Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 6648351, 12 pages
https://doi.org/10.1155/2021/6648351

Research Article

Network Intrusion Detection System Based on the Combination of Multiobjective Particle Swarm Algorithm-Based Feature Selection and Fast-Learning Network

Sajad Einy,[1,2] **Cemil Oz**,[1] **and Yahya Dorostkar Navaei**[3]

[1]*Computer Engineering Department, Sakarya University, Turkey*
[2]*Application and Research Center for Advanced Studies, Istanbul Aydin University, Turkey*
[3]*Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran*

Correspondence should be addressed to Yahya Dorostkar Navaei; y.dorostkar@qiau.ac.ir

Received 19 November 2020; Revised 8 May 2021; Accepted 2 June 2021; Published 16 June 2021

Academic Editor: Yanhui Guo

Given the growth of wireless networks and the increase of the advantages and applications of communication networks, especially mobile ad hoc networks (MANETs), this type of network has attracted the attention of users and researchers more than before. The benefit of these types of networks in various kinds of networks and environments is that MANET does not require to hardware infrastructure to communicate and send and receive data packets within the network. It is one of the main reasons for using these MANET in various fields. On the other hand, the increased popularity of these networks has led to many challenges, one of the most important of which is network security. In this regard, a lack of regulatory and security infrastructure in MANETs has caused some problems in sending and receiving data, where intrusion in the network has been recognized as one of the most important issues. In MANETs, wireless notes act as a link between the source and destination nodes and play the role of relays and routers in the network. Therefore, malicious node penetration and the destruction of information packages become feasible. Today, intrusion detection systems (IDSs) are used as a solution to deal with the problem through remote monitoring of the performance and behaviors of nodes existing in wireless sensor networks. In addition to detecting malicious nodes in the network, IDSs can predict the behavior of malicious nodes in the future in most cases. Therefore, the present study introduced a network IDS (NIDS) entitled MOPSO-FLN by using a combination of multiobjective particle swarm optimization algorithm-(MOPSO-) based feature subset selection (FSS) and fast-learning network (FLN). In this work, we used the KDD Cup99 and dataset to select features, train the network, and test the model. According to the simulation results, this method was able to improve the performance of the IDS in terms of evaluation criteria, compared to other previous methods, by creating a balance between the objectives of the number of representative features and training errors based on the evolutionary power of MOPSO.

1. Introduction

Mobile ad hoc networks (MANETs) are a group of mobile nodes that communicate over wireless links without any backbone. MANETs do not have centralized control mechanism, and each mobile node acts as routers for transferring data packages to other specific nodes of the network in addition to being a terminal in the network [1]. Security is a paramount concern in MANETS due to having a dynamic topology and nodes that can easily enter or exit the network at any time and have access to data flow throughout the network. In addition, several mobile nodes are limited to resources in terms of computational power and energy source [2, 3]. As such, the presence of permanent security monitoring nodes in the network is almost impossible due to limited resources, and there is a need for remote control of nodes' behavior in the network and determining security necessities in MANET [4, 5].

Network intrusion detection systems (NIDS) are used to monitor node activity or network traffic activity. The main goal of NIDSs is detecting malicious nodes and predicting possible future attacks on the network [6]. An alert is

generated for further action when detecting a malicious node in the network. Various techniques have been proposed for detecting attacks by NIDS, and it is notable that an IDS's success depends on the type of technique used in this regard [7]. One of the key factors in NIDS performance is the selection of representative features from the main dataset [8]. Reducing the number of features existing in the data set (e.g., the behavior of nodes and network traffic) without affecting the classification precision can play an important role in IDS performance optimization [9].

In the proposed method, the feature selection approach based on MOPSO (multiobjective particle swarm optimization) is responsible for selecting representative features of the main dataset. The selected features are entered into a fast-learning network (FLN) as a solution. Using rapid training, FLN evaluates solutions and determines the model's error based on the selected features. The main goal of the present study was eliminating unrelated features and attributes and plugins to reduce the dimension of the data and complexity of the model while increasing its classification accuracy in determining the model of malicious nodes and network attacks at a higher pace. Therefore, the proposed method included the feature selection approach based on MOPSO to determine important features.

In this technique, features are considered as the primary particles of a multiobjective particle swarm optimization algorithm (PSO). In addition, the target function is applied to minimize the number of features used and the class error in the proposed method. Moreover, the primary particles are selected as a subset of the entire features in the database. The particles are sent to FLN in MOPSO as a solution in order to estimate the value of the fit function. The particles with the smallest function value in each repetition of the MOPSO algorithm are selected as expert particles and the optimal solutions in that round and are stored in the repository of solutions. In the next round, the location and speed of other particles are updated in line with the tendency towards expert particles. At the end of the algorithm, the expert particles that were stored in the repository of solutions are sorted by the value of the fit function, and a solution with the smallest value is selected as the optimal solution. Moreover, the features determined by the optimal solution are determined as the behavioral pattern of malicious nodes and used to predict future malicious nodes in the network.

2. Related Works

The need for high-speed services in providing network services has always been a necessity of networks and no further emphasis is required in this regard. NIDS is an important tool for protecting the network [1]. These systems analyze the entry paths of nodes into the system regarding protected systems and decide whether the entry routes contain nodes from an attack or not [10, 11]. NIDS raises an alert in case of detecting an attack. Therefore, this part of the article is dedicated to the assessment of some NIDSs considered in publications.

A monitored NIDS is a system that can learn from training samples during previous attacks to detect new attacks. Application of intrusion detection based on artificial neural networks (ANNs) is effective for reducing the number of samples that are classified falsely (false positive negative or false negative) owing to ANNs' ability to learn from actual samples. In 2019, Ali et al. proposed a developed training model for neural networks entitled a fast-learning network (FLN). The method is presented based on feature selection according to optimized PSO with the title of PSO-FLN [12]. Selvakumar and Muneeswaran (2019) developed a feature subset selection (FSS) approach with the use of a firefly algorithm, which affects the speed of classification model analysis [13]. In 2018, Chiba et al. proposed a NIDS based on the backpropagation neural network (BPNN) using an advanced learning algorithm. These scholars applied a new architecture for the network, the function of which affected anomaly detection, including feature selection, normalization of the data, architecture of the neural network, and activation function [14]. Al-Azam et al. (2020) proposed a wrapper-based feature selection algorithm for IDS, which used a feature selection pigeon optimizer algorithm. A new method has been offered for achieving a feature selection optimization technique to be combined with classification methods and has been compared with traditional feature selection methods based on collective intelligence algorithms [15]. In 2020, Zhou et al. presented a new framework based on feature selection and group learning techniques to detect intrusion. In this method, a metaheuristic algorithm entitled CFS-BA is suggested for reducing sizes. Afterwards, a group classification approach is combined with C4.5, random forest (RF), and feature penalty-based forest (Forest PA), and the classification voting method was applied to detect the attack [16]. In 2010, Senthilnayaki et al. presented a new feature selection algorithm using the max-dependency max-significance algorithm. The algorithm is used to select a minimum number of features from the data set. In addition, a new algorithm is proposed based on k-nearest neighbors to classify datasets [17].

3. Proposed Method

In this article, a method is proposed for NIDS based on the combination of MOPSO and FLN-based FSS. The proposed method used the KDD Cup 99 dataset to determine intrusion detection patterns in the network and evaluate the model. Features are primarily selected to decrease classification difficulty and increase classification accuracy by selecting related features. In single-objective feature selection tasks, feature selection has a goal for optimization. Feature selection is mainly carried out to find the best combination of features for the most optimal classification performance. Multiobjective feature selection (MOFS) is responsible for feature selection by turning it into a multiobjective optimization problem, where the goal is to create a balance for optimizing multiple goals. The main objectives of this optimization method for FSS include reducing the number of features based on the class label and decreasing attack detection errors in the network, which will increase the accuracy of predicting test samples. As a result, a solution for the multiobjective feature selection optimization problem is a set of dominant

solutions, in which each solution is a vector of two components, number of features, and classification error rate. The goal was to minimize the number of unrelated features by using the feature selection problem as a minimization issue, thereby minimizing the classification error rate. The proposed method is formulated below. The flowchart of the proposed method is shown in Figure 1.

3.1. Formulation of Proposed Method. Kennedy and Eberhart first reported PSO in 1995, inspired by flock behavior. The main objective of PSO is to find the optimal solution in the search space of a target function similar to a flock's search pattern in the quest for the best food source. A set of generated particles randomly search for the best solutions in PSO. In this regard, a search is carried out by particles' adjustment of their own flight speed and direction based on Equations (1) and (2), respectively [18].

$$x_{id}(\mathrm{t}+1) = x_{id}(t) + v_{id}(t+1), \tag{1}$$

$$\begin{aligned} v_{id}(t+1) = {} & w \times v_{id}(t) + r_1 \times c_1 \times [p_{id}(t) - x_{id}(t)] \\ & + r_2 \times c_2 \times [g_{id}(t) - x_{id}(t)], \end{aligned} \tag{2}$$

where id is the number of dimensions, w is the inertia weight, which controls particle exploration, r_1 and r_2 are random numbers in the range of zero-one ($r_1, r_2 \in [0, 1]$), and c_1 and c_2 are acceleration constants used to control the effect of personal and overall best particles. In addition, P_{id} is the best personal position for a particle (P_{best}), and g_{id} shows the best overall position found by neighbors ($\mathrm{g}_{\mathrm{best}}$).

Obviously, PSO has high-speed convergence ability in single-objective problems, which is a favorable choice for MOPs. The Pareto-optimal solution is used in the design of MOPSO to generate a set of leaders who control the direction of particle flight and direct the search process toward optimal condition. In addition, the dominant solutions found are stored in the overall external memory (called repository) and are later used by particles as global leaders. The global guidance is selected using the roulette wheel method and based on hypercubes. Moreover, MOPSO adopts a geography-based strategy to maintain solution diversity.

In fact, the external repository "archive" includes two sections: a controller and a network. The goal of the controller is deciding whether a new solution can be added to the archive or not; updating or pruning the archive depends on the dominant relationship. Nevertheless, an adaptive network method is called whenever the archive is full. In contrast, the network is used to promote diversity among solutions. In fact, the target space is divided into areas known as a hypercube. In general, the hypercube is geographical regions encompassing a number of solutions created based on target functions. A fit function is assigned to each hypercube based on the number of existing particles. Therefore, hypercubes with a very large number of particles have less fit value. A hypercube is selected using the roulette wheel method, followed by selecting a random particle from the hypercube. Therefore, the network facilitates the process of selecting solutions in low-population areas in the target space, compared to samples locating in swarm regions [18, 19]. The speed updating function is as follows:

$$\begin{aligned} v_{id}(t+1) = {} & w \times v_{id}(t) + r_1 \times c_1 \times [p_{id}(t) - x_{id}(t)] \\ & + r_2 \times c_2 \times [REP(h) - x_{id}(t)]. \end{aligned} \tag{3}$$

In Equation (3), REP(h) is a dominant solution for selection from the repository, where the index h is selected based on the value of the fit function of hypercubes. For example, we can consider h as a set of features whose REP(h) is the accuracy of classification problem with these features.

In this research, the feature selection problem was considered as a multiobjective optimization issue resolved by using a MOPSO. Accordingly, the number of features selected shows the dimensions of the problem independently with a binary search space in the range of zero-one. Given the fact that MOPSO is an initial population-based metaheuristic method, the primary population complies with the potential subset of features that are directly related to class labels. In the two-dimensional search space in the problem, the first dimension of x_1 is a real positive number that shows the error rate, whereas the second dimension of x_2 is a real positive number that exhibits the number of features. In addition, F function leads to a balanced set of decision-making vectors, which reduce both error rate and the number of features presented by $[F(x_1), F(x_2)]$.

In the proposed method, the initial population is adjusted to the features existing in the KDD-CUP dataset. Therefore, each particle is a binary vector, the element of which is equal to the number of features, and each element refers to one feature in the dataset. Therefore, each particle existing in the swarm shows feature selection with a value of one. As such, the length of a particle is equal to the number of features existing in the dataset. Figure 2 exhibits an example of a representation of a particle, where the number of features existing in the dataset was estimated at 41.

As shown, each particle in PSO is considered a set of features existing in the dataset. In this regard, a number of elements of the vector can be randomly zero and one. The elements with zero value indicate unselected features, whereas elements with one value show feature selection related to the element. As a result, it is clear that the selected subset of features includes the feature set of $[F_2, F_6, F_8, F_{41}]$.

In the proposed method, the transfer function of Sigmoid (S) was used to define the probability of feature selection or lack of selection to select features for primary particle vectors. The function of this function is such that if the random probability is less than the threshold value of the transfer function, which is generally considered to be 0.5, the value of zero will be allocated to this feature in the related element. Otherwise, the value one will be recorded for the features, and the desired feature will be assessed based on the objective function. In this method, sigmoid is used as a random function to select or not to select a feature in a solution. So, we used this threshold value to remove features that are less than half probability to be selected and features that are more than half probability to be selected. The primary location and velocity of each particle are determined by assessment functions following selecting the initial population based on the nature

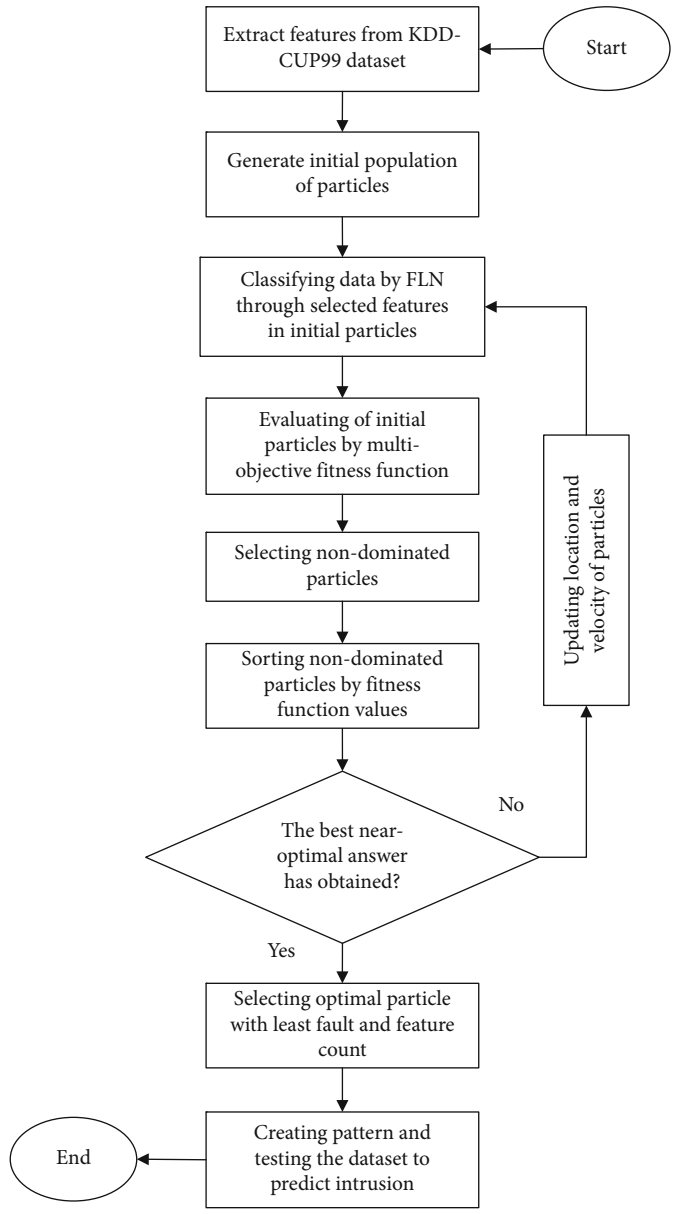

Figure 1: Flowchart of proposed method.

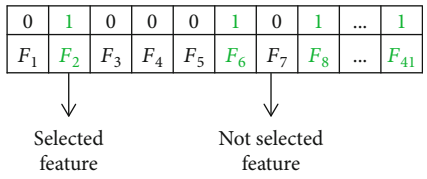

Figure 2: A representation of the initial population vector of the dataset sample.

of PSO. In this technique, the location of each particle is considered as features selected from the features existing in the dataset, and the velocity of each particle is regarded as the convergence rate to high classification rate and reduction of classification error. The features with the highest assessment function value and higher surrounding particle swarm are selected as the output of the primary feature selection stage. The best particle location and velocity results are stored at this stage and the particle position is updated. The process continues until reaching a final response that creates a balance between goals.

3.2. Proposed Objective Function. As mentioned before, the proposed method applied the MOPSO to select a subset of features selected for the class label. In this technique, multiple objectives are combined, and two general categories of features are obtained in the end in the form of minimization. Moreover, the selected subsets of features are assessed based on two main objectives of reducing the number of features and classification error rates. The fit function is presented in the form of Equation (4) in line with the evaluation of the initial population, selection of the population of experts,

and finding particles with the largest weight.

$$\text{Minimize } F(x) = \begin{cases} f_1(x) = \dfrac{L}{A}, & L \in A,\ A \in \mathbb{R}^+, \\ f_2(x) = 1 - \dfrac{FP + FN}{P + N}, & (P + N) \in \mathbb{R}^+, \end{cases} \tag{4}$$

where L is the number of features selected from datasets and A is the total number of features. The criteria related to the confusion matrix was used to evaluate the error rate of each particle based on the features selected in each step, where true positive (TP) is used to show the category of normal nodes that are accurately detected to be normal by the classification model and based on the selected features. In addition, false positive (FP) is applied to demonstrate the category of normal nodes that are falsely detected as an intrusion by the classification model and based on the selected features. Moreover, true negative (TN) is used to show the category of intrusion nodes that are accurately detected by the classification model and based on the selected features. Finally, false negative (FN) is applied to indicate the category of intrusion nodes that are falsely detected to be normal by the classification model and based on the selected features. In Equations (3)–(11), P is equal to TP + FN, and N is equal to FP + TN. The first target function $f_1(x)$ is related to the ratio of selected features to the total features existing in the dataset, whereas the second function $f_2(x)$ is used to evaluate the classification error rate.

3.3. FLN-Based Classification. FLN is a parallel connection from a leading single-layer network and a three-layer neural network containing three inputs, hidden and output layers. In general, FLN is an artificial neural network, which is a double parallel forward neural network (DPFNN), where the classification error rate is estimated using an analysis approach called the least-square technique [12]. This describes a multilayer parallel connection and a single-layer neural network. As discussed earlier, optimal particles in MOPSO show the subset of features selected from the dataset, which are entered into FLN as input. In the proposed method, FLN is located at the core of MOPSO and is responsible for estimating the classification errors of solutions. In FLN, the coded solutions are transferred to the middle layer through the input layer, where they are weighted and trained. The main difference between FLN and neural networks is the lack of waiting for training weights by all hidden players in FLN and transferring bias amounts and weights to the output layer in each layer of hidden layers where the amount of error is less than the specified threshold. This allows an increase in the speed of learning and classification of solutions in addition to preventing overfitting in the network.

In the proposed method, initial solutions that are randomly selected in MOPSO are classified by FLN in addition to assessing the number of features and the importance of selected features in the fitness function, and their error value is specified as the second goal in the multiobjective fitness function. The particles with the lowest number of important features and lowest classification error rates are selected as expert particles and nondominated solutions in each stage and are stored in the expert solutions repository. In the next stage of the proposed algorithm, the error threshold amount is considered based on the error rate of optimal particles in the previous phase. The solutions entered into FLN in each hidden layer that applies to the threshold condition are directed toward the exit.

As mentioned, the proposed method uses fast-learning networks to identify malicious nodes. Fast-learning networks are expanded from artificial neural networks, so, it has inherited the training properties of the model. In neural networks, for the purpose of classifying or detecting malicious nodes, each input feature is examined, and a weight is assigned to each input feature. These weights in each neuron indicate the importance of the value of that feature in determining malicious node. Thus, the amount of weights in neurons is considered as a pattern for identifying malicious nodes. Given that the nodes are in two classes, normal and malicious nodes, a pattern is created for each class by the proposed fast-learning networks.

In the proposed method, the fitness function is used to select a feature subset in the training set. Once, the optimal feature subset in the training data set has been selected, and the minimum amount of intrusion detection error using this data set has been ensured; the same feature subset selection pattern can be used in the test dataset. In other words, the feature subset that has been selected in the training set as the optimal feature subset is used as a template, and among the test data set, only this feature subset is selected to predict intrusions in test set by FLN.

3.4. Complexity Analyzing of Proposed Method. The time complexity of the proposed method can be analyzed as follows. If N is the total number of features in the dataset, each particle as a solution selects n ($1 \le n \le N$) features of all features in the dataset. If the number of iterative generations to find the optimal answer be M for the proposed problem, so a maximum of $M * N$ iterations is required for the problem. On the other hand, since a fast-learning neural network is used at the core of the feature selection method, the complexity of neural networks is order of $O(n^2)$ [20], where n is equal to the number of input features. The fast-learning network method transmits the results to the output layer when the desired accuracy is achieved and does not wait for the completion of the training steps. Hence, the time order of this method is less than $O(n^2)$. Therefore, it can be concluded that the time complexity in the proposed method is the maximum the order of $O(\mathrm{MN}^2)$. In case of more features in the dataset, the time complexity of the proposed method will also increase.

4. Implementation

In the proposed method, primary particles are first selected randomly from the dataset of KDD-CUP99 [21]. In this data set, there are more than 54,000 instances of node connections in the network in which each has 42 features and has classified into two classes of normal nodes and malicious nodes.

Table 1: A part of the initial particle population matrix.

Particle #	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}
1	1	0	1	0	0	1	1	1	1	1	1	1	1	0
2	1	1	0	1	0	1	1	1	0	0	0	1	0	1
3	0	0	1	0	1	1	1	0	1	0	1	0	1	1
4	1	1	1	0	0	0	1	0	0	0	0	1	0	1
5	1	0	0	1	1	0	1	0	0	0	1	1	0	0
6	0	1	1	0	1	0	1	0	1	1	1	1	1	1
7	0	0	1	1	1	1	1	1	0	0	0	0	0	0
8	1	1	1	1	0	1	1	0	1	1	0	0	1	1
9	1	1	1	1	0	0	0	1	1	0	0	1	0	1
10	1	1	1	0	0	1	1	0	1	1	0	1	1	1

The size of the initial population is defined by the number of features existing in the dataset and as an n-dimensional vector. A 100×42-dimensional matrix of random numbers in the range of (0,1) is created to determine the initial population. The $A_{i,j}$ element of the matrix shows the possibility of the presence of the j_{th} feature in the i_{th} solution (particle). According to the Sigmoid function applied in the proposed method, the values of the initial population matrix elements are converted to binary based on a threshold of 0.5. In other words, if the $A_{i,j}$ element has a value below the threshold, the j_{th} feature will not exist in the i_{th} solution (particle). On the other hand, the j_{th} feature of one of the subsets of selected features will have the ith solution (particle) if the value of the mentioned element is above the threshold. Table 1 shows initial particle population.

As observed in Table 1, the initial particle population is distributed in the proposed method in a binary form, and each of the initial populations shows a solution for selecting a subset of features existing in the KDD dataset. According to the adjusted parameters, the initial population matrix is used as an input of MOPSO. In addition, the proposed algorithm evaluates the initial population based on the fitness function in the first step while considering the adjusted parameters and the initial population. Afterwards, it obtains the level of competency of each solution. Therefore, the number of solutions found will increase with more repetitions of the stages, and iterations will continue until reaching the cessation condition. Ultimately, the presented solutions are assessed, and the best solution is selected among the existing solutions. MOPSO receives the initial population and evaluates its competency. In the first step, the algorithm finds the nondominated solutions or dominant solution, saving them in the solutions repository. In the next phase, other solutions and particles are directed toward the solution. Accordingly, a number of dominant solutions may be found in each stage, the value of the fitness function of which might be higher than the threshold and are stored in the respiratory. It is notable that in the proposed method, the amount of competency is equal to the aggregation of two objectives, and the higher the competency of a solution, the lower the number of selected features and the highest the accuracy of classification and intrusion detection by using the features based on FLN class. Therefore, the dominant solutions improve both objectives used in the proposed method. Figure 3 shows the distribution of solutions in the problem space and the dominant solutions in the first step.

As observed in Figure 3, the solutions are randomly distributed in the problem space in the first step of MOPSO in the proposed method. The problem space includes two objectives of F1 and F2, with the former existing to reduce the number of features selected from all features in the dataset and the latter corresponding to the vertical axis of Figure 3 to reduce classification error rate based on the selected features. Given the random selection of the initial particle population and since the nature of particles is binary, the existence or lack of existence of a feature in each particle can affect the results obtained for each solution. Therefore, with regard to the problem space, the Pareto front tends to the origin of the coordinates where both objectives are minimal.

Eventually, it could be expressed that the 6 important features based on the multiobjective fitness function in the MOPSO has selected. The selected features have indexes {2,7,13,19,26,27} entitled {"Srv_count", "Count", "Wrong_Fragment", "land", "ds_host_srv_serror_rate", and "dst_same_srv_rate"}. These features have the greatest impact on the node class label, and based on these features, network intrusion can be detected with the highest accuracy and least complexity.

Due to the fact that different feature selection methods use different policies to select the feature subset, they can therefore select different subsets of feature as the representative feature subset. In the proposed method, different solutions have been created in the initial population, but by examining the rate of classification error related to the subset of selected features, the optimal solution can be selected. Solutions created during iterations can solve the intrusion detection problem, but their intrusion detection accuracy will be less than the near-optimal solution. The proposed method has selected the best solution by evaluating the solutions in terms of intrusion detection accuracy.

With regard to the implementation of the proposed method on the initial population, the mentioned solutions were selected as an expert generation, were present in all

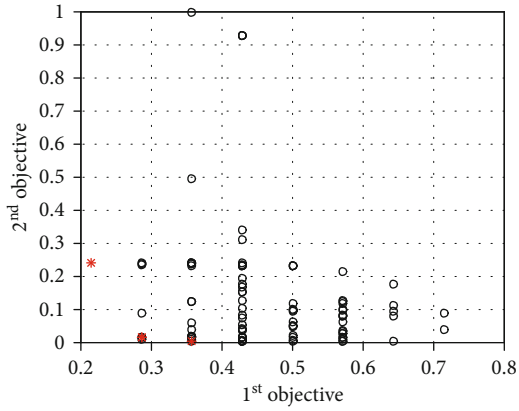

Figure 3: Distribution of expert solutions.

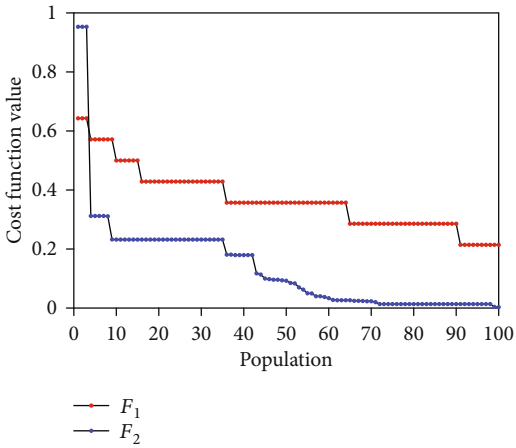

Figure 4: The convergence of target functions' values toward the optimal amount.

iterations, and were improved accordingly. In the last generation of iteration, most particles tended to the expert particles or dominant particles that were repeated in the previous steps and were reserved in the repository. Therefore, the values of the fitness function were close to optimal for all particles in the last generation of iteration. As mentioned before, the value of the fitness function was obtained from a combination of two F_1 and F_2 functions. Accordingly, the final solutions decreased the errors of the proposed method in addition to reducing the number of features in the subset of selected features. Figure 4 shows the convergence of F_1 and F_2 functions toward the optimal point.

As shown in Figure 4, the optimal state of the two objectives had lower values in functions F_1 and F_2 considering that the nature of both target functions was minimized. Therefore, the data presented in Figure 3 revealed that the diagram related to function F_1, which showed the number of selected features, was gradually decreased and reached 0.2 in the end. The diagram showed that an intrusion system could be established with 20% of the total KDD data, and the rest of the data had no use in determining the group related to nodes in the dataset. In addition, with regard to the diagram related to the F_2 function in Figure 4, the errors of

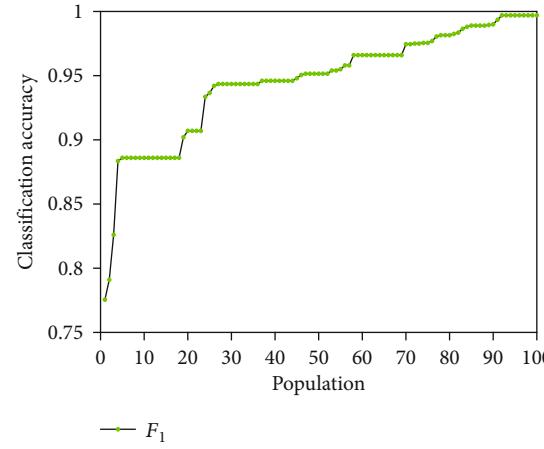

Figure 5: FLN accuracy for intrusion detection patterns.

the proposed method decreased gradually and ultimately reached zero.

A general interpretation of Figure 4 demonstrated that the proposed technique was able to reduce the errors of the intrusion detection system to the lowest level by decreasing the number of features existing in the dataset and selecting the most appropriate features. Given the use of FLN in the method, it could be expressed that the values obtained in F_2 fitness function can be used as an FLN classification error in the proposed technique. As such, the accuracy of the FLN method has shown in Figure 5 for training the intrusion detection techniques in the network.

As observed in Figure 5, the accuracy of the proposed method tended to be optimal with regard to the tendency of the solutions toward the dominant solutions of MOPSO, reaching 99.7%. Therefore, we extracted the most adequate solution obtained from the proposed PSO algorithm, according to which the test dataset was predicted.

4.1. Proposed Model Evaluation. As observed in the previous section, FLN was developed in the present article for classification and prediction of destructive nodes in the network, and the simulation results were determined based on the selection of a feature subset according to MOPSO. There are several criteria for assessing the performance of the proposed method, the most important of which are performance criterion, error rate criterion, and sample the correct prediction rate criterion. The performance criterion monitors the performance of the developed model, meaning that the ideal performance could be drawn by having the label of the data's classes. In addition, evaluation criteria based on the confusion matrix could be used for a two-class problem in order to assess the quality of the proposed method in detecting intrusion and reducing node classification errors in the network. These criteria included accuracy, recall, precision, classification rate (CR), detection rate (DR), false-positive rate (FPR), and F-measure, defined as follows [22]:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \quad (5)$$

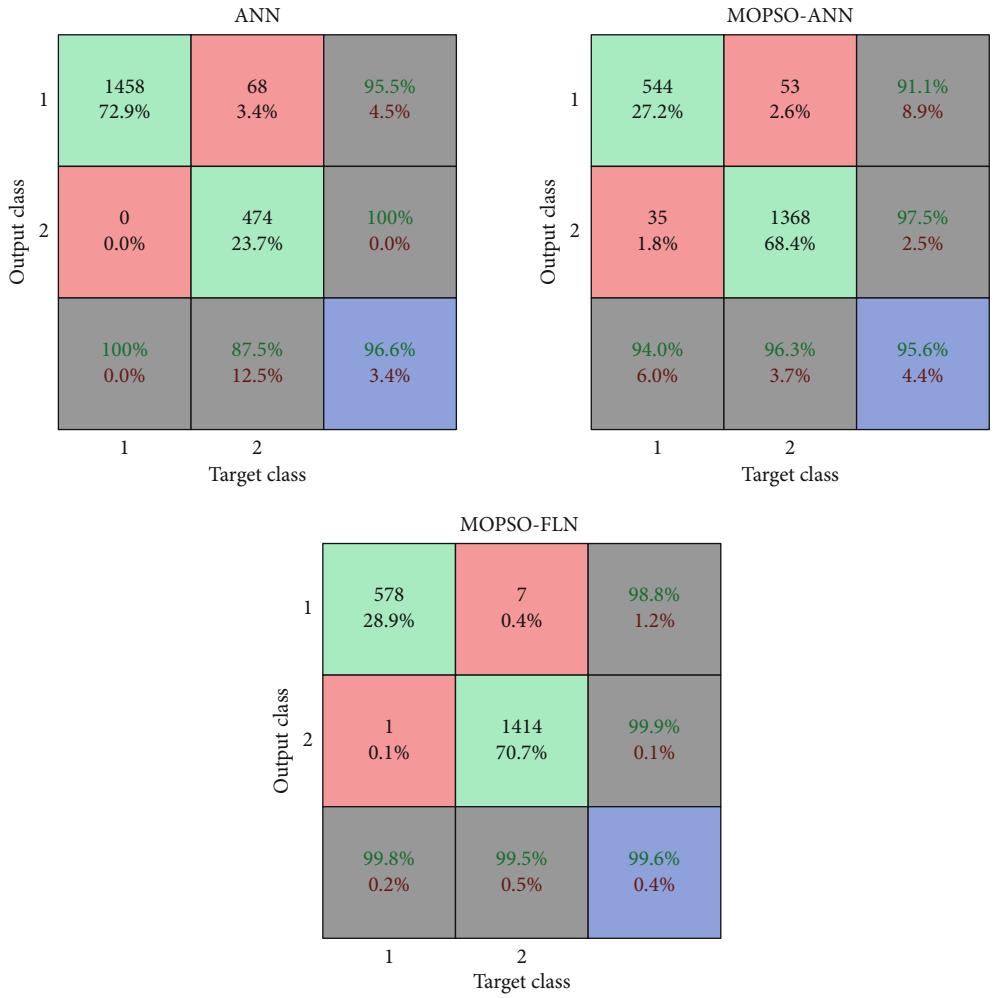

Figure 6: Comparison of the confusion matrix of the proposed method and neural network.

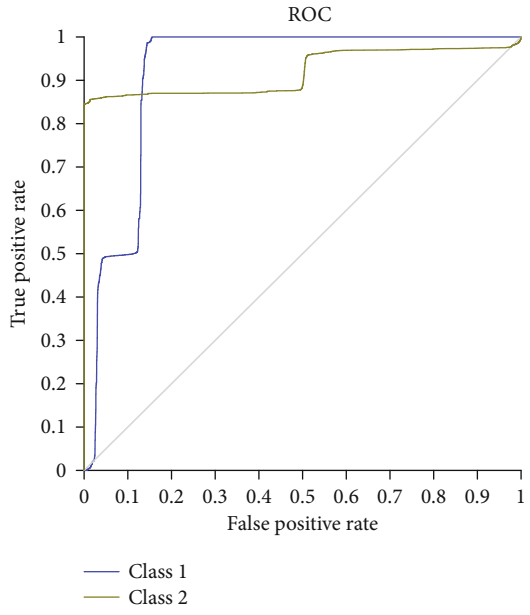

Figure 7: ROC curve of the proposed method.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{6}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{7}$$

$$\text{Classification Rat(CR)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \tag{8}$$

$$\text{Detection rate} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{9}$$

$$\text{False Positive rate (FPR)} = \frac{\text{FP}}{\text{FP} + \text{TN}}, \tag{10}$$

$$F - \text{measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{11}$$

The mentioned evaluation criteria are used as a tool to assess the efficiency of the proposed method and compare it with other existing techniques. Therefore, we compare the proposed method with neural networks without using feature subset selection and neural network approach according to MOPSO-based feature selection. Figure 6 shows the

Table 2: Comparison of values related to the evaluation criteria.

Method	CR (accuracy)	False-positive rate (FPR)	Precision	DR (recall)	*F*-measure
MOPSO-FLN	**99.6**	**0.0137**	**99.44**	99.79	**99.61**
MOPSO-ANN	95.6	0.0888	96.27	97.51	96.88
ANN	96.6	0.0446	84.7	**99.99**	91.71

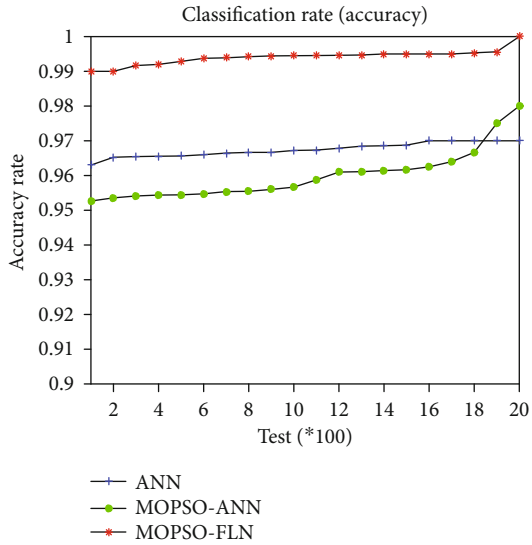

Figure 8: Comparison of the classification rate criterion (accuracy).

comparison of the confusion matrix related to the proposed method and the neural network.

As shown in Figure 6, 99.6% of the total data was classified accurately in the proposed method. Meanwhile, 96.6% and 95.6% of the data were classified correctly in the ANN and MOPSO-ANN methods, respectively.

Another criterion used to evaluate the proposed method is the ROC curve. The ROC curve shows the relation of the true-positive rate to the true-negative rate. The ROC curve of proposed method has shown in Figure 7.

As shown in Figure 7. The proposed method has identified normal and malicious nodes with high accuracy. Table 2 shows a comparison of the values related to the proposed method, ANN, and MOPSO-ANN.

As observed in Table 2, the proposed method had a more adequate performance in terms of the evaluation criteria, compared to the ANN and MOPSO-ANN methods. Figure 8 illustrates the comparison of the classification rate criterion (accuracy) between MOPSO-FLN, ANN, and MOPSO-ANN in 10 steps of 10-fold cross-validation.

According to Figure 8, there were improvements in the proposed method regarding the classification rate (accuracy), compared to ANN and MOPSO-ANN. In the neural network method, the model may experience overfitting given that the training process is carried out completely. In this phenomenon, the model focuses on the training samples and learns all features and relations among the training samples. In addition, its accuracy is maximized in the classification of the training samples. Now, when new test samples that were not previously observed by the model are entered into the system, the model may lack the sufficient accuracy to detect the relations between the features of the new samples that are different from the training samples, which leads to the less efficient performance of the system. Accordingly, the proposed method continues the training steps until reaching the desired accuracy in order to prevent overfitting and increase an IDS's performance. As such, it seems that the FLN method had better accuracy, compared to the method of artificial neural networks. In fact, the proposed method was able to detect a higher percentage of attack and healthy nodes accurately. Figure 8 depicts a diagram that compares the detection rate criteria (recall) between MOPSO-FLN, ANN, and MOPSO-ANN.

The accuracy rate in the proposed method may be in the form of the ratio of the detected healthy nodes among all healthy nodes existing in the dataset that might be detected accurately or be among the false-negative samples. The classification rate is in fact a sum of true-positive samples on all true-positive and false-negative samples. True positive refers to healthy nodes that are detected accurately, whereas false negative refers to healthy nodes that are falsely detected as attack nodes. This relationship shows the proposed model's ability to detect healthy nodes accurately. The higher the value of this relationship, the lower the number of samples related to false negative, where the predicted class has negative nodes, and the actual class has healthy nodes. The lower values of false-negative samples increase the performance of the proposed method in detecting healthy nodes.

According to Figure 9, the proposed method was improved in terms of the detection rate (recall), compared to ANN and MOPSO-ANN. FLN has a higher detection rate, compared to neural networks, considering its lower training than the mentioned technique.

With regard to the structure of accurate learning networks, the training process is discontinued, and the results are transferred to the output when the desired accuracy is reached. Therefore, the features of healthy nodes may not be fully learned but overfitting is avoided in the process, which is an advantage of the proposed method. In fact, the proposed technique has high accuracy for new and unknown samples. In addition to the detection rate in the proposed method, the criterion of the positive error rate of discovered samples is of paramount importance. Figure 10 shows a diagram related to the comparison of the positive error rate criterion between MOPSO-FLN, ANN, and MOPSO-ANN.

As observed in Figure 10, the proposed method had a lower value in terms of the positive error rate, compared to ANN and MOPSO-ANN. In this regard, the positive error rate in the proposed method referred to attacks that could not be detected by IDS. In fact, FLN had a lower positive error rate, compared to the neural networks method,

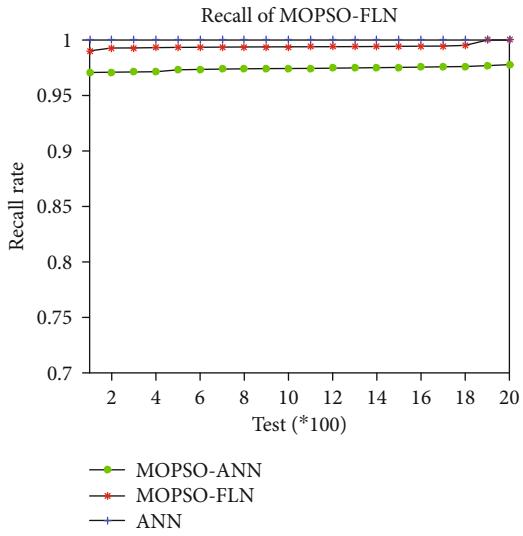

Figure 9: Comparison of detection rate criterion (recall).

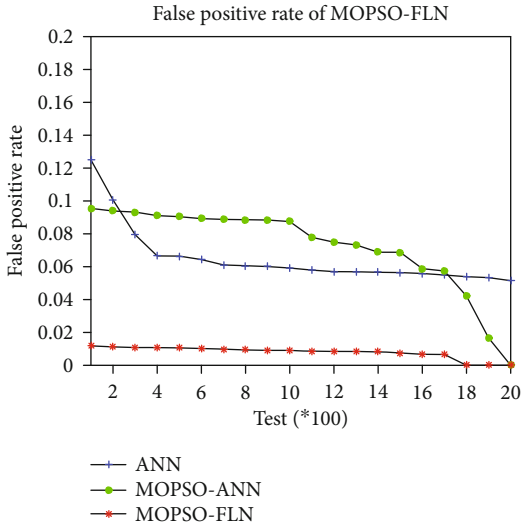

Figure 10: Comparison of the positive error rate criterion.

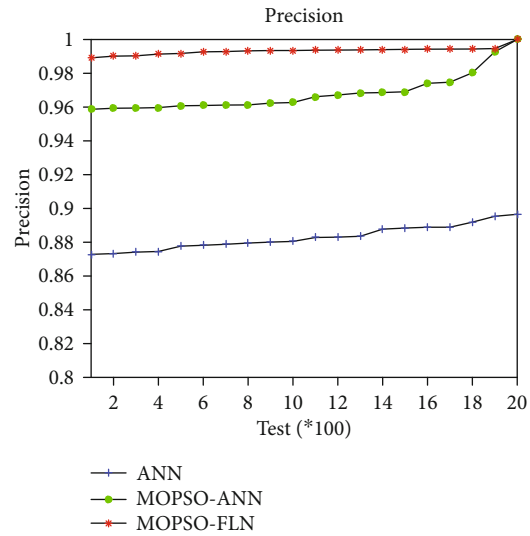

Figure 11: Comparison of the accuracy criterion of the proposed method and neural network.

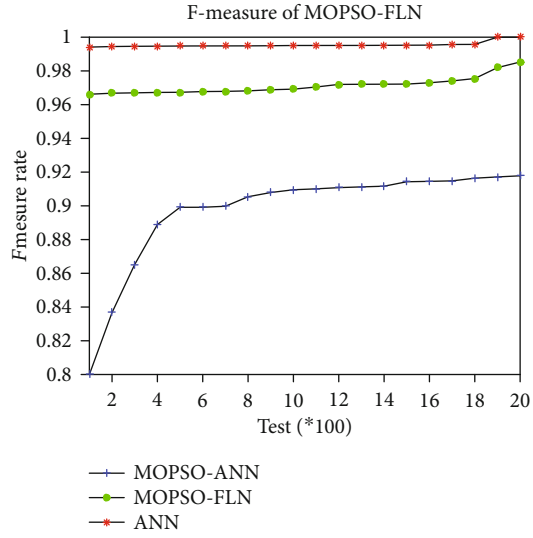

Figure 12: F-measure comparison.

considering its focus on attacks. In spite of full training on attacks existing in the training dataset, neural networks fail to detect some new attacks, about which they had no previous training. Meanwhile, the proposed method was able to detect new attacks by creating a balance between learning training samples and the network's speed. After a positive error rate, we evaluated the attack detection accuracy of the proposed method. In IDSs, accuracy is the form of the ratio of true-positive samples to true-positive samples and true-negative samples, which estimates a reflection of attack detection ability in the classification methods. The higher this value, the higher the classification method's ability to detect and identify new attacks. Figure 11 illustrates a diagram related to the comparison of the positive accuracy criterion between MOPSO-FLN, ANN, and MOPSO-ANN.

According to Figure 11, the proposed method was improved in terms of the accuracy criterion, compared to ANN and MOPSO-ANN. The accuracy of the proposed method improved considering the focus of FLN on attacks and its ability to considerably detect new attacks, compared to neural networks. Figure 11 is a complete representation of the presence of overfitting in neural networks and the lack of presence of this phenomenon in the proposed method. In general, overfitting decreases a model's accuracy per new samples. In fact, FLN can detect most attacks that are among new samples and have not been previously observed in the model. The final criterion assessed in the present study was F-measure, which was a combination of two accuracy and detection rate criteria. The criterion was recognized as a general criterion of the performance of classification methods and IDSs. The higher the value of the criterion, the higher the IDS's ability to classify healthy samples and predict attacks in the training dataset and new attacks that enter

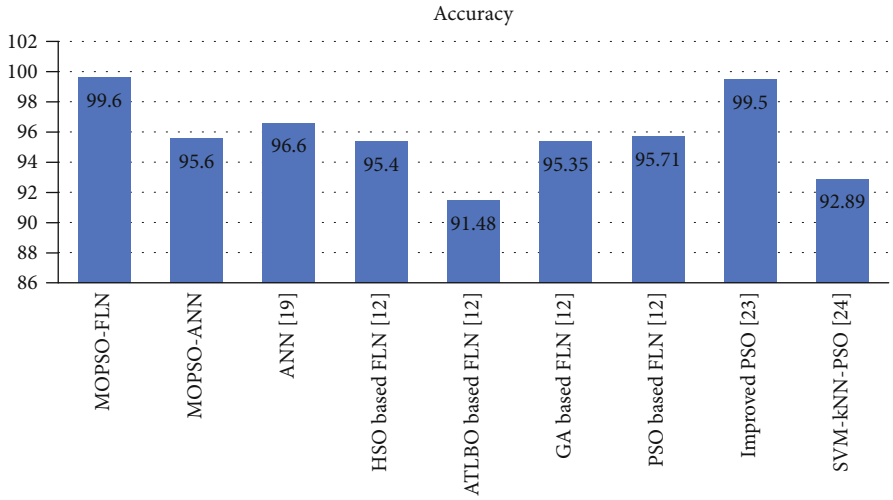

Figure 13: Comparison of the proposed method with previous techniques. All methods compared to the proposed method used a different feature subset selection approach. It can be seen that the accuracy of the proposed method is higher than other methods due to the selection of the optimal feature subset. According to Figure 13, MOPSO-FLN had higher intrusion detection and prediction accuracy, compared to previous methods.

the system. Figure 12 shows a diagram related to the comparison of MOPSO-FLN, ANN, and MOPSO regarding F-measure.

According to Figure 12, the proposed method improved in terms of F-measure criterion, compared to ANN and MOPSO-ANN. In other words, FLN had a better performance in detecting healthy nodes and new attacks in the network when combined with MOPSO-based feature subset selection, compared to neural networks.

4.2. Comparison of Proposed Method with Previous Techniques. In this subset, we compared MOPSO-FLN with other methods existing in publications [12–24] to assess the validity of the proposed method in terms of the prediction accuracy criterion. Figure 13 shows a comparison of the proposed method with previous techniques.

5. Discussion and Conclusion

With regard to the use of the MOPSO-based feature selection approach, it was aimed at reducing performance errors in the classification model and prediction of test samples in addition to finding the best features representing all the features in the dataset. Optimal features extracted in the proposed method by MOPSO were evaluated in each stage of optimization algorithm iteration in order to increase the speed of particles' movement toward particles with high values at Pareto front and reduce data classification errors based on these features in each step. Therefore, selecting these features least to simple distinguishing of the samples related to classes by the classification model and high classification accuracy.

Furthermore, the proposed method was compared to other popular approaches presented in articles. According to the results, the neural network had a relatively lower accuracy, compared to the proposed method, if used independently and without selecting important features subset from the dataset related to intrusion in wireless networks. The difference of about 4% in the accuracy of the proposed method and the neural network-based method was another evidence of the importance of selecting a subset of features using FLN.

The proposed method was also compared to several other approaches that use metaheuristic optimization algorithm-based feature subset selection. According to the results, the proposed method had higher test sample prediction accuracy, compared to the intrusion detection approach, which was a combination of MOPSO-based feature selection and fast neural network. Accordingly, it could be concluded that the proposed method was able to extract important features by using MOPSO-based feature subset selection. In addition, the model yielded acceptable results in terms of the detection and prediction of intrusion in wireless networks by using an evaluation function, which was a combination of a number of features and classification error.

Data Availability

This research and proposed methodology was simulated, and all data are included already in the paper. So, there is no need for extra data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. K. Saxena, S. Sinha, and P. Shukla, "A review on intrusion detection system in mobile ad-hoc network," in *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, pp. 549–554, Bhopal, India, 2018.

[2] S. Muruganandam, J. A. Renjit, and R. S. Kumar, "A survey: comparative study of security methods and trust manage solutions in MANET," in *2019 Fifth International Conference on*

Science Technology Engineering and Mathematics (ICONSTEM), pp. 125–131, Chennai, India, 2019.

[3] V. Singh, D. A. Singh, and M. M. Hassan, "Survey: black hole attack detection in MANET," in *Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE)*, Sultanpur, UP, India, 2019.

[4] A. Gupta and A. Dubey, "A survey on various applications and blackhole attack in mobile ad hoc network," *Recent Trends in Parallel Computing*, vol. 5, pp. 1–6, 2018.

[5] R. Fotohi and S. Jamali, "A comprehensive study on defence against wormhole attack methods in mobile ad hoc networks," *International journal of Computer Science & Network Solutions*, vol. 2, pp. 37–56, 2014.

[6] G. Kumar Ahuja and G. Kumar, "Evaluation metrics for intrusion detection systems-a study," *Evaluation*, vol. 2, pp. 11–17, 2014.

[7] P. Yang, Z. Li, P. Yang, and Y. Dong, "Information-centric mobile ad hoc networks and content routing: a survey," *Ad Hoc Networks*, vol. 58, pp. 255–268, 2017.

[8] A. Sultana and M. A. Jabbar, "Intelligent network intrusion detection system using data mining techniques," in *2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, pp. 329–333, Bangalore, India, 2017.

[9] S. Sindhuja and R. Vadivel, "A study on intrusion detection system of mobile ad-hoc networks," in *Soft Computing for Problem Solving*, pp. 307–316, Springer, 2020.

[10] E. Rosas, N. Hidalgo, V. Gil-Costa et al., "Survey on simulation for mobile ad-hoc communication for disaster scenarios," *Journal of Computer Science and Technology*, vol. 31, no. 2, pp. 326–349, 2016.

[11] D. Rajalakshmi and K. Meena, "A survey of intrusion detection with higher malicious misbehavior detection in MANET," *International journal of civil engineering and technology*, vol. 8, no. 10, pp. 99–110, 2017.

[12] 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.

[13] B. Selvakumar and K. Muneeswaran, "Firefly algorithm based feature selection for network intrusion detection," *Computers & Security*, vol. 81, pp. 148–155, 2019.

[14] Z. Chiba, N. Abghour, K. Moussaid, A. El Omri, and M. Rida, "A novel architecture combined with optimal parameters for back propagation neural networks applied to anomaly network intrusion detection," *Computers & Security*, vol. 75, pp. 36–58, 2018.

[15] H. Alazzam, A. Sharieh, and K. E. Sabri, "A feature selection algorithm for intrusion detection system based on pigeon inspired optimizer," *Expert systems with applications*, vol. 148, p. 113249, 2020.

[16] Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Computer Networks*, vol. 174, 2020.

[17] B. Senthilnayaki, K. Venkatalakshmi, and A. Kannan, "Intrusion detection system using fuzzy rough set feature selection and modified KNN classifier," *The International Arab Journal of Information Technology*, vol. 16, no. 4, pp. 746–753, 2019.

[18] M. Habib, I. Aljarah, H. Faris, and S. Mirjalili, "Multi-objective particle swarm optimization: theory, literature review, and application in feature selection for medical diagnosis," in *Evolutionary Machine Learning Techniques*, pp. 175–201, Springer, Singapore, 2020.

[19] C. A. Coello Coello and M. S. Lechuga, "MOPSO: a proposal for multiple objective particle swarm optimization," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No. 02TH8600)*, vol. 2, pp. 1051–1056, Honolulu, HI, USA, 2002.

[20] C. H. Dagli, "Complexity analysis of multilayer perceptron neural network embedded into a wireless sensor network," *Procedia Computer Science*, vol. 36, pp. 192–197, 2014.

[21] KDD Cup, *Computer Network Intrusion Detection*, 1999, https://www.kdd.org/kdd-cup/view/kdd-cup-1999.

[22] M. Almseidin, M. Alzubi, S. Kovacs, and M. Alkasassbeh, "Evaluation of machine learning algorithms for intrusion detection system," in *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*, pp. 277–282, Subotica, Serbia, 2017.

[23] A. Dickson and C. Thomas, "Improved PSO for optimizing the performance of intrusion detection systems," *Journal of Intelligent Fuzzy Systems*, vol. 38, pp. 6537–6547, 2020.

[24] A. Aburomman and R. Mamun, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," *Applied Soft Computing*, vol. 38, pp. 360–372, 2016.