01	1.00	1.00	66.9%	0.7%	0.000	8	9	1
5	DEMLP-IDS	0	3556	1463	184	0.00	0.95	0.00	68.3%	0.0%	0.049	5	10	10
6	EHOMLP-IDS	21	3468	1714	0	0.01	1.00	1.00	67.1%	1.2%	0.000	7	6	1
7	GSAMLP-IDS	17	3400	1718	68	0.01	0.98	0.20	65.7%	1.0%	0.020	9	7	8
8	HADMLP-IDS	27	3468	1708	0	0.02	1.00	1.00	67.2%	56.0%	0.000	6	5	1
9	MFOMLP-IDS	1722	3461	13	7	0.99	1.00	1.00	99.6%	99.3%	0.002	1	2	5
10	SCAMLP-IDS	1628	3425	107	43	0.94	0.99	0.97	97.1%	93.8%	0.012	4	4	6
11	WOAMLP-IDS	1640	3468	95	0	0.95	1.00	1.00	98.2%	94.5%	0.000	3	3	1

ALOMLP-IDS and MFOMLP-IDS are here the best performers with an accuracy of 98.7% and 99.96%, respectively, a detection rate of 100% and 99.89%, respectively, and zero false alarms were recorded by these models CSMLP-IDS, EHOMLP-IDS, and WOAMLP-IDS. WOAMLP-IDS followed closely with an accuracy of 98.2%, a detection rate of 94.5% and zero false alarms. The HADMLP-IDS show relatively inferior performance compared to their previous scores and to other models with respect to this final subset of data. The ABCMLP-IDS and ACOMLP-IDS with the 17th day dataset recorded worst ACC, DR and FAR scores for ABCMLP-IDS are 62.2%, 0% and 0.066, respectively, while those for ACOMLP-IDS are 63.8%, 0.8% and 0.047, respectively. These last results suggest an important point: although the HADMLP-IDS shows less impressive results than the other models with respect to the ISCX2012 dataset, it is generally more consistent, across the various data subsets, and sometimes even the absolute best. This conclusion is also consistent with the results from the other benchmarking datasets.

Figure 9 demonstrates the comparative performance of the 11 algorithms against the whole ISCX 2012 dataset in terms of the average of ACC, DR, and FAR measurements, which is a tabular layout to visualize the performance of the supervised classifiers. The content of this matrix is the basic measurements of TP, TN, FN, and FP, which is caused by the mapping between the number of correct and wrong predictions of the classifier for both positive (attack) and negative (normal) instances of the testing data. The general template of a confusion matrix is demonstrated in Table 5.

Figure 9 demonstrates the average performance of the 11 models against the whole datasets of ISCX 2012 (ISCX 2012-12, ISCX 2012-13, ISCX 2012-14, ISCX 2012-15, and ISCX 2012-17). The figure indicates that the proposed model outperformed all other models against all datasets of type ISCX 2012. This was measured in terms of accuracy and detection rate. It was ranked second with respect to accuracy at 91.14% score, the second best with respect to detection rate at 88.96% score, and the third best with respect to false alarm rate at a score of 0.024. Moreover, the CSMLP-IDS and EHOMLP-IDS outperformed all models in terms of false alarm rate, where the CSMLP-IDS was ranked the first with score of 0.0106 and the EHOMLP-IDS was ranked the second with score of 0.021, However,

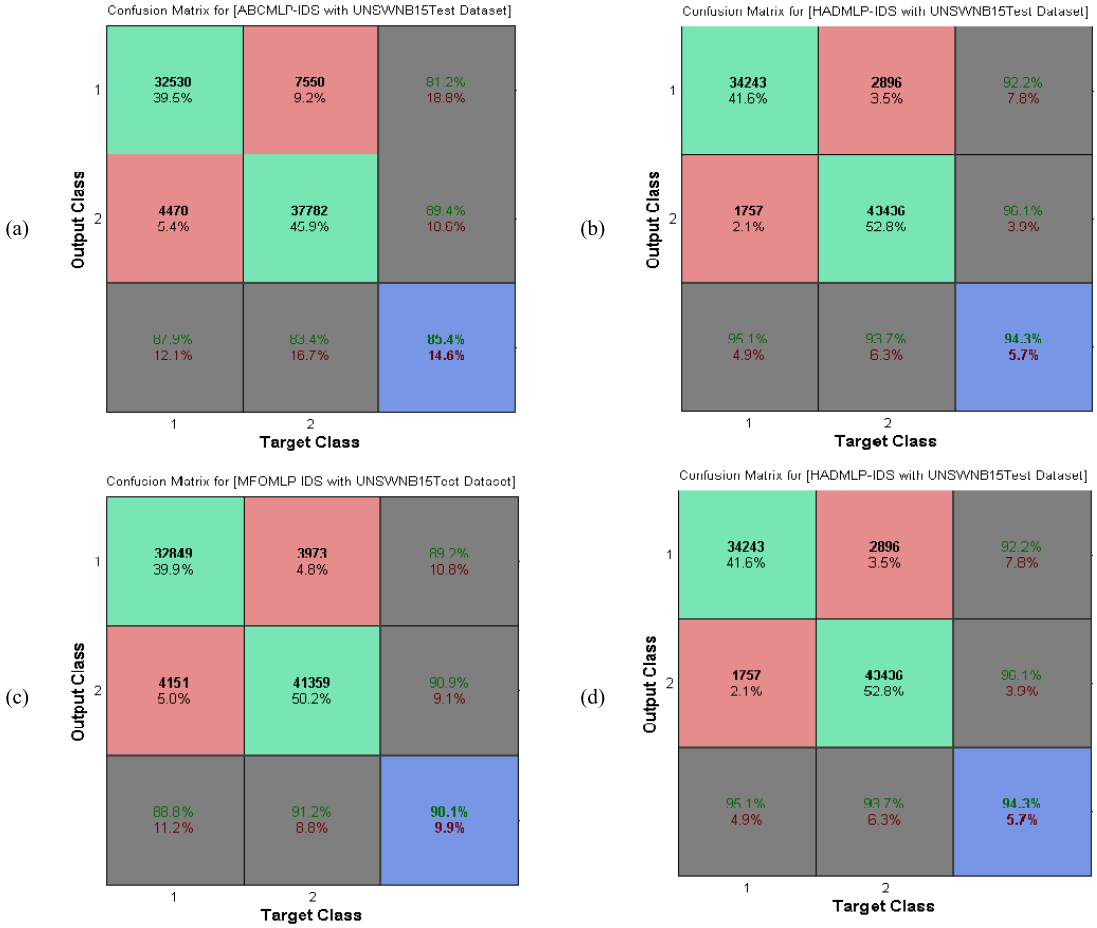

FIGURE 8. The confusion matrices against the UNSW-NB15 dataset.

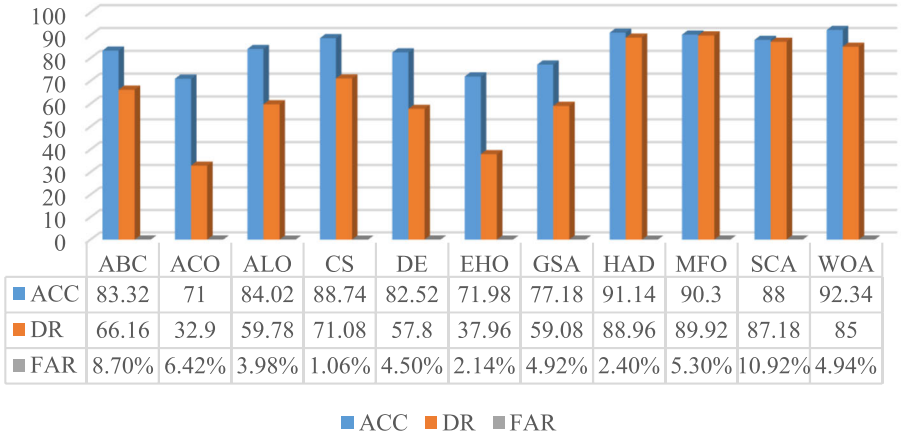

FIGURE 9. The average of the evaluation variables (ACC, FAR, and DR) of 11 MLP trainer algorithms for all ISCX 2012 dataset.

the performance of these two models in terms of accuracy and detection rate is quite poor compared to our model. We can conclude from these results that overall, the outstanding performance of our model is more stable in terms of the three measurement criteria. For example, although WOAMLP-IDS showed better ACC than our model, it was weak in terms of DR and FAR.

The confusion matrices of the best performing models are presented in Tables (14-18). Figure 10 shows the best and worst results for the proposed intrusion detector. Figure 10 (a) and (b) separately indicate the binary classification performance of the proposed IDS against the ISCX 2012-12th and ISCX 2012-13th datasets, respectively. Figure 10 (a) shows the result obtained from ISCX 2012-12th

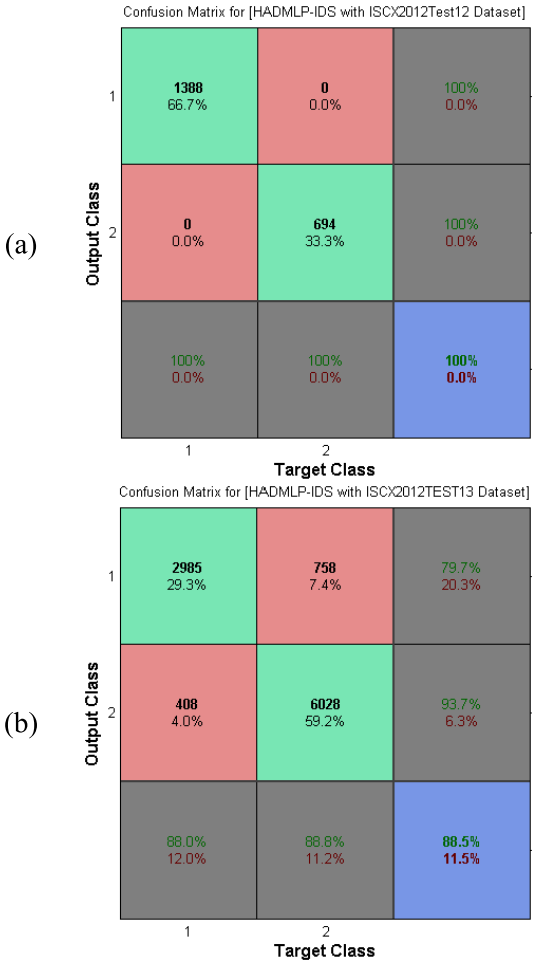

FIGURE 10. The confusion matrices against the ISCX 2012 dataset.

with HADMLP-IDS model which correctly detects 100% of all attack records and 100% of all normal records without false alarms. Figure 10 (b) shows the result obtained from ISCX 2012-13th with the HADMLP-IDS model, which detects 93.7% of all attack records and does not detect 6.3% of all attacks. In the case of normal traffic, 79.7% of all normal records are detected correctly and 20.3% of them are detected wrongly.

Figure 11 shows the convergence curves of the MSEs within 100 iterations for the ISCX 2012-12th. This figure confirms that, in terms of MSE, our proposed HADMLP algorithm has the best convergence rate and the least classification error compared with the rest of the algorithms. The algorithm for training MLPs doesn't needs only robust exploration ability, but also rigorous exploitation ability. The results of the classification accuracy, detection rate, and false positive rate obtained by HADMLP-IDS model and the rest of the models, it is shown that HADMLP-IDS performs better than the rest of the models due to the more precise exploitation ability of the HAD algorithm, while, the rest of the models still suffers from the problem of becoming trapped in local minima that means has leading to an unstable perfor-

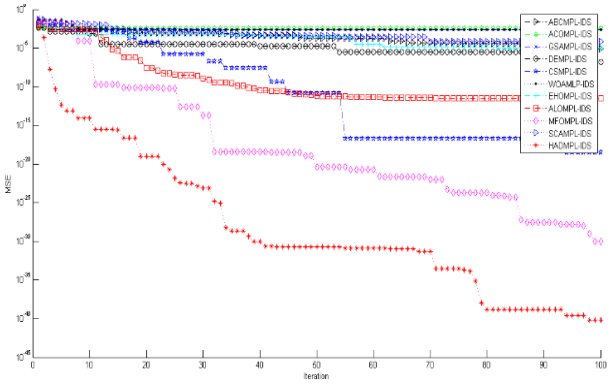

FIGURE 11. Convergence curves of all models based on averages of MSE for the benchmark classification ISCX 2012-12th day datasets.

mance. The results obtained from all ISCX 2012 datasets by the proposed model prove that it has both strong exploitation and good exploration abilities. These results mean that HADMLP-IDS is capable of solving the problem of becoming trapped in local minima and that it gives a fast convergence speed.

Therefore, the proposed method offers the best robust exploration and rigorous exploitation capabilities. The proposed training algorithm HAD is effective and feasible for application to IDS research.

E. COMPARISON OF PROPOSED METHOD WITH EXISTING STATE-OF-ARTS

The comparisons between the performance results of the proposed method with the more recent proposed techniques of intrusion detection systems from literature are listed in Table 19. This comparison shows the contribution and superiority of our method on publicly available datasets including the KDDCup 99, NSL-KDD, ISCX 2012 and UNSW-NB15 datasets. The proposed model has the best performance in terms of ACC, DR, and FAR. The data were correctly classified by the proposed approach compared to those classified by the static approaches. Moreover, HADMLP-IDS exhibited a significantly lower FAR than some of the recent state-of-arts.

VI. CONCLUSION AND FUTURE WORK

This research introduced a new model for an intrusion detection system, called the HADMLP-IDS model which is based on training MLP using a fresh hybrid metaheuristic that combines the Artificial Bee Colony (ABC) algorithm with the Dragonfly Algorithm (DA). In this work, the new model has been evaluated by using the confusion matrix based on TP, TN, FN, and FP that was obtained using the KDD Cup 99, NSL-KDD, UNSW-NB15, and ISCX2012 datasets. The performance of the new model was also assessed against a number of other intrusion detection models designed with a similar principle. In this work ten models have been used to train MLP, namely ABC, ACO, ALO, CS, DE, EHO,

TABLE 19. Comparison results with other methods.

No.	Ref	Year	Dataset	Method	ACC	DR	FAR
1	[48]	2020	UNSW NB15	1D-CNN	90.91	N/A	N/A
2	[48]	2020	UNSW NB15	1D-CNN+LSTM	89.93	N/A	N/A
3	[53]	2020	UNSW NB15	KPCA-DEGSA-HKELM	89.01	N/A	2.41
4	[54]	2018	UNSW NB15	Dendron	84.33	N/A	2.61
5	[55]	2018	UNSW NB15	CAI	82.74	N/A	36.46
6	[56]	2020	UNSW NB15	T2 Hotelling's	91.01	N/A	0.2748
7	[49]	2020	UNSW NB15	Sigmoid_PIO	91.3	N/A	0.052
8	[50]	2020	UNSW NB15	MSCNN-LSTM	89.8	N/A	0.474
9	[49]	2020	UNSW NB15	Cosine_PIO	91.7	N/A	0.034
10	[46]	2019	UNSW-NB15	SVM-rbf	65.3	N/A	N/A
11	[46]	2019	UNSW-NB15	DNN	78.4	N/A	N/A
Proposed model			UNSW-NB15	HADMLP-IDS	94.4	93.7	0.049
1	[57]	2019	ISCX 2012	AdaBoost	83	73	N/A
2	[52]	2018	ISCX 2012	ACNN	0.8351	N/A	N/A
3	[80]	2017	ISCX 2012	SLFN	N/A	88.18	5.56
4	[14]	2019	ISCX 2012	DeepFullRange	0.826	N/A	N/A
5	[51]	2019	ISCX 2012	ELM50	58.76	N/A	0.513
6	[51]	2019	ISCX 2012	MLP50	87.22	N/A	0.145
Proposed model			ISCX 2012	HADMLP-IDS	91.14	88.96	0.024
1	[49]	2020	KDDCUP 99	Sigmoid_PIO	94.7	N/A	0.097
2	[49]	2020	KDDCUP 99	Cosine_PIO	96	N/A	0.076
3	[46]	2019	KDDCUP 99	SVM-rbf	N/A	87.7	N/A
4	[46]	2019	KDDCUP 99	DNN	N/A	92.9	N/A
Proposed model			KDDCUP 99	HADMLP-IDS	88.7	90.2	0.141
1	[46]	2019	NSL-KDD	SVM-rbf	83.7	N/A	N/A
2	[46]	2019	NSL-KDD	DNN	80.1	N/A	N/A
3	[47]	2019	NSL-KDD	Deep Neural Network	0.772	N/A	N/A
4	[47]	2019	NSL-KDD	PCA + Deep Neural Network	0.793	N/A	N/A
5	[56]	2020	NSL-KDD	T2 Hotelling's	91.71	N/A	0.0624
6	[49]	2020	NSL-KDD	Sigmoid_PIO	0.869	N/A	0.064
7	[47]	2020	NSL-KDD	Cosine_PIO	0.883	N/A	0.088
8	[59]	2018	NSL-KDD	Deep belief networks	N/A	88.1	N/A
9	[58]	2020	NSL-KDD	Hybrid model	N/A	86.2	0.134
Proposed model			NSL-KDD	HADMLP-IDS	91.7	93.6	0.108

N/A: Not available

GSA, MFO, SCA, and WOA. The HADMLP-IDS model trained with the KDD Cup 99, NSL-KDD, UNSW-NB15, and ISCX2012 datasets has achieved a detection rates of 90.2%, 93.6%, 93.7%, and 89% as well as the false alarm rate of 0.141, 0.108, 0.049, and 0.024 respectively. Our model has attained better results than those obtained by other models. The result has shown the potential efficacy and capability of the model for developing practical IDSs. Nevertheless, this work has only evaluated the model using intrusion detection datasets where an adequate feature selection technique has not been included. Therefore, future work should focus on minimizing the number of selected features and application of the proposed model to develop an effective IDS.

REFERENCES

[1] J. P. Anderson, *Computer Security Threat Monitoring and Surveillance*. Fort Washington, PA, USA: James P. Anderson Co, 1980.

[2] D. E. Denning, "An intrusion-detection model," *IEEE Trans. Softw. Eng.*, vol. SE-13, no. 2, pp. 222–232, Feb. 1987.

[3] W. A. H. M. Ghanem and B. Belaton, "Improving accuracy of applications fingerprinting on local networks using NMAP-AMAP-ETTERCAP as a hybrid framework," in *Proc. IEEE Int. Conf. Control Syst., Comput. Eng.*, Nov. 2013, pp. 403–407.

[4] J. Kevric, S. Jukic, and A. Subasi, "An effective combining classifier approach using tree algorithms for network intrusion detection," *Neural Comput. Appl.*, vol. 28, no. S1, pp. 1051–1058, Dec. 2017.

[5] A. Patcha and J.-M. Park, "An overview of anomaly detection techniques: Existing solutions and latest technological trends," *Comput. Netw.*, vol. 51, no. 12, pp. 3448–3470, Aug. 2007.

[6] H. Lu, R. Setiono, and H. Liu, "Effective data mining using neural networks," *IEEE Trans. Knowl. Data Eng.*, vol. 8, no. 6, pp. 957–961, 1996.

[7] O. I. Abiodun, A. Jantan, A. E. Omolara, K. V. Dada, A. M. Umar, O. U. Linus, H. Arshad, A. A. Kazaure, U. Gana, and M. U. Kiru, "Comprehensive review of artificial neural network applications to pattern recognition," *IEEE Access*, vol. 7, pp. 158820–158846, 2019.

[8] R. Agrawal, T. Imielinski, and A. Swami, "Database mining: A performance perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 5, no. 6, pp. 914–925, Dec. 1993.

[9] F.-Y. Leu, K.-L. Tsai, Y.-T. Hsiao, and C.-T. Yang, "An internal intrusion detection and protection system by using data mining and forensic techniques," *IEEE Syst. J.*, vol. 11, no. 2, pp. 427–438, Jun. 2017.
[10] C. Zhang, J. Jiang, and M. Kamel, "Comparison of BPL and RBF network in intrusion detection system," in *Proc. Int. Workshop Rough Sets, Fuzzy Sets, Data Mining Granular-Soft Comput.*, Berlin, Germany: Springer, 2003, pp. 466–470.
[11] J. Jiang, C. Zhang, and M. Kamel, "RBF-based real-time hierarchical intrusion detection systems," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2003, pp. 1512–1516.
[12] M. Alauthaman, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "A P2P botnet detection scheme based on decision tree and adaptive multilayer neural networks," *Neural Comput. Appl.*, vol. 29, no. 11, pp. 991–1004, Jun. 2018.
[13] A. K. Ghosh and A. Schwartzbard, "A study in using neural networks for anomaly and misuse detection," in *Proc. USENIX Secur. Symp.*, vol. 99, 1999, p. 12.
[14] H. Li, "Research on prediction of traffic flow based on dynamic fuzzy neural networks," *Neural Comput. Appl.*, vol. 27, no. 7, pp. 1969–1980, Oct. 2016.
[15] K. L. Fox, R. R. Henning, J. H. Reed, and R. P. Sitnonian, "A Neural network approach towards intrusion detection," Harris Corp., Government Inf. Syst. Division, Tech. Rep. FL32902, 1990.
[16] W. Wang, X. Guan, X. Zhang, and L. Yang, "Profiling program behavior for anomaly intrusion detection based on the transition and frequency property of computer audit data," *Comput. Secur.*, vol. 25, no. 7, pp. 539–550, Oct. 2006.
[17] H. Yang and F. Wang, "Wireless network intrusion detection based on improved convolutional neural network," *IEEE Access*, vol. 7, pp. 64366–64374, 2019.
[18] K. Li and G. Teng, "Unsupervised SVM based on p-kernels for anomaly detection," in *Proc. 1st Int. Conf. Innov. Comput., Inf. Control (ICICIC)*, vol. 1, Aug. 2006, pp. 59–62.
[19] B. M. Aslahi-Shahri, R. Rahmani, M. Chizari, A. Maralani, M. Eslami, M. J. Golkar, and A. Ebrahimi, "A hybrid method consisting of GA and SVM for intrusion detection system," *Neural Comput. Appl.*, vol. 27, no. 6, pp. 1669–1676, Aug. 2016.
[20] C. A. Catania, F. Bromberg, and C. G. Garino, "An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection," *Expert Syst. Appl.*, vol. 39, no. 2, pp. 1822–1829, Feb. 2012.
[21] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," *Inf. Sci.*, vol. 177, no. 18, pp. 3799–3821, Sep. 2007.
[22] Z. Zhang and H. Shen, "Online training of SVMs for real-time intrusion detection," in *Proc. 18th Int. Conf. Adv. Inf. Netw. Appl. AINA*, Mar. 2004, pp. 568–573.
[23] D. Martens, B. Baesens, and T. Fawcett, "Editorial survey: Swarm intelligence for data mining," *Mach. Learn.*, vol. 82, no. 1, pp. 1–42, 2001.
[24] C. Kolias, G. Kambourakis, and M. Maragoudakis, "Swarm intelligence in intrusion detection: A survey," *Comput. Secur.*, vol. 30, no. 8, pp. 625–642, Nov. 2011.
[25] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2006, pp. 1–738.
[26] C. Ozturk and D. Karaboga, "Hybrid artificial bee colony algorithm for neural network training," in *Proc. IEEE Congr. Evol. Comput. (CEC)*, Jun. 2011, pp. 84–88.
[27] X. Yao, "Evolving artificial neural networks," *Proc. IEEE*, vol. 87, no. 9, pp. 1423–1447, Sep. 1999.
[28] M. Sheikhan and M. Sharifi Rad, "Gravitational search algorithm-optimized neural misuse detector with selected features by fuzzy grids-based association rules mining," *Neural Comput. Appl.*, vol. 23, nos. 7–8, pp. 2451–2463, Dec. 2013.
[29] K. Socha and C. Blum, "An ant colony optimization algorithm for continuous optimization: Application to feed-forward neural network training," *Neural Comput. Appl.*, vol. 16, no. 3, pp. 235–247, May 2007.
[30] K. Mehrotra, C. K. Mohan, and S. Ranka, *Elements of Artificial Neural Networks*. Cambridge, MA, USA: MIT Press, 1997.
[31] D. R. Hush and B. G. Horne, "Progress in supervised neural networks," *IEEE Signal Process. Mag.*, vol. 10, no. 1, pp. 8–39, Jan. 1993.
[32] D. Karaboga, B. Akay, and C. Ozturk, "Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks," in *Proc. Int. Conf. Modeling Decisions Artif. Intell.*, Berlin, Germany: Springer, 2007, pp. 318–329.
[33] M. Carvalho and T. B. Ludermir, "Hybrid training of feed-forward neural networks with particle swarm optimization," in *Proc. Int. Conf. Neural Inf. Process.*, Berlin, Germany: Springer, 2006, pp. 1061–1070.
[34] M. H. Ali, B. A. D. Al Mohammed, A. Ismail, and M. F. Zolkipli, "A new intrusion detection system based on fast learning network and particle swarm optimization," *IEEE Access*, vol. 6, pp. 20255–20261, 2018.
[35] A. Yang, Y. Zhuansun, C. Liu, J. Li, and C. Zhang, "Design of intrusion detection system for Internet of Things based on improved BP neural network," *IEEE Access*, vol. 7, pp. 106043–106052, 2019.
[36] M. Moradi and M. Zulkernine, "A neural network based system for intrusion detection and classification of attacks," in *Proc. IEEE Int. Conf. Adv. Intell. Syst.*, Nov. 2004, pp. 15–18.
[37] W. A. H. M. Ghanem and A. Jantan, "A cognitively inspired hybridization of artificial bee colony and dragonfly algorithms for training multi-layer perceptrons," *Cognit. Comput.*, vol. 10, no. 6, pp. 1096–1134, Dec. 2018.
[38] V. K. Ojha, A. Abraham, and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Eng. Appl. Artif. Intell.*, vol. 60, pp. 97–116, Apr. 2017.
[39] T. Wang, L. Wei, and J. Ai, "Improved BP neural network for intrusion detection based on AFSA," in *Proc. Int. Symp. Comput. Informat.*, Jan. 2015, pp. 373–380.
[40] L. Shi, Y. Yang, and J. Lv, "PCA-PSO-BP neural network application in IDS," in *Proc. Int. Power, Electron. Mater. Eng. Conf.*, May 2015, pp. 1–6.
[41] M. Sheikhan and Z. Jadidi, "Flow-based anomaly detection in high-speed links using modified GSA-optimized neural network," *Neural Comput. Appl.*, vol. 24, nos. 3–4, pp. 599–611, Mar. 2014.
[42] W. Tian and J. Liu, "Network intrusion detection analysis with neural network and particle swarm optimization algorithm," in *Proc. Chin. Control Decis. Conf.*, May 2010, pp. 1749–1752.
[43] L. Wang, G. Yu, G. Wang, and D. Wang, "Method of evolutionary neural network-based intrusion detection," in *Proc. Int. Conf. Info-Tech Info-Net.*, Oct. 2001, pp. 13–18.
[44] R. Xu, R. An, and X. Geng, "Research intrusion detection based PSO-RBF classifier," in *Proc. IEEE 2nd Int. Conf. Softw. Eng. Service Sci.*, Jul. 2011, pp. 104–107.
[45] Y. Zeng, H. Gu, W. Wei, and Y. Guo, "*Deep − full − range* : A deep learning based network encrypted traffic classification and intrusion detection framework," *IEEE Access*, vol. 7, pp. 45182–45190, 2019.
[46] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al-Nemrat, and S. Venkatraman, "Deep learning approach for intelligent intrusion detection system," *IEEE Access*, vol. 7, pp. 41525–41550, 2019.
[47] S. Rawat, A. Srinivasan, and V. R, "Intrusion detection systems using classical machine learning techniques versus integrated unsupervised feature learning and deep neural network," 2019, *arXiv:1910.01114*. [Online]. Available: http://arxiv.org/abs/1910.01114
[48] M. Azizjon, A. Jumabek, and W. Kim, "1D CNN based network intrusion detection with normalization on imbalanced data," in *Proc. Int. Conf. Artif. Intell. Inf. Commun. (ICAIIC)*, Feb. 2020, pp. 218–224.
[49] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer," *Expert Syst. Appl.*, vol. 148, Jun. 2020, Art. no. 113249.
[50] J. Zhang, Y. Ling, X. Fu, X. Yang, G. Xiong, and R. Zhang, "Model of the intrusion detection system based on the integration of spatial-temporal features," *Comput. Secur.*, vol. 89, Feb. 2020, Art. no. 101681.
[51] M. Monshizadeh, V. Khatri, B. G. Atli, R. Kantola, and Z. Yan, "Performance evaluation of a combined anomaly detection platform," *IEEE Access*, vol. 7, pp. 100964–100978, 2019.
[52] Y. Yang, C. Kang, G. Gou, Z. Li, and G. Xiong, "TLS/SSL encrypted traffic classification with autoencoder and convolutional neural network," in *Proc. IEEE 20th Int. Conf. High Perform. Comput. Commun., IEEE 16th Int. Conf. Smart City, IEEE 4th Int. Conf. Data Sci. Syst. (HPCC/SmartCity/DSS)*, Jun. 2018, pp. 362–369.
[53] L. Lv, W. Wang, Z. Zhang, and X. Liu, "A novel intrusion detection system based on an optimal hybrid kernel extreme learning machine," *Knowl.-Based Syst.*, vol. 195, May 2020, Art. no. 105648.
[54] D. Papamartzivanos, F. Gómez Mármol, and G. Kambourakis, "Dendron : Genetic trees driven rule induction for network intrusion detection systems," *Future Gener. Comput. Syst.*, vol. 79, pp. 558–574, Feb. 2018.
[55] C.-R. Wang, R.-F. Xu, S.-J. Lee, and C.-H. Lee, "Network intrusion detection using equality constrained-optimization-based extreme learning machines," *Knowl.-Based Syst.*, vol. 147, pp. 68–80, May 2018.

[56] M. Ahsan, M. Mashuri, M. H. Lee, H. Kuswanto, and D. D. Prastyo, "Robust adaptive multivariate Hotelling's t2 control chart based on kernel density estimation for intrusion detection system," *Expert Syst. Appl.*, vol. 145, May 2020, Art. no. 113105.
[57] M. Mazini, B. Shirazi, and I. Mahdavi, "Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 31, no. 4, pp. 541–553, Oct. 2019.
[58] B. Rababah and S. Srivastava, "Hybrid model for intrusion detection systems," 2020, *arXiv:2003.08585*. [Online]. Available: http://arxiv.org/abs/2003.08585
[59] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 1, pp. 41–50, Feb. 2018.
[60] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2016.
[61] W. A. H. Ghanem and A. Jantan, "New approach to improve anomaly detection using a neural network optimized by hybrid abc and Pso algorithms," *Pakistan J. Statist.*, vol. 34, no. 1, pp. 1–14, 2018.
[62] A. Jantan, W. A. H. Ghanem, and S. A. Ghaleb, "Using modified bat algorithm to train neural networks for spam detection," *J. Theor. Appl. Inf. Technol.*, vol. 95, no. 24, pp. 1–12, 2017.
[63] W. A. H. Ghanem and A. Jantan, "Using hybrid artificial bee colony algorithm and particle swarm optimization for training feed-forward neural networks," *J. Heoretical Appl. Inf. Technol.*, vol. 67, no. 3, pp. 1–11, 2014.
[64] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," *ACM Trans. Inf. Syst. Secur. (TISSEC)*, vol. 3, no. 4, pp. 227–261, Nov. 2000.
[65] S. Ganapathy, K. Kulothungan, S. Muthurajkumar, M. Vijayalakshmi, P. Yogesh, and A. Kannan, "Intelligent feature selection and classification techniques for intrusion detection in networks: A survey," *EURASIP J. Wireless Commun. Netw.*, vol. 2013, no. 1, Dec. 2013.
[66] D. S. Terzi, R. Terzi, and S. Sagiroglu, "Big data analytics for network anomaly detection from netflow data," in *Proc. Int. Conf. Comput. Sci. Eng. (UBMK)*, Oct. 2017, pp. 592–597.
[67] A. Zainal, M. A. Maarof, and S. M. Shamsuddin, "Feature selection using rough-DPSO in anomaly intrusion detection," in *Proc. Int. Conf. Comput. Sci. Appl.*, Berlin, Germany: Springer, 2007, pp. 512–524.
[68] Z. A. Othman, Z. Muda, L. M. Theng, and M. R. Othman, "Record to record feature selection algorithm for network intrusion detection," *Int. J. Advancements Comput. Technol.*, vol. 6, no. 2, p. 163, 2014.
[69] H. H. Jebur, M. A. Maarof, and A. Zainal, "Identifying generic features of KDD cup 1999 for intrusion detection," *JurnalTeknologi*, vol. 74, no. 1, pp. 1–9, 2015.
[70] W. A. H. M. Ghanem and A. Jantan, "A new approach for intrusion detection system based on training multilayer perceptron by using enhanced bat algorithm," *Neural Comput. Appl.*, 2019. [Online]. Available: https://doi.org/10.1007/s00521-019-04655-2
[71] W. A. H. M. Ghanem and A. Jantan, "Training a neural network for cyberattack classification applications using hybridization of an artificial bee colony and monarch butterfly optimization," *Neural Process. Lett.*, vol. 51, no. 1, pp. 905–946, Feb. 2020.
[72] L. M. Ibrahim, D. T. Basheer, and M. S. Mahmod, "A comparison study for intrusion database (KDD99, NSL-KDD) based on self-organization map (SOM) artificial neural network," *J. Eng. Sci. Technol.*, vol. 8, no. 1, pp. 107–119, 2013.
[73] G. Kumar and K. Kumar, "A multi-objective genetic algorithm based approach for effective intrusion detection using neural networks," in *Intelligent Methods for Cyber Warfare*. Cham, Switzerland: Springer, 2015, pp. 173–200.
[74] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1–6.
[75] M. Ahmed, A. Naser Mahmood, and J. Hu, "A survey of network anomaly detection techniques," *J. Netw. Comput. Appl.*, vol. 60, pp. 19–31, Jan. 2016.
[76] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Comput. Secur.*, vol. 31, no. 3, pp. 357–374, May 2012.
[77] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, Nov. 2015, pp. 1–6.
[78] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set," *Inf. Secur. J., Global Perspective*, vol. 25, nos. 1–3, pp. 18–31, Apr. 2016.
[79] N. Moustafa and J. Slay, "The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems," in *Proc. 4th Int. Workshop Building Anal. Datasets Gathering Exper. Returns Secur. (BADGERS)*, Nov. 2015, pp. 25–31.
[80] H. Huang, Y. Cai, and H. Yu, "Distributed-neuron-network based machine learning on smart-gateway network towards real-time indoor data analytics," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, 2016, pp. 231–263.

WAHEED ALI H. M. GHANEM received the B.Sc. degree in computer sciences and engineering from Aden University, Yemen, in 2003, and the M.Sc. degree in computer science and the Ph.D. degree in network and communication protocols from Universiti Sains Malaysia, in 2013 and 2019, respectively. His research interests include computer and network security, cybersecurity, machine learning, artificial intelligence, swarm intelligence, optimization algorithm, and information technology.

AMAN JANTAN is currently an Associate Professor with the School of Computer Sciences, Universiti Sains Malaysia, Malaysia. He has published more than 100 articles in reputed journals. His research interests include digital forensic, artificial intelligence, malware, intrusion detection systems, computer security, cryptography, and computer and network security. He received national and international recognition in some of his work.

SANAA ABDULJABBAR AHMED GHALEB received the bachelor's degree from Aden University, Yemen, in 2011, the master's degree from Universiti Sains Malaysia, Malaysia, in 2017. She is currently pursuing the Ph.D. degree with the Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin. Her general research interests include technology-enhanced learning, and instructional design and technology. Her research interests include computer and network security, cybersecurity, machine learning, artificial intelligence, swarm intelligence, and optimization algorithm.

ABDULLAH B. NASSER received the B.Sc. degree from Hodeidah University, Yemen, in 2006, the M.Sc. degree from Universiti Sains Malaysia, Malaysia, in 2014, and the Ph.D. degree from University Malaysia Pahang, in 2018, all in computer science. He is currently a Senior Lecturer with Universiti Malaysia Pahang. His research interests include software testing and optimization algorithms.