Received: October 4, 2022. Revised: December 3, 2022. 410

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Heuristic Initialization Using Grey Wolf Optimizer Algorithm for Feature Selection in Intrusion Detection

Hussein Fouad Almazini[1]* Ku Ruhana Ku-Mahamud[2] Hassan Almazini[1]

[1]*Computer Technology Engineering, Shatt Al-Arab University College, Basra, Iraq*
[2] *School of Computing, Universiti Utara Malaysia, Malaysia*
[2]*Shibaura Institute of Technology, Tokyo, Japan*
* Corresponding author's Email: hussein.f.abbas@sa-uc.edu.iq

Abstract: Anomaly detection deals with identification of items that do not conform to an expected pattern or items present in a dataset. The performance of the various mechanisms that are employed to execute anomaly detection is strongly dependent on the set of features that are utilized. Thus, not every feature in the dataset may be employed in the classification operation since certain characteristics may result in poor solution quality. Feature selection (FS) may reduce the size of high-dimensional datasets by eliminating unimportant features. Modified binary grey wolf optimizer (MBGWO) is a successful metaheuristic that has been used for FS in anomaly detection. Nonetheless, the MBGWO is a randomization population-based algorithm that has an issue in finding a good quality solution during the initial population procedure. Thus, this study proposes a heuristic modified binary grey wolf optimizer (heuristic MBGWO) algorithm for FS in intrusion detection to enhance the initial population of the MBGWO using a heuristic-based ant colony optimization algorithm (ACO). The heuristic MBGWO algorithm was evaluated on NSL-KDD benchmark dataset from the University California Irvine (UCI) repository against five (5) benchmark metaheuristic algorithms. experimental results of the heuristic MBGWO algorithm on the NSL-KDD dataset in terms of the number of chosen attributes and classification accuracy are superior to other benchmark optimization algorithms, where it obtained the best features with 99.85% classification accuracy. The proposed heuristic MBGWO algorithm can be used for FS in anomaly detection tasks that involve any dataset size from various application domains.

Keywords: Grey wolf optimizer, Classification, Feature selection, Optimization, Anomaly detection.

1. Introduction

The activity of detecting abnormalities or assaults in information systems is known as intrusion detection [1]. In intrusion detection, two types of detection are known as anomaly-based detection and signature-based detection. The signature-based detection process is efficient in detecting known attacks/anomalies and operates by checking the data in a system's memory or network traffic for specific patterns. Anomaly detection systems function by watching the behaviour of the whole system, traffic, data or objects, and then comparing it to expected or normal behaviour. Thus, any behaviour that is different from the norm is seen as a possible attack. However, anomaly detection has become the preferred method since it is difficult to discern between various sorts of attacks in high-dimensional data [1].

Attackers have adept at producing malware with the ability to alter their structure (polymorphism). Moreover, after an attack is discovered, time is needed to observe and produce an action within the anomaly detection system. Machine learning and artificial intelligence are required for anomaly detection [2, 3]. Classification in anomaly detection systems helps to detect a pattern to differentiate between normal and non-normal data. The performance of various machine learning techniques used to identify anomalies in systems or data is largely dependent on the features that are used.

Feature selection (FS) is a technique for removing noisy, irrelevant, and redundant data while recognizing useful features [4]. Feature selection can

speed up data mining algorithms and improves predictive accuracy. Redundant features supply no more knowledge than the irrelevant features, and chosen features provide no usable information. Feature selection has been used effectively in different application domains. Better detection accuracy may be achieved by applying FS techniques to the data before the anomaly detection system analyses the features [5].

The advantages of FS in data mining classification are as follows: i) develops predictive accuracy, ii) improve the process time of an algorithm for data mining, iii) enhances inductive learning, iv) increases comprehensibility, and v) reduces the complexity of model. In general, there are three techniques for FS: wrapper, embedding, and filtering [6]. In the FS context, the filter is faster but poorly in providing the quality of solution and more complicated than the wrapper [7, 8]. In the embedded approach, inside its framework the learning algorithm creates its own optimization tools [9]. The wrapper approach is widely utilized due to its efficiency in handling more massive and complex datasets than the filter and embedded approaches [10]. However, different approaches have been suggested for solving FS issues, such as metaheuristic, heuristic, and evolutionary[11].

Using metaheuristic algorithms in solving FS problems is popular because it provides near-optimal solutions [12]–[16]. Metaheuristic algorithms are computational intelligence models specifically utilized for solving complicated optimization problems.

Grey wolf optimizer (GWO) was first suggested in 2015 for data extraction [17], which is a process in FS and classification [18]. The success of the GWO drives researchers to use this algorithm to resolve different kinds of optimisation problems as it takes one vector of location which needs less memory compared to other swarm intelligence methods like particle swarm optimisation (PSO) that requires memory to hold velocity and position vectors [19]. Moreover, the GWO algorithm has become more interesting is solving FS issues as shown in Fig 1.

The modified binary GWO (MBGWO) which has been proposed for FS selection [20] has an issue for identifying an acceptable quality solution involving reduced exploration. To search a vast area, each wolf in the MBGWO algorithm needs to explore more areas. The procedure of the initial solution in the MBGWO is influenced by random selection whereby selecting even one wrong feature will affect the solution's quality [20–22].

This paper proposed a heuristic MBGWO algorithm to overcome the problem of random initial

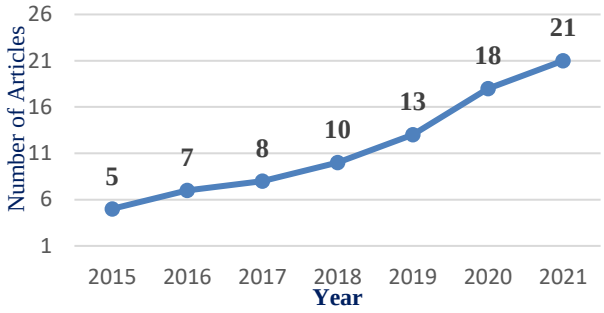


Figure. 1 Grey wolf social structure

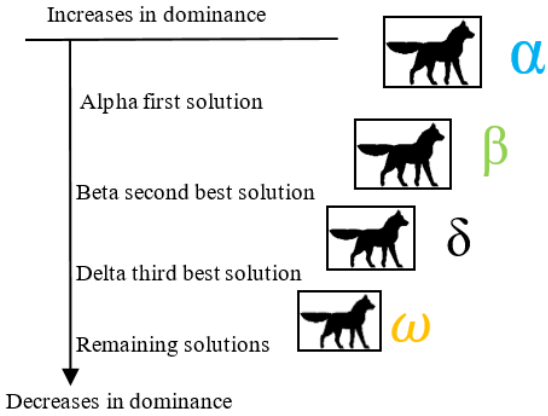


Figure. 2 Grey wolf social structure

population. Main advantages of this paper are the heuristic approach in choosing the features subset. The main benefit of this approach is that it can find an ideal or nearly ideal solution in a high-dimensional dataset. The suggested algorithm is assessed using NSL-KDD dataset from University California Irvine (UCI) repository together with five benchmark metaheuristic algorithms.

This paper is organized as follows. Section 2 describes the mathematics of MBGWO optimization. Section 3 describes the proposed heuristic method. Section 4 presents the data and experimental design. The results are presented in section 5, and finally, section 6 summarizes the findings and future research.

2. Modify grey wolf optimizer

The MBGWO is a modern mechanism in the stock of swarm intelligence (SI) algorithms supported by hunting and leadership behaviour. There is a collective of social hierarchy that recognises the power and dominance in every set of wolves in the pack. The MBGWO relies on the hunting and leadership behaviour of grey wolf packs. The most potent and effective wolf in feeding migration and hunting is called alpha (α), this wolf guides and leads the full pack. The next dominant

Table 1. List of notations

Symbol	Description
W	Best solution/location
A&C	Coefficient vectors
r	Random vectors in [0, 1]
a	Linearly decrement parameter
bi	Number of samples in every attribute
F-score (F_i)	Heuristic information
c	Class numbers in the dataset
ni	Number of patterns in class
xi	The average of all the samples
xki & rki	Variance and mean of class
Sw & SB	Scatter matrix
μ_j	Pattern average vector of all class
Mo	Pattern average vector of each class entire data points
$\tau_i(t)$	Quantity of pheromone
$\Delta\tau_i^k$	Extra increment of pheromone
FS^k	Attribute subset founded by ant
NF	Features subset
AC	Accuracy

wolf, called beta (β), will be a leader if alpha is sick or dead. Other wolves which are less effective than alpha and beta, are called omega (ω) and delta (δ). This characteristic of swarm intelligence is the MBGWO algorithm's main motivation. The wolves' social hierarchy is shown in Fig. 2, where the simulation stringent in hunting and leadership classifies the wolves into four kinds depending on their leadership (fitness) [17].

The MBGWO algorithm uses a set of random locations to start the optimization process. At the same time, there is a vector for each position that preserves the amounts for the parameters. The initial phase of each repetition is to calculate the fitness value of each location. Thus, every position is prepared with a variable to save its fitness. The four vectors and three variables save the fitness value and positions of the best wolves in the memory. In the MBGWO the four vectors alpha, beta, delta and omega agents are required to be updated before the position updating process. Distances between the three wolves/variables and the present solution must be determined before a new wolf position can be determined. The four variables in updating the location of the solutions are then calculated. In this paper, the notations used are summarized in Table 1.

New wolf locations are computed based on the four best places/solutions, as shown below:

$$\overrightarrow{W}_{(t+1)} = \frac{\overrightarrow{W_1}+\mathrm{W}_2+\mathrm{W}_3+\overrightarrow{W_4}}{4} \quad (1)$$

where $\overrightarrow{W}_1, \mathrm{W}_2, \overrightarrow{W}_3, \mathrm{W}_4$ are defined as:

$$\begin{aligned} \overrightarrow{W}_1 &= |\mathrm{W}_\alpha - \vec{A}_1.\vec{D}_\alpha| \\ \overrightarrow{W}_2 &= |\overrightarrow{W}_\beta - \vec{A}_2.\vec{D}_\beta| \\ W_3 &= |\overrightarrow{W}_\delta - \vec{A}_3.\vec{D}_\delta| \\ \mathrm{W}_4 &= |\overrightarrow{W}_\omega - \vec{A}_4.\vec{D}_\omega| \end{aligned} \quad (2)$$

$$\begin{aligned} \vec{D}_\alpha &= |\vec{C}_1.\overrightarrow{W}_\alpha - \overrightarrow{W}| \\ \vec{D}_\beta &= |\vec{C}_2.\overrightarrow{W}_\beta - \overrightarrow{W}| \\ \vec{D}_\delta &= |\vec{C}_3.\overrightarrow{W}_\delta - \overrightarrow{W}| \\ \vec{D}_\omega &= |\vec{C}_4.\overrightarrow{W}_\omega - \overrightarrow{W}| \end{aligned} \quad (3)$$

The variables, $\overrightarrow{W}_\alpha, \overrightarrow{W}_\beta, \overrightarrow{W}_\delta, \overrightarrow{W}_\omega$ are the four best solutions at iteration *t*, $\vec{C}_1, \vec{C}_2, \vec{C}_3, \vec{C}_4$ and $\vec{A}_1, \vec{A}_2, \vec{A}_3, \vec{A}_4$ are the coefficient vectors calculated as in Eqs. (4) & (5) [17]:

$$\vec{A} = 2\vec{a}.\vec{r}_1\vec{a} \quad (4)$$

$$\vec{C} = 2\vec{r}_1 \quad (5)$$

During the iterations $\vec{r}_1$, $\vec{r}_2$ are the random vectors in [0, 1],and $\vec{a}$ decrease linearly from 2 to 0. Updating the value of the parameter (a) is performed as in Eq. (6) [17]:

$$\mathrm{a}=2-t(\tfrac{2}{T}) \quad (6)$$

Consequently, it is hard to implement a solution to FS problems without making certain modifications. Thus, to find solutions to FS issues, there should be a procedure that can convert the general method into its binary versions. Many binary versions have been proposed in the literature, such as crossover [18], sigmoid [23], and tanh functions [24].

In SI algorithms, initialization for the population is a crucial part that can significantly impact their performance. In addition, this initialization is dependent on the issue to be addressed, since it may cause the algorithm to provide low-quality solutions. Thus, finding an acceptable quality solution with less exploration is a challenge for the MBGWO. Each wolf in the MBGWO algorithm needs to search more region in order to cover the search space. Random selection plays a role in the MBGWO initial solution process because choosing one incorrect feature might have an impact on the outcome. Random initialization techniques have been used for the SI to progress its performance. A binary particle swarm optimization (BPSO) for FS to achieve a good subset of features was proposed for better classification accuracy and shorter computational time [25] [26]. The proposed algorithm shows that the BPSO is powerful to select relevant features from different

dataset sizes.

Enhancement of the SVM classifier using a binary dragonfly (BDF) was proposed in 2017 [27], for FS which demonstrates that the algorithm compatibity to binary bat algorithm and BPSO. The results show that the BDF algorithm obtains a smaller number of selected attributes and obtains a high classification performance. It also produces better solution quality and adaptively converges faster.

An enhanced binary harris hawks optimizer (BHHO) for FS proposed by Thaher [28] is a potentially effective strategy for handling high-dimensional real-world datasets. The wrapper approach was used to obtain the best subset of features. The classification accuracy was improved with the lowest number of attributes.

In all earlier published studies, the procedure of the initial solution is influenced by random selection whereby selecting even one wrong feature will affect the solution's quality [20]–[22]. In this study, a method to obtain an initial population with good quality (classification accuracy) by using the heuristic from the ant colony optimization (ACO) algorithm is proposed. The ACO memory has an advantage that can help to learn from previous solutions to avoid the limitation for the random initial population in the MBGWO and enhance the MBGWO for FS in anomaly detection.

3. Proposed heuristic initialization mechanism for MBGWO

The heuristics population initialization mechanism is based on the ACO heuristic concept. The mechanism consists of a heuristic function, probabilistic rule and pheromone update strategy. In the initial population mechanism for the family of grey wolf optimizer algorithm which include the GWO, modify GWO, binary GWO and MBGWO algorithms, the population were randomly initialized which has affected the solution's quality [29].

Fig. 3 presents a low-level description of the ACO heuristic population initialization mechanism which aims to produce solutions by selecting features that maximise classification accuracy.

The first step of the ACO heuristic initialization is to initialize the pheromone array with an amount of pheromone that is inversely proportionate to the amount of features in the dataset using Equation (7) [30]:

$$\tau_n(t=0) = \frac{1}{\sum_{i=1}^{a} bi} \tag{7}$$

where *a* is the total amount of attributes, and *bi* is

```
Input: training set
Output: features subset
Step 1- Initialize pheromone
        Initialize parameters A, C & a
Step 2- For i = 1 to no. of features
            Compute F-score for each feature

Step 3- Initialize features subset size randomly
         For i = 1 to subset size
            Compute probability for each feature
            Select feature with highest probability
         End
Step 4- Evaluate feature subset
Step 5- Update pheromone
Step 6- Find best four features subsets
Step 7- while "not converge" do
           For each agent
              Update positions according to the best
                        wolves

           Update parameters A, C & a
           Evaluate wolves' positions
           Update the best four positions
       End while
Step 8-Return best solution
```

Figure. 3 Heuristics initial population mechanism

the overall number of samples in every attribute. The second step is a loop which will be executed until a termination case is met. In this study, the number of attributes in the dataset is similar to the number of iterations used. The heuristic information for every attribute is measured by F-score (F_i) from the filter-based FS using Eq. (8) [31]:

$$F\_Score(F_i) = \frac{\sum_{k=1}^{c} n_i(x_i^k - x_i)^2}{\sum_{k}^{c} n_i(\sigma_i^k)^2} \tag{8}$$

where *c* is the class numbers in the dataset, *ni* is the number of patterns in class *i*, *xi* displays the average of all the samples identical to feature *Fi*, and also *xki* and *rki* denote the variance and mean of class *k* identical to feature *Fi*. A big *F-Score* value means that feature *Fi* has a greater characteristic capability. Another iteration is performed in the third step which is to randomly initialize the feature subset size. There is another loop to compute probability and select features which is equal to feature subset size. Then, the ants will choose an attribute which is the least similar to the previous chosen attribute. The new chosen attribute reflects the highest dependence on the target class and dependence, $n(F_i|, VF_i)$, is computed using Eq. (9) [31]:

$$n(F_i|, VF_i) = [F - Score(F_i) - \frac{1}{|VF_k|}\sum_{F_x \in VF_k} sim(F_i, F_x)] \quad (9)$$

where sim *(Fi, Fx)* denotes the similarity value between features *Fx* and feature *Fi*. In the probabilistic process, the kth ant chooses the next attribute *Fj* with a probability of *Pk (VFk, Fj)* this is calculated as in Eq. (10) [31]:

$$P_k(F_j, VF_k) = \frac{[\tau_j]^{\alpha}[n(F_j, VF_k)]^{\beta}}{\sum_{u \in UF_i^k}[\tau_u]^{\alpha}[n(F_u, VF_k]^{\beta}} \quad (10)$$

Every feature has the same chance of being chosen as its probability value, that is calculated as in Equation (10). The fitness of each ant's solution is assessed in the fourth phase using a specifically formulated separability index that was realized in getting the optimum linear grouping of attributes in the case of two class difficulties. The separability index is described by Eq. (11) [31]:

$$\gamma(FS) = trace(\frac{W^T S_B W}{W^T S_W W}) \quad (11)$$

where *W* the diversions matrix from the unique l-dimensional subspace to the n-dimensional space identical to the chosen subset FS, *Sw*, within scatter matrix, *SB* between scatter matrix that are measured as in Eqs. (12) & (13) [31]:

$$S_w = \sum_{j=1}^{c} \pi_j \varepsilon_j \quad (12)$$

$$S_B = \sum_{j=1}^{c} (\mu_j - M_0)(\mu_j - M_0)^T \quad (13)$$

where π_j is the a priori possibility in which a sample related to a specific class *j*, $\sum_j$ is the pattern covariance matrix of class *j*, $\mu_j$is the pattern average vector of class and *Mo* is the pattern average vector of the class entire data points measured as in Eq. (14) [31]::

$$Mo \sum_{j=1}^{C} \pi_j \mu_j \quad (14)$$

The fifth step is for the pheromone update of every feature. This is updated at the end of every iteration, only when all the search ants have finished their traverses on the search space, the pheromone level of every attribute is updated using the following Eq. (15) [31]:

$$\tau_i(t+1) = (1-\rho)\tau_i(t) + \sum_{k=1}^{A} \Delta_i^k(t) \quad (15)$$

where ρ the pheromone decay parameter, $\tau_i(t)$ and $\tau_i(t+1)$ is the quantity of pheromone on attribute Fi at times *t* and *t + 1*, sequentially, *A* is the ant's number, and $\Delta\tau_i^k(t)$ is the extra increment of the pheromone to the attribute Fi by ant *k*, is defined as in Eq. (16) [31]:

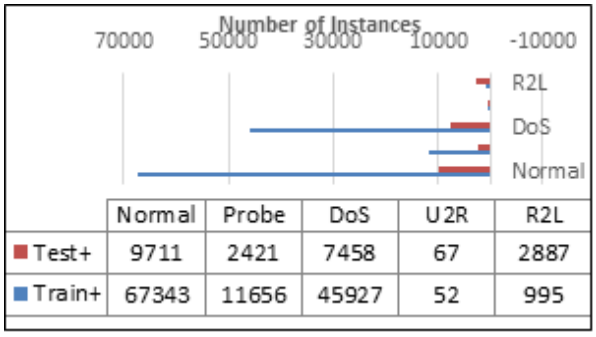
$$\Delta_i^k(t) = \gamma(FS^k(t)) \quad (16)$$

where $FS^k(t)$ is the attribute subset founded by ant *k* at iteration *t*, and $\gamma(FS^k(t))$ is the assessment function which determines the fineness of solution $FS^k(t)$.

In the sixth step, the best subsets from the ACO

#	Feature	#	Feature
1	duration	22	is_guest_login
2	protocol_type	23	Count
3	service	24	srv_count
4	flag	25	serror_rate
5	src_bytes	26	srv_serror_rate
6	dst_bytes	27	rerror_rate
7	land	28	srv_rerror_rate
8	wrong_fragment	29	same_srv_rate
9	Urgent	30	diff_srv_rate
10	Hot	31	srv_diff_host_rate
11	num_failed_logins	32	dst_host_count
12	logged_in	33	dst_host_srv_count
13	num_compromised	34	dst_host_same_srv rate
14	root_shell	35	dst_host_diff_srv rate
15	su_attempted	36	dst_host_same_src port rate
16	num_root	37	dst_host_srv_diff host rate
17	num_file_creations	38	dst_host_serror_rate
18	num_shells	39	dst_host_srv_serror rate
19	num_access_files	40	dst_host_rerror_rate
20	num_outbound cmds	41	dst_host_srv_rerror rate
21	is_host_login		

Figure. 4 The NSL-KDD dataset features



	Normal	Probe	DoS	U2R	R2L
Test+	9711	2421	7458	67	2887
Train+	67343	11656	45927	52	995

Figure. 5 NSL-KDD dataset class distribution

heuristic process will be sorted in descending order, and the first best subsets (equal to the size of the wolf pack) will be selected as the final initial population for the heuristic MBGWO. The remaining processes of the algorithm are to update the location of current wolf giving to the best wolves' positions and to provide the best solution (best feature subset).

4. Experimental setup

Performance of the enhanced MBGWO algorithm has been assessed on the NSL-KDD dataset which contains 41 feaatures [32]. The KDDTest+ (for testing) and KDDTrain+ (for training) datasets are subset of NSL-KDD dataset and contain approximately 125,973 and 22,544 instances respectively. For the experiment, a combination of 20% from KDDTrain+ and 20% from KDDTest+ datasets giving a total of 29,702 instances was used. From this total, 80% was used for training and 20% for testing [26]. The hold-out method is used in the experiments [20]. Fig. 4 shows the features of the NSL-KDD dataset while Fig. 5 shows the class distribution. Every attack is classified under one of the four classes: DoS, Probe, U2R, and R2L. A network connection (e.g., protocol type, service and flag), in each NSL-KDD sample with 41 known attributes labelled as an attack (e.g., DoS, Probe, R2L and U2R) or as normal.

The performances of the benchmark algorithms BPSO [25] [26], BDA [27], BHHO [28], and MBGWO were compared to the performance of heuristic MBGWO based on features selected and accuracy. The benchmark algorithms were chosen because they are classified as metaheuristic and have distinct beginning procedures (random initial population) which limits the algorithm ability to select the optimal features during the initial population procedure. Furthermore, these algorithms from the SI family are benchmark algorithms for FS. The fitness function that has been applied in this phase is calculated as in Eq. (17) [20]:

$$Fitness = AC.a + (1/NF).b, \quad (17)$$

where NF is the features subset and AC is the accuracy and the values of parameters a, and b are between 0 and 1. In evaluating the proposed algorithm, the SVM classifier is used because it is commonly used for classification of anomaly detection [33], [34].

To achieve the appropriate balance between the features selected and classification accuracy, the most common test used in the literature is the nonparametric Friedman test conducted with the Holm post-hoc test [35]. The outcome of the nonparametric Friedman test and Holm's post-hoc test to find the average number of FS rank versus the average classification accuracy rank for the algorithms. The goal of this test is to determine the best algorithm that can produce a good balance between the features selected and accuracy.

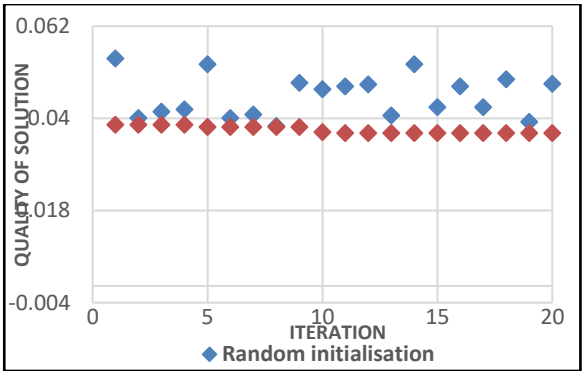


Figure. 6 Population initialization for the heuristic and random initializations

5. Experimental results and analysis

Fig. 6 presents samples of the population initialization outcomes for the heuristic initialization in the proposed method and random initialization in MBGWO on NSL-KDD dataset. As shown in the figure, the heuristic MBGWO seeks to overcome the problem of random initialization (scattered) in the MBGWO to the solution that contain a poor quality which challenges in obtaining the ideal or near ideal solution. The heuristic initialization is demonstrating the quality is maintained.

Tables 2 and 3 display the results of the average classification accuracy (Acc) and average number of selected features (ANF). The best algorithm is the one with the highest accuracy and the least number of features. Results in Table 1 shows that in three classes the suggested heuristic MBGWO showed the best classification accuracy (Normal, U2R and R2L) while the BPSO algorithm achieved the best classification accuracy in the remaining classes (Dos and Probe). The standard deviation (std) values of the proposed heuristic MBGWO indicate that the algorithm is stable in performing the FS task.

Results display in Table 3 shows that the MBGWO algorithm is the best algorithm in terms of selected features in four of the datasets. One of the results is being shared with the heuristic MBGWO algorithm. The heuristic MBGWO managed to obtain the best results in only the Normal and Dos datasets.

The ranks of the average accuracy and selected features using the Friedman test with Holm's post-hoc test results are shown in Table 4. The smallest value

Table 2. Average classification accuracy using SVM classifier

DATA		1	2	3	4	5
Nor	Acc	**98.74 %**	98.26 %	97.15 %	98.41 %	97.29 %
	Std	0.002	-	0.002	-	0.001
	Rank	**1**	3	5	2	4
Dos	Acc	99.62 %	99.42 %	99.66 %	**99.73 %**	99.71 %
	Std	0.000	-	0.001	-	0.000
	Rank	4	5	3	**1**	2
Probe	Acc	98.76 %	98.66 %	98.67 %	**98.88 %**	98.74 %
	Std	0.000	-	0.001	-	0.000
	Rank	2	5	4	**1**	3
U2R	Acc	**99.85 %**	99.59 %	99.77 %	99.77 %	99.77 %
	Std	0.000	-	0.000	-	0.000
	Rank	**1**	5	3	3	3
R2L	Acc	**97.97 %**	97.36 %	97.71 %	97.58 %	97.76 %
	Std	0.001	-	-	0.001	0.001
	Rank	**1**	5	3	4	2

1.Proposed heuristic MBGWO **3. BDA**
2.MBGWO **4.BPSO**
5.BHHO

Table 3. Average number of selected features using SVM classifier

DATA		1	2	3	4	5
Nor	ANF	**20**	**20**	24	26	21
	Rank	**1.5**	**1.5**	4	5	3
Dos	ANF	**16**	17	25	23	23
	Rank	**1**	2	5	3.5	3.5
Probe	ANF	17	**16**	25	22	21
	Rank	2	**1**	5	4	3
U2R	ANF	18	**12**	23	18	19
	Rank	2.5	**1**	5	2.5	4
R2L	ANF	19	**18**	22	23	21
	Rank	2	**1**	4	5	3

1.Proposed heuristic MBGWO **3. BDA**
2.MBGWO **4.BPSO**
5.BHHO

Table 4. Performance rank on NSL-KDD subset with SVM classifier

	1	2	3	4	5
Accuracy	**1.8**	4.6	3.6	2.2	2.8
Selected features	1.8	**1.3**	4.6	4	3.3

indicates the best rank which refers to the highest accuracy or the least number of features. In this test,

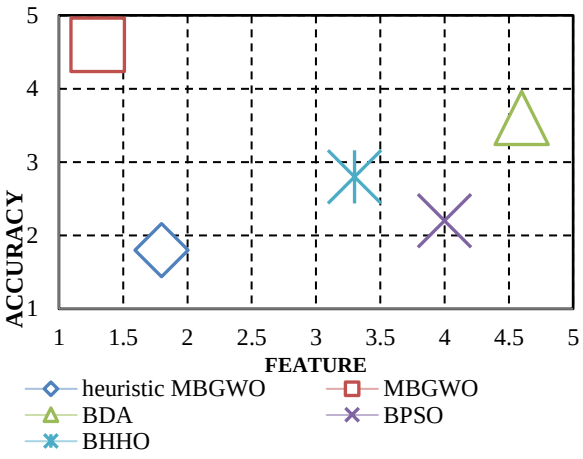


Figure. 7 Performance rank plot using SVM classifier on NSL-KDD subset

the heuristic MBGWO and MBGWO are the best performed algorithms for the average accuracy and average selected features respectively.

The results in Table 4 are also shown in Fig. 7 where the performance metric feature is plotted against accuracy. In this display, the algorithm that is the closest to the origin is the best algorithm to obtain a balance between the features selected and accuracy. In this figure, the best algorithm is the heuristic MBGWO algorithm.

In summary, the heuristic MBGWO obtained the best balance between number of chosen features and accuracy. This balance was able to be obtained because the best number of features were selected which increased classification accuracy. A high number of features might include redundant features resulting in noise data while a low number of features might not include enough information for the classification task. This will affect classification accuracy. Thus, the heuristic MBGWO helps to avoid the non-optimal results in the initial solution by selecting the best features subset that increases the quality of solutions (classification accuracy). The proposed mechanism aims to find the efficient optimal solution for each search wolf in the initial population by considering the heuristic and the feedback in the ACO algorithm.

6. Conclusion and future work

This paper proposes a heuristic MBGWO to enhance the initialization of the wolves' population which is based on the ACO algorithm concept. The general advantage of this mechanism is to select an appropriate feature that maximises classification accuracy in the initial population step. The experimental results show that the heuristic-based initial population mechanism, performs better than other variant of GWO algorithms and to other state-

of-the-art algorithms for FS tasks that use the random initial population approach, where it obtained the best features with 99.85% classification accuracy. The heuristic MBGWO algorithm for FS can be classified as another variant of the GWO algorithm, which can be used for FS in anomaly data. Finally, future studies can focus at other approaches to compute an initial population in order to reach a good compromise between the GWO's exploration and exploitation processes.

Conflicts of interest

The authors declare no conflict of interest.

Author contributions

The first and second authors prepared the draft, while the third author did the review and editing.

Acknowledgments

The researchers thank the Malaysian Ministry of Higher Education for financially supporting this study under the Grant Scheme (TRGS /1/2018/UUM/02/3/3 (S/O code14163).

References

[1] A. Alghuried, "A Model for Anomalies Detection in Internet of Things (IoT) Using Inverse Weight Clustering and Decision Tree", *Dublin Institute of Technology*, 2017.

[2] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection", *IEEE Commun. Surv. Tutorials*, Vol. 18, No. 2, pp. 1153–1176, 2016.

[3] E. Alpaydin, *Introduction to machine learning*, The MIT Press, 2020.

[4] H. Almazini and K. R. Ku-Mahamud, "Adaptive technique for feature selection in modified graph clustering-based ant colony optimization", *Int. J. Intell. Eng. Syst.*, Vol. 14, No. 3, pp. 332–345, 2021, doi: 10.22266/ijies2021.0630.28.

[5] J. Song, "Feature selection for intrusion detection system", Aberystwyth University, 2016.

[6] M. S. Packianather and B. Kapoor, "A wrapper-based feature selection approach using Bees Algorithm for a wood defect classification system", In: *Proc. of 10th Syst. Syst. Eng. Conf. SoSE 2015*, pp. 498–503, 2015.

[7] D. T. Pham, M. Mahmuddin, and S. Otri, "Application of the Bees Algorithm to the Selection Features for Manufacturing Data", *Manuf. Eng.*, No. January, 2007.

[8] P. Gupta, S. Jain, and A. Jain, "A Review of Fast Clustering-Based Feature Subset Selection Algorithm", *Int. J. Sci. Technol. Res.*, Vol. 3, No. 11, pp. 86–91, 2014.

[9] V. Kumar and S. Minz, "Feature Selection : A literature Review", *SmartCR*, Vol. 4, No. 3, 2014.

[10] Y. Sun, S. Todorovic, and S. Goodison, "Local-learning-based feature selection for high-dimensional data analysis", *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 32, No. 9, pp. 1–18, 2010.

[11] H. N. K. A. Behadili and K. R. Ku-Mahamud, "Fuzzy Unordered Rule Using Greedy Hill Climbing Feature Selection Method: An Application to Diabetes Classification", *J. Inf. Commun. Technol.*, Vol. 20, No. 3, pp. 391–422, 2021.

[12] V. Agrawal and S. Chandra, "Feature selection using Artificial Bee Colony algorithm for medical image classification", In: *Proc. of International Conf.on Contemporary Computing*, 2015, pp. 171–176.

[13] L. Y. Chuang, C. H. Yang, and J. C. Li, "Chaotic maps based on binary particle swarm optimization for feature selection", *Appl. Soft Comput.*, Vol. 11, No. 1, pp. 239–248, 2011.

[14] H. M. Zawbaa, E. Emary, A. E. Hassanien, and B. Parv, "A wrapper approach for feature selection based on swarm optimization algorithm inspired from the behavior of social-spiders", In: *Proc. of International Conf.on Soft Computing and Pattern Recognition*, 2015, pp. 25–30.

[15] M. Sudana, R. Nalluri, Saisujana, H. Reddy, and V. Swaminathan, "An Efficient Feature Selection using Artificial Fish Swarm Optimization and SVM Classifier", In: *Proc. of International Conf. On Networks & Advances in Computational Technologies*, 2017, No. 7, pp. 407–411.

[16] J. A. G. Sargo, S. M. Vieira, J. M. C. Sousa, and and C. J. A. B. Filho, "Binary fish School search applied to featureselection: Application to ICU readmission", In: *Proc. of Intl. Conf. on Fuzzy Systems*, pp. 1366-1373, 2014.

[17] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Advances in Engineering Software Grey Wolf Optimizer", *Adv. Eng. Softw.*, Vol. 69, pp. 46–61, 2014.

[18] E. Emary, "Binary Gray Wolf Optimization Approaches for Feature Selection", *Neurocomputing*, Vol. 172, No. 8, pp. 371–381, 2015.

[19] H. Faris, I. Aljarah, M. A. A. Betar, and S.

Mirjalili, "Grey wolf optimizer : a review of recent variants and applications", *Neural Comput. Appl.*, No. November, 2017.
[20] Q. M. Alzubi, M. Anbar, Z. N. M. Alqattan, and M. Azmi, "Intrusion detection system based on a modified binary grey wolf optimisation", *Neural Comput. Appl.*, pp. 1–13, 2020.
[21] Q. Gu, X. Li, and S. Jiang, "Hybrid genetic grey wolf algorithm for large-scale global optimization", *Complexity*, pp. 1–18, 2019.
[22] Z. J. Teng, J. L. Lv, and L. W. Guo, "An improved hybrid grey wolf optimization algorithm", *Soft Comput.*, 2018.
[23] Q. A. Tashi, S. J. Abdulkadir, H. Rais, and S. Mirjalili, "Binary Optimization Using Hybrid Grey Wolf Optimization for Feature Selection", *IEEE Access*, Vol. 7, pp. 1–9, 2019.
[24] A. Sahoo and S. Chandra, "Multi-objective Grey Wolf Optimizer for improved cervix lesion classification", *Appl. Soft Comput. J.*, Vol. 52, pp. 64–80, 2017.
[25] S. Lin, K. Ying, S. Chen, and Z. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines", *Expert Syst. Appl.*, Vol. 35, pp. 1817–1824, 2008.
[26] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, "Feature selection based on rough sets and particle swarm optimization", *Pattern Recognit. Lett.*, Vol. 28, No. 4, pp. 459–471, 2007.
[27] S. Ahmed, T. Ait, A. Benyettou, and M. Ouali, "Kernel-based learning and feature selection analysis for cancer diagnosis", *Appl. Soft Comput. J.*, Vol. 51, pp. 39–48, 2017.
[28]T. Thaher, A. A. Heidari, M. Mafarja, J. S. Dong, and S. Mirjalili, "Binary harris hawks optimizer for high-dimensional, low sample size feature selection", January, pp. 251–272, 2020.
[29] H. Almazini and K. R. Ku-Mahamud, "Grey Wolf optimization parameter control for feature selection in anomaly detection", *Int. J. Intell. Eng. Syst.*, Vol. 14, No. 2, pp. 474–483, 2021, doi: 10.22266/ijies2021.0430.43.
[30] H. N. K. A. Behadili, Ku-Mahamud, and R. Sagban, "Adaptive parameter control strategy for ant-miner classification algorithm", *Indones. J. Electr. Eng. Informatics*, Vol. 8, No. 1, pp. 149–162, 2020.
[31] P. Moradi and M. Rostami, "Integration of graph clustering with ant colony optimization for feature selection", *Knowledge-Based Syst.*, Vol. 84, pp. 144–161, 2015.
[32] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set", In: *Proc. of International Conf. On Computational Intelligence for Security and Defense Applications*, 2009, pp. 1–6.
[33] E. Emary, H. M. Zawbaa, and A. E. Hassanien, "Binary ant lion approaches for feature selection", *Neurocomputing*, Vol. 213, pp. 54–65, 2016.
[34] N. Kunhare, R. Tiwari, and J. Dhar, "Particle swarm optimization and feature selection for intrusion detection system", *Sadhana*, Vol. 45, No. 109, pp. 1–14, 2020.
[35] J. Demšar, "Statistical comparisons of classifiers over multiple data sets", *J. Mach. Learn. Res.*, Vol. 7, pp. 1–30, 2006.