Computer Systems Science & Engineering
DOI: 10.32604/csse.2023.036119
Article

Tech Science Press

An Improved Reptile Search Algorithm Based on Cauchy Mutation for Intrusion Detection

Salahahaldeen Duraibi*

Department of Computer and Network Engineering, Jazan University, Jazan 82822-6649, Saudi Arabia
*Corresponding Author: Salahahaldeen Duraibi. Email: Sduraibi@jazanu.edu.sa
Received: 17 September 2022; Accepted: 21 December 2022

Abstract: With the growth of the discipline of digital communication, the topic has acquired more attention in the cybersecurity medium. The Intrusion Detection (ID) system monitors network traffic to detect malicious activities. The paper introduces a novel Feature Selection (FS) approach for ID. Reptile Search Algorithm (RSA)—is a new optimization algorithm; in this method, each agent searches a new region according to the position of the host, which makes the algorithm suffers from getting stuck in local optima and a slow convergence rate. To overcome these problems, this study introduces an improved RSA approach by integrating Cauchy Mutation (CM) into the RSA's structure. Thus, the CM can effectively expand search space and enhance the performance of the RSA. The developed RSA-CM is assessed on five publicly available ID datasets: KDD-CUP99, NSL-KDD, UNSW-NB15, CIC-IDS2017, and CIC-IDS2018 and two engineering problems. The RSA-CM is compared with the original RSA, and three other state-of-the-art FS methods, namely particle swarm optimization, grey wolf optimization, and multi-verse optimizer, and quantitatively is evaluated using fitness value, the number of selected optimum features, accuracy, precision, recall, and F1-score evaluation measures. The results reveal that the developed RSA-CM got better results than the other competitive methods applied for FS on the ID datasets and the examined engineering problems. Moreover, the Friedman test results confirm that RSA-CM has a significant superiority compared to other methods as an FS method for ID.

Keywords: Feature selection; intrusion detection; metaheuristic algorithms; reptile search algorithm; cauchy mutation

1 Introduction

Due to the increased internet usage rate caused by the widespread computer networks, security has become one of the most critical areas for research because of the threats and attacks on these networks, which are now more aggressive than before [1]. Several security technologies are employed to deal with and prevent attacks, such as firewalls, authentication, and encryption. Despite the powerful capabilities of these technologies, they have limitations in reaching the desired level of attack detection. ID system

and intrusion prevention system can analyze data passing the networks in greater depth compared to other security systems, are used to overcome the issue of these technologies,.

With the increase in the number of attacks, cybersecurity companies focus on developing sensitive systems besides traditional security methods [2–4]. As a result, proactive cybersecurity systems such as network behavior analysis, threat analysis, and Machine learning (ML) are also developed. ID systems are frequently used technology that has become more sensitive to cyber threats. ID system is a software package that is responsible for detecting threats across the network or system.

In order to achieve optimal security requirements of a network, researchers have focused on the use of ML approaches to develop an ID system that can detect such types of attacks more accurately [5,6]. ML techniques gained special attention in ID in recent years because of their capabilities to classify hundreds of features into normal system behavior or attack attempt [7,8]. The primary purpose of Feature Selection (FS) as a technique is to select an Optimal Feature Subset (OFS) in a given dataset, thus, optimizing the learning process by the ML techniques.

Selecting OFS in a given dataset facilitates learning by ML techniques to achieve better prediction, and classification results for ID. Nature-inspired algorithms are mostly Meta-Heuristics (MH) optimization methods inspired by nature. They gained special attention from scholars in different applications due to their great potential to specify OFS. These methods are effective, and reliable gradient-free stochastic optimization techniques that have been successful in various numerical, and combinatorial optimization problems with diverse frameworks [9–11]. MH inspiration sources are broken down into three types [12]: swarm-based algorithms, evolutionary-based algorithms, and physics-based algorithms. Some of the more popular MH methods include Multi-Verse Optimizer (MVO) [13], Particle Swarm Optimization (PSO) [14], Genetic Algorithm (GA) [15], Salp Swarm Algorithm (SSA) [16], Whale Optimization Algorithm (WOA) [17], Gray Wolf Optimizer (GWO) [18], and Reptile Search Algorithm (RSA) [19].

MH algorithms can be combined to achieve better results for FS in different applications. The authors in [20] combined RSA with Remora Optimization Algorithm (ROA) for data clustering. In another work [21], RSA is combined with deep learning for ID. In [22], chaotic-map, and simulated annealing are used to improve RSA for FS in Medical field. In [23], the authors used Levy flight to improve the capability of the RSA for vehicle cruise control system design. In [24], ant colony optimization's capability is boosted by RSA for churn prediction. In [25], the mean transition mechanism is used to improve RSA for constrained engineering problems. In [26], an enhanced GA based FS method, named GbFS, is presented to increase classification detection accuracy. In [27], the authors proposed a hybrid model based on the correlation feature selection (CFS) with three different search techniques: Best-first, greedy stepwise and GA for ID. In another work [28], the authors used Intelligent Water Drops (IWD) method to choose OFS in KDD-CUP99 dataset.

These methods use two principles that are characteristic in all optimization techniques, which are exploration and exploitation. In exploration, the algorithm tries to find different regions in the search area, while the second principle, exploitation, and the method searches around the obtained solution from the first phase to find the best solutions. In this paper, an improved version of RSA, named RSA-CM for ID is introduced. The RSA-CM combines the original RSA with CM to enhance the exploration capability and maintain a balance between exploration, and exploitation of the RSA. The main contributions of this work could be summarized as follows:

- An improved version of RSA using CM named RSA-CM is introduced for ID.
- CM strategy is used to boost the search mechanism of the RSA during the search process.

- The RSA-CM is examined using five open access datasets for ID, and two popular engineering optimization problems.
- The results confirm the efficacy of the RSA-CM compared to other MH methods and the engineering problems as well.

This paper is organized as follows: Section 2 provides a brief idea of RSA and CM, followed by a description of the developed method presented in Section 3. The experimental results, and statistical comparison with other FS methods are shown in Section 4, and Section 5 concludes this paper.

2 Method

2.1 Reptile Search Algorithm (RSA)

In 2022, Abualigah *et al*. [19] reported a MH method inspired by the hunting behavior of Crocodiles and is known as RSA. The method initializes the *i*th set of candidate OFS $x_{i,j}$ randomly as follows:

$$x_{i,j} = rand_{\in U(0,1)} * \left(UB_j - LB_j\right) + LB_j \; i \in \{1, \ldots, N\} \; and \; j \in \{1, \ldots, M\} \tag{1}$$

where LB_j and UB_j and are minimum, and maximum values of the *j*th feature, $rand_{\in U(0,1)}$ generates a random number from uniform distribution, N is a maximum number of sets of candidate OFS, and M is the total number of input features.

The crocodiles' food search is implemented in RSA using two separate strategies namely, exploration and exploitation. For sequential implementation of these two strategies, the maximum number of iterations is split into four stages. In the first half of the total number of stages, the crocodile's encircling behavior is implemented using the high and the belly walking movements of the crocodile to effectively explore the region. This stage can mathematically be written as:

$$x_{i,j}(g+1) = \begin{cases} \left[-n_{i,j}(g) \cdot \gamma \cdot Best_j(g)\right] - \left[rand_{\in[1,N]} \cdot R_{i,j}(g)\right], g \leq \frac{T}{4} \\ ES(g) \cdot Best_j(g) \cdot x_{(rand_{\in[1,N]},j)}, g \leq \frac{2T}{4} and \; g > \frac{T}{4} \end{cases} \tag{2}$$

where, for *g*th iteration, *i*th candidate OFS, and *j*th feature $Best_j(g)$ is the best solution, $n_{i,j}$ is the hunting operator (Eq. (3)), and $ES(g)$ is Evolutionary Sense (Eq. (7)) which reduces from 2 to −2 over the total number of iterations, and γ is set as 0.1 for controlling the exploration accuracy. The $R_{i,j}$, computed as in Eq. (6), reduces the search region, and $rand_{\in[1,N]}$ randomly selects one of the candidate OFS.

$$n_{i,j} = Best_j(g) \times P_{i,j} \tag{3}$$

where $P_{i,j}$, calculated as in Eq. (4), is the normalized difference between the *j*th feature value of the *i*th candidate OFS and average value of the *i*th solution. It is calculated as:

$$P_{i,j} = \theta + \frac{x_{i,j} - \mu(x_i)}{Best_j(g) \times \left(UB_j - LB_j\right) + \epsilon} \tag{4}$$

where θ controls the sensitive of the exploration, and ϵ is a minimum floor value. It is defined as:

$$\mu(x_i) = \frac{1}{n}\sum_{j=1}^{n} x_{i,j} \tag{5}$$

$$R_{i,j} = \frac{Best_j(g) - x_{(rand_{\in[1,N]},j)}}{Best_j(g) + \epsilon} \tag{6}$$

$$ES(g) = 2 \times rand_{\in\{-1,1\}} \times \left(1 - \frac{1}{T}\right) \tag{7}$$

where the value 2 acts as a multiplier to provide correlation values in the range [0, 2], and $rand_{\in\{-1,1\}}$ is a random integer between $\{-1, 1\}$.

Crocodiles' hunting coordination and cooperation are implemented to exploit the search space. The exploitation stage can be mathematically represented as:

$$x_{i,j}(g+1) = \begin{cases} rand_{\in[-1,1]} \cdot Best_j(g) \cdot P_{i,j}(g), & g \leq \frac{3T}{4} and\ g > \frac{2T}{4} \\ \left[\epsilon \cdot Best_j(g) \cdot n_{i,j}(g)\right] - \left[rand_{\in[-1,1]} \cdot R_{i,j}(g)\right], & g \leq Tandg > \frac{3T}{4} \end{cases} \tag{8}$$

The algorithm terminates after T iterations while the performance of each set of candidate OFS is evaluated using a predefined Fitness Function (FF). The OFS is a candidate feature set with the smallest FF.

2.2 Cauchy Mutation (CM)

Several mutation operators are introduced in the literature to escape the problem of premature convergence and to improve the performance. Among them, CM shows a powerful capability due to its extended tail probability distribution function, which can enrich the performance, and prevent getting, stuck in any optimization method's local optima.

CM is a continuous probability distribution having two parameters, where p_0 indicates the location parameter and γ is the scale parameter used to determine the shape of the Cauchy distribution [29–31]. CM aims to solve the premature convergence problem and local stagnation problem of any optimization algorithm by taking controlled small steps. The Probability Distribution Function (PDF) of CM is defined as follows:

$$f(p: p_0, \gamma) = \frac{1}{\pi\gamma\left[+\left(\frac{p-p_0}{\gamma}\right)^2\right]} = \frac{1}{\pi}\left[\frac{\gamma}{(p-p_0)^2 + \gamma^2}\right] \tag{9}$$

where γ is set to 1, p equals 0, and p_0 is a random number between [0, 1]. The CM operator is calculated as:

$$\gamma = tan\left(\pi\left(p_0 - \frac{1}{2}\right)\right) \tag{10}$$

3 Proposed Method

In RSA, the exploration phase is performed by encircling the prey, and exploitation is done in the subsequent stages. However, this may result in the method suffering from premature convergence. Accordingly, CM is integrated into the RSA structure to escape from being trapped in local solutions by allowing RSA to jump, and visit new locations in the search space. This will help the RSA control, and balance the exploration, and exploitation abilities during the search process. The flowchart of the RSA-CM is provided in Fig. 1, and the pseudo-code is given in Algorithm 1.

For gth iteration and mth dimension, a Cauchy's parameter (p), generated using Eqs. (9) and (10), is added to the best possible candidate solution of RSA $\left(x_{best}^{RSA}\right)$ as follows:

$$x_{best,m}(g) = x_{best,m}^{RSA}(g) + p_j(g) \tag{11}$$

The performance of the updated solution is calculated using FF, as shown in Eq. (12). It uses a K-Nearest Neighbor (KNN) classifier with five neighbors, and a threshold value of 0.5 as recommended by [24,32]. The candidate OFS that has minimum features, and attains maximum accuracy attains smallest fitness.

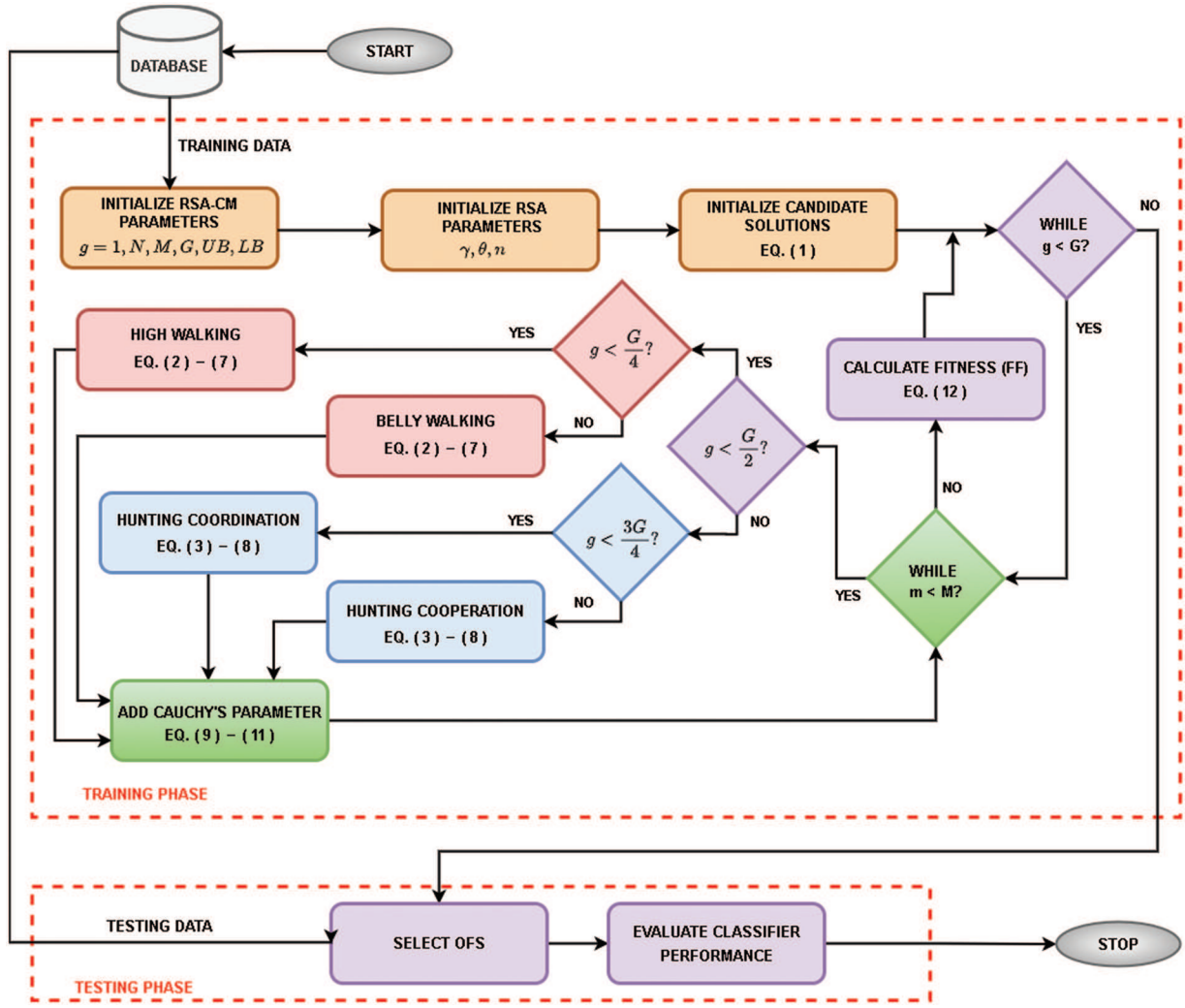

Figure 1: Flow diagram of the developed RSA-CM algorithm

$$FF\,(x_i) = \alpha \times E + (\alpha - 1) \times \frac{|OFS_i|}{M} \tag{12}$$

where E is the classification error rate of the K-Nearest Neighbor (KNN) classifier with five neighbors, $|OFS_i|$ is the cardinality of optimum feature set and M is the cardinality of input feature set of the dataset, and α controls the relative importance of classification error and number of selected features. The value of α varies in the range of [0,1], and is set to 0.99 in this work [32].

Algorithm 1: Pseudo-code for the developed RSA-CM algorithm

1. Cluster the dataset into two exclusive and exhaustive sets for training and testing
Training Phase
2. Load training examples
3. Calculate UB and LB and define fitness function FF (f)
4. Initialize RSA parameters γ, θ, n
5. Initialize candidate OFS Eq. (1) and iteration g = 1
6. while $g < T$ do
7. Initialize feature index $m = 1$
8. while m < M do
9. Calculate the best candidate solution for the current dimension
10. Calculate revised solution for current dimension using Eq. (2)
11. Calculate Cauchy's coefficient for the current dimension using Eq. (9) and (10)
12. Update RSA-CM solutions for the current dimension using Eq. (11)
13. $m = m + 1$
14. Evaluate revised candidate solutions using Eq. (12)
15. $g = g + 1$
16. Use a threshold of 0.5 to choose OFS with smaller FF
Testing Phase
17. Load testing examples
18. Select only OFS
19. Evaluate the classifier performance

4 Experimental Results and Discussion

The capability of the interdicted RSA-CM method to determine the OFS is assessed using five datasets for ID and comparing it with other FS methods: PSO [14], GWO [18], MVO [13], and RSA [19].

4.1 Experimental Setup

Python is used to implement all the methods used in this work and they are executed on a 3.13 GHz PC with 16 GB RAM and Windows 11 operating system. The parameter settings for all the methods are provided in Table 1. These methods are implemented based on their implementations in original work. For all methods, the population of 32 and the maximum iterations of 100 are selected empirically. Each algorithm is executed 20 times independently to obtain reliable analysis and convincing results.

Table 1: Method's parameter settings

Method	Parameters
PSO	$c_1 = c_2 = 2$, $w_{min} = 0.1$ and $w_{max} = 0.9$
GWO	C = random in [0, 2], α & A decrease linearly in range [2, 0] & [1, −1]
MVO	$WEP_{max} = 1$, $WEP_{min} = 0.2$, α decreases from 2 to 0 and $p = 6$
RSA	$\gamma = 0.9, \theta = 0.5$, UB & LB vary based on the features in the dataset
Common settings	Population size = 32, number of runs = 20, & number of iterations = 100

4.2 Datasets Description

Five real datasets from ID applications are selected to assess RSA-CM efficiency. These datasets are widely used for ID [22,23] and they include KDD-CUP99, NSL-KDD, UNSW-NB15, CIC-IDS2017, and CIC-IDS2018. The main characteristics of those datasets are given in Table 2.

Table 2: The datasets characteristics

Dataset	Source	No. of features	No. of samples
KDD-CUP99	[33]	43	494,020
NSL-KDD	[34]	43	125,973
UNSW-NB15	[35]	49	540,044
CIC-IDS2017	[36]	78	2,827,876
CIC-IDS2018	[36]	80	1,048,575

The datasets contain huge number of records for normal activities and network attacks. Using an iterative FS such as MH methods will be computationally expensive. Hence, only 10% of the dataset is used for FS evaluation while maintaining the ratio of natural activities and network attacks.

4.3 Evaluation Metrics

The quantitative evaluation measures employed to compare the proposed RSA-CM and the other MH methods are as follows:

- Fitness values are used to compute the quality of the solution, which is used to guide the searching process by the RSA-CM method.
- The number of OFS is used to illiterate RSA-CM's ability to reduce number of features in a given dataset.
- Accuracy (AC): It calculates the accuracy over the total number of runs and in this work number of runs is 20:

$$AC = \frac{TP + TN}{TP + TN + FN + FP} \tag{13}$$

Precision (P): It measures the actual positives which are actually positive:

$$P = \frac{TP}{TP + FP} \tag{14}$$

Recall (R): It measures the proportion of actual positives which are correctly identified:

$$R = \frac{TP}{TP + FN} \tag{15}$$

F-measure (F): is the harmonic mean of recall and precision measures and it is defined as:

$$F = \frac{2PR}{P + R} \tag{16}$$

where True Positive and (TP) and True Negative (TN) denote the samples of customers correctly detected as churner or not, while False Negative (FN) and False Positive (FP) represents the number of misclassified positive and negative cases, respectively.

4.4 Experimental Results and Discussion

To examine the efficacy of the RSA-CM as an FS method, the real-world datasets provided in Table 1 are used and compared against other MH methods.

Table 3 gives the results of the introduced RSA-CM and the other MH methods using mean and standard deviation (Std) of fitness. From Table 3, the RSA-CM got the lowest mean and Std values in four out of five datasets compared to other methods. The PSO method outperformed the other methods in the CIC-IDS2017 dataset.

Table 3: The fitness values of the RSA-CM against other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	0.0335	0.0220	0.0199	0.0094	**0.0081**
	Std	0.0096	0.0093	0.0073	0.0078	**0.0066**
NSL-KDD	Mean	0.0602	0.0746	0.0687	0.0593	**0.0539**
	Std	0.0081	0.0102	0.0092	0.0093	**0.0088**
UNSW-NB15	Mean	0.0372	0.0318	0.0354	0.0308	**0.0303**
	Std	0.0075	0.0057	0.0052	0.0071	**0.0049**
CIC-IDS2017	Mean	**0.0136**	0.0261	0.0250	0.0151	0.0208
	Std	**0.0060**	0.0084	0.0066	0.0090	0.0082
CIC-IDS2018	Mean	0.0340	0.0300	0.0402	0.0303	**0.0256**
	Std	0.0072	0.0094	0.0093	0.0091	**0.0061**

The results of the proposed RSA-CM and the other MH algorithms based on the mean and standard deviation (Std) of the number of optimum features selected by the corresponding MH algorithm are provided in Table 4. In Table 4, the RSA-CM selected the least-average OFS for three out of five datasets, while for KDD-CUP99, both RSA and RSA-CM selected the least number of features. In the case of CIC-IDS2017, PSO selected least OFS, followed by RSA, RSA-CM, MVO, and GWO. Similarly, Std of number of OFS is least for RSA-CM for three out of five datasets, indicating better stability. For UNSW-NB15, both MVO and RSA-CM show similar Std, while for CIC-IDS2018, RSA and RSA-CM show similar Std. In the case of CIC-IDS2017, PSO shows the least Std of number of OFS, followed by RSA-CM, GWO, MVO, and RSA.

Table 4: The number of OFS of the RSA-CM and other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	40	35	41	**22**	**22**
	Std	5	9	6	7	**3**
NSL-KDD	Mean	38	34	39	37	**31**
	Std	4	6	5	5	**3**

(Continued)

Table 4: Continued

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
UNSW-NB15	Mean	33	29	37	23	**25**
	Std	10	9	**4**	6	**4**
CIC-IDS2017	Mean	23	63	49	25	61
	Std	3	6	7	7	5
CIC-IDS2018	Mean	45	49	71	55	**43**
	Std	10	10	9	**8**	**8**

Table 5 compares different MH algorithms in terms of mean and Std of accuracy. The proposed RSA-CM shows highest mean accuracy for all five datasets. The Std of accuracy is least for the proposed RSA-CM for four out of five datasets indicating high stability of the trained model. In the case of KDD-CUP99, GWO achieves the least Std, followed by the proposed RSA-CM.

Table 5: The accuracy of the RSA-CM and other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	0.9756	0.9860	0.9895	0.9957	**0.9970**
	Std	0.0314	**0.0271**	0.0385	0.0342	0.0294
NSL-KDD	Mean	0.9481	0.9326	0.9398	0.9488	**0.9528**
	Std	0.0231	0.0327	0.0353	0.0726	**0.0120**
UNSW-NB15	Mean	0.9702	0.9747	0.9729	0.9743	**0.9753**
	Std	0.0420	0.0368	0.0391	0.0205	**0.0182**
CIC-IDS2017	Mean	0.9917	0.9884	0.9863	0.9906	**0.9933**
	Std	0.0744	**0.0535**	0.0697	0.0835	**0.0535**
CIC-IDS2018	Mean	0.9762	0.9812	0.9761	0.9823	**0.9842**
	Std	0.0486	0.0529	0.0584	0.0308	**0.0307**

Table 6 compares MH algorithms in terms of mean and Std of precision. The proposed RSA-CM shows the highest mean precision for all five datasets. The Std of precision is least for the proposed RSA-CM for three out of five datasets, indicating consistency of the trained model in detecting the cyber-attacks. In the case of KDD-CUP99, GWO achieves the least Std followed by PSO, RSA-CM, RSA, and MVO. RSA achieves the least Std for CIC-IDS2017 followed by GWO, RSA-CM, PSO, and MVO.

Table 6: The precision of the RSA-CM and other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	0.9846	0.9821	0.9867	0.9916	**0.9968**
	Std	0.0331	**0.0327**	0.0581	0.0464	0.0366
NSL-KDD	Mean	0.9165	0.9181	0.9138	0.9256	**0.9266**
	Std	0.0519	0.0672	0.0616	0.0346	**0.0325**
UNSW-NB15	Mean	0.9633	0.9727	0.9736	0.9745	**0.9753**
	Std	0.0822	0.0529	0.0869	0.0573	**0.0522**
CIC-IDS2017	Mean	0.9801	0.9789	0.9753	0.9779	**0.9909**
	Std	0.0517	0.0226	0.0852	**0.0222**	0.0337
CIC-IDS2018	Mean	0.9738	0.9752	0.9656	0.9734	**0.9765**
	Std	0.0907	0.0620	0.0420	0.0450	**0.0408**

The mean and Std of recall for different MH algorithms are compared in Table 7. The proposed RSA-CM shows the highest mean recall for all five datasets, indicating that the trained model understands cyber-attacks well. The Std of recall is least for the proposed RSA-CM for three out of five datasets showing consistency of the model's understanding and the actual pattern of the cyber-attacks. MVO and GWO achieve the least Std of recall for CIC-IDS2017 and CIC-IDS2018 datasets, respectively.

Table 7: The recall of the RSA-CM and other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	0.9780	0.9793	0.9865	0.9946	**0.9970**
	Std	0.0282	0.0303	0.0270	0.0182	**0.0115**
NSL-KDD	Mean	0.9507	0.9463	0.9468	0.9566	**0.9584**
	Std	0.0541	0.0916	0.0455	0.0517	**0.0432**
UNSW-NB15	Mean	0.9685	0.9736	0.9734	0.9743	**0.9755**
	Std	0.0566	0.0495	0.0465	0.0734	**0.0366**
CIC-IDS2017	Mean	0.9949	0.9832	0.9895	0.9953	**0.9979**
	Std	0.0418	0.0257	**0.0148**	0.0239	0.0227
CIC-IDS2018	Mean	0.9615	0.9571	0.9653	0.9633	**0.9681**
	Std	0.0572	**0.0558**	0.0694	0.0634	0.0703

Table 8 compares different MH algorithms in terms of mean and Std of F1-score. The proposed RSA-CM shows the highest mean F1-score for all five datasets. The Std of F1-score is least for the proposed RSA-CM for two out of five datasets. In the case of KDD-CUP99 and CIC-IDS2018, GWO achieves the least Std followed by RSA-CM; in the case of UNSW-NB15, MVO achieves the least Std followed by RSA-CM.

Table 8: The F1-score of the RSA-CM and other MH algorithms

Dataset	Measure	Method				
		PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	Mean	0.9813	0.9807	0.9866	0.9931	**0.9969**
	Std	0.0863	**0.0126**	0.0726	0.0391	0.0345
NSL-KDD	Mean	0.9333	0.9320	0.9323	0.9414	**0.9417**
	Std	0.0220	0.0257	0.0213	0.0197	**0.0139**
UNSW-NB15	Mean	0.9659	0.9731	0.9735	0.9744	**0.9754**
	Std	0.0737	0.0099	**0.0051**	0.0918	0.0704
CIC-IDS2017	Mean	0.9874	0.9810	0.9823	0.9865	**0.9944**
	Std	0.0435	0.0611	0.0703	0.0838	**0.0199**
CIC-IDS2018	Mean	0.9676	0.9661	0.9654	0.9683	**0.9723**
	Std	0.0790	**0.0068**	0.0491	0.0759	0.0329

Comparative analysis of convergence of RSA-CM and different MH methods is shown in Fig. 2 after 20 independent runs for each method. In Fig. 2, the developed RSA-CM improves the convergence rate towards optimal solutions much better than the other MH algorithms in almost all the used datasets, which reflects the stability of the proposed RSA-CM as an FS method for ID.

Boxplot is used to visualize representations of data distribution of the results in terms of accuracy in three quartiles: lower, middle, and upper. A boxplot of all MH algorithms over five datasets is shown in Fig. 3. This figure shows that the median accuracy of RSA-CM is higher than other MH algorithms for all the five datasets, while upper accuracy is higher in four out of five datasets.

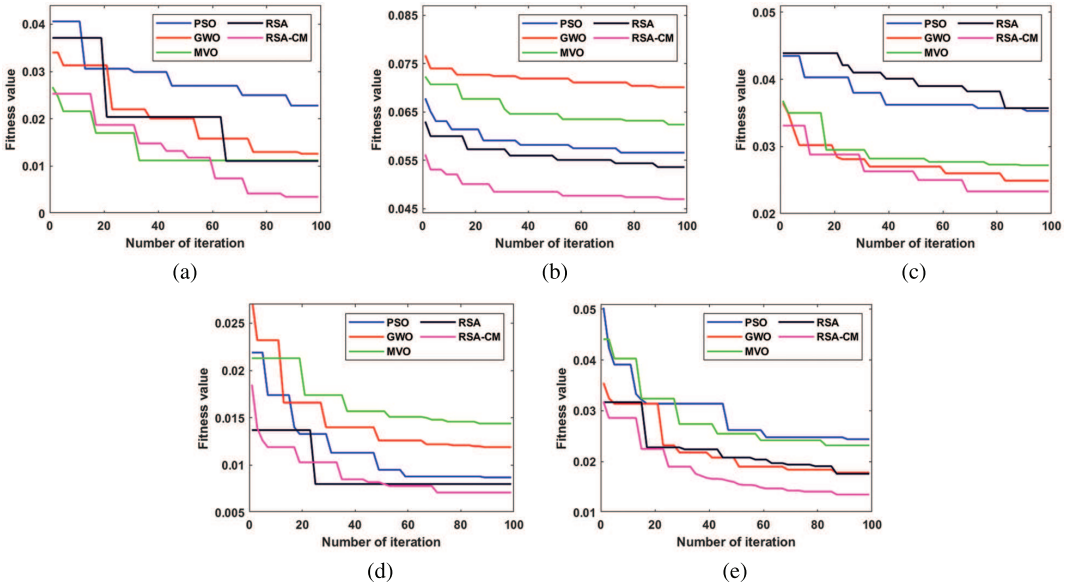

Figure 2: RSA-CM and the other MH methods convergence curves for (a) KDD-CUP99, (b) NSL-KDD, (c) UNSW-NB15, (d) CIC-IDS2017, and (e) CIC-IDS2018

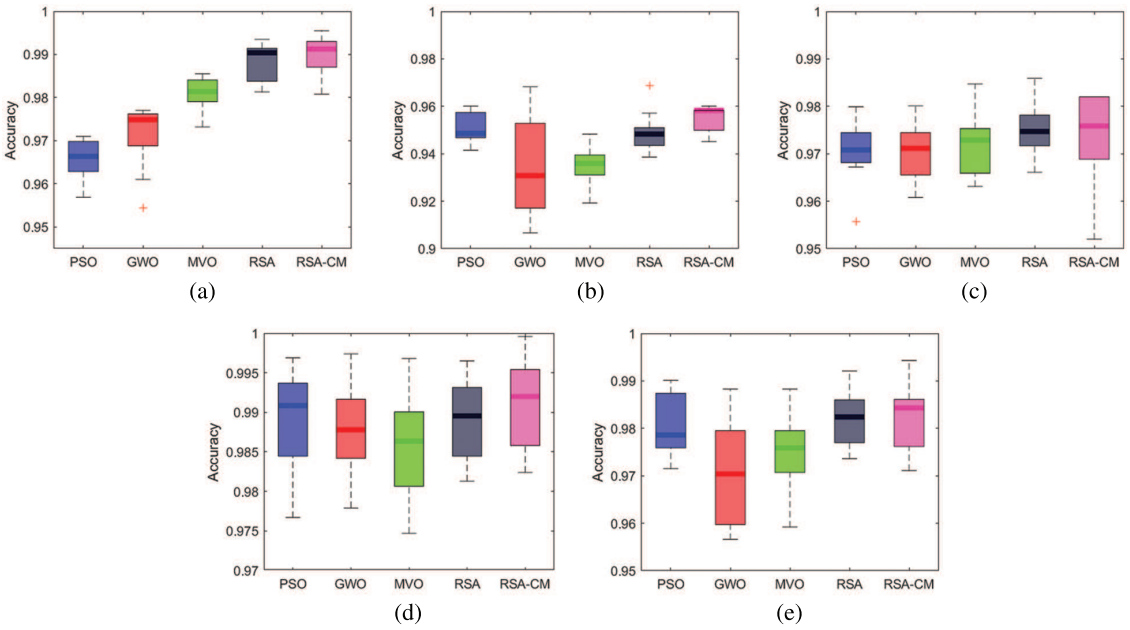

Figure 3: Boxplots of the RSA-CM and the other MH algorithms for (a) KDD-CUP99, (b) NSL-KDD, (c) UNSW-NB15, (d) CIC-IDS2017, and (e) CIC-IDS2018

4.5 Real-World Engineering Problems

The RSA-CM method is employed to solve two engineering problems with constraints, including Pressure Vessel Design (PVD) and Three-bar Truss Design, and the results are provided in this section.

4.5.1 Pressure Vessel Design (PVD)

In this problem, the PVD seeks to minimize the welding cost of the pressure vessel using the constraints on material and shipping. It consists of four variables, as illustrated in Fig. 4. These variables comprise T_s as the shell thickness, T_h as the head thickness, R as the inner radius, and L as the cylindrical-section length. The objective function of the PVD can be represented as:

Minimize

$$f(x) = 0.6224x_1x_2x_3 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \tag{17}$$

Subject to

$$g_1(x) = -x_1 + 0.0193x_3 \leq 0,$$

$$g_2(x) = -x_3 + 0.00954x_3 \leq 0,$$

$$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1,296,000 \leq 0,$$

$$g_4(x) = x_4 - 240 \leq 0, \tag{18}$$

where $(0 \leq x_i \leq 100, i = 1.2)$ and $(10 \leq x_i \leq 200, i = 3.4)$.

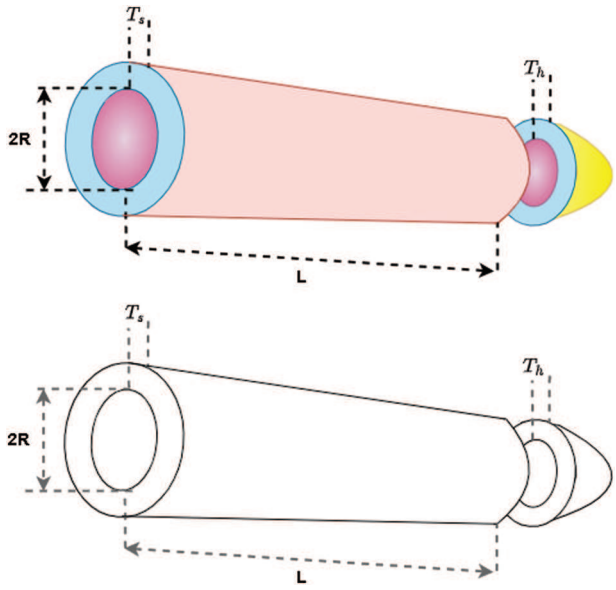

Figure 4: The PVD problem

Table 9 shows the welding cost for different methods used in this work. From this table, one can observe that the RSA-CM has the smallest weight of 2100.7202 compared to PSO, GWO, MVO and RSA, followed by the GWO with an optimal cost of 2101.866 and the PSO ranked last since it gained the highest optimal cost.

Table 9: Welding cost of PVD using different MH methods

Method	Optimal values				Optimal cost
	T_s	T_h	R	L	
PSO	1.0000	0.0000	120.0000	10.5012	2414.0478
GWO	1.2591	0.0000	65.2298	10.0000	2101.8663
MVO	1.2614	0.0000	65.2280	10.1553	2110.2778
RSA	1.0000	0.0000	110.0000	9.5346	2212.5875
RSA-CM	1.2588	0.0000	65.2252	10.0000	**2100.7202**

4.5.2 Three-Bar Truss Design (TBD)

A TBD's optimal design seeks to reduce the structure weight subject to support total load acting vertically downward. The structural geometry of the problem is given in Fig. 5 and its objective function can be written as:

Minimize

$$f(x) = \left(2\sqrt{2x_1} + x_2\right) * l \tag{19}$$

Subject to

$$g_1(x) = \frac{\sqrt{x_1}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0$$

$$g_2(x) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0$$

$$g_3(x) = \frac{1}{\sqrt{2}x_2 + x_1}P - \sigma \leq 0 \tag{20}$$

where $l = 100\ cm, P = 2\ kN/cm^2, \sigma = 2\ kN/cm^2, 0 \leq x_i \leq 1$, and $i = 1.2$.

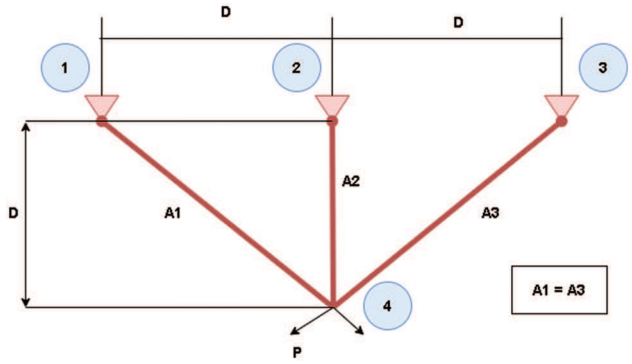

Figure 5: TBD problem

The RSA-CM results for solving the problem of TBD are provided in Table 10. From this table, the RSA-CM gives the best outcomes since it gained 317.3389, which is the smallest weight in comparison to other MH methods. Then GWO method ranked second while MVO ranked last for the problem of TBD.

Table 10: Structure weight using different MH methods in TBD problem

Method	Optimal values for variables		Optimal weight
	A_1	A_2	
PSO	1.3240	0.0000	325.4535
GWO	1.2591	0.0000	317.3767
MVO	1.2614	0.0000	317.6665
RSA	1.2613	0.0000	317.6539
RSA-CM	1.2588	0.0000	**317.3389**

4.6 Statistical Test

The Friedman test, a widely used non-parametric two-way analysis of variances by ranks [47], is performed to identify the significance of the performance evaluation measures on five datasets and five MH algorithms with 20 independent runs. The test assumes a null hypothesis (H_0) as the equal performance of the comparative methods while the alternative hypothesis (H_1) assumes the difference

in the performance of the comparative MH algorithms. The highest rank for accuracy refers to the best algorithm as the larger value is preferred. On the other hand, the lowest rank is best for OFS and fitness as the smaller values are selected.

Table 11 shows average ranks for different MH algorithms with significance level $\alpha = 0.05$. The highest p-value calculated using Friedman's test for all five datasets was 0.0166, which is less than α, indicating that the results are statistically significant. The proposed RSA-CM gained the best accuracy, OFS, and fitness value as compared to PSO, GWO, MVO, and RSA in three out of five datasets. In the case of CIC-IDS201, GWO achieved the best OFS and fitness, followed by the proposed RSA-CM. In the case of CIC-IDS2018, PSO performed the best OFS, but the proposed RSA-CM achieved the best accuracy and fitness value.

Table 11: Friedman ranking results for the ACO-RSA and the other MH algorithms across all metrics

Dataset	Metric	PSO	GWO	MVO	RSA	RSA-CM
KDD-CUP99	ACC	2.4	3.3	2.65	3.25	**3.45**
	OFS	2.7	3.15	2.95	2.89	**2.65**
	Fitness	3.2	3.25	3.15	3	**2.75**
NSL-KDD	ACC	3.05	3.25	2.65	3.45	**3.6**
	OFS	2.8	3.05	3.05	3.4	**2.5**
	Fitness	2.95	3	2.9	2.9	**2.85**
UNSW-NB15	ACC	2.5	2.45	2.5	2.45	**3**
	OFS	2.95	3.45	2.9	3.2	**2.8**
	Fitness	3.3	3.6	3.25	2.9	**2.55**
CIC-IDS2017	ACC	2.6	2.9	3.2	3.25	**3.55**
	OFS	3.6	**2.95**	3.35	3.05	3.05
	Fitness	3.35	**2.7**	3.45	2.85	2.85
CIC-IDS2018	ACC	2.9	3.25	**2.6**	3	**2.6**
	OFS	**2.5**	3	3	2.85	2.7
	Fitness	3.2	3.5	3.05	2.65	**2.45**

Note: Highlight (bold) denotes the best performance of the corresponding metric.

5 Conclusion and Future Work

Several security solutions based on ML have been developed in recent years, including ID systems. However, the existence of irrelevant or redundant data affects the performance of ML methods and their performance. Therefore, a novel FS method to improve the capability of the original RSA in exploration and exploitation using CM is presented. The CM is used to expand search capability of the RSA, which in turns prevent the RSA from getting stuck in local optima and improve its convergence speed. The developed RSA-CM efficiency is validated using five open-access datasets in the ID domain and two engineering problems. Its efficiency is also compared with PSO, GWO, MVO, and RSA methods. The results show that the RSA-CM performs better than the other methods on almost the datasets and the tested engineering problems in terms of several evaluation metrics used in this work. Moreover, Friedman test outcomes show that the proposed RSA-CM has the most significant results compared to other methods. These results make introduced RSA-CM superior to other comparative

methods and more suitable to be used as a FS approach for the application of ID. In future work, we will attempt to use developed RSA-CM as an FS method in other applications such as text mining, image segmentation, and IoT.

Funding Statement: The author received no specific funding for this study.

Conflicts of Interest: The author declare that they have no conflicts of interest to report regarding the present study.

References

[1] I. H. Sarker, A. S. M. Kayes, S. Badsha, H. Alqahtani, P. Watters *et al.,* "Cybersecurity data science: An overview from machine learning perspective," *Journal of Big Data*, vol. 7, no. 1, pp. 1–29, 2020.

[2] E. Jaw and X. Wang, "Feature selection and ensemble-based intrusion detection system: An efficient and comprehensive approach," *Symmetry*, vol. 13, no. 10, pp. 1764, 2021.

[3] H. Kure and S. Islam, "Cyber threat intelligence for improving cybersecurity and risk management in critical infrastructure," *Journal. Universal Computer Science*, vol. 25, pp. 1478–1502, 2019.

[4] I. Lee, "Internet of Things (IoT) cybersecurity: Literature review and IoT cyber risk management," *Future Internet*, vol. 12, no. 9, pp. 157, 2020.

[5] V. Ford and A. Siraj, "Applications of machine learning in cyber security," in *27th Int. Conf. on Computing Application Industry & Engineering*, Kota Kinabalu, Malaysia, pp. 1–6, 2014.

[6] A. Gupta, R. Gupta and G. Kukreja, "Cyber security using machine learning: Techniques and business applications," *Application Artificial Intelligence in Business Education & Healthcare, Studies Computer Intelligence*, vol. 954, pp. 385–406, 2021.

[7] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions Emerging Telecommunication Technology*, vol. 32, no. 1, pp. e4150, 2021.

[8] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *Applied. Sciences*, vol. 9, no. 20, pp. 4396, 2019.

[9] M. Banaie-Dezfouli, M. H. Nadimi-Shahraki and Z. Beheshti, "R-GWO: Representative-based Grey wolf optimizer for solving engineering problems," *Applied Soft Computing*, vol. 106, no. 4, pp. 107328, 2021.

[10] R. Khalid and N. Javaid, "A survey on hyperparameters optimization algorithms of forecasting models in smart grid," *Sustainable Cities & Society*, vol. 61, no. 6, pp. 102275, 2020.

[11] L. Abualigah and A. Diabat, "A comprehensive survey of the Grasshopper optimization algorithm: Results, variants, and applications," *Neural Computing & Applications*, vol. 32, no. 19, pp. 15533–15556, 2020.

[12] R. Khalid and N. Javaid, "A survey on hyperparameters optimization algorithms of forecasting models in smart grid," *Sustainable Cities & Society*, vol. 61, no. 6, pp. 102275, 2020.

[13] S. Mirjalili, S. M. Mirjalili and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Computing & Applications*, vol. 27, no. 2, pp. 495–513, 2016.

[14] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Int. Conf. on Neural Networks*, Perth, WA, Australia, pp. 1942–1948, 1995.

[15] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[16] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris *et al.,* "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Advances Engineering Software*, vol. 114, pp. 163–191, 2017.

[17] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances Engineering Software*, vol. 95, no. 12, pp. 51–67, 2016.

[18] S. Mirjalili, S. M. Mirjalili and A. Lewis, "Grey wolf optimizer," *Advances Engineering Software*, vol. 69, pp. 46–61, 2014.

[19] L. Abualigah, M. Abd Elaziz, P. Sumari, Z. W. Geem and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer," *Expert Systems with Applications*, vol. 191, no. 11, pp. 116158, 2022.

[20] K. H. Almotairi and L. Abualigah, "Hybrid reptile search algorithm and remora optimization algorithm for optimization tasks and data clustering," *Symmetry*, vol. 14, no. 3, pp. 116158, 2022.

[21] A. Dahou, M. Abd Elaziz, S. A. Chelloug, M. A. Awadallah, M. A. Al-Betar *et al.,* "Intrusion detection system for IoT based on deep learning and modified reptile search algorithm," *Computer Intelligence Neuroscience*, vol. 2022, no. 4, pp. 6473507, 2022.

[22] Z. Elgamal, A. Q. M. Sabri, M. Tubishat, D. Tbaishat, S. N. Makhadmeh *et al.,* "Improved reptile search optimization algorithm using Chaotic map and Simulated annealing for feature selection in medical field," *IEEE Access*, vol. 10, pp. 51428–51446, 2022.

[23] S. Ekinci and D. Izci, "Enhanced reptile search algorithm with Lévy flight for vehicle cruise control system design," *Evolutionary Intelligence*, vol. 2022, pp. 1–13, 2022.

[24] I. Al-Shourbaji, N. Helian, Y. Sun, S. Alshathri and M. Abd Elaziz, "Boosting ant colony optimization with reptile search algorithm for churn prediction," *Mathematics*, vol. 10, no. 7, pp. 1031, 2022.

[25] K. H. Almotairi and L. Abualigah, "Improved reptile search algorithm with novel mean transition mechanism for constrained industrial engineering problems," *Neural Computing Applications*, vol. 2022, no. 20, pp. 1–21, 2022.

[26] Z. Halim, M. N. Yousaf, M. Waqas, M. Sulaiman, G. Abbas *et al.,* "An effective genetic algorithm-based feature selection method for intrusion detection systems," *Computers & Security*, vol. 110, no. 34, pp. 102448, 2021.

[27] M. H. Kamarudin, C. Maple and T. Watson, "Hybrid feature selection technique for intrusion detection system," *International Journal High Performance Computing & Networking*, vol. 13, no. 2, pp. 232–240, 2019.

[28] N. Acharya and S. Singh, "An IWD-based feature selection method for intrusion detection system," *Soft Computing*, vol. 22, no. 13, pp. 4407–4416, 2018.

[29] H. Wang, C. Li, Y. Liu and S. Zeng, "A hybrid particle swarm algorithm with Cauchy mutation," in *IEEE Swarm Intelligence Symp*. Honolulu, HI, USA, pp. 356–360, 2007.

[30] A. Kollu and S. Vadlamudi, "Eagle strategy with Cauchy mutation particle swarm optimization for energy management in cloud computing," *International Journal Intelligent. Engineering & Systems*, vol. 13, no. 6, pp. 42–51, 2020.

[31] M. P. Behera, A. Sarangi and S. Mishra, "Analysis of Gaussian and Cauchy mutations in k-means particle swarm optimization algorithm for data clustering," *Technical Advancements Machine Learning Healthcare, Studies Computer Intelligence*, vol. 936, pp. 103–117, 2021.

[32] I. Al-Shourbaji, P. H. Kachare, S. Alshathri, S. Duraibi, B. Elnaim *et al.,* "An efficient parallel reptile search algorithm and snake optimizer approach for feature selection," *Mathematics*, vol. 10, no. 13, pp. 2351, 2022.

[33] M. Tavallaee, E. Bagheri, W. Lu and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *IEEE Symp. Computer Intelligence Security & Defense Applications*, Ottawa, ON, Canada, pp. 1–6, 2009.

[34] S. Sapre, P. Ahmadi and K. Islam, "A robust comparison of the KDDCup99 and NSL-KDD IoT network intrusion detection datasets through various machine learning algorithms," ArXiv:1912.13204, 2019.

[35] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in *Int. Conf. on Military Communication & Information Systems*, Canberra, ACT, Australia, pp. 1–6, 2015.

[36] I. Sharafaldin, A. H. Lashkari and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in *Int. Conf. on Information Systems & Security*, Libson, Portugal, pp. 108–116, 2018.