Enhanced Network Security Through Optimized Feature Subset Selection Using GTO Algorithm

Abderrezak Benyahia, Ouahab Kadri, Moumen Hamouma, and Adel Abdelhadi

Original scientific article

***Abstract*—Several attacks are carried out daily to steal sensitive data or make servers inaccessible. Currently, Optical Burst Switching (OBS) networks are among the most widely used in the world. Hackers regularly resort to Burst Header Packet Flooding (BHPF) techniques due to vulnerabilities in the network architecture. Identifying BHPF attacks prevents server applications from being disrupted or stopped. Our solution comprises three main steps: learning, detection, and diffusion of the model. We used an Extreme Learning Machine (ELM), a highly accurate and fast classifier. We proposed a new feature selection algorithm that combines the Fisher score to calculate variable relevance and the Gorilla Troops Optimizer (GTO) to avoid exhaustive searches. The type of attack is shared using the MQTT protocol to enhance network security. The experimental results show that our approach achieves the best precision while maintaining competitive accuracy, compared to Ant-Tree, Naive Bayes, Nearest Neighbor, Artificial Neural Networks (ANN), SVM with Linear Kernel (SVM-LN), and SVM with Radial Basis Function (SVM-RBF).**

***Index terms*—machine learning, feature selection, gorilla troops optimizer, predictive models, optical burst switching.**

I. Introduction

The Internet of Things (IoT) is a network of interconnected physical devices and sensors that communicate and exchange data over the Internet to provide automated services. The number of IoT devices is increasing exponentially. The combination of this infrastructure with intelligence aims to facilitate various tasks and makes our lives more sophisticated. Optical networks have attracted considerable attention since their inception due to their massive bandwidth [1]. The three essential parts of this architecture are the entry nodes, the central nodes, and the exit nodes. The functions that ensure the proper functioning of an OBS network are signaling, scheduling, and conflict resolution. The first function consists of identifying the means to ensure the effective transmission of data. The second function provides scheduling by controlling wavelengths. The third function aims to resolve contention in the OBS system, which is the result of several simultaneous bursts occurring from different input ports to a single output port [2]. Currently, the dominant infrastructure is Optical Burst Switching. This technology replaced Optical Circuit Switching (OCS) [3].

Currently, Internet of Things (IoT)-based networks and Optical Burst Switching (OBS) networks are widely employed for high-speed data transmission. IoT data is frequently carried over optical backbone infrastructures—such as Optical Burst Switching (OBS) networks—that aggregate and handle large volumes of IoT traffic. Ensuring secure data transmission across these infrastructures has become a major challenge in the field of cybersecurity. One of the most critical threats is the Burst Header Packet Flooding (BHPF) attack, which can severely disrupt network operations by saturating the available bandwidth and causing Denial of Service (DoS) incidents. Traditional detection systems typically rely on simple signature-based comparisons or the analysis of individual parameters, rendering them ineffective against dynamic and complex attack patterns.

Our work addresses the urgent need for a robust, scalable, and intelligent intrusion detection solution capable of adapting to diverse attack scenarios while maintaining high detection accuracy and low computational overhead. To meet this challenge, we propose a hybrid detection framework that integrates the Gorilla Troops Optimizer (GTO) for optimal feature selection with the Extreme Learning Machine (ELM) for fast and accurate classification. Additionally, we incorporate the Message Queuing Telemetry Transport (MQTT) protocol to facilitate lightweight and real-time communication of detected threats across network components. The ultimate objective is to provide a highly efficient and real-time defense mechanism that can be deployed in modern OBS and IoT network environments.

The most popular attack against OBS is the BHP flood attack [4]. Its principle consists of sending several BHP simultaneously to saturate the bandwidth of a network, as seen in Figure 1. After this attack, the nodes cannot ensure the transfer of legitimate BHP since the information created by the attackers mobilizes all the network resources.

This phenomenon is called denial of service (DoS) [5]. To limit the damage caused by this attack, it is important to implement a detection system. Classical methods do not detect all types of attacks, knowing that there are several scenarios for the same type of attack. In most cases, classical methods rely on a limited set of parameters, which restricts their detection capability. Our approach overcomes this limitation by exploiting a richer and more informative feature space. The proposed solution must be effective and not reduce network performance.

Manuscript received July 23, 2025; revised September 19, 2025. Date of publication December 23, 2025. Date of current version December 23, 2025. The associate editor prof. Hrvoje Karna has been coordinating the review of this manuscript and approved it for publication.

Authors are with the University of Batna 2, Algeria (e-mails: {a.benyahia, o.kadri, hamouma.moumen, A.Abdelhadi}@univ-batna2.dz).

Digital Object Identifier (DOI): 10.24138/jcomss-2025-0146

A good solution should allow learning from limited and incomplete data. It must provide a model with acceptable prediction accuracy. In this paper, we propose a solution to this type of attack based on an Extreme Learning Machine (ELM) and Gorilla Troops Optimizer. The principle of our proposal is to reduce the number of entries and publish the attack source to improve detection against future attacks.

This article presents the following key contributions:

- A novel intrusion detection system architecture is proposed for OBS networks, combining Gorilla Troops Optimizer (GTO) with Extreme Learning Machine (ELM).
- Introduction of a hybrid feature selection method that fuses the Fisher score with GTO to identify the most relevant network parameters for classification.
- Implementation of a lightweight and efficient MQTT-based communication protocol to broadcast attack alerts for real-time decision-making.
- Extensive experiments conducted using a custom-built dataset based on National Science Foundation Network (NSFNET) topology simulations generated via National Chiao Tung University Network Simulator (NCTUns).
- Comparative evaluation against six benchmark models (Anttree, Naive Bayes, Nearest Neighbor, ANN, SVM-LN, and SVM-RBF), demonstrating superior accuracy (91%), precision (92%), recall (76%), and Harmonic mean of Precision and Recall (F1-score) (83%).

The remainder of this manuscript is organized as follows: Section II presents recent techniques for detecting BHP Flooding Attacks. Section III presents the general architecture of the detection system and outlines the algorithms used, namely Gorilla Troops Optimizer and Extreme Learning Machine. In section IV, the results and discussions of the experiment are detailed. A conclusion and some perspectives are provided in Section V.

II. Related Work

We conducted a non-exhaustive search on Google Scholar. The results show that there are several methods for detecting BHP Flooding attacks. Almost all of these solutions are based on three essential elements: the different entries, the types of attacks, and the steps followed by the attacker to occupy the nodes of the network. Flood attacks, circulating burst attacks, spoofing, and replay attacks are the most well-known attacks that target optical burst-switching networks. Attack history is key to any machine learning-based solution.

Several studies in this field have combined metaheuristic algorithms with machine learning to tackle the problem of intrusion detection in IoT and high-speed networks. For example, a hybrid Convolutional Neural Network (CNN) - Bidirectional Gated Recurrent Unit (BiGRU) - Bidirectional Long Short-Term Memory (BiLSTM) architecture optimized using Particle Swarm Optimization (PSO) was proposed in [6], achieving an accuracy exceeding 98% in identifying attacks within IoT environments. Similarly, the work in [7] employed a deep learning model enhanced by a multi-objective Gorilla Troops Optimizer (GTO) for feature selection, reaching near-perfect accuracy on benchmark datasets such as NSL-KDD and TON-IoT. While these approaches offer high accuracy, they often rely on complex and computationally demanding models. To overcome this limitation, lightweight yet efficient metaheuristic methods like GTO have gained attention. The study in [8] surveys various GTO variants and their applications, reinforcing its effectiveness for feature selection in security-related contexts. Furthermore, a recent study [9] demonstrated that swarm-based feature selection methods outperform traditional techniques in intrusion detection tasks. In parallel, [10] proposed an innovative intrusion detection system architecture for Industry 4.0 using graph-based deep learning models to respond to evolving threats in modern network infrastructures. In contrast to these deep and resource-intensive approaches, our work presents a lightweight and scalable hybrid solution based on GTO and Extreme Learning Machine (ELM), delivering fast learning, real-time classification, and reduced computational overhead while maintaining competitive detection performance.

In [11], they created four classes for attacks: Misbehaving-Block, Behaving-No Block, Misbehaving-No Block, and Misbehaving-Wait. They proposed a system that uses Sequential Minimal Optimization (SMO) algorithms and the K-Star algorithm. OBS sends control information and primary data using different channels. This communication strategy increases the likelihood that the network will be subjected to an attack that either completely blocks or partially degrades the quality of service. Sudarshan proposed a solution for these problems based on monitoring network traffic on control and data channels. Flooding attacks consist of occupying bandwidth, which leads to limiting network performance. In [4], they used PSO and SVM to detect these types of attacks. PSO is used to optimize the parameters of the SVM. They used the dataset provided by the UCI warehouse to train and test the model. In [12], they analyzed the results obtained by using four semisupervised machine learning (SSML) algorithms to identify the source of the attacks. This study includes the K-means algorithm, the Gaussian mixture model (GMM), the selflearning model (ST), and the selflearning model (MST). They also tackled the problem of manually labeling a large dataset. In several cases, the class of observation is absent. SSML is used to find the missing class id. Using an example composed of four categories and some hundred observations, the MST gave good results. Software-Defined Networking (SDN) is used for managing complex networks. In [13], to limit the risks of these threats, the authors proposed a classification system to identify Distributed Denial of Service (DDoS) attacks by knowing the different categories of observations. The target of this attack is the SDN controller. Its principle is to carry out several checks to verify the existence of a change in the behavior of the network. This information is used as an indicator of network status, as well as to detect DDoS attacks. The complex and dynamic nature of IoT networks makes security management very challenging. One of the main threats is the Low-Rate Denial of Service (LR DoS) attack. Its principle is based on sending bursts of specific traffic that target Transmission Control Protocol (TCP) datagrams and waste time by sending the same information several times. This type of attack is difficult to detect because

it imitates the traffic signature of a trusted node, that is, the characteristic pattern of its legitimate communication, which makes the malicious traffic appear authentic. In [14], they developed an anomaly identification system based on a Feedforward Convolutional Neural Network (FFCNN) to identify LR DoS menaces in IoT-SDN. They used the Denial-of-Service 2017 (CIC DoS 2017) dataset from the Canadian Institute for Cybersecurity [15]. An iterative parameters selection based on SVM is applied to find the most relevant parameters in the detection process. IoT adopts a communication system that is open and fragile to computer attacks. The main objective of a security system is to identify and isolate malicious nodes. The LWCTS model ensures secure and qualitative information communication between devices on the Nets. Rajendra and Krishna proposed a set of simulation experiments on the model and compared the performance to existing models [16].

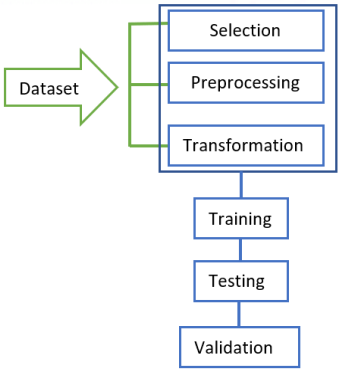

Fig. 1. Architecture of the detection model.

III. Proposed Approach

In this work, we propose a system for detecting BHP Flooding Attacks in OBS Networks using GTO and ELM. This work represents a continuation of the research carried out by the team of Adel Rajab at the University of South Carolina [17] [18].

We have made a selection of parameters that combines Fisher score with a recent method based on the smart behavior of gorilla troops, called Artificial Gorilla Troops. During the classification phase, we dealt with the problem of overfitting. We solved this problem by adjusting the values of the kernel function hyperparameters through several tests on a limited set of data. Additionally, we proposed a publication/subscription paradigm system for the dissemination of information to ensure the effective exploitation of the obtained results.

A. Data preparation

Before discussing the choice of the classification model, it is necessary to go through a data preparation phase. It is composed of three main steps which are: data selection, data pre-processing, and data transformation.

Several parameters are used to identify the state of the network. We cite for example the number of the sending node, Used Bandwidth, Packet Drop Rate, Initial reserved Bandwidth, Average Delay Time, and Total received packets [19]. We have proposed a parameter selection algorithm that combines the Fisher score and the Gorilla Troops Optimizer algorithm. The use of this algorithm is not limited to the problem treated in this paper but we could apply it to resolve other data classification problems.

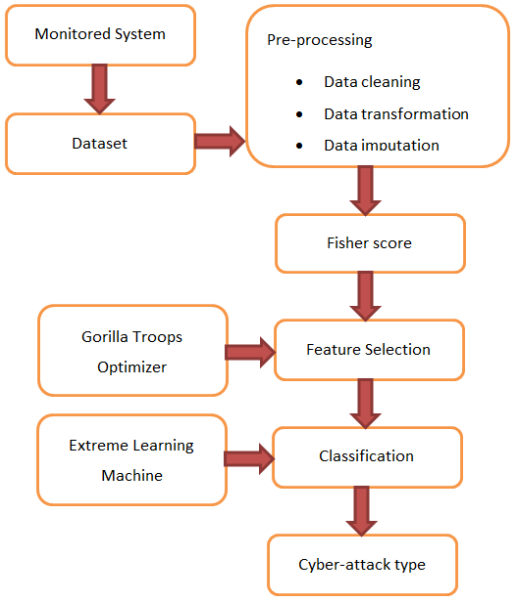

Fig. 2. The overall block of the GTO-ELM model.

B. Gorilla Troops Optimizer

Gorilla Troop optimization (GTO) represents the modeling of five behaviors observed in Gorilla groups which are: Moving to an anonymous region, moving to other groups, moving to a region already visited, continuing to follow the leader, and striving for the adult females [20].

$$GX(t+1) = \begin{array}{c} (UB - LB) \times r_1 + LB, \ \ rand < p, \\ (r_2 - C) \times X_r(t) + L \times H, rand \geq 0.5 \\ X(i) - L \times \left(X(i) - GX_r(i)\right) + \\ r_3 \times \left(X(i) - GX_r(i)\right), rand < 0.5 \end{array} \quad (1)$$

where *GX(t)* is updated solution at iteration t, *X(t)* is current solution, *C* is persistence factor of an attribute, *Q* is influence force parameter, and *E* is exploration factor simulating aggressive gorilla behavior

The other values are calculated by the following equations:

$$C = F \times \left(1 - \frac{It}{MaxIt}\right) \quad (2)$$

$$F = cos\,(2 \times r_4) + 1 \quad (3)$$

$$L = C \times l \quad (4)$$

$$H = Z \times X(t) \quad (5)$$

$$L = C \times l \quad (6)$$

We can define the determined number of repetitions as well as the minimum value of the random behavior. If the value

obtained by GX(t) is lower than X(t) then the latter is considered the best solution.

Depending on the result of the comparison of the value of C with W, equation (6) or else equation (8) is applied.

$$\mathrm{GX}\ (\mathrm{t}\ +\ 1) = X_{s\_b} - \left(X_{s_b} \times\ \mathrm{Q} - X(t) \times\ \mathrm{Q}\right) \times \mathrm{A} \tag{7}$$

$$Q\ =\ 2\ \times\ r_5\ -\ 1 \tag{8}$$

$$A\ =\ \beta\ \times\ E \tag{9}$$

$$E = \begin{cases} N_1, rand \geq 0.5 \\ N_2, rand < 0.5 \end{cases} \tag{10}$$

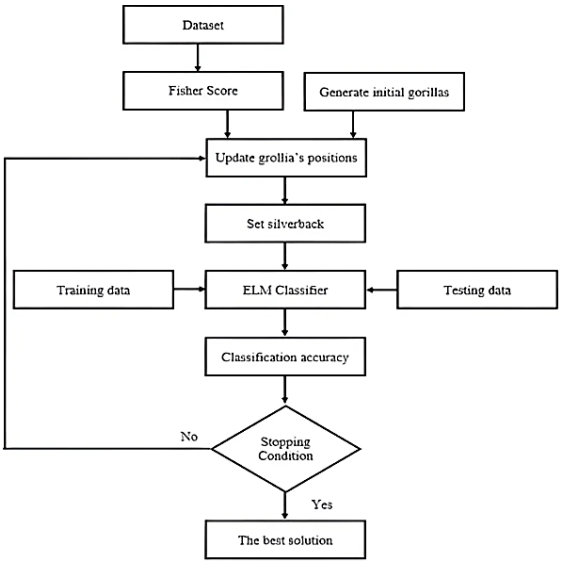

Fig. 3. GTO-ELM feature selection algorithm.

$$Fitness_i = a \times (1 - C_i) + (1 - a) \times \frac{|BX_i|}{D} \tag{11}$$

Q is the influence force. E is responsible for imitating the violent influence on the resolutions of equation (10). At the end of this step, GX(t) will replace X(t) if the target value of GX(t) is less than X(t). All parameters are the inputs of the GTO algorithm. In this step, we calculate the retention of each attribute and its probability of appearing in the final subset of selected parameters. The Fisher score is used to calculate the fitness function [21], [22].

The variable C represents the persistence of an attribute obtained by the Fisher score. D is the dimension of the input training dataset. We use the Min function to determine the best. All parameters are adjusted at the end of each iteration. The GTO algorithm terminates using two criteria which are the maximum number of iterations and a number depending on the results of iterations without improvement. This algorithm eliminates irrelevant parameters and returns the subset of parameters that will be used as input for the ELM classifier.

C. Fisher score

Fisher's criterion is applied to each parameter. It allows for to evaluation of the separation between classes using a single parameter [23]. In the case of binary classification, the Fisher criterion is expressed as follows:

$$J(a) = \frac{m_1(a) - m_2(a)}{N_1\sigma_1^2 - N_2\sigma_2^2} \tag{12}$$

where $m_i(a)$ (i=1,2) is class center of gravity w_i considering only the parameter a.

ó: component variance of class vectors w_i:

$$m_1(a) = \frac{1}{N_i}\sum_{k_i=1}^{N_i} \quad X_{k_i}(\alpha) \tag{13}$$

$$\sigma_i^2(\alpha) = \frac{1}{N_i}\sum_{j=1}^{N_i} \quad \left[X_{ij}(\alpha) - m_i(\alpha)\right]^2 \tag{14}$$

In general, for M classes, J(a) is written:

$$J(a) = \sum_{i=1}^{M} \quad \sum_{j=1}^{M-1} \quad \frac{m_i(a) - m_j(a)}{N_i\sigma_i^2 - N_j\sigma_j^2}. \tag{15}$$

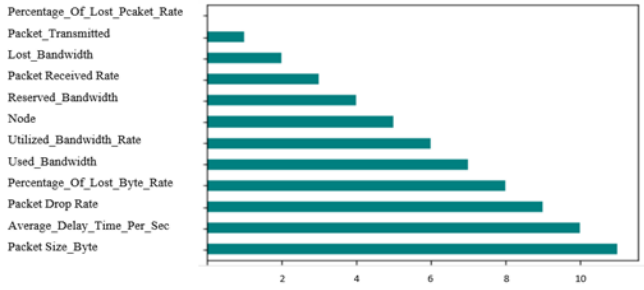

Fig. 4. Comparison of parameters pertinence.

The numerator of the previous formula translates the separation of the classes (interclass variance) while the denominator reflects the compactness of the classes (by analogy with the intra-classe variance). As long as the criterion is large then the classes are more separated [24].

The solution for the selection of the parameters consists of taking into account the first parameters verifying:

$$(\alpha_1) \geq J(\alpha_2) \geq \cdots \geq J\left(\alpha_{d'}\right) \geq J(\alpha_d) \tag{16}$$

To determine the number of parameters to be used in the classification phase, we performed several simulations. figure 4 shows a good example. We created a table with 200 rows and 12 columns. We have ranked the 12 parameters according to their importance. We notice that the Packet_size_byte parameter is the most important while the Percentage_Of_Lost_Packet_Rate parameter is of little importance.

D. Extreme Learning Machine

In the learning step, we calculate the least squares value of the linear model [20]. This method aims to train a single-layer feedforward neural network (SLFNs). It was created by [11]. ELM initializes randomly the hidden layer weights defined by the outputs of the hidden layer. The general architecture of an ELM is shown in Figure 5:

The different steps of the learning algorithm are as follows:

- Randomly assign weight wi and bi, i=1, …L

- Resolve hidden layer output H
- Resolve output weight matrix β = H†Y
- Use β to predict new data T=H β

Our dataset is composed of N distinct observations knowing that the xi represents the entries and the yi are the values corresponding to the different classes.

The equation of the function that allows calculating the targets of an SLFN with M hidden nodes can be written as:

$$\hat{y_j} = \sum_{i=1}^{M} \quad B_i f\left(w_i x_j + b_i\right),\ j \in [1,\ N] \tag{17}$$

Knowing that $\hat{y}_j$ represents the estimation to y_j, f represents the activation function, w_i is the input weight vector, b_i is the hidden layer bias, and β_i is the output weight conforming to the i^{th} neuron in the hidden layer.

If the algorithm converges and the gap between the correct value and the assessed value is close to zero, the new equation is written as follows:

$$\sum_{i=1}^{M} \quad B_i f\left(w_i x_j + b_i\right) = y_j,\ j \in [1,\ N] \tag{18}$$

which can be written compactly as Hβ = Y, and where H is the hidden layer output matrix defined as:

$$H = \begin{pmatrix} f(w_1 x_1 + b_1) & \cdots & f(w_M x_1 + b_M) \\ \vdots & \ddots & \vdots \\ f(w_1 x_N + b_1) & \cdots & f(w_M x_N + b_M) \end{pmatrix} \tag{19}$$

For the solutions of the previous equation to be acceptable, they must satisfy three properties:

1. It represents a least-squares solution to the given problem, so the smallest learning mistake can be attained with this answer;
2. It represents the answer with the minimum norm between the least-squares possibilities;
3. There is one smallest norm solution between the least squares possibilities and equals β=H†Y.

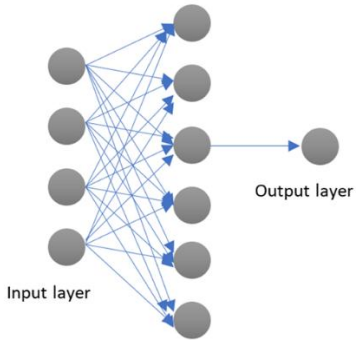

Fig. 5. Extreme Learning Machine.

B. MQTT protocol

Message Queuing Telemetry Transport (MQTT) is a very used communication protocol in IoT. It is based publish/subscribe model. This protocol is lightweight and functions effectively under the constraints imposed by IoT technology, such as low bandwidth and limited resources in terms of computing power and storage [25].

The MQTT protocol consists of 4 main components which are: Message, Client, Server, and Subject. An MQTT message consists of a header, which contains information about the message type, as well as the data published by a client.

The first type of message is Connect, which is used to initialize communication between the client and the server. The Disconnect message completes the data exchange before the client disconnects. The Publish message manages the distribution of information across the network, while the Subscribe message allows clients to register for specific topics or services.

We can distinguish two types of clients: Publishers and Subscribers. Publishers are responsible for sending messages to the broker, with messages categorized under well-defined topics. Subscribers receive messages from the broker based on their topic of interest. In this protocol, communication between clients is indirect and facilitated by the broker. The MQTT Broker acts as the server, ensuring message dissemination based on subscription types and the order of client requests. The broker employs various authentication and encryption techniques to ensure secure communication and data storage. MQTT does not directly enhance network security; instead, it facilitates the lightweight and reliable dissemination of attack alerts, thereby improving the system's overall responsiveness and coordination.

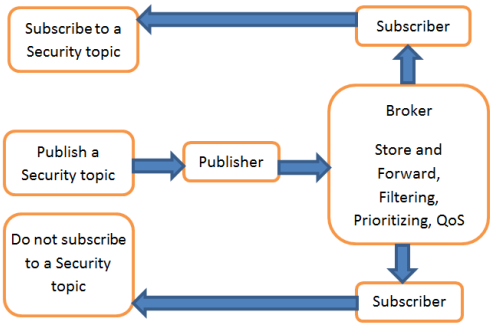

Fig. 6. MQTT Protocol

IV. Experiments and Results

A. Dataset

The most important step in the development process is data preparation. In all the experiments we performed, we used a database created specifically for the development of BHP flood attack detection models. The database is composed of 1075 data records and each one is in turn composed of 22 attributes [15]. The main attributes of the dataset are vertex id, assigned bandwidth, rate of data transfer, Bandwidth missing, datagram diffused, datagram lost, datagram acknowledged, emitted data, received data, latency, and the rate of Burst Header Packet Flood Attack.

Data was collected using the NCTUns simulator. Several simulations were carried out on a network under the NSFNET topology. This topology is the most used in the installation of modern networks. This type of network has the advantage of being extensible and flexible.

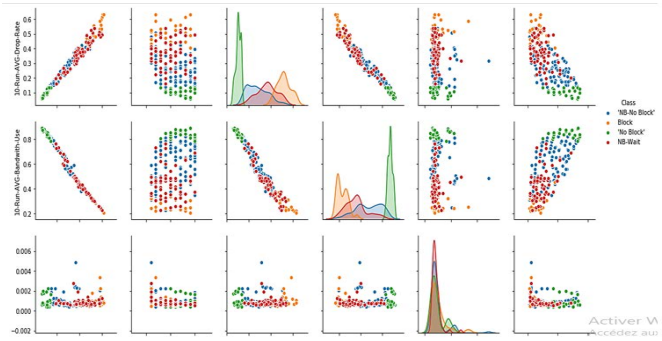

Fig. 7. Class positions by parameters.

The information is recorded by carrying out several scenarios. In each attack, the location and number of nodes are changed to simulate the attackers' real behavior. Using this simulator, it is possible to configure in detail the various network elements, e.g. data burst (timeout, smallest burst measurement, extreme file dimension), channel allocation wavelength (regulator and data channels), periodic wave conversion, reservation scheme, control datagram treatment period, resolution method conflicting control packets and conflicting burst used method.

The data is divided into four classes which are: Misbehaving-No Block (M-No Block), Behaving-No Block (No Block), Misbehaving-Block (Block), and Misbehaving-Wait (M-Wait). Figure 6 gives an idea about the distribution of some parameters.

B. Feature selection

The experiments were conducted using a machine running Microsoft Windows 10; a 64-bit operating system equipped with an Intel Core i3 2.13 GHz CPU and 8 GB of RAM.

To indicate the importance of each parameter to Gorilla Troops Optimizer, we created a procedure that uses the dataset and returns the Fisher score as a result. The results of this operation are shown in Table I.

TABLE I
FISHER SCORE

PARAMETER	DESIGNATION	FISHER SCORE
P1	NODE	0.02
P2	UTILIZED BANDWIDTH RATE	0.07
P3	PACKET DROP RATE	0.07
P4	RESERVED_BANDWIDTH	0.02
P5	AVERAGE_DELAY_TIME_PER_SE	0.04
P6	PERCENTAGE_OF_LOST_PCAKET_	0.07
P7	PERCENTAGE_OF_LOST_BYTE_R	0.07
P8	PACKET RECEIVED RATE	0.07
P9	USED_BANDWIDTH	0.04
P10	LOST_BANDWIDTH	0.04
P11	PACKET SIZE_BYTE	0.04
P12	PACKET_TRANSMITTED	0.04
P13	PACKET_RECEIVED	0.04
P14	PACKET_LOST	0.04
P15	TRANSMITTED_BYTE	0.02
P16	RECEIVED_BYTE	0.04
P17	10-RUN-AVG-DROP-RATE	0.10
P18	10-RUN-AVG-BANDWIDTH-USE	0.07
P19	10-RUN-DELAY	0.04
P20	NODE STATUS' {B, NB, P NB}	0.04
P21	FLOOD STATUS	0.18

The GTO algorithm selected the optimal set of parameters because it not only selects the parameters with the largest fisher score but also removes redundant parameters by finding the relationship between them. The parameters selected by GTO are {P2, P3, P5, P6, P7, P8, P9, P10, P13, P14, P16, P17, P18, P19}.

C. Classification

After completing the parameter selection phase by Gorilla Troops Optimizer, we used ELM to classify the data and detect the type of attack. We have carried out a comparative study with four other models which are SVM, Ant Colony Optimization (ACO) - SVM, ELM, and ACO-ELM. These methods are widely used to classify the different BHP flood attacks in the OBS network. We can use several measures to evaluate our parameter selection model. Among the measures we used, we quote the true positive rate (TPR). Based on our previous research in this area, we chose the confusion matrix. This matrix makes it possible to calculate four interesting metrics. The confusion matrix shows the classification quality of our approach. We also obtained a very good accuracy rate using the other algorithms. The algorithm based on the use of a parameter selection phase is faster compared to other algorithms.

TABLE II
CONFUSION MATRIX GTO-ELM MODEL USING A LIST OF 400 ROWS

		PREDICTED CLASS			
		NB-NO BLOK	BLOK	O BLOK	B-WAIT
TRUE CLASS	NB-NO BLOK	94	0	0	6
	BLOK	0	100	0	0
	NO BLOK	0	0	100	0
	NB-WAIT	6	0	0	94

As noticed in Table III, the GTO-ELM algorithm obtained a very good value in the four measures. The reduced number of parameters obtained by GTO participated in a major way in this result.

Another factor is the ELM kernel function hyperparameters. After several simulations using restricted data sets, we found the values suitable for our models. The search interval for parameter C is $[2^3, 2^{11}]$ and the search interval for parameter γ is $[2^{-12}, 2^2]$. During each simulation, we used a different value combination. The best combination for the pair (C, γ) was $(2^3, 2^{-3})$.

TABLE III
IMPACT OF PERFORMANCE INDEX RESULTS OF SEVEN METHODS

METHODS	ACCURACY	PRECISION	RECALL	F1-SCORE
ANTTREE	0,55	0,6	0,3	0,4
NAIVE BAYES	0,69	0,70	0,43	0,53
NEAREST NEIGHBOR	0,62	0,66	0,39	0,49
RNA	0,61	0,7	0,35	0,46
SVM LN	0,81	0,8	0,58	0,67
SVM RBF	0,92	0,9	0,80	0,84
GTO-ELM	0,91	0,92	0,76	0,83

The dataset was split into a training set and a test set: 675 rows were used for training, while the remaining 400 rows were reserved for testing.

A pre-treatment operation was carried out. Its purpose is to reduce the influence of missing values on the final result. This operation increases the classification speed while maintaining the same accuracy rate.

To evaluate our proposed approach, we used four metrics: accuracy, precision, recall, and the F1-score. Accuracy measures the proportion of correctly classified cases out of the total number of data points. Precision reflects the model's ability to correctly identify positive cases without including false positives. Recall indicates the proportion of actual positive cases that were correctly detected by the model. Finally, the F1-score, defined as the harmonic mean of precision and recall, provides a balanced measure that considers both false positives and false negatives.

As shown in Table III, our proposed GTO-ELM method achieves competitive performance, with the best precision (92%) and comparable accuracy to SVM (RBF), while being faster to train and less computationally expensive, making it more suitable for real-time IoT environments.

According to Figure 8, our method converges to the optimal solution after less than 20 iterations which proves its speed. Compared to traditional classifiers such as Naive Bayes and Nearest Neighbor, our method significantly improves detection metrics, particularly precision and F1-score. While SVM with RBF kernel achieves slightly higher accuracy (92%), it requires longer training time and is less suitable for deployment in resource-constrained environments. The GTO-ELM model achieves a balanced trade-off between detection performance and computational efficiency.

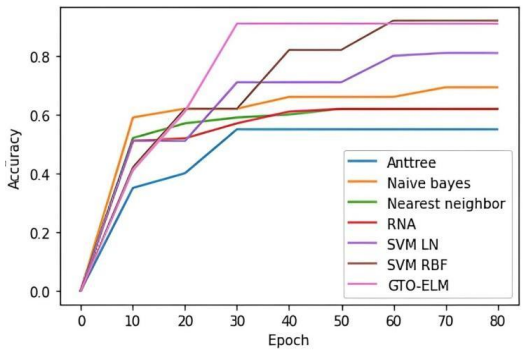

Fig. 8. Classification rate and convergence speed for the testing dataset.

Furthermore, unlike methods that rely heavily on labeled data or require extensive hyperparameter tuning (e.g., SVM), the proposed GTO-ELM model demonstrates robustness across varying data volumes and maintains high performance even when some features are missing or noisy.

The computational complexity of the ELM algorithm is lower than that of SVM-RBF. The training complexity of ELM is approximately $O(N \times L^3)$, where N is the number of data samples and L is the number of hidden neurons. In contrast, the training complexity of SVM-RBF can reach $O(N^3)$ in the worst case. This difference explains why our proposed method is less demanding in terms of computational and storage resources, making it more suitable for real-time detection.

D. Communication Protocol Testing and Validation

For the proposed communication protocol, we performed a set of tests to prove that our implementation conforms well to the MQTT protocol specification. The first test is to send Subscribe/Publish requests with different messages.

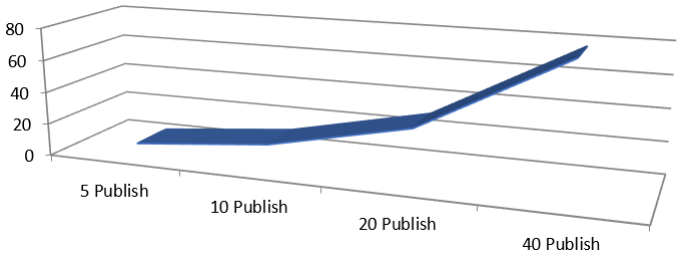

Fig. 9. Functional test "MQTT-Publish.

Fig. 10. Functional test "MQTT Subscribe

We also performed a functional test. Its objective is to analyze the behavior concerning the specifications (non-compliant, missing functions, errors, etc.) and to check compliance with constraints (performance, memory, delay, etc.). Quality is verified by inspecting portability, maintainability, and documentation.

All the established tests prove that our implementation conforms well to the specifications of the MQTT protocol.

IV. Conclusion

In this paper, we proposed a new approach for BHP Flooding attack detection. The architecture of our proposal is composed of three levels. First, we made a selection of parameters using Gorilla Troops Optimizer. Then, ELM is used for classification. Finally, we have created a quick and simple communication protocol for the dissemination and reuse of the results. The NCTUns simulator was used to create the training and test data. This data is collected by performing several attacks on an NSFNET topology network. The obtained results were compared with other methods. We validated our proposal by carrying out several scenarios. Therefore, this architecture could also be of interest to the industrial field because it helps engineers improve the diagnosis of materials.

Unlike traditional hybrid methods, such as PSO-SVM, the main contribution of this work lies in the novel combination of the Fisher score with the Gorilla Troops Optimizer, which enhances the relevance and compactness of the feature subset while reducing computation time. Furthermore, the integration of MQTT enables model reuse and strengthens the operational

and real-time applicability of our solution, representing a significant advancement for deployment in constrained OBS and IoT environments. Overall, our approach establishes itself as a lightweight and scalable alternative capable of overcoming key limitations of existing metaheuristic-plus-ML IDS systems.

To improve our proposal, we will test the implementation of an extreme learning machine on several nodes. Other experiments will be carried out to detect other types of attacks.

REFERENCES

[1] A. Ottino, J. Benjamin, and G. Zervas, "RAMP: A flat nanosecond optical network and MPI operations for distributed deep learning systems," *Optical Switching and Networking*, vol. 51, Art. no. 100761, 2024, doi: 10.1016/j.osn.2023.100761.

[2] C. Benrebbouh et al., "A lightweight security scheme to defend against quantum attack in IoT-based energy internet," *Int. J. Sensor Netw.*, vol. 43, no. 1, pp. 13–26, 2023, doi: 10.1504/ijsnet.2023.133813.

[3] K. C. Claffy, G. C. Polyzos, and H.-W. Braun, "Traffic characteristics of the T1 NSFNET backbone," in *Proc. IEEE INFOCOM*, 1993, doi: 10.1109/infcom.1993.253279.

[4] E. Efeoğlu and G. Tuna, "Performance evaluation of sequential minimal optimization and K* algorithms for predicting burst header packet flooding attacks on optical burst switching networks," *Balkan J. Electr. Comput. Eng.*, vol. 9, no. 4, pp. 342–347, 2021, doi: 10.17694/bajece.892150.

[5] M. K. Hossain, M. M. Haque, and M. A. A. Dewan, "A comparative analysis of semi-supervised learning in detecting burst header packet flooding attack in optical burst switching network," *Computers*, vol. 10, no. 8, Art. no. 95, 2021, doi: 10.3390/computers10080095.

[6] X. Zhou and L. Wang, "Hyperparameter optimization of convolutional neural network combined with bidirectional LSTM using particle swarm optimization for remaining useful life prediction of mechanical equipment," *Machines*, vol. 12, no. 5, p. 342, May 2024. [Online]. Available: https://doi.org/10.3390/machines12050342.

[7] A. Ahmed, V. Kumar, and M. Bedi, "An Intrusion Detection System on the Internet of Things Using Deep Learning and Multi-objective Enhanced Gorilla Troops Optimizer," *J. Bionic Eng.*, vol. 21, no. 4, pp. 489–507, Apr. 2024. doi:10.1007/s42235-024-00575-7.

[8] A. Mouassa, S. B. Pandya, K. Kalita, A. Kumar, P. Jangir, S. Chakraborty, and L. Abualigah, "An in-depth survey of the artificial gorilla troops optimizer: outcomes, variations, and applications," *Artificial Intelligence Review*, vol. 58, no. 2, pp. 503–545, Feb. 2024, doi: 10.1007/s10462-02410838-8.

[9] N. Shabir, M. Shahid, and A. Abbasi, "PSO-Driven Feature Selection and Hybrid Ensemble for Network Anomaly Detection," *Big Data and Cognitive Computing*, vol. 6, no. 4, p. 137, Nov. 2022. doi:10.3390/bdcc6040137.

[10] Z. Yan, W. Li, X. Chen, and L. Du, "A Survey on Graph Neural Networks for Intrusion Detection Systems: Methods, Trends and Challenges," *Computers & Security*, vol. 141, p. 103821, Jun. 2024, doi: 10.1016/j.cose.2024.103821.

[11] H. C. Leung et al., "Extreme learning machine for estimating blocking probability of bufferless OBS/OPS networks," *J. Opt. Commun. Netw.*, vol. 9, no. 8, pp. 682–692, 2017, doi: 10.1364/jocn.9.000682.

[12] S. Liu, X. Liao, and H. Shi, "A PSO-SVM for burst header packet flooding attacks detection in optical burst switching networks," *Photonics*, vol. 8, no. 12, Art. no. 555, 2021, doi: 10.3390/photonics8120555.

[13] H. H. Jazi et al., "Detecting HTTP-based application layer DoS attacks on web servers in the presence of sampling," *Comput. Netw.*, vol. 121, pp. 25–36, 2017, doi: 10.1016/j.comnet.2017.03.018.

[14] S. Wang et al., "Detecting flooding DDoS attacks in software defined networks using supervised learning techniques," *Eng. Sci. Technol. Int. J.*, vol. 35, Art. no. 101176, 2022, doi: 10.1016/j.jestch.2022.101176.

[15] L. Zhang, Y. Chen, and M. Li, "Resilient predictive control for cyberphysical systems under denial-of-service attacks," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 69, no. 1, pp. 144–148, 2021, doi: 10.1109/TCSII.2021.3076764.

[16] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, 2006, doi: 10.1016/j.neucom.2005.12.126.

[17] H. S. Ilango, M. Ma, and R. Su, "A feedforward–convolutional neural network to detect low-rate DoS in IoT," *Eng. Appl. Artif. Intell.*, vol. 114, Art. no. 105059, 2022, doi: 10.1016/j.engappai.2022.105059.

[18] A. Rajab, C.-T. Huang, and M. Al-Shargabi, "Decision tree rule learning approach to counter burst header packet flooding attack in optical burst switching network," *Optical Switching and Networking*, vol. 29, pp. 15–26, 2018, doi: 10.1016/j.osn.2018.03.001.

[19] A. Rajab et al., "Countering burst header packet flooding attack in optical burst switching network," in *Proc. Int. Conf. Inf. Security Pract. Exp. (ISPEC)*, Springer, 2016, pp. 279–293, doi: 10.1007/978-3-319-49151-6_22.

[20] A. M. Sambo, M. A. Bagiwa, Y. S. Ali, et al., "Enhancing Intrusion Detection in IoT Platforms Using a Novel Hybrid Gorilla Troops and Bird Swarm Optimization Algorithm," *Dutse J. Pure Appl. Sci.*, vol. 11, no. 1c, pp. 68–82, 2025.

[21] M. T. Seddik et al., "Detection of flooding attack on OBS network using Ant Colony Optimization and machine learning," *Computación y Sistemas*, vol. 25, no. 2, pp. 423–433, 2021, doi: 10.13053/cys-25-2-3939.

[22] P. Ghadekar, M. R. Pradhan, D. Swain, and B. Acharya, "EmoSecure: Enhancing Smart Home Security With FisherFace Emotion Recognition and Biometric Access Control," *IEEE Access*, vol. 12, pp. 93133–93144, 2024, doi: 10.1109/ACCESS.2024.3423783.

[23] Y. Omae et al., "Deep learned features selection algorithm: Removal operation of anomaly feature maps (RO-AFM)," *Appl. Soft Comput.*, vol. 134, Art. no. 111809, 2024, doi: 10.1016/j.asoc.2024.111809.

[24] M. K. I. Rahmani et al., "Security in optical wireless communication-based vehicular ad hoc networks using signature and certificate revocation," *J. Nanoelectron. Optoelectron.*, vol. 19, no. 1, pp. 112–119, 2024, doi: 10.1166/jno.2024.3544.

[25] I. R. Nugraha, W. H. N. Putra, and E. Setiawan, "A comparative study of HTTP and MQTT for IoT applications in hydroponics," *J. RESTI*, vol. 8, no. 1.

Abderezzak Benyahia received his Engineering and Master degrees respectively in 2008 and 2012 from UHL Batna. He is currently a Ph.D. student, member of LaSTIC laboratory and Assistant Professor in Computer Science Department of University of Batna 2. His research subjects include Information Systems, Computer networks, Mobile Wireless ad hoc networks (MANets), Wireless Sensor Networks, Modeling and simulation, and Internet of Things.

Ouahab Kadri was born in Batna, Algeria in 1978. He received his Magister degree from the Department of Computer Science, University of Batna, Algeria. He received his PhD from the Department of Industrial Engineering, University of Batna, in 2013, and his Habilitation from the same department in 2018. He was previously an Assistant Professor in the Department of Mathematics and Computer Science at the University of Khenchela, Algeria. He is currently a Professor in the Department of Computer Science at the University of Batna2, Algeria. He has published five books and over 20 papers. His current research interests include evolutionary computation and artificial intelligence.

Hamouma Moumen is currently a Professor with the University of Batna 2, where he is also the Dean of the Faculty of Mathematics and Informatics. His main research interest includes the basic principles of distributed computing systems. He has published papers in the IEEE NCA, OPODIS, PODC, and ICDCN international conferences and articles in international renowned journals, such as the Journal of the ACM and Theoretical Computer Science Journal (Elsevier). He was a recipient of the Best Paper Award at ACM PODC 2014.

Adel Abdelhadi was born in Batna, Algeria in 1978. Adel Abdelhadi received, his Habilitation from the Department of Industrial Engineering, University of Batna, Algeria, in 2021. He received, his PhD from the Department of Industrial Engineering, University of Batna, in 2015. He is currently an Assistant Professor in the Department of Computer Science at the University of Batna2, Algeria. He was an Assistant Professor in the Department of Mathematics and Computer Science at the University of Khenchela, Algeria. He received his Magister degree from Department of Computer Science, University of Batna, Algeria. He has published six books and over 15 papers. His current research interests include artificial intelligence, scheduling and Maintenance.