Neural Computing and Applications
https://doi.org/10.1007/s00521-019-04103-1

ORIGINAL ARTICLE

Intrusion detection system based on a modified binary grey wolf optimisation

Qusay M. Alzubi[1] · Mohammed Anbar[1] · Zakaria N. M. Alqattan[1] · Mohammed Azmi Al-Betar[2] · Rosni Abdullah[1]

Received: 23 July 2018 / Accepted: 15 February 2019

Abstract
One critical issue within network security refers to intrusion detection. The nature of intrusion attempts appears to be nonlinear, wherein the network traffic performance is unpredictable, and the problematic space features are numerous. These make intrusion detection systems (IDSs) a challenge within the research arena. Hence, selecting the essential aspects for intrusion detection is crucial in information security and with that, this study identified the related features in building a computationally efficient and effective intrusion system. Accordingly, a modified feature selection (FS) algorithm called modified binary grey wolf optimisation (MBGWO) is proposed in this study. The proposed algorithm is based on binary grey wolf optimisation to boost the performance of IDS. The new FS algorithm selected an optimal number of features. In order to evaluate the proposed algorithm, the benchmark of NSL-KDD network intrusion, which was modified from 99-data set KDD cup to assess issues linked with IDS, had been applied in this study. Additionally, the support vector machine was employed to classify the data set effectively. The proposed FS and classification algorithms enhanced the performance of the IDS in detecting attacks. The simulation outcomes portrayed that the proposed algorithm enhanced the accuracy of intrusion detection up to 99.22% and reduction in the number of features from 41 to 14.

Keywords Intrusion detection system · Anomaly-based detection · Modified binary grey wolf optimisation · Grey wolf optimisation

1 Introduction

The enhanced intrusion detection system (IDS) can detect forms of damaging attacks in existing environments. The IDS is placed within the network that it protects and collects network packets promiscuously in the same manner as a network sniffer. IDS detects hostile, harmful, and network-damaging events. The collected packets are analysed by the IDS, which signals the system administrator to block the connections of the attack so that more network damages are prevented due to malicious attacks. The system is also connected to firewall, which is a key technological aspect that maintains network security [1, 2].

The IDS can be classified into two major groups based on detection techniques. The first group refers to signature-based detection or the 'misuse detection', while the second group is 'anomaly-based detection' [3]. Misuse detection system depends on signatures and can only work efficiently in known attacks scenarios. Unknown attacks, however, cannot be detected by this system. The anomaly detection system, on the other hand, draws upon the attacker's action in comparison with the actions displayed by a normal user. Anomaly detection works in unknown attack scenarios with rates that are highly false and positive. The IDS can function at many stages, such as data collection, pre-processing, feature selection (FS), and classification. Since IDS addresses massive amounts of data, the feature must be demanding.

The attributes in IDS could contain correlations that are false that would eventually prevent the process from running smoothly or hinder the learning task. Certain features

✉ Mohammed Anbar
anbar@nav6.usm.my

[1] National Advanced IPv6 Centre of Excellence (NAv6) 6th Floor, School of Computer Sciences, Universiti Sains Malaysia, Penang, Malaysia

[2] Information Technology Dept, Al-huson University College, Al-Balqa Applied University, Irbid, Jordan

Published online: 27 February 2019

could be redundant or even irrelevant, hence decreasing the classifier accuracy and increasing computation time. Thus, such classification is appropriate for FS approaches, which are centred in gaining features subset to address the issue without jeopardising its performance. FS is used to choose features that are informative and relevant, apart from reducing feature set and general data, as well as enhancing performance and comprehension of data.

The selection process of important features is referred to as FS, while classification incorporates selected significant features that constitute a subset. In IDS, the significant computation overhead is crucial [4], wherein accuracy is a concern [5] by minimising computation time [6], as IDS has the capability of identifying various intrusions in real time. Thus, ranking and selecting a subset of extremely discriminating features is the most challenging task in designing an efficient IDS [7]. FS approaches and optimisation methods have recently been the focus of attention in selecting important features [8].

In precise, not all features are regarded as significant or even relevant in detecting intrusion. Some features may be noisy, irrelevant, and redundant, thus should be rejected. Based on the classification, the FS refers to pre-processing task that minimises the number of features and error rate of classification—the typical dual contradicting objectives. FS addresses these issues to enhance the level of accuracy [9].

Such challenge cannot be easily addressed by using classical optimisation approaches. Alternative optimisation techniques are available, such as swarm-based algorithms that offer ideal solutions. The idea of swarm-based algorithms is generated from the natural and social behaviours of species. It is worth mentioning that various computational techniques have been proposed by researchers to mimic species' behaviours in search of food, which represents an optimal solution to the problem [10]. Several IDS-based FS algorithms have been proposed in the literature, such as particle swarm optimisation (PSO) [11], artificial bee colony (ABC) [12], and ant colony optimiser (ACO) with PSO [13].

A wide range of heuristic methods tends to imitate nature's biological and physical systems' behaviours as worldwide optimisation robust methods. One of the many evolutionary computation (EC) methods is PSO that accomplishes practical solutions depending on position and fitness that are calculated based on position, as well as the speed of the practical to detect direction when the practical is moving [10]. Grey wolf optimiser (GWO) improvements were carried out by researchers to enhance the accuracy of IDS [14–16]. However, most boosts have suffered from low rates of data set coverage and accuracy due to the massive data set volume.

The GWO algorithm is proposed in [17] on the basis of modelling grey wolf social hierarchy and hunting habits towards finding prey, as represented in the solution to the optimisation problem. The social hierarchy is simulated by categorising the population of search agents into four types of individuals: alpha, beta, delta, and omega, based on their fitness. The search process is modelled to mimic the hunting behaviour of grey wolfs via three stages: searching, encircling, and attacking the prey. The first two stages are dedicated to exploration, while the last one is exploitation. The reduced number of search parameters is an advantage of the GWO algorithms reflected in varied applications.

This study contributes to the following three assertions:

1. The proposed modified binary grey wolf optimisation (MBGWO) embeds the omega wolf in deciding to update the next location within the original GWO.
2. The MBGWO is proposed to identify network intrusions that display better accuracy and the smallest number of features.
3. The MBGWO was tested by using NSL-KDD data set based on true positive, true negative, and accuracy.

The following section presents the related scholarly studies. Background information regarding IDS, GWO, and binary grey wolf optimisation (bGWO) is depicted in Sect. 3. The proposed MBGWO approach is discussed in Sect. 4. The new approach assessment and the evaluation outcomes are provided in Sect. 5. The conclusions and future researches are given in Sect. 6.

2 A review of related studies

The IDS-based anomaly techniques, which are previously mentioned, are still implemented for certain reasons as briefly reviewed in the following scholarly studies.

Both bGWO and neural network classifier can be applied to select the significant features as a new method for the IDS network. The irrelevant features are removed by using GWO on the NSL-KDD data set to boost the accuracy rate and to reduce the feature set. However, the false alarm rate and the feature select subset have been neglected [18].

The GWO refers to a metaheuristics swarm intelligence approach that has been vastly applied to overcome numerous optimisation issues because of its exceptional traits, when compared to other approaches, including fewer parameters and nil requirement of derivative information for the first search. Additionally, this approach is flexible, simple, easy, and scalable, and can offer the right balance between exploitation and exploration at the searching phase, thus leading to convergence that is favourable. With that, the GWO has gained much attention across multiple domains in no time [19].

The most attractive operator within optimisation algorithm refers to the selection task, which mimics the

survival of the fittest principle. Upon greedy selection, the search is biased towards exploitation and the pressure becomes higher. On the contrary, upon random selection, exploration is emphasised, and the pressure is lower. Being driven by three best solutions, the GWO selection process becomes greedy [20].

GWO is a technique that is based on swarm optimisation to detect the features' subset so that the SVM accuracy and the naïve classification are enhanced, while minimising the number of features. First, the model incorporated the filter-based principle so that redundancy by mutual information was decreased. Second, the wrapper approach was used to guide the classifier performance. However, the false positive rate was neglected [21].

The PSO model works based on several linear programming principles to upgrade the attacks' detection accuracy rates. Multiple criteria linear programming (MCLP) is a technique of classification based on mathematical programming. The MCLP has the ability to solve real-life data mining issues. The PSO approach is robust and simple for implementation by enhancing the performance of the MCLP classifier. In a study, the outcomes included detection rate, false alarm rate, and running time. Nonetheless, the selected subset of features is neglected [12].

The ABC and AdaBoost algorithms are used for selection, classification, and evaluation of features. They are hybrid approaches used for an anomaly network-based IDS to obtain the highest detection rate and the lowest false rate based on simulations carried out on NSL-KDD and ISC-XIDS2012. These hybrid methods have been reported to achieve high accuracy and detection rates. However, the number of features was high, which contributed to high-dimensional space [13].

In order to address the most challenging problem related to the high number of features in detecting intrusion, a study proposed a new method by combining PSO and information entropy minimisation method with hidden naïve base classifier. The performance of the proposed method was evaluated on NSL-KDD data set. The obtained results reduced the number of features, although it did not enhance the accuracy rate [22].

Both ACO and PSO were utilised by Parsian et al. [14] in intrusion detection. The suggested hybrid technique decreased the false alarm rate and attained higher accuracy rate. The IDS effectiveness had been based on data collection. The researchers confirmed that the data, which led to the intrusion detection engine, determined the IDS quality. The intrusion detection employed the hybrid swarm intelligence algorithm (PSO/ACO). A comparison of the findings was performed with the outcomes retrieved from SVM algorithm.

A hybrid algorithm that detects network intruders was proposed by Amudha et al. [23] by combining ABC with improved PSO to specifically identify network traffic irregular patterns. Friedman and ANOVA tests verified the classifier accuracy.

As for this present study, a genetic algorithm (GA)- and SVM-based IDS is proposed. GA is combined with SVM so that the overall performance of SVM-based IDS is enhanced and the best SVM classifier detection model is determined. The study outcomes demonstrated that the SVM-based IDS did not only select the best parameters for SVM classifier, but also selected the best features amongst the feature set as a whole. There was an improvement in all attack types detection rate. It displayed better separation between normal and attack. Nevertheless, the empirical work excluded false alarm rate [24].

3 Background

The theoretical background of IDS based on FS method is outlined in this section. It also depicts a brief overview of FS methods employed in intrusion detection based on network traffic. The most well-known IDS approaches are also reviewed.

3.1 Theoretical background of IDS based on FS method

Although FS and dimensionality reduction are very close, they differ. It selects an optimum subset of relevant features that represents an original feature set with the least rates of error to design a classification model [25].

The framework architecture of IDS, which is based on features selection using SVM classifier, is illustrated in Fig. 1 [25]. Network traffic refers to input (NSL-KDD data set) and attack detection, whereas deduction and alarms are final outputs. The IDS model is composed of three significant steps: training, FS, and classification.

In the first step, the input served as the trained data set to generate features positions and to assess each position. Positions of the highest classification AC and the smallest features number were selected. Such optimised features' positions of each attack and normal traffic represented the final output in this step. In intrusion detection, training is regarded as the first step, wherein SVM is trained with training data set. The SVM creates support vectors based on training data set classification so that more accurate detection in the following step is achieved. The traffic data set is classified into normal traffic or anomaly traffic [25].

3.2 Grey wolf optimisation (GWO)

Grey wolves inspired GWO as a novel metaheuristic that imitates the hierarchy of leadership and grey wolves'

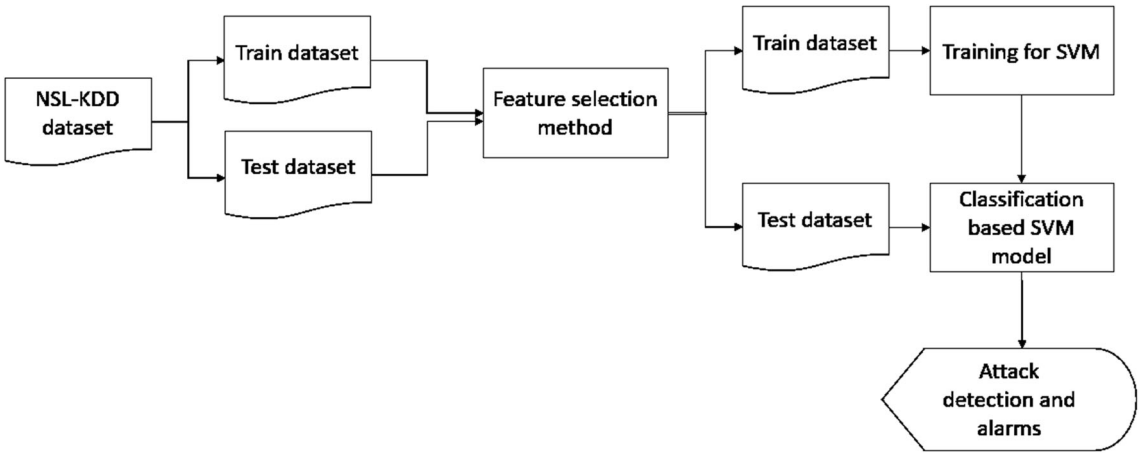

Fig. 1 Framework of IDS based on features selection method

hunting mechanism in the nature. In the attempt of simulating the hierarchy of leadership, GWO used four kinds of grey wolves: alpha, beta, delta, and omega [17]. Grey wolves often prefer living within a pack. The average size of the group ranges between 5 and 12. Their hierarchy is an extremely strict social dominant one. Figure 2 illustrates the features contained in the grey wolf pack.

For a mathematical model of GWO, the most suitable result is alpha (α) in the first place. Accordingly, beta (β) and delta (δ) represent appropriate results in the second and third places, respectively. Another expected result is the omega (ω). Based on GWO algorithm, α, β, and δ guide the hunt (optimisation), whereas ω wolves track three wolves. The grey wolf encircles its prey, and this is the main phase during the hunt. Therefore, Eqs. (1) and (4) are used to mathematically model the encircling behaviour.

$$W(iter+1) = P(iter) - A \cdot D_i, \tag{1}$$

where $\boldsymbol{D_i}$ is calculated in Eq. 2, **iter** is the current iteration, **P** is the prey's position, and **Y** is the grey wolf's position.

$$D_i = |C \cdot P(t) - W(t)|, \tag{2}$$

where **A** and **C** are coefficient vectors and are calculated in Eqs. (3) and (4).

$$A = 2b \cdot r_1 - b \tag{3}$$

$$C = 2 \cdot r_2, \tag{4}$$

where the components of **b** are decreased (2–0) in a linear pattern during iterations, while r_1 and r_2 are random vectors in [0, 1]. Hunting is often directed by alpha. Beta and delta occasionally take part in the hunt. Nonetheless, in an abstract search space, the location of the optimum (prey) is unknown. Thus, in order to carry out a mathematical simulation of grey wolves' hunt action, it was predicted that alpha (the best candidate solution), beta, and delta would exhibit more

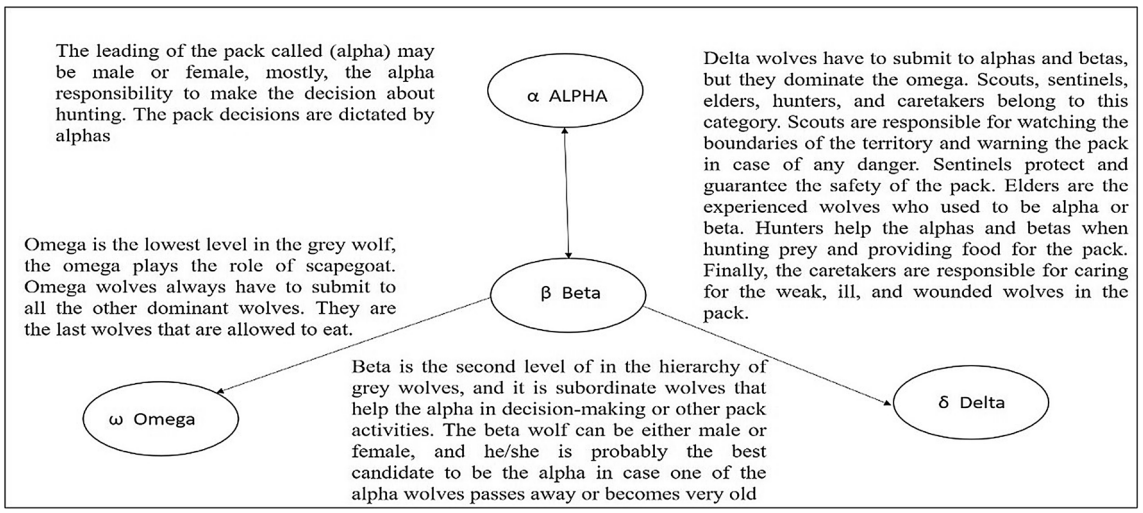

Fig. 2 Features contained in the grey wolf pack [17]

acquaintance of where the prey could potentially be located. Accordingly, the best three results attained were reserved. Searching agents (including omegas) are obliged to update locations based on the best search agents' location. To update the wolves' location, Eq. (5) is used.

$$W(iter+1) = \frac{W_1 + W_2 + W_3}{3}, \tag{5}$$

where $\boldsymbol{y_1}, \boldsymbol{y_2}$, and $\boldsymbol{y_3}$ are defined in Eqs. (6)–(8), respectively.

$$W_1 = |W_\alpha - A_1 \cdot D_{i\alpha}| \tag{6}$$

$$W_2 = \left|W_\beta - A_2 \cdot D_{i\beta}\right| \tag{7}$$

$$W_3 = |W_\delta - A_3 \cdot D_{i\delta}|, \tag{8}$$

where W_α, W_β, and W_δ are the first three best solutions in the grey wolf at a given iteration iter, A_1, A_2, and A_3 are defined in Eq. (3), and $D_{i\alpha}$, $D_{i\beta}$, and $D_{i\delta}$ are defined in Eqs. (9)–(11), respectively.

$$D_{i\alpha} = |C_1 \cdot W_\alpha - W| \tag{9}$$

$$D_{i\beta} = \left|C_2 \cdot W_\beta - W\right| \tag{10}$$

$$D_{i\beta} = \left|C_3 \cdot W_\beta - W\right|, \tag{11}$$

where C_1, $\mathbf{C_2}$, and $\mathbf{C_3}$ are defined in Eq. (4).

In conclusion, in order to emphasise exploration and exploitation, parameter **b** was decreased from 2 to 0 according to Eq. (12).

$$b = 2 - iter\frac{2}{MAXIter}, \tag{12}$$

where ***iter*** is the current iteration and ***MAXIter*** is the total number of iterations allowed in the algorithm. The GWO algorithm was outlined by Algorithm 1.

Initialize *the grey wolf population Wi(i=1,2,……n)*
Initialize *b, A, and C*
Calculate *the fitness of each search agent*
w_α *= the first best search agent*
w_β *= the second best search agent*
w_δ *= the third best search agent*
While *(iter < Max number of iterations)*
  For *each search agent*
    Update *the position of the current search agent by equation* ***(5)***
  End for
  Update *b, A, and C*
  Calculate *the fitness of all search agent*
  Update w_α, w_β, w_δ
iter=iter+1
end while
return w_α

3.3 Binary grey wolf optimisation (bGWO)

In adhering to GWO, wolves often practice changing locations to some places in the space. Regarding specific challenges pertaining to FS, the obtained results were limited to the binary {0, 1} values, which inspired a special version of GWO [26]. The work proposed a novel bGWO applied for FS tasks. The wolves that updated the equation represented a three-location vector function: w_α, w_β, and w_δ, which worked on inviting each of the wolves to the most suitable three results. Based on the GWO principle, the location of a given wolf is incorporated while keeping the binary restriction according to Eq. (5). The researcher applied the update on GWO, which is described in detail in Eqs. (13)–(23). The central updating equation is framed in Eq. (13).

$$W_i^{t+1} = \text{``Crossover''}(w_1 \cdot w_2 \cdot w_3) \tag{13}$$

w_1, w_2, and w_3 are binary vectors that represent the impact of the wolf's move towards alpha, beta, and delta grey wolves based on a particular sequence. Equations (14), (17), and (20) give the mathematically calculated w_1, w_2, and w_3, respectively.

$$w_1^d = \begin{cases} 1 & \text{if } \left(w_\alpha^d + \text{stepb}_\alpha^d\right) \geq 1 \\ 0 & \text{otherwise} \end{cases}, \tag{14}$$

where $\boldsymbol{w_\alpha^d}$ is the alpha wolf location vector in **d** and $\mathbf{stepb_\alpha^d}$ is a binary step in **d.** They are mathematically determined in Eq. (15).

$$\text{stepb}_\alpha^d = \begin{cases} 1 & \text{if } \text{stepc}_\alpha^d \geq \text{rand} \\ 0 & \text{otherwise} \end{cases}, \tag{15}$$

where **rand** is randomly selected from uniform distribution $\in [\mathbf{0} \cdot \mathbf{1}]$ and $\mathbf{stepc_\alpha^d}$ is the dimension **d** continued valued step size that is mathematically determined by the sigmoidal function in Eq. (16).

$$\text{stepc}_\alpha^d = \frac{1}{1 + e^{-10\left(A_1^d Di_\alpha^d - 0.5\right)}}, \tag{16}$$

where $\boldsymbol{A_1^d}$ and $\boldsymbol{D_\alpha^d}$ are calculated using Eqs. (3) and (9) in dimension **d**.

$$w_2^d = \begin{cases} 1 & \text{if}\left(w_\beta^d + \text{stepb}_\beta^d\right) \geq 1 \\ 0 & \text{otherwise} \end{cases}, \tag{17}$$

where $\mathbf{w_\beta^d}$ is the beta wolf location vector in **d** and $\mathbf{stepb_\beta^d}$ is a binary step in **d**, determined mathematically in Eq. (18).

$$\text{stepb}_\beta^d = \begin{cases} 1 & \text{if } \text{stepc}_\beta^d \geq \text{rand} \\ 0 & \text{otherwise} \end{cases}, \tag{18}$$

where **rand** is randomly selected from uniform distribution $\in [\mathbf{0} \cdot \mathbf{1}]$ and $\mathbf{cstep_\beta^d}$ is dimension **d** continued valued step size, determined by the sigmoidal function according to Eq. (19).

$$\text{cstep}_{\beta}^{d} = \frac{1}{1 + e^{-10\left(A_2^d Di_{\beta}^d - 0.5\right)}}, \tag{19}$$

where $\boldsymbol{A_2^d}$ and $\boldsymbol{Di_{\beta}^d}$ are mathematically determined in **d** according to Eqs. (3) and (9).

$$w_3^d = \begin{cases} 1 & \text{if } \left(w_{\delta}^d + \text{stepb}_{\delta}^d\right) \geq 1 \\ 0 & \text{otherwise} \end{cases}, \tag{20}$$

where $\boldsymbol{w_{\delta}^d}$ is the delta wolf location vector in **d** and $\mathbf{stepb_{\delta}^d}$ is a binary step in **d**, which are mathematically determined by Eq. (21).

$$\text{stepb}_{\delta}^d = \begin{cases} 1 & \text{if stepc}_{\boldsymbol{\delta}}^d \geq \text{rand} \\ 0 & \text{otherwise} \end{cases}, \tag{21}$$

where **rand** is randomly chosen from uniform distribution $\in [\mathbf{0} \cdot \mathbf{1}]$ and $\mathbf{stepc_{\delta}^d}$ is the continued valued step size for **d**. They are determined by the sigmoidal function according to Eq. (22).

$$\text{stepc}_{\boldsymbol{\delta}}^d = \frac{1}{1 + e^{-10\left(A_3^d Di_{\boldsymbol{\delta}}^d - 0.5\right)}}, \tag{22}$$

where $\boldsymbol{A_3^d}$ and $\boldsymbol{Di_{\delta}^d}$ are mathematically obtained in d according to Eqs. (3) and (9).

A simple strategy of random probability distribution crossover was implemented per dimension to crossover w_1, w_2, and w_3 outputs in accordance with Eq. (23).

$$w_d = \begin{cases} w_1^d & \text{if rand} < \dfrac{1}{3} \\ w_2^d & \dfrac{1}{3} \leq \text{rand} < \dfrac{2}{3} \\ w_3^d & \text{otherwise} \end{cases}. \tag{23}$$

The bGWO algorithm is profiled by Algorithm 2.

Initialize *the grey wolf population Wi(i=1,2,……n)*
Initialize *b, A, and C*
Calculate *the fitness of each search agent*
$\boldsymbol{w_{\alpha}}$ *= the first best search agent*
$\boldsymbol{w_{\beta}}$ *= the second best search agent*
$\boldsymbol{w_{\delta}}$ *= the third best search agent*
While *(iter < Max number of iterations)*
  For *each search agent*
  Calculate *w1, w2, w3 using the Equation (14), (17), and (20)*
  Calculate wd using Equation (23)
  End for
  Update *b, A, and C*
  Calculate *the fitness of all search agent*
  Update w_{ω}, w_{β}, w_{δ}
iter=*iter+1*
end while
return w_{α}

4 Modified binary grey wolf optimisation (MBGWO)

In MBGWO, the wolves keep changing their position based on four best solutions: α, β, δ, and $\boldsymbol{\omega}$, to reduce the impact of the best solutions. The bGWO used the crossover approach to change the position of the grey wolves to a point in the space based on three best solutions: α, β, and δ, in GWO to perform FS. Figure 3 illustrates the flow chart of the proposed MBGWO FS algorithm.

The bGWO was unrestricted to GWO for FS problem. However, in MBGWO, the next position changed based on the four best solutions of α, β, δ, and ω using the crossover approach. It motivated the special version of GWO by adding the omega wolf to participate in changing the positions of the grey wolves via Eq. (23). The attempt of reducing the impact rate of any of the best solutions by increasing the number of the wolves that participated in the decision led to reduction in the impact rate of the decision of any wolf from 0.33 to 0.25. This is described in detail in Eqs. (24)–(28). The central updating equation is formed in Eq. (24).

$$W_i^{t+1} = \text{``Crossover''}(w_1 \cdot w_2 \cdot w_3 \cdot w_4) \tag{24}$$

where w_1, w_2, w_3, and w_4 are binary vectors that represent the wolf move impact on alpha, beta, delta, and **omega** grey wolves in sequence. w_1, w_2, w_3, and w_4 were mathematically determined in Eqs. (14), (17), (20), and (25), respectively.

$$w_4^d = \begin{cases} 1 & \textit{if}\left(w_{\omega}^d + \text{stepb}_{\omega}^d\right) \geq 1 \\ 0 & \text{otherwise} \end{cases}, \tag{25}$$

where $\mathbf{w_{\omega}^d}$ is the location vector of the omega wolf in **d** and $\mathbf{stepb_{\omega}^d}$ is a binary step in dimension **d.** Equation (26) is used to determine these.

$$\text{stepb}_{\omega}^d = \begin{cases} 1 & \text{if stepc}_{\omega}^d \geq \text{rand} \\ 0 & \text{otherwise} \end{cases}, \tag{26}$$

where rand is a randomly selected number from uniform distribution $\in [\mathbf{0} \cdot \mathbf{1}]$ and $\mathbf{stepc_{\omega}^d}$ is the continuous valued step size for dimension **d.** Equation (27) is used to calculate the sigmoidal function.

$$\text{stepc}_{\omega}^d = \frac{1}{1 + e^{-10\left(A_4^d Di_{\omega}^d - 0.5\right)}}, \tag{27}$$

where $\boldsymbol{A_4^d}$ and $\boldsymbol{D_{i\omega}^d}$ are mathematically determined by Eqs. (3) and (9) in dimension **d**.

A simple strategy of random probability distribution crossover was implemented per dimension to crossover w_1, w_2, w_3, and w_4 outcomes, as illustrated in Eq. (28).

$$w_d = \begin{cases} w_1^d & \text{if rand} < \frac{1}{4} \\ w_2^d & \frac{1}{4} \leq \text{rand} < \frac{2}{4} \\ w_3^d & \frac{2}{4} \leq \text{rand} < \frac{3}{4} \\ w_4^d & \text{otherwise} \end{cases}. \quad (28)$$

Algorithm 3 outlines the modified binary grey wolf optimisation (MBGWO) algorithm.

Initialize *the grey wolf population Wi(i=1,2,……n)*
Initialize *b, A, and C*
Calculate *the fitness of each search agent*
w_α *= the first best search agent*
w_β *= the second best search agent*
w_δ *= the third best search agent*
w_ω *=the fourth best search agent*
While *(iter < Max number of iterations)*
  For *each search agent*
  Calculate *w1, w2, w3 using the Equation (14), (17), (20), and (25)*
  Calculate *wd using Equation (28)*
  End for
  Update *b, A, and C*
  Calculate *the fitness of all search agent*
  Update w_ω, w_β, w_δ, *and* w_ω
iter=iter+1
end while
return w_α

The MBGWO can be considered as an enhanced version of the original bGWO in [26]. However, it displayed varying solution update (next position move) process based on positions of four wolves, instead of three. The MBGWO also exerted different decision impact rate calculation, which exhibited a great impact on the algorithm's search performance. Finally, although the new position update process of the MBGWO algorithm increased the overall processing time of the algorithm, it did improve the overall performance of the algorithm.

4.1 Proposed fitness function

The basic component of MBGWO is fitness function, which is used to evaluate if a subset meets the objectives. The significant parameters of accuracy and number of features have been applied by researchers for fitness function and to assess each feature subset. The general performance of IDS is based on how efficiently it detects intrusions and the accurate diagnosis of the attacks. The detection rate or classification accuracy, as well as the number of features, is a significant factor in detecting intrusions in network security. The fitness function, which is used in the model, is as follows:

$$\text{Fitness} = P \cdot a + \left(\frac{1}{\text{NF}}\right) \cdot b, \quad (29)$$

where NF is the number of subset features for classification, while *P, a,* and *b* depend on the empirical scope. The discussion of the important use of the expression $\left(\frac{1}{NF}\right)$ in Eq. (29) is illustrated in the following example:

Suppose that the value of **P** depends on the accuracy of classifier and is equal to 99.55%, the value of **a** is equal to 0.6 depending on the empirical scope, the value of **b** is equal to 0.4 based on the value of **a**, and the value of NF is equal to 18. Based on Eq. (29), the fitness is equal to 59.752. However, if the number of features is increased to 21, the value of fitness will be equal to 59.749. This indicates that the decreasing number of features increases the value of fitness.

5 Evaluation of the proposed algorithm

The proposed algorithm is evaluated using different evaluation scenarios as shown in Sect. 5.1. The IDS performance metrics such as accuracy (AC), detection rate (DR), and false positive rate (FPR) are presented in Sect. 5.2.

5.1 Evaluation scenarios

This section evaluates the robustness of the proposed algorithm using NSL-KDD (KDD99 modified version) data set with various attack-based scenarios. The characteristic of NSL-KDD data set is provided in Sect. 5.1.1, while attack-based scenarios are presented in Sect. 5.1.2

5.1.1 NSL-KDD data set

The NSL-KDD data set was used in this study because it is an effective data set in comparing different intrusion detection methods. The number of training and testing data sets in NSL-KDD seemed reasonable. The experiments were performed in the whole data set, and not in a selected portion. The NSL-KDD has certain advantages:

- Redundant records are excluded from the data set and, therefore, there are fewer chances of the classifier biasing towards frequent records.
- A better detection rate as there are less/no duplicate records in the data set.
- Different evaluation results can be compared as it is affordable to run the testing and training data sets. The NSL-KDD data set is, therefore, used for testing and training purposes.

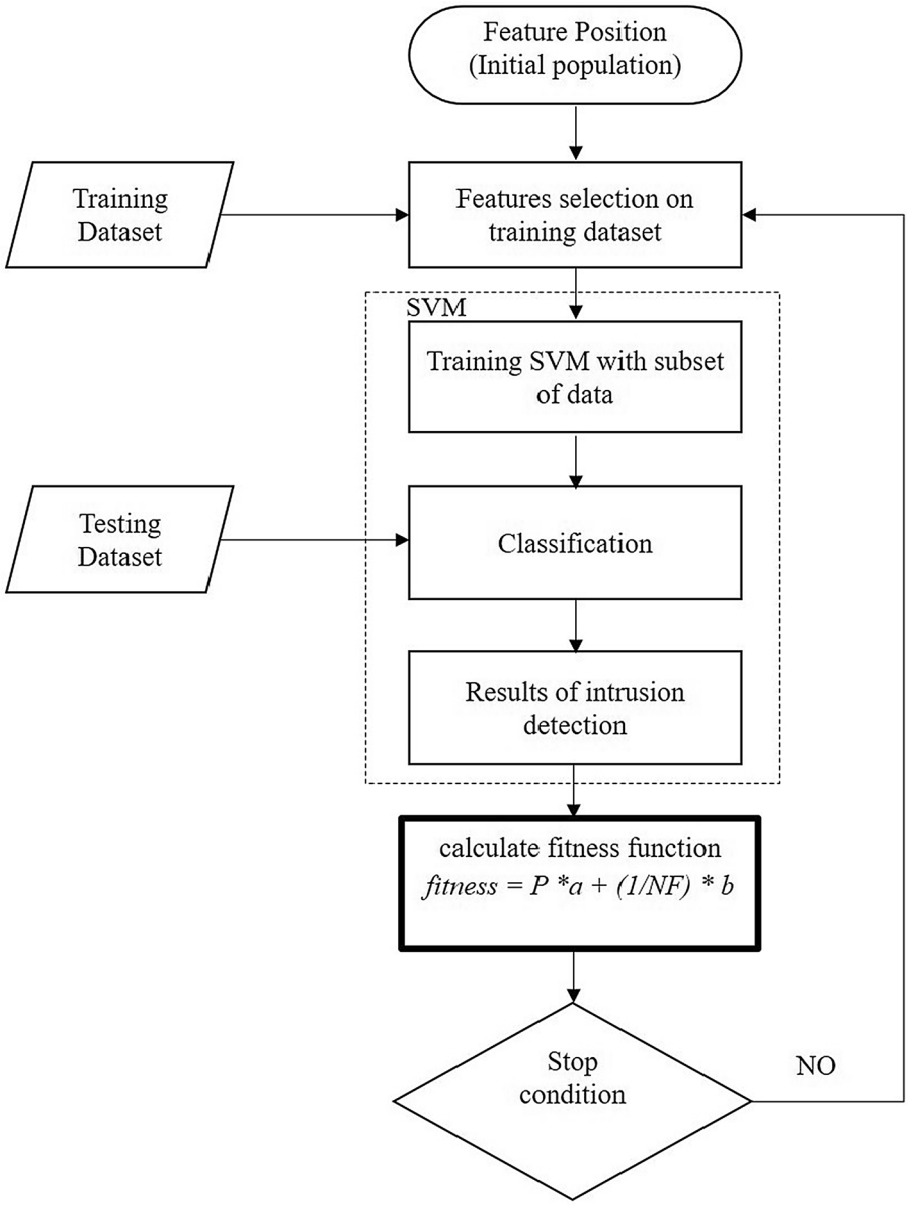

Fig. 3 Flow chart of the proposed MBGWO feature selection algorithm [25]

The NSL-KDD training data set was made up of 21 different attacks, and 37 attacks were present in the testing data set. In addition to known attacks, 14 unknown attacks were found in the testing data set, which were absent in the training data set and made it more difficult for any conventional ID method. Each attack was classified under one of the four categories: DoS, Probe, U2R, and R2L.

- Denial of service attack (Dos): in this attack, the attacker keeps memory resources too busy that memory is not available to handle the request of legitimate users, such as back, land, neptune, pod, smurf, and teardrop.
- Probe attack: the attacker collects information about computer network by sending probing messages to them to access security controls of the machine, such as ipsweep, nmap, portsweep, and satan.
- User to root attack (U2R): the attacker starts by obtaining some legitimate users' credits and exploits system weaknesses to obtain root users' rights, such as buffer_overflow, loadmodule, perl, and rootkit.
- Remote to local attack (R2L): the attacker sends packets to a machine on the network by exploiting system loopholes (vulnerable points) and becomes a user of the remote machine, such as ftp_write, guess_-passwd, imap, multihop, phf, spy, warezclient, and warezmaster.

Figure 4 portrays the major types of attacks in both training and testing data sets.

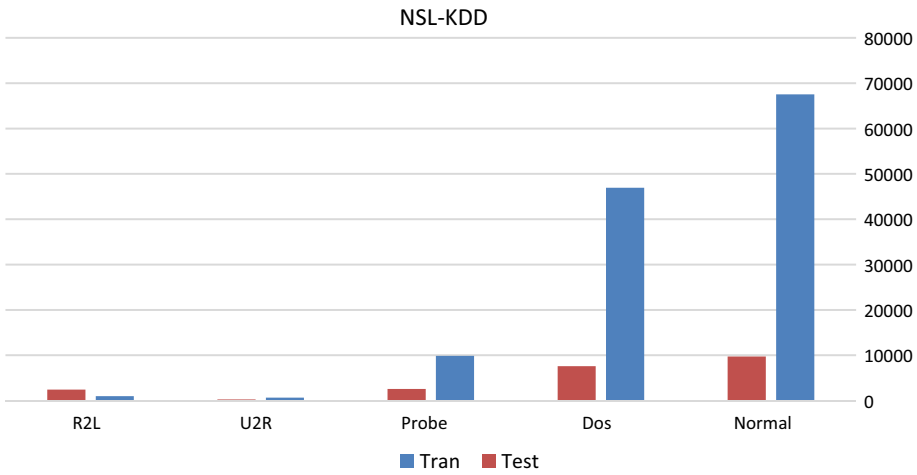

Fig. 4 Major types of attacks in both training and testing data sets

5.1.2 Scenarios

The proposed approach parameters for varying scenarios using NSL-KDD are presented in Table 1. The simulations were performed in MATLAB using SVM classifier with simulation parameters, as tabulated in Table 1.

5.1.2.1 Scenario 1 This scenario compared the four FS methods: GWO, MGWO, bGWO, and MBGWO, based on the average accuracy performance (AAP) and the average of the subset selected features (ANF), which were calculated using Eqs. (30) and (31), respectively.

$$\mathrm{AAP} = \frac{\sum_{i=1}^{\mathrm{run}} \mathrm{AC}_i}{\mathrm{run}} \tag{30}$$

$$\mathrm{ANF} = \frac{\sum_{i=1}^{\mathrm{run}} \mathrm{NF}_i}{\mathrm{run}}, \tag{31}$$

where *AC* is the accuracy rate and *NF* is the number of features.

Based on this comparison, the SVM was used as a classifier that applied the separated *KDD train* with 125,973 records and *KDDTest+* with 22,544 records, where the fitness function, which depended on the detection rate and the number of selected features, is used in Eq. (32).

$$\mathrm{Fitness} = \mathrm{DR} \cdot a + \left(\frac{1}{\mathrm{NF}}\right) \cdot b, \tag{32}$$

where DR is the detection rate, NF is the number of features, and the values of a and b are as given in Table 1 based on the toning for the scope of scenario. It is worth noting that based on the conducted experimental tests, the detection rate for the separated data set was low and, therefore, the researchers suggested using it as the target for the fitness function measurement (for this scenario), instead of accuracy (see Sect. 4.1). This step had significantly improved the accuracy results.

5.1.2.2 Scenario 2 This scenario validated the proposed MBGWO approach. The testing average performance accuracy, the average number of features, as well as the best and the worst accuracy evaluation parameters of the proposed system, had been compared with other different FS methods, such as bGWO [26], binary PSO [27], and binary BAT [28]. Based on this scenario, the data set merged between *KDDTrain-20Percent* and *KDDTest-21*. Next, 80% were selected from the data set as train data and 20% as test data. The fitness function used in this scenario relied on the accuracy rate as a fitness parameter according to Eq. (33).

$$\mathrm{Fitness} = \mathrm{AC} \cdot a + \left(\frac{1}{\mathrm{NF}}\right) \cdot b, \tag{33}$$

where AC is accuracy, NF is the number of features, and the values of a and b are given in Table 1.

5.1.2.3 Scenario 3 For an intensive evaluation of the proposed detection method, the MBGWO approach was compared with other state-of-the-art algorithms from the literature. The comparison focused on the four main evaluation parameters for the IDS issue: AC, DR, FPR, and the number of features selected. The outputs of the compared algorithms are presented in Table 8.

5.2 Evaluation metric

Several criteria for IDS performance, such as DR, FPR, and AC, were calculated to approve the proposed method in Eqs. 34–36.

$$\mathrm{AC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{34}$$

$$\mathrm{DR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{35}$$

Table 1 Simulation parameters settings

Parameters	Scenario 1	Scenario 2	Scenario 3
Number of run method **(run)**	20 runs	20 runs	20 runs
Number of population **(np)**	12	12	12
Number of iterations **(iter)**	20 iterations	20 iterations	20 iterations
Kernel function to SVM	RBF	RBF	RBF
a in fitness function equal	0.6	0.6	0.6
b in fitness function equal	0.4	0.4	0.4

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{TN} + \mathrm{FP}}, \tag{36}$$

where TP, TN, FP, and FN are the confusion matrices that represent true and false classification results.

- True positive (TP): intrusions that are successfully detected by the IDS.
- False positive (FP): normal behaviour that is wrongly classified as intrusive by the IDS.
- True negative (TN): normal behaviour that is successfully labelled as normal by the IDS.
- False negative (FN): intrusions that are missed by the IDS and classified as normal.

The following are the possibilities of classifying events, as portrayed in Table 2.

Table 2 Confusion matrices

Actual	Predicted	
	Abnormal	Normal
Abnormal	TP	FN
Normal	FP	TN

5.3 Evaluation results

5.3.1 Scenario 1

The experimental test of scenario 1 was performed. The results of comparing anomaly-based IDS FS algorithms with AAP and ASF are given in Table 3.

Table 3 Results of feature selection methods for Scenario 1

Algorithm	AAP (%)	ANF
GWO	79.66	28
MGWO	79.66	**24**
bGWO	81.07	26
MBGWO	**81.58**	26

Based on the results presented in Table 3, the compared algorithm that included the original grey wolf GWO generated AAP at 79.66% with ANF of 28 features. The MGWO achieved a similar AC with a smaller number of 24 features. The bGWO achieved 81.07% as AAP with the average of 26 features. The MBGWO attained 81.58% with a similar number of 26 features. Based on the results, the modification made to the original GWO enhanced the next location selection process. In MGWO, the opinion of the fourth grey wolf (omega) was considered for the next location decision-making. In bGWO, stochastic crossover strategy, which was added to the original GWO, improved the next position selection (refer to Sect. 3.3, Eq. 23). The MBGWO outcomes emphasised omega wolf significant improvement for the decision-making process when it was combined with bGWO. In conclusion, based on Table 3, the experimental outputs highlight the powerful impact of the two added modifications (binary decision and omega) on the original GWO for the IDS problem.

5.3.2 Scenario 2

Tables 4, 5, 6, and 7 illustrate the validation results of the anomaly-based IDS FS approach using NSL-KDD data set (separated training and testing data) for Scenario 2. Different classes of attacks were evaluated to detect AC with two state-of-the-art methods so as to critically evaluate the proposed MBGWO performance of the challenging IDS problem.

Based on Tables 4, 5, 6, and 7, the experimental results indicated that the proposed MBGWO outperformed the other state-of-the-art methods in terms of ANF with reasonably close values to AAP. In precise, MBGWO can balance between searching for a significant AC and a smaller number of features. This reflects the advantage of multi-objective fitness function (see Sect. 4.1), as applied in MBGWO. For example, the AAP for detecting a normal class versus anomaly class using MBGWO is 98.26 with 20 features only. The AAPs for the same classes using binary GWO, binary PSO, and binary BAT methods were 98.21, 98.41, and 98.54% with 23, 26, and 25 features, respectively. In comparing the best detection AC that was achieved by the proposed MBGWO (98.31), 16 features only with that of bGWO, binary PSO, and binary BAT (98.35%, 98.51%, and 98.57% with 20, 23, and 23 features), a slight difference was found in terms of AC.

Table 4 Results of MBGWO on NSL-KDD data set

Class	AAP	ANF	Best accuracy	Feature number	Worst AC	Feature number
Normal	98.26	20	98.31	16	97.86	14
Dos	99.42	17	99.55	14	98.95	13
Probe	98.66	16	98.94	14	98.00	11
U2R	99.59	12	99.61	10	99.52	10
R2L	97.36	18	97.48	14	96.67	11

Table 5 Results of the bGWO on NSL-KDD data set

Class	AAP	ANF	Best AC	Feature number	Worst AC	Feature number
Normal	98.23	21	98.35	20	97.87	16
Dos	99.40	18	99.48	16	99.55	20
Probe	98.58	18	98.52	14	98.59	20
U2R	99.59	12	99.64	9	99.61	14
R2L	97.33	19	97.24	15	96.97	13

Table 6 Results of binary PSO on NSL-KDD data set

Class	AAP	ANF	Best AC	Feature number	Worst AC	Feature number
Normal	98.41	26	98.51	23	98.28	23
Dos	99.73	23	99.79	24	99.62	25
Probe	98.88	22	98.94	19	98.79	20
U2R	99.77	18	99.81	18	99.71	20
R2L	97.58	23	97.64	25	97.51	25

Table 7 Results of the binary BAT on NSL-KDD data set

Class	AAP	ANF	Best AC	Feature number	Worst AC	Feature number
Normal	98.54	25	98.57	23	98.40	20
Dos	99.79	23	99.83	20	99.70	23
Probe	98.96	21	99.04	20	98.79	20
U2R	99.80	18	99.87	16	99.68	21
R2L	97.74	22	97.89	21	97.54	18

However, as for features, MBGWO exhibited a significant variance.

In general, based on the compared results of all classes of attacks, MBGWO reduced the number of features required for the detection process for an acceptable AC. For instance, the best AC value, which was achieved by MBGWO for the five evaluated classes of attacks, ranged between 97.48 and 99.61%, while the number of features selected for the best accuracy ranged between 10 and 16 features only. However, for bGWO, binary PSO, and binary BAT methods, the best AC values ranged between 97.24 and 99.64%, 97.64 and 99.81%, and 97.89 and 99.87%, while the numbers of features selected ranged between 9 and 20, 18 and 25, and 16 and 27, respectively. In conclusion, there was no significant variance between the best accuracy, which was achieved by the compared methods for all types of attacks, in comparison with the big difference in the number of the used features by the proposed method. The proposed MBGWO method drove the optimisation search towards maximum detection AC and a minimal number of features employed for the detection.

5.3.2.1 Convergence of MBGWO This section presents the performance of convergence for MBGWO and bGWO. The bGWO has always been ineffective in addressing FS with multiple objectives as it could only detect a handful of optimal solutions within a single run, which implies running for a number of times at attaining a certain number if features. The MBGWO algorithm presented in this present study is based on population with the metaheuristic approach that can seek numerous solutions within a run. Figure 5 illustrates the optimum sets of features gained by bGWO and MBGWO upon addressing NSL-KDD data sets.

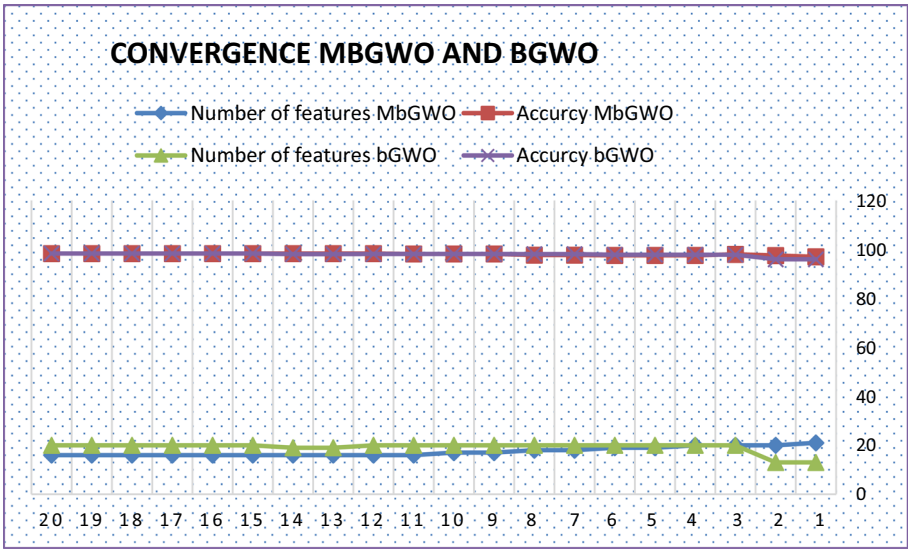

Fig. 5 Number of feature sets gained by MBGWO and bGWO

Based on Fig. 5, upon selecting 16 and 17 features, the MBGWO gave better outcomes than bGWO for AC. Nevertheless, the values of AC in MBGWO increased from 97.01 to 98.31% with the decrease in the number of features from 21 until 16. On the contrary, the bGWO displayed unstable curve, wherein the AC values hit approximately 98.35% with increment in the number of features.

5.3.3 Scenario 3

Table 8 shows the MBGWO approach, in comparison with other state-of-the-art algorithms derived from the literature.

Based on Table 8, the results of the proposed MBGWO approach revealed significant improvements in the algorithm in terms of balancing between the maximised AC and DR, the minimised FPR, and the number of features. In precise, the proposed algorithm displayed the capability of attaining better AC, DR, and FPR with smaller number of features, when compared to the other modified algorithms derived from the literature.

The multi-objective fitness function, which was incorporated in MBGWO, demonstrated a significant performance. It enhanced the overall performance of the approach by enabling the algorithm to search for more precise features to reduce the number of features applied for the detection process. Based on Table 7, the outputs indicated that the algorithm achieved 99.22% AC with 99.10% DR, and FPR of 0.006% using 14 features only. This exhibited an exceptional performance considering the two conflicting objectives of the IDS problem, i.e. to increase AC of classification and to decrease the number of features so as to minimise high-dimensional search space.

Table 8 MBGWO, in comparison with state-of-the-art algorithms from the literature

Algorithm	AC%	DR%	FPR%	Number of features
Binary GWO	**99.50**	–	–	24
AdaBoost	98.90	**99.61**	0.014	25
PSO-discretize-HNB	98.20	98.00	0.014	**11**
MBGWO	99.22	99.10	0.0064	14

6 Conclusion and future work

An MBGWO that addresses IDS problem is proposed in this study. A fourth wolf schema was added to the GWO with a binary formulation for the IDS FS process. The new proposed algorithm was tested on the NSL-KDD data set. Varied scenarios were performed to carry out a thorough evaluation upon MBGWO and bGWO. The scenarios were based on the type of data set separation, where the subsets of training and testing data played a key role in FS and intrusion detection accuracy. Varied evaluation metrics were applied for the algorithm experimental evaluation process. These metrics included accuracy, detection rate, false positive rate, and the number of features selected, each of which reflected a specific performance measurement. The findings concluded that the MBGWO is indeed a practical method to address IDS problems. The ability of the algorithm to increase the accuracy value and to decrease the number of features for the detection process significantly enhanced the performance of the IDS. It was discovered that the added multi-objective fitness function with the fourth grey wolf had a direct impact on the algorithm's next position selection process. The obtained

experimental outcomes, when compared with other modified algorithms, demonstrated that the proposed MBGWO had a higher impact on the IDS problem.

Other aspects can be considered as improvements for future studies. The next location decision can be further enhanced by adapting the velocity parameter of the PSO algorithm, where three wolves of the original GWO may participate in the next location decision, similar to local-best and global-best locations, upon participating in PSO algorithm. In precise, the next location update equation, which is used in the PSO, could be amalgamated into the original GWO, so that the three wolves could participate in the velocity calculation process.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Kim G, Lee S, Kim S (2014) A novel hybrid intrusion detection method integrating anomaly detection with misuse detection. Expert Syst Appl 41(4):1690–1700
2. Anbar M, Abdullah R, Hasbullah IH, Chong YW, Elejla OE (2016) Comparative performance analysis of classification algorithms for intrusion detection system. In: 2016 14th annual conference on privacy, security and trust (PST). IEEE, pp 282–288
3. Hamed T, Ernst JB, Kremer SC (2018) A survey and taxonomy on data and pre-processing techniques of intrusion detection systems. Computer and network security essentials. Springer, Cham, pp 113–134
4. Debar H, Dacier M, Wespi A (2000) A revised taxonomy for intrusion-detection systems. Ann Telecommun 55(7):361–378
5. Balasaraswathi VR, Sugumaran M, Hamid Y (2017) Feature selection techniques for intrusion detection using non-bio-inspired and bio-inspired optimization algorithms. J Commun Inf Netw 2(4):107–119
6. Pereira LAM, Rodrigues D, Almeida TN, Ramos CC, Souza AN, Yang XS, Papa JP (2014) A binary cuckoo search and its application for feature selection. In: Cuckoo search and firefly algorithm. Springer, Cham, pp 141–154
7. Kumar K, Batth JS (2016) Network intrusion detection with feature selection techniques using machine-learning algorithms. Int J Comput Appl 150(12):1–13
8. Kabir M, Shahjahan M, Murase K (2013) Ant colony optimization toward feature selection. In: Ant colony optimization-techniques and applications. InTech
9. Xue B, Fu W, Zhang M (2014) Multi-objective feature selection in classification: a differential evolution approach. In: Asia-Pacific conference on simulated evolution and learning. Springer, Cham, pp 516–528
10. Shoghian S, Kouzehgar M (2012) A Comparison among wolf pack search and four other optimization algorithms. World Acad Sci Eng Technol 6(12):447–452
11. Bamakan SMH, Amiri B, Mirzabagheri M, Shi Y (2015) A new intrusion detection approach using PSO based multiple criteria linear programming. Proc Comput Sci 55:231–237
12. Mazini M, Shirazi B, Mahdavi I (2018) Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2018.03.011
13. Sailaja M, Kumar RK, Murty PSR, Prasad PESNK (2012) A novel approach for intrusion detection using swarm intelligence. In: Proceedings of the international conference on information systems design and intelligent application held in Visakhapatnam, India. January 2012. Springer, Berlin, Heidelberg, pp 469–479
14. Parsian A, Ramezani M, Ghadimi N (2017) A hybrid neural network-gray wolf optimization algorithm for melanoma detection. Biomed Res 28(8):3408–3411
15. Mittal N, Singh U, Sohi BS (2016) Modified grey wolf optimizer for global engineering optimization. Appl Comput Intell Soft Comput 8:1–16
16. Zhu A, Xu C, Li Z, Wu J, Liu Z (2015) Hybridizing grey wolf optimization with differential evolution for global optimization and test scheduling for 3D stacked SoC. J Syst Eng Electron 26(2):317–328
17. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61
18. Seth JK, Chandra S (2016) Intrusion detection based on key feature selection using binary GWO. In: 3rd international conference on computing for sustainable global development, Mar 2016. IEEE, pp 3735–3740
19. Faris H, Aljarah I, Al-Betar MA, Mirjalili S (2018) Grey wolf optimizer: a review of recent variants and applications. Neural Comput Appl 30:413–435
20. Al-Betar MA, Awadallah MA, Faris H, Aljarah I, Hammouri AI (2018) Natural selection methods for Grey Wolf Optimizer. Expert Syst Appl 113:481–498
21. Devi R, Suganthe RC (2017) Feature selection in intrusion detection grey wolf optimizer. Asian J Res Soc Sci Humanit 7(3):671–682
22. Elngar AA, El DA, Mohamed A, Ghaleb FFM (2013) A real-time anomaly network intrusion detection system with high accuracy. Inf Sci Lett 2(2):49–56
23. Amudha P, Karthik S, Sivakumari S (2015) A hybrid swarm intelligence algorithm for intrusion detection using significant features. Sci World J 1:1–16
24. Kim DS, Nguyen HN, Park JS (2005) Genetic algorithm to improve SVM based network intrusion detection system. In: 19th international conference on advanced information networking and applications (AINA), vol 2. IEEE, pp 155–158
25. Gharaee H, Hosseinvand H (2016) A new feature selection IDS based on genetic algorithm and SVM. In: 8th international symposium on telecommunications (IST). IEEE, pp 139–144
26. Emary E, Zawbaa HM, Hassanien AE (2016) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172:371–381
27. Lee S, Soak S, Oh S, Pedrycz W, Jeon M (2008) Modified binary particle swarm optimization. Prog Nat Sci 18(9):1161–1166
28. Mirjalili S, Mirjalili SM, Yang XS (2014) Binary bat algorithm. Neural Comput Appl 25(3–4):663–681