This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

IEEE Open Journal of the Communications Society

Received XX Month XXXX; revised X Month XXXX; accepted XX Month XXXX. Date of publication XX Month XXXX; date of current version XX Month XXXX.

Digital Object Identifier 10.1109/OJCOMS.2020.1234567

An Improved Binary Spider Wasp Optimization Algorithm for Intrusion Detection for Industrial Internet of Things

Mousa'b Shtayat[1], Mohammad Kamrul Hasan[1*], Senior Member, IEEE, Anil Kumar Budhati[2], Senior Member, IEEE Rossliawati Solaiman[1], Shayla Islam[3,], Senior Member, B. Pandey[4], Huda Saleh Abbas[5], Mamoon M Saeed[6]

[1]Faculty of Information Science and Technology, University Kebangsaan Malaysia, 43600, Bangi, Selangor, Malaysia
[2]Department of ECE,koneru Lakshmaiah Education Foundation, Hyderabad,India & ICSDI,UCSI University, Kuala Lumpur,Malaysia.E-mail:anilbudati@gmail.com".
[3]Institute of Computer Science and Digital Innovation, UCSI university Malaysia, Malaysia (e-mail: shayla@ucsiuniversity.edu.my).
[4]Department Department of Intelligent System and Cyber Security, Astana IT University, Kazakstham (Email: bk.pandey@astanait.edu.kz; l.rzayeva@astanait.edu.kz)
[5]Department of Computer Science, Computer Science and Engineering College, Taibah University Madinah, Madinah, Madinah, Saudi Arabia 42353.
[6] Department of Communications and Electronics Engineering, Faculty of Engineering, University of Modern Sciences (UMS)
Yemen
CORRESPONDING AUTHOR: Mohammad Kamrul Hasan (e-mail: mkhasan@ukm.edu.my); Mamoon M Saeed(e-mail: dr.mamoon@ums-edu.com);

ABSTRACT Ensuring network security, particularly within the Industrial Internet of Things (IIoT), has become paramount with the escalating reliance on Internet applications across diverse sectors, emphasizing the critical need for robust feature selection techniques in IIoT Intrusion Detection Systems (IDS). This paper introduces the Improved Binary Spider Wasp Optimizer (IBSWO) algorithm to address this pressing need. By merging the Spider Wasp Optimizer (SWO) with Genetic Algorithms (GAs) and leveraging flat crossover, the algorithm aims to enhance feature selection efficacy. Validation of the methodological framework was conducted using publicly available real-world datasets, including UNSW-NB15, TON_IoT, and NCTUKM-IIOT. The results demonstrate the superior classification accuracy, precision, recall, and F1-measure of IBSWO compared to established Metaheuristic (MH) algorithms and machine learning techniques. Furthermore, the incorporation of flat crossover and transfer functions presents promising advancements in feature selection methodologies for IIoT IDS, offering implications for enhancing network security, and effectively detecting and mitigating evolving cyber threats.

INDEX TERMS Spider Wasp Optimizer, Flat crossover, Industrial Internet of Things, Intrusion Detection System, TON_IoT, UNSW-NB15, NCTUKM-IIOT, IBSWO.

I. INTRODUCTION

Network security and privacy, particularly in the context of the IIoT, have become paramount due to the widespread use of Internet applications and services in fields such as management and e-commerce, smart cities, and healthcare [1] [2]. The increasing deployment of advanced technologies across these sectors has led to the emergence of various malware and cyberattacks aimed at compromising data, evading access controls, and disrupting software systems or IIoT networks. To combat these threats, a range of protective measures such as encryption, firewalls, and anti-malware tools are employed [3]. These methods are particularly effective in identifying cyberattacks and zero-day attacks.

An IDS is a security tool available as software or hardware that monitors network traffic to identify suspicious and malicious activities, generating alerts and reports as necessary [4]. IDSs can be categorized based on various criteria, including data sources and detection methods. The two main types of IDSs based on data sources are host-based and network-based systems. For detection methods, IDSs primarily use either anomaly-based or signature-based techniques. Signature-based IDSs detect threats by comparing observed activities with known patterns stored in a database, which requires regular updates to recognize new attacks. Anomaly-based IDSs, on the other hand, identify unusual activities by comparing them to established profiles of normal behavior [5]. IDSs handle large amounts of network data, often containing redundant, noisy, or irrelevant information, which can impact performance and resource efficiency. Therefore, feature selection is a critical task to improve the effectiveness and efficiency of IDSs [6].

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

Feature Selection (FS) is a crucial step in the preprocessing stage of constructing effective machine learning models [7]. It involves identifying the most pertinent features that adequately represent the entire dataset, which is vital for data mining tasks. The performance of IDS is greatly influenced by the selected features [8]. FS techniques can be divided into two main categories: filter-based and wrapper-based methods. The filter-based approach assesses the relationship between features and their associated class labels independently of the learning algorithm. In contrast, the wrapper-based approach incorporates the learning algorithm to evaluate feature subsets during the optimization process. While the wrapper-based method is generally more effective, it is also more computationally intensive [9]–[11].

In the realm of IDSs, meta-heuristic (MH) algorithms are frequently harnessed within the wrapper-based approach for FS, primarily owing to their efficacy in enhancing model accuracy [12], [13]. Since the FS is conceptualized as an optimization task operating within a binary search space, researchers commonly leverage binary-based operators alongside various transfer functions. Moreover, additional operators such as crossover and mutation are integrated at the MH algorithm level to fortify the optimization process and circumvent potential entrapment in local optima. The selection of an optimal initial population technique is pivotal, as it directly impacts the convergence rate and the ability to attain optimal fitness levels in the initial iterations. Diverse MH algorithms are deployed to augment the learning process of the wrapper-based FS approach in IDSs, encompassing methodologies such as the Grey Wolf Optimizer (GWO), Reptile Search Algorithm (RSA), hybrid GWO coupled with Particle Swarm Optimizer (PSO), and others highlighted in prior studies [14]–[19].

The Spider Wasp Optimizer (SWO) is a recently developed metaheuristic (MH) algorithm introduced in 2023 by Abdel-Basset et al [20]. It mimics the behavior of female spider wasps in seeking, constructing nests, and mating. This algorithm employs a population of "virtual wasps" to systematically explore optimal solutions within a specified search space. These virtual wasps execute actions such as Levy flights (exploratory jumps), tracking fitter individuals (exploitation), and laying eggs (generating new solutions) to efficiently traverse the search space and converge toward promising positions. One notable advantage of this algorithm is its adaptable control parameters, rendering it applicable to a diverse range of optimization problems with varying requirements. Despite its novelty, SWO has thus far found application primarily in the field of photovoltaic cells and modules [21]. Efforts have also been made to improve this algorithm further [22].

Since SWO is primarily designed for continuous domains, adjustments are necessary to enable its application in FS, which operates within a binary search space. Indeed, FS poses a formidable NP-hard challenge, as highlighted in [23]. The quest to identify the optimal (minimal) feature subset becomes particularly daunting in high-dimensional data scenarios. For instance, exploring all possible subsets in datasets with N features entails evaluating 2N subsets to ascertain the most suitable feature subset for data differentiation [24]. The exhaustive nature of this approach quickly renders it impractical and computationally burdensome, particularly when dealing with high-dimensional datasets. To enhance the efficiency of FS methods, various search strategies can be employed, including the adoption of diverse transfer functions, the implementation of alternative crossover and mutation operators, and the integration of intelligent initial population mechanisms.

In this paper, an improved Binary SWO (IBSWO) algorithm is proposed to select the most appropriate features for IDS in the IIoT domain. The improvements include these contributions:

1. A transfer function is embedded in the SWO optimization framework to map the continuous solutions of the binary domain. This yields to the binary SWO (BSWO).
2. BSWO is merged with the Genetic algorithm (GA) using its original crossover operator and mutation to improve the evaluation process of BSWO.
3. Flat crossover operator is utilized instead of the original crossover of GA for more improvement of the evaluation process of BSWO. This yields the IBSWO.

The proposed iterations of IBSWO undergo evaluation using three publicly accessible real-world datasets related to IDSs and IIoT. Comparative assessments against established MH algorithms are carried out, demonstrating IBSWO's effectiveness in terms of classification accuracy, precision, recall, and F1 measurements. Additionally, comparative evaluations against various machine learning (ML) methods are conducted, favoring the proposed approach. Furthermore, to validate its efficacy, IBSWO's performance is compared with that of BSWO in both binary and multiclass classification scenarios.

This paper is structured as follows: Section II presents a review of related work, discussing previous methods employed in the domains of IDS and optimization techniques in the IIoT. The Background Section (Section III) delves into the original version of the Spider Wasp Optimizer. The proposed method, involving modifications to the SWO, is outlined in Section IV. Section V presents the experimental results and subsequent discussion. Finally, Section VI concludes with a summary of the main findings and suggestions for future directions.

The flowchart of the full methodology is illustrated in Figure 1.

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

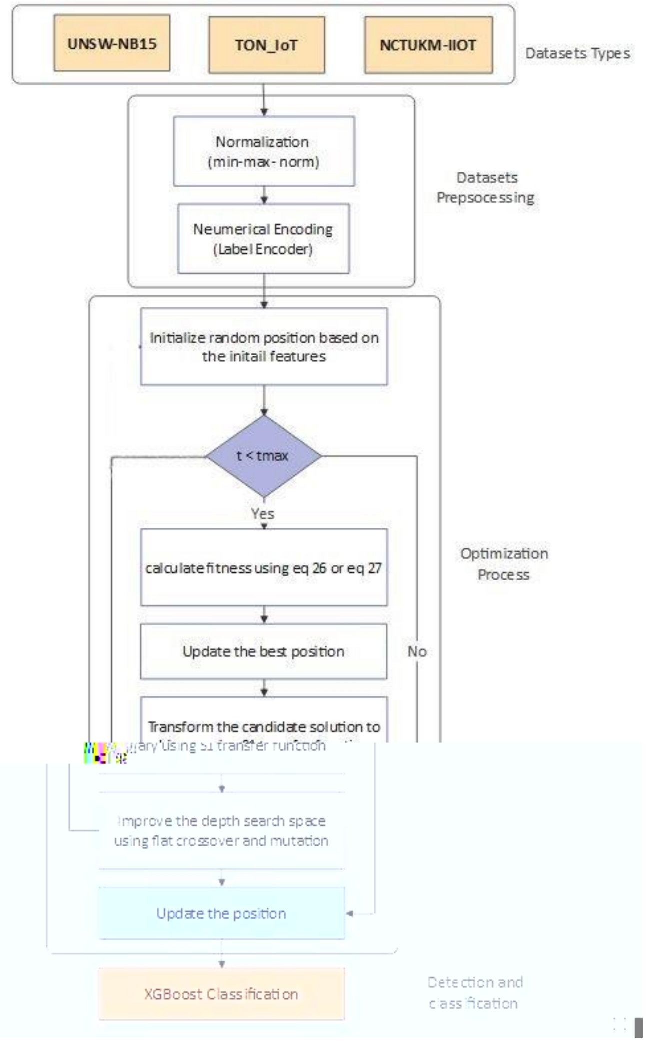

FIGURE 1. Proposed methodology

II. RELATED WORK

In recent years, the evolution of network infrastructures has spurred the development of sophisticated intrusion detection systems (IDS) tailored to specific industry contexts. One such domain experiencing notable advancements is Agriculture 4.0, where network security assumes a pivotal role in ensuring the efficiency and sustainability of farming practices. The advent of optimization algorithms has also emerged as a viable solution in intrusion detection for Industrial Internet of Things (IIoT) environments. A plethora of optimization algorithms, including the Grey Wolf Optimizer (GWO), Salp Swarm Algorithm (SSA), Particle Swarm Optimization (PSO), and Whale Optimization Algorithm (WOA), have been deployed in IDSs to enhance their effectiveness and efficiency. In the subsequent sections, we will delve into further details regarding the state-of-the-art solutions in intrusion detection, elucidating the intricacies of these optimization algorithms and their applications in securing network infrastructures across various industry domains.

Shaik et al. (2023)proposes an Enhanced SVM (EMSVM) model with Orthogonal Learning Chaotic Grey Wolf Optimization (OLCGWO) for intrusion detection in Agriculture 4.0 networks. The authors claim that EMSVM with OLCGWO achieved superior performance compared to other methods like SVM, GWO-SVM, and a hybrid approach, using the TON_IoT dataset for evaluation. They report that for binary classification, EMSVM with OLCGWO achieved a precision of 0.963, recall of 0.9414, and F1 score of 0.9519. For multiclass classification, the reported performance metrics are: precision of 0.9624, recall of 0.9871, and F1 score of 0.9746 [25].

However, study's reliance on a limited dataset raises concerns about generalizability. Additionally, the article doesn't compare the proposed approach with established intrusion detection techniques specifically designed for Agriculture 4.0, making it difficult to assess its unique value. Finally, the computational complexity of OLCGWO might be a drawback in resource-constrained agricultural environments. noting a potential avenue for future improvements.

Lilhore et al. (2023) presents a novel "HIDM" model tailored specifically for intrusion detection within Industry 4.0 networks. This model integrates an optimized Convolutional Neural Network (CNN) with a Long Short-Term Memory (LSTM) network and leverages transfer learning to enhance its efficacy. Specifically, the optimized CNN utilizes a Grey Wolf Optimizer for parameter fine-tuning, while pre-trained weights from a substantial image dataset are incorporated to expedite training and enhance initial performance. The authors highlight promising outcomes on the ToN-IoT and UNSW-NB15 datasets. In binary classification, the HIDM model achieved an accuracy of 0.97 with a precision of 0.96 and recall of 0.74, along with an F1 score of 0.76 on the ToN-IoT dataset. For multiclass classification, the model attained an accuracy of 0.944, with precision, recall, and F1 scores of 0.927, 0.5239, and 0.566, respectively [26].

Similarly, on the UNSW-NB15 dataset, the HIDM model demonstrated robust performance, achieving an accuracy of 0.97 in binary classification, with a precision of 0.96, recall of 0.72, and an F1 score of 0.74. In multiclass classification, it achieved an accuracy of 0.95, with precision, recall, and F1 scores of 0.92, 0.51, and 0.54, respectively. However, the article may lack explicit discussions on other crucial metrics such as true negative rate and detailed comparisons with existing intrusion detection methods within the Industry 4.0 context. This limitation makes it challenging to comprehensively evaluate the unique value proposition of HIDM.

Gad et al. (2021b) proposes a machine learning-based intrusion detection system (IDS) for securing vehicular ad hoc networks (VANETs). The system leverages the ToN-IoT dataset, known for its realistic and diverse nature, to train and evaluate its attack detection capabilities. The authors employed various techniques, including data preprocessing, feature selection, and machine learning algorithm evaluation. XGBoost emerged as the most effective

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

algorithm for this specific dataset and task. The resulting IDS achieved impressive performance in both binary and multiclass classification, reaching an accuracy of 0.982 and 0.979, respectively. Additionally, precision, recall, and F1-score metrics also remained high, indicating the model's ability to accurately identify different attack types with low false positives and negatives [27].

However, the evaluation solely relied on the ToN-IoT dataset, raising concerns about the generalizability of these findings to real-world VANET scenarios with diverse network configurations and attack types. Furthermore, the article might not delve into the interpretability of the model, making it difficult to understand which features are most crucial for accurate attack detection.

Ahmad et al. (2022) introduces a novel intrusion detection system (IDS) for Industrial Internet of Things (IIoT) networks. It leverages a Deep Random Neural Network (DRaNN) for robust traffic analysis and combines it with Particle Swarm Optimization (PSO) to optimize the DRaNN's performance. DRaNN offers advantages like distributed learning and improved generalization, making it suitable for complex IIoT data. PSO, inspired by swarm intelligence, helps fine-tune the DRaNN's internal settings for optimal attack detection [28].

The researchers evaluated the DRaNN-PSO system using the TON-IoT dataset, known for its realistic and diverse nature. The system achieved impressive results, demonstrating strong performance in both binary and multi-class classification tasks. For binary classification on the TON-IoT dataset, the accuracy, precision, recall, and F1-score reached 0.9957, 0.9966, 0.9959, and 0.9962, respectively. Multi-class classification performance was also high, with 0.989 accuracy, 0.991 precision, 0.990 recall, and 0.990 F1-score. The system was also evaluated using the UNSW-NB15 dataset, achieving similar results. Binary classification on UNSW-NB15 yielded an accuracy of 0.9912, precision of 0.9927, recall of 0.9908, and F1-score of 0.9917. Multi-class classification on UNSW-NB15 resulted in an accuracy of 0.987, precision of 0.989, recall of 0.987, and F1-score of 0.988.

Some limitations require further exploration. Firstly, the generalizability of these findings to various real-world scenarios with diverse attack types and network configurations needs verification using a wider range of datasets. Secondly, while the article reports high-performance metrics, it might not delve into the interpretability of the model's decisions. Understanding which features are most influential for attack detection could be valuable. Finally, the computational complexity of the DRaNN-PSO approach might limit its use on resource-constrained IoT devices.

Awadallah, et al. proposed Binary Enhanced RSO with Crossover Operators (BERSOC). An S-shaped transfer function is used in this method to translate RSO solutions into binary representation. BERSOC also incorporates the local search method from PSO within the RSO loop to enhance its convergence characteristics. To further increase diversity in the population, BERSOC applies three crossover techniques: one-point, two-point, and uniform crossover [29].

Mafarja et al. proposed a Whale Optimization Algorithm (WOA) enhanced with feature-selection for detecting IoT attacks. This enhanced WOA can handle binary data since it has several V- and S-shaped transfer functions. Results from experiments using the N-BaIoT dataset showed that the Augmented WOA with S-shaped functions performed better than the ones with V-shaped functions, indicating that S-shaped functions are more useful in IoT intrusion detection systems [30].

The Reptile Swarm Algorithm (RSA) is a metaheuristic that imitates the hunting behavior of crocodiles. [31] suggested leveraging data from Internet of Things environments to tackle the feature selection (FS) issue in intrusion detection systems by fusing RSA with deep learning. To assess the performance of RSA with deep learning, they used datasets designed for IoT applications, including KDDCup-99, NSLKDD, CICIDS 2017, and BoT-IoT. Comparisons with other established algorithms, like Grey Wolf Optimization (GWO) and the Bat Algorithm (BAT), revealed that RSA with deep learning yielded competitive outcomes in these evaluations.

In another research to solve the FS problem, [32] developed the Improved Sticky Binary Particle Swarm Optimization (ISBPSO) technique. An initialization method that considers feature weighting data produced from mutual information is incorporated into ISBPSO. Furthermore, ISBPSO employs a technique based on a dynamic bit-masking strategy to gradually reduce the search space during the optimization process. Experiments conducted using twelve datasets from the UCI repository demonstrated that ISBPSO outperforms other PSO variations in terms of performance.

Multi-population-based Particle Swarm Optimization (MPPSO) is a contemporary variant of PSO designed for addressing the FS problem [33]. Multiple populations are produced with MPPSO, and PSO is applied to each of them concurrently. The performance of MPPSO was assessed using 26 UCI and 3 Arizona State University (ASU) datasets, comparing it with other algorithms like traditional PSO. The results indicated that MPPSO demonstrated higher accuracy compared to these other algorithms.

A new hybrid algorithm called SCHHO was created by combining the Sine Cosine Algorithm (SCA) with the Harris Hawks Optimization (HHO) method [34]. Within SCHHO, SCA functions as an exploration technique to expand HHO's search capabilities. To enhance exploitation, SCHHO dynamically adjusts candidate solutions to help the HHO algorithm avoid becoming trapped in local optima.

[35] proposed a hybrid optimization algorithm that merges the Salp Swarm Algorithm (SSA) with the Grey Wolf Optimization (GWO) algorithm, abbreviated as SSA-FGWO. This technique was created to handle FS difficulties as well as ongoing optimization issues. In SSA-FGWO, the GWO's update mechanism is used to adjust the follower candidate solutions, while the SSA's update method is used for updating the leader candidate solutions. Using eighteen real-world datasets, SSA-FGWO was compared against popular optimization methods such as GWO and SSA. The

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

simulation results showed that SSA-FGWO is a viable method.

Improved HHO (IHHO) is the improved version of the Harris Hawks Optimization (HHO) method created by Hussien and Amin [36] to address the early convergence issue. To increase the quality of the result, this updated algorithm combines Chaotic Local Search, Opposition-Based Learning (OBL), and a self-adaptive strategy. IHHO was tested on seven datasets from the UCI library in feature selection tasks. IHHO outperformed industry-leading algorithms including HHO, Crow Search Algorithm (CSA), Particle Swarm Optimization (PSO), and Whale Optimization Algorithm (WOA) in terms of both performance and solution quality, according to a comparative analysis.

The Multi-objective Binary Genetic Algorithm with an Adaptive Operator Selection mechanism, or MOBGA-AOS, was proposed by [37]. This algorithm makes use of five distinct crossover operators, each designed to handle a different optimization challenge: uniform crossover, shuffle crossover, reduced surrogate crossover, two-point crossover, and single-point crossover. Using a roulette wheel mechanism, MOBGA-AOS chooses a crossover operator at each iteration based on predetermined probabilities, producing new candidate solutions for later iterations. MOBGA-AOS produced the best accurate results among the studied algorithms, as evidenced by evaluation against five other multi-objective binary algorithms utilizing ten UCI datasets.

Gad et al. tested various ML methods for both binary and multi-class classification problems, incorporating the Chi-square (Chi2) technique for feature selection and the Synthetic Minority Over-sampling Technique (SMOTE) for class balancing. The experimental results showed that the XGBoost method outperformed other ML methods. This work proposed adopting the ToN-IoT dataset to better represent contemporary attack patterns and recommended using the XGBoost method for enhanced security in VANETs [38].

Shtayat et al. proposed an explainable ensemble deep learning (DL)-based Intrusion Detection System (IDS) to enhance the security of Industrial Internet of Things (IIoT) systems. This model addresses common issues in IDSs, such as high false-positive rates and opaque decision-making. By incorporating Shapley additive explanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME), the framework provides valuable insights into the decision-making process of DL-based IDSs, aiding cybersecurity professionals in improving system effectiveness. The proposed system, evaluated using the ToN_IoT dataset, combines CNN models with an ensemble strategy, achieving accuracy rates over 99%. This approach improves transparency, fosters trust, and supports continuous improvement of the system. However, a limitation of this research is the reliance on a single dataset (ToN-IoT), which may not fully capture the variety of real-world IIoT scenarios. Future work will focus on refining model architecture, exploring diverse CNN setups, and incorporating advanced techniques like autoencoders and real-time monitoring. Enhanced interpretability and continuous updates will be crucial for maintaining effectiveness against sophisticated attacks [1].

To enhance IDS performance, Aziza, et al. performed three machine learning algorithms—decision jungle (DJ), random forest (RF), and support vector machine (SVM)—were evaluated based on their accuracy, precision, and recall using the CIC-IDS2017 dataset and KDD methodology. The SVM achieved the highest average accuracy (98.18%) and precision (98.74%), while RF had the highest recall (97.62%). Overall, SVM was found to be the most effective algorithm for detecting intrusions. However, the performance of these algorithms is lower compared to other contemporary solutions. Additionally, the KDD dataset used is one of the oldest for IoT, which limits the relevance and effectiveness of the results [39]

III. BACKGROUND INFORMATION

In this section, we elaborate on the procedural aspects of the binarization strategy proposed for utilization in conjunction with the SWO algorithm. Following the generation of the search space, we will expound upon the methodology employed to establish its binary representation. Subsequently, we will proceed to develop the binary variant of the SWO algorithm and incorporate it into the binary search matrix.

A. Inspiration of the Spider Wasp Optimizer

Define The new optimization technique known as Spider Wasp Optimizer (SWO) is inspired by the hunting and nesting habits of some wasp species, especially those that engage in obligatory brood parasitism, in which females lay a single egg in each spider's abdomen[20]. First, female wasps search their environment for suitable spiders, immobilizing and carrying them to nests that have already been set up. The suggested algorithm, SWO, is primarily inspired by this behavior. The female wasp deposits an egg on the spider's abdomen before closing the nest after finding a suitable victim and nest and bringing the prey to the nest. A group of female wasps is randomly distributed around the search space in their suggested SWO technique. Then, using the hunting and tracking strategies unique to their species, each wasp moves continually around the area in search of a spider that corresponds to the sex of its progeny, as determined by the haplodiploid sex-determination system common to hymenopterans [40], [41]. When the female wasps locate a suitable spider, they begin to forage around the spider's web and repeatedly search the ground for any fallen spiders. They then paralyze the meal and carry it to the ready-made nest, where they seal it and deposit an egg within the spider's abdomen.

To summarize, the simulated behaviors of the wasps in our algorithm include:

- Searching behavior: Initiates the optimization process by seeking suitable prey for larval growth.
- Tracking and evading behavior: Once prey/spiders are located, they may attempt to flee, prompting the

female wasp to pursue, immobilize, and transport the most suitable candidate.

- Nesting behavior: Simulates the process of transporting prey to appropriately sized nests for egg deposition.
- Mating behavior: Represents the properties of offspring produced through hatching, using the crossover rate (CR), which is a uniform crossover operator between male and female wasps with a given probability.

In the subsequent subsection, we will present a mathematical model detailing these behaviors along with a more comprehensive description.

B. *The Mathematical models of SWO*

Based on the behavioral patterns observed in Spider Wasps, this section begins by addressing the setup procedure for the Spider Wasp Optimizer (SWO). It then proceeds to detail the mechanisms involved in continuously updating the position of the waterwheel during both the exploration and exploitation phases of the optimization process. Algorithm 1 explains the Pseudo-code of SWO.

1) Generation of the Initial Population in SWO

Within the current generation, each female spider-wasp is a solution in the suggested algorithm and may be encoded into a D-dimensional vector using the following expressions:

$$M_i^{\rightarrow} = [x_1, x_2, x_3, \ldots, x_D] \tag{1}$$

Using the following procedure, a set of N vectors can be randomly produced inside the upper initial parameter bound $H^{\rightarrow}$ and the lower initial parameter bound $L^{\rightarrow}$.

$$M_{Pop} = \begin{bmatrix} M_{1.1} & \cdots & M_{1.D} \\ \vdots & \ddots & \vdots \\ M_{N.1} & \cdots & M_{N.D} \end{bmatrix} \tag{2}$$

where M_{Pop} represents the initial population of spider wasps. The subsequent equation can be utilized to randomly generate any solution within the search space:

$$M_i^{\rightarrow t} = L^{\rightarrow} + r^{\rightarrow} * (H^{\rightarrow} - L^{\rightarrow}) \tag{3}$$

where i is the population index (i = 1, 2, ..., N); t is the generation index; and $r^{\rightarrow}$ is a vector of D-dimension randomly initialized values between 0 and 1. The behaviors of the spider wasps will then be mathematically recreated in order to present a unique metaheuristic algorithm for solving optimization issues. The following is an outline of the behaviors:

- The habits of hunting and nesting
- Behavior in mating

2) The habits of hunting and nesting

The female spider wasp initiates its activity with an initial exploration phase aimed at identifying potential prey, followed by transitioning to an exploitation stage to approach and attack the target upon its discovery. The mathematical intricacies of these two phases are outlined below.

a) Search stage (Exploration)

As previously mentioned, the female spider wasp activates this operator at the onset of the search process to locate its desired prey. This behavior can be represented mathematically by the following expression:

$$M_i^{\rightarrow t+1} = M_i^{\rightarrow t} + \mu_1 * (M_a^{\rightarrow t} - M_b^{\rightarrow t}) \tag{4}$$

Where $M_a^{\rightarrow t}$ and $M_b^{\rightarrow t}$ represent two randomly selected solutions from the current population. The consistent forward velocity of the female wasp is computed using an adaptive factor termed μ_1, which is defined mathematically in the following equation:

$$\mu_1 = |rn| * r_1 \tag{5}$$

Where r_1 represents a random number chosen from a uniform distribution between zero and one, while rn is a random number sampled from a normal distribution. When prey falls from the orb, it may be lost if female wasps fail to capture it. To retrieve the lost prey, they utilize an alternative exploration strategy, which is mathematically described as follows:

$$M_i^{\rightarrow t+1} = M_c^{\rightarrow t} + \mu_2 * (L^{\rightarrow} + r_2 * (H^{\rightarrow} - L^{\rightarrow})) \tag{6}$$

$$\mu_2 = B * \cos(2\pi l) \tag{7}$$

$$B = \frac{1}{1+e^{l}} \tag{8}$$

Where $M_c^{\rightarrow t}$ denotes a randomly selected solution from the current population, representing the position of the dropped prey. $L^{\rightarrow}$ denotes the lower bound, while $H^{\rightarrow}$represents the upper bound. r_2 is a vector comprising random values generated within the interval [0, 1], and l is a random number selected from the range 1 to -2. Ultimately, the following equation depicts the compromise between equations (4) and (6), facilitating the forward movement of the ith solution.

$$M_i^{\rightarrow t+1} = \begin{cases} Eq.4 & r_3 < r_4 \\ Eq.6 & Otherwise \end{cases} \tag{9}$$

Where r_3 and r_4 represent two arbitrary numbers selected from the range between zero and one.

b) Following and escaping stage

Spider wasps employ the following formula to compute new positions relative to the spiders, enabling them to capture their prey effectively.

$$M_i^{\rightarrow t+1} = M_i^{\rightarrow t} + C * | 2 * r_5^{\rightarrow} * M_a^{\rightarrow t} - M_i^{\rightarrow t} | \tag{10}$$

$$C = (2 - 2 * \left(\frac{t}{t_{max}} \right)) * r_6 \tag{11}$$

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

Where t and t_{max} denote the current function evaluation and maximum function evaluation, respectively. $\vec{r_5}$represents a vector containing numerical values randomly generated between 0 and 1, following a uniform distribution. r_6 is a random numerical value generated between 0 and 1, also adhering to a uniform distribution. However, there exists a chance for the spiders to evade the female wasps, causing the distance between them to gradually increase. To simulate this behavior in SWO, the following equation is employed:

$$M_i{}^{\to t+1} = M_i{}^{\to t} * \; vc^{\to} \tag{12}$$

Where $vc^{\to}$ represents a vector containing numerical values arbitrarily generated using the normal distribution, with the values ranging between k and $-k$. The value of k is derived by applying the following formula:

$$k = 1 - \left(\frac{t}{t_{max}}\right) \tag{13}$$

The following equation can be utilized to achieve a satisfactory compromise between equations (10) and (12):

$$M_i{}^{\to t+1} = \begin{cases} Eq.10 & r_3 < r_4 \\ Eq.12 & Otherwise \end{cases} \tag{14}$$

In SWO, the following equation is employed to strike a balance between (9) and (13):

$$M_i{}^{\to t+1} = \begin{cases} Eq.9 & p < k \\ Eq.13 & Otherwise \end{cases} \tag{15}$$

Where p is a randomly selected number from the range [0, 1], following the characteristics of the uniform distribution.

c) Nesting behavior (exploitation)

Female wasps retrieve the incapacitated spider into their nest. In addition to using pre-existing nests or cavities, spider wasps may dig and create cells in soil and build mud nests in leaves or rocks. Given these varied nesting habits, SWO employs two equations to model them. The first equation involves attracting the spider to the optimal location for nest creation, where the immobilized spider and egg are situated over its abdomen, as outlined in the following formula:

$$M_i{}^{\to t+1} = M^{\to *} + \cos(2\pi l) * (M^{\to *} - M_i{}^{\to t}) \tag{16}$$

Where $M^{\to *}$ represents the optimal solution obtained thus far. Building the nest at the site of a female spider chosen at random from the population is the second equation. To avoid building two nests in the same location, this equation additionally includes an extra step size. Mathematically, this equation is described as follows:

$$M_i{}^{\to t+1} = M_a{}^{\to t} + r_3 * |\gamma| * \left(M_a{}^{\to t} - M_i{}^{\to t}\right) + (1 - r_3) * U^{\to} * \left(M_b{}^{\to t} - M_c{}^{\to t}\right) \tag{17}$$

Where γ is a randomly chosen numerical value based on the levy flight, and U is a vector containing binary values that determine whether the additional step size is employed during the updating process. The decision to use the additional step size is determined by the following defined factor:

$$U^{\to} = \begin{cases} 1 & \vec{r_4} > \vec{r_5} \\ 0 & Otherwise \end{cases} \tag{18}$$

where $\vec{r_4}$ and $\vec{r_5}$ are two random vectors obtained from a uniform distribution, each containing numerical values ranging from zero to one. To update each solution during optimization, equations (16) and (17) are interchanged based on the equation below:

$$M_i{}^{\to t+1} = \begin{cases} Eq.16 & r_3 < r_4 \\ Eq.17 & Otherwise \end{cases} \tag{19}$$

Ultimately, the balance between hunting and nesting behaviors is attained through Eq. (20), wherein all spider wasps initiate the optimization process by searching for their respective spiders. Subsequently, The wasps take their appropriate spiders and carry them to the nests that have already been set up.

$$M_i{}^{\to t+1} = \begin{cases} Eq.15 & i < N * k \\ Eq.19 & Otherwise \end{cases} \tag{20}$$

3) Mating behavior

In the SWO, the mating behavior of wasps is considered. The capacity to identify gender in spider wasps based on the size of the host in which an egg is laid is one of their most important characteristics. Wasps are characterized by their size; females are greater in size, and males are smaller. A possible solution in the current generation is represented by each spider wasp, and a newly created potential solution within that generation is represented by the spider wasp egg. The generation of new solutions/spider wasp eggs follows the equation below:

$$M_i{}^{\to t+1} = Crossover\,(M_i{}^{\to t}, M_m{}^{\to t}, Cr) \tag{21}$$

Where $M_m{}^{\to t}$ and $M_i{}^{\to t}$ represent two vectors for the female and male spider wasps, respectively, and Crossover is the uniform crossover operator applied to $M_m{}^{\to t}$ and $M_i{}^{\to t}$ with a probability denoted as Cr. To differentiate male spider wasps from females, the following formula is employed in SWO:

$$M_m{}^{\to t+1} = M_i{}^{\to t} + e^l * |\beta| * v_1{}^{\to} + (1 - e^l) * |\beta_1| * v_2{}^{\to} \tag{22}$$

where β and β_1 are two randomly selected numbers obtained from the normal distribution, and $v_1{}^{\to}$ and $v_2{}^{\to\wedge\longrightarrow}$ are two vectors generated using the following formula:

$$v_1{}^{\to} = \begin{cases} M_a{}^{\to} - M_i{}^{\to} & f(M_a{}^{\to}) < f(M_i{}^{\to}) \\ M_i{}^{\to} - M_a{}^{\to} & Otherwise \end{cases} \tag{23}$$

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

$$\vec{v_2} = \begin{cases} \vec{M_b} - \vec{M_c} & f(\vec{M_b}) < f(\vec{M_c}) \\ \vec{M_c} - \vec{M_b} & Otherwise \end{cases} \quad (24)$$

Indexes a, b, and c are three selected solutions from the population to ensure their uniqueness as a ≠ i ≠ b ≠ c. Through crossover, genetic material from two spider wasp parents is combined to produce an offspring inheriting traits from both. Hunting and mating behavior balance is controlled by a predefined factor called the tradeoff rate (TR).

4) Decreased population and preservation of memory

After laying her eggs, the female spider will seal the nest and move to a new location, indicating her optimization role is done. Eliminating some wasps in the population can speed up convergence time and allow remaining wasps to conduct more evaluations. Population size is adjusted dynamically during optimization using a specific formula.

$$N = N_{min} + (N - N_{min}) * k \quad (25)$$

Where N denotes the size of the population and $Nmin$ is the minimum population size needed to avoid local minima. Furthermore, SWO includes a memory retention method where the best-ranked wasp is passed on to the next generation. Essentially, each wasp's suggested new position is compared to its current one, and if the new solution is inferior, it is substituted.

Algorithm 1 Pseudo-code of SWO

Input: N, $\boldsymbol{N_{min}}$, CR, TR, $\boldsymbol{t_{max}}$
Output: $\vec{\boldsymbol{M}}^{*}$

```
Initialize N female wasps, $\vec{M_i}^{t} (i = 1, 2, ……, N)$, using Eq. 3
Evaluate each $\vec{M_i}^{t}$ and finding the one with the best fitness in $\vec{M}^{*}$
$t = 1$; //the current function evaluation
while ($t < t_{max}$)
    $r_6$: random number between 0 and 1
    if($r_6 < TR$)%% Hunting and Nesting Behavior
        for $i = 1 : N$
            Applying Eq. 20
            $t = t + 1$
        End for
    Else %% Mating Behavior
        for $i = 1 : N$
            Applying Eq. 21
            $t = t + 1$
        End for
    End if
    Applying Memory Saving
    Updating N using Eq. 25
End while
```

IV. THE PROPOSED IBSWO ALGORITHM

In this section, we present the improved Binary Spider Wasp Optimizer (IBSWO) to solve the FS then attacks detection process in IIoT. IBSWO unfolds in two distinct phases. The initial step involves binary improvement, followed by the subsequent integration of SWO with the genetic algorithm. In the second phase, there is a Flat crossover operator will utilized instead of the original crossover operator in the GA. Figure 2 shows the flowchart of the proposed IBSWO.

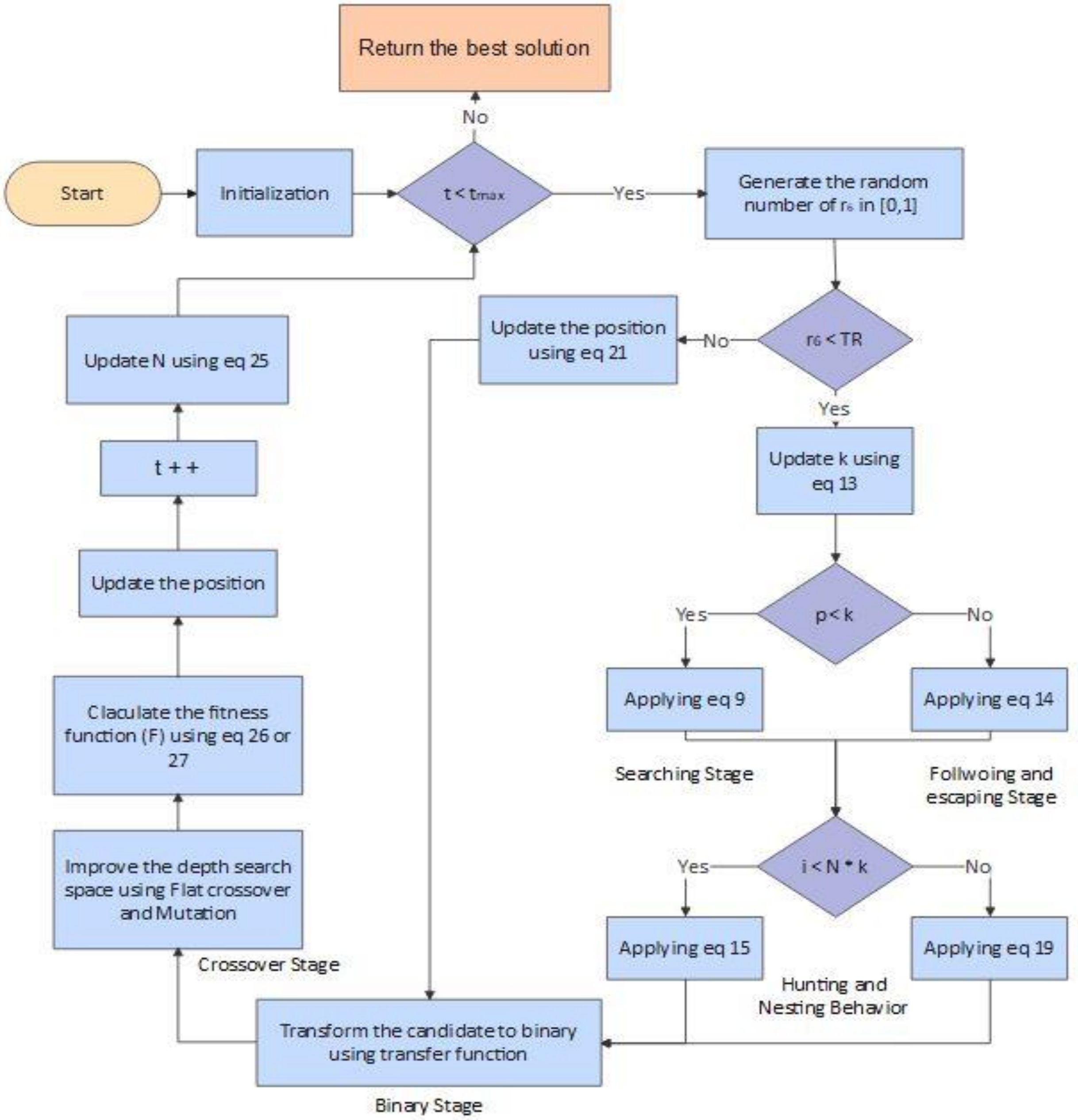

FIGURE 2. Proposed improved Binary Spider Wasp Optimizer

A. *Binary improvement*

Since the IBSWO algorithm creates spider wasps with continuous values, a two-step transfer function is needed to convert continuous spider wasp into binary spider wasp. To identify the most effective transfer function for our experiment, we evaluated eight options (four S-shaped and four V-shaped) using IBSWO simulations (refer to Table 1 for details). The results indicated that function s1 achieved the highest accuracy, and consequently, it was chosen in the experiments described later in this paper.

In S1, the decision variable $M_i^{\boldsymbol{j}}$ in spider wasp $M_i^{j} = < M_i^{\mathbf{1}}, M_i^{\mathbf{2}}, \dots, M_i^{\boldsymbol{m}} >$ at iteration t to calculate the probability of changing $M_i^{\boldsymbol{j}}$ to 0 or 1. The probability is calculated using the equation below:

$$\mathrm{T}\left(M_i^{\,j}(\mathrm{t})\right) = 1/(1 + e^{-2s}) \tag{26}$$

Then we change $M_i^{\,j}(\boldsymbol{t})$ to 0 or 1 as follows:

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

$$M_i^{\mathrm{j}}(\mathrm{t}+1) = \begin{cases} 1 - M_i^{j}(\mathrm{t}) & r < \mathrm{T}\left(M_i^{j}(\mathrm{t})\right) \\ M_i^{j}(\mathrm{t} \quad \mathrm{t} & \end{cases}$$

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

The Offspring's Journey: After the crossover, the newly created individual can be evaluated and potentially integrated into the population for further genetic operations like selection, mutation, or even more crossovers.

Algorithm 2 Pseudo-code of Flat Crossover

```
Select two parents $x^{(t)}$ and $y^{(t)}$ from a parent pool
Create one offspring $x^{(t+1)}$ as follow:
For $i$ = 1 to $n$ do
    Choose a uniform random real number
    $a \in <\min(x_i^t, y_i^t), \max(x_i^t, y_i^t)>$
    $x_i^{(t+1)} = a$
End do
```

C. Fitness Function

By employing Flat crossover techniques, we strive to accomplish a two-fold objective: minimizing the number of chosen features while simultaneously enhancing the accuracy of both classification and detection.

To realize these objectives, our proposed approach utilizes the following fitness function [42].

$$F(s) = \alpha * ERR + \beta * \frac{|R|}{|N|} \tag{28}$$

Where:

- F(s): the fitness function of spider wasp s.
– ERR: the error rate achieved by a classifier (XGBoost classifier) with s as input.
– |R|: the number of features in s.
– |N|: the number of features in the dataset.
– α: the weight of ERR.
– β=1- α: the weight for the selection ratio (|R|/|N|)

An alternative equation to Eq. 28 is as follow:

$$F(s) = maximize(\, Acc(s) + s_f * \left(1 - \frac{L_f}{L_t}\right)) \tag{29}$$

Where:

– F(s): the fitness function of spider wasp s.
– ERR: the accuracy achieved by XGBoost classifier
– s_f : a scaling factor between 0 and 1.
– L_f : the number of attributes in s.
– L_t: the number of features in the given dataset

Algorithm 3 Pseudo-code of IBSWO

```
Input: N, $N_{min}$, CR, TR, $t_{max}$
Output: $M^{\to *}$
Initialize N female wasps, $M_i^{\to t}(i = 1, 2, \ldots\ldots, N)$, using Eq. 3
Evaluate each $M_i^{\to t}$ and finding the one with the best fitness in $M^{\to *}$
$t = 1$; //the current function evaluation
while ($t < t_{max}$)
    $r_6$: random number between 0 and 1
    if($r_6 < TR$)%% Hunting and Nesting Behavior
        for $i = 1: N$
            Applying Eq. 20
            $t = t + 1$
        End for
    Else %% Mating Behavior
        for $i = 1: N$
            Applying Eq. 21
            $t = t + 1$
        End for
    End if
    ---------Transferring solutions to Binary Ones ----
    Apply the transfer function as in Table I to the updated candidate solutions.
    ----------Crossover operator------------------
    Select two parents $x^{(t)}$ and $y^{(t)}$ from a parent pool
    Create one offspring $x^{(t+1)}$ as follow:
    For $i = 1: N$ do
        Choose a uniform random real number
        $a \in <\min(x_i^t, y_i^t), \max(x_i^t, y_i^t)>$
        $x_i^{(t+1)} = a$
        Mutation operation
    End do
    Calculate the fitness function (F)
    $t = t + 1$
    Applying Memory Saving
    Updating N using Eq. 25
End while
```

V. Experimental Results

This section details the experiments conducted to evaluate the proposed algorithm's performance. We assess its efficiency and reliability using various evaluation metrics and real-world application data. Section A delves into the real-world datasets used as benchmarks, while Section B details the evaluation metrics chosen to assess efficiency and accuracy. Section C then analyzes how efficiently IBSWO converges on optimal solutions. Next, Section D compares IBSWO's performance against leading optimization algorithms, highlighting its competitive edge.

A. Parameter settings

Table 3 outlines the parameter settings for IBSWO. Although these settings are recommended by existing research, we determined suitable parameter ranges through an analysis of the algorithm's mathematical properties and design principles. We conducted multiple algorithm runs with various parameter combinations and compared their outcomes. The performance metrics included convergence speed, solution quality, robustness, and computational efficiency. Additionally, we performed sensitivity analysis by systematically varying one parameter at a time while keeping others fixed, and then observing the changes in performance metrics. This approach helped us identify critical parameters and their impact on the algorithm's convergence, exploration-exploitation balance, and overall effectiveness.

To maximize the performance of IBSWO, the right parameter settings must be used. While expanding the population can enhance the exploration of the search space, it will also result in longer calculation times and higher memory use. In a similar vein, increasing the number of runs can lower the probability of local optima but also increases computational costs, and more iterations can improve solution accuracy but increase computational costs. To narrow the search space and keep people in realistic areas, the lower and upper limits (lb and ub) must have the right values. TR regulates the possibility of trade-offs between hunting and mating activities. The performance of IBSWO can be greatly impacted by changing these variables. Therefore, in order to find the best values for resolving the feature selection issue in IDSs, a sensitivity analysis was carried out.

TABLE 2. Parameters settings

Parameter	Value
Population size	*20*
# of iterations	*100*
Dimension	*# of features*
# of runs	*5*
t_{max}	*20*
Tradeoff Rate (TR).	*0.3*
Crossover probability	*0.2*
lb	*-1*
up	*1*
β	*1.5*

B. DATASET

Three real-world IIoT and IDS datasets were used to validate our procedure: UNSW-NB15, TON_IoT,a nd NCTUKM-IIOT datasets. The research community often relies on publicly available datasets like UNSW-NB15 and TON_IoT for evaluating Intrusion Detection Systems (IDS). These datasets are popular choices due to their recent updates reflecting modern attack scenarios and their accessibility for researchers. In my thesis, I've developed a new dataset, NCTUKM-IIOT, that caters to real-world requirements and is currently under copyright protection. This new dataset offers a valuable contribution alongside established options like UNSW-NB15 and TON_IoT.

UNSW-NB15: A widely used benchmark dataset for network intrusion detection, offering a comprehensive set of labeled network traffic data [43].

TON_IoT: A dataset specifically designed for anomaly detection in IoT environments, containing various attack scenarios and network configurations [44].

NCTUKM-IIoT: A novel dataset created by the authors, capturing real-world IIoT network traffic characteristics, further enriching the evaluation process. These datasets are described in Table 3 concerning the number of features, instances, and classes.

TABLE 3. Datasets Description

Dataset	# of features	# of instances	# of classes
UNSW-NB15	45	257473	9
TON_IoT	45	461043	9
NCTUKM-IIoT	40	718716	16

Figure 4, 5 and 6 are illustrated the attacks and instances details for the three datasets.

The preprocessing steps for dataset preperations are include:

Dataset Normalization using min-max normalization technique as in the following equation:

$$\boldsymbol{new\ x_{ij}} = \frac{\boldsymbol{x_{ij} - \min(x_i)}}{\boldsymbol{\max(x_i) - \min(x_i)}} \tag{30}$$

Where $\boldsymbol{new\ x_{ij}}$ the new value of each is is feature, and $\boldsymbol{x_{ij}}$ is the old value of the feature. $\boldsymbol{\min(x_i)}$ is the min value of x and $\boldsymbol{\max(x_i)}$ is the max value of x.

Numerical Encoding using label encoder to convert the categorical values to numerical values.

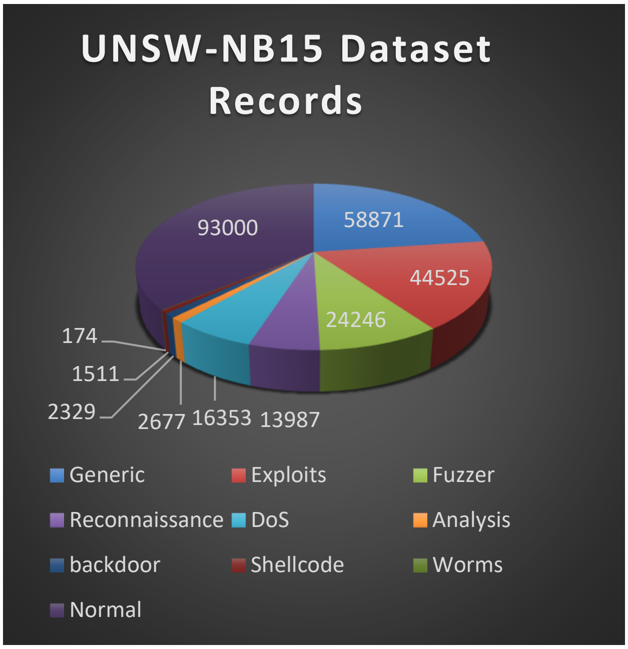

FIGURE 3. UNSW-NB15 dataset attacks and records

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

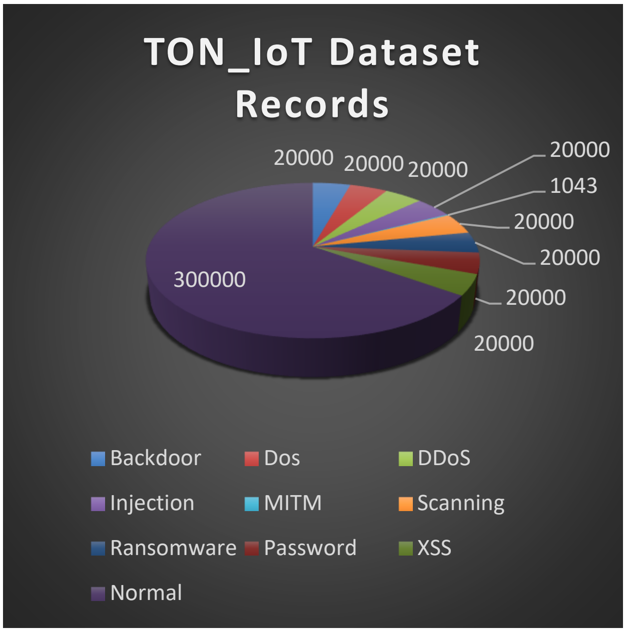

FIGURE 4. TON_IoT dataset attacks and records

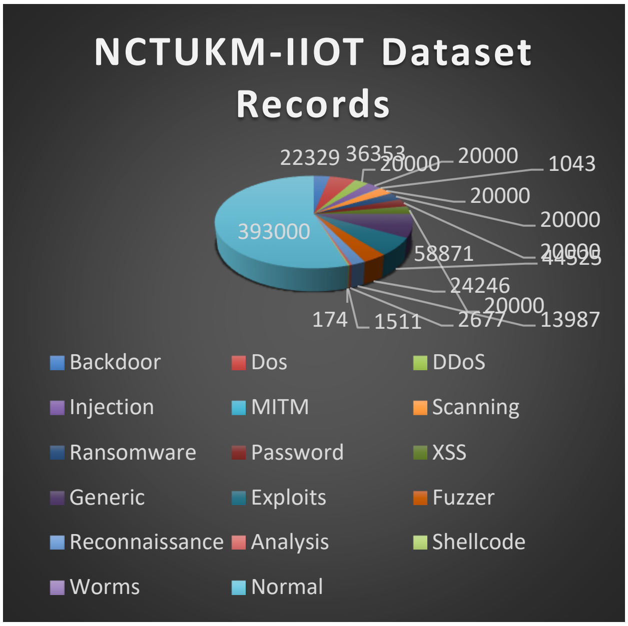

FIGURE 5. NCTUKM-IIOT dataset attacks and records

C. EVALUATION MEASURES

To assess the effectiveness of the proposed algorithms, including IBSWO, we compare their performance using various evaluation metrics. Here, we'll focus on how recent algorithms stack up against IBSWO based on the following criteria:

The average classification accuracy for IBSWO. Measured by the following equation.

$$\text{Accuracy}= (TP+TN)/(TP+FP+TN+FN) \tag{31}$$

Where TP is the True Positive rate represent the correctly identified class, FP is the False Positive rate represents the false identified class

The proportion of cases identified as positive that actually turned out to be positive (Precision).

$$\text{Precision}=TP/(TP+FP) \tag{32}$$

The proportion of actual positive cases that the algorithm correctly identified as positive (Recall)

$$\text{Recall}= TP/(TP+FN) \tag{33}$$

The harmonic mean of Precision and Recall. (F1-score).

$$\text{F1 score}=2* (P*R)/(P+R) \tag{34}$$

D. CONVERGENCE BEHAVIOR OF IBSWO

The research delves into IBSWO to assess how the introduced changes affect the core algorithm. Figures 6 and 7 compare the convergence behavior of IBSWO, visualizing how its fitness values (a measure of solution quality) improve over time. This comparison helps to evaluate the effectiveness of the proposed IBSWO towards optimal solutions compared with the BSWO.

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

FIGURE 6. The BSWO and IBSWO binary convergence curves over the UNSW-NB15, TON_IoT, and NCTUKM-IIOT datasets.

Figures 6 and 7 illustrate IBSWO's impressive convergence speed. In Figure 6 (binary classification), IBSWO consistently reaches good solutions faster than BSWO within a few iterations. Similarly, Figure 7 (multiclass classification) demonstrates IBSWO's superior convergence trend across various datasets compared to BSWO. This rapid convergence suggests that IBSWO effectively balances exploration and exploitation throughout the optimization process.

During the initial stages of the simulation, IBSWO excels in exploration, likely due to the utilization of the "flat crossover operator." This operator effectively expands the search space, allowing IBSWO to identify promising areas for potential solutions. As the simulation progresses, IBSWO seamlessly transitions to a more exploitative phase, focusing on refining the solutions within the identified promising areas. This shift might be attributed to the "transfer function" incorporated into the algorithm.

Finally, the integration of XGBoost classification plays a critical role. XGBoost's high accuracy ensures IBSWO converges towards optimal solutions, its regularization techniques prevent overfitting to the training data, and its scalability allows efficient handling of large datasets – all contributing to IBSWO's superior performance.

D. Comparison between IBSWO and the state of the art optimization and Machine Learning (ML) algorithms

This section benchmarks IBSWO against the state-of-the-art algorithms and existing IDS for IIoT applications.

While other algorithms could be compared to IBSWO, we've chosen tested ones because they offer a solid basis for comparison with our algorithm. Moreover, these algorithms have achieved the best results in recent years, making them the most suitable benchmarks.

Table 4 presents the average detection and classification performance for IBSWO and other state-of-the-art methods

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

on the UNSW-NB15 dataset. The comparison considers four key metrics: accuracy, precision, recall, and F1-score. IBSWO outperforms all other methods in terms of detection and classification performance on this dataset.

Table 5 presents the average classification accuracy achieved by IBSWO and other leading methods on the TON_IoT dataset. This table again utilizes the same four key metrics: accuracy, precision, recall, and F1-score. IBSWO continues its impressive performance on the TON_IoT dataset. It outperforms all other methods in terms of classification accuracy. Table 6 highlights the effectiveness of IBSWO on the new NCTUKM-IIOT dataset. It compares the performance of IBSWO against two other algorithms: the original Binary BSWO and the ensemble CNN IDS model we previously published using the TON-IoT dataset [1]. The table shows IBSWO's superiority across all four key metrics: accuracy, precision, recall, and F1-score. This impressive performance on a new dataset further strengthens the case for IBSWO as a robust IDS solution. IBSWO stands out for its consistently high precision across various IIoT and network datasets (compared to other algorithms).

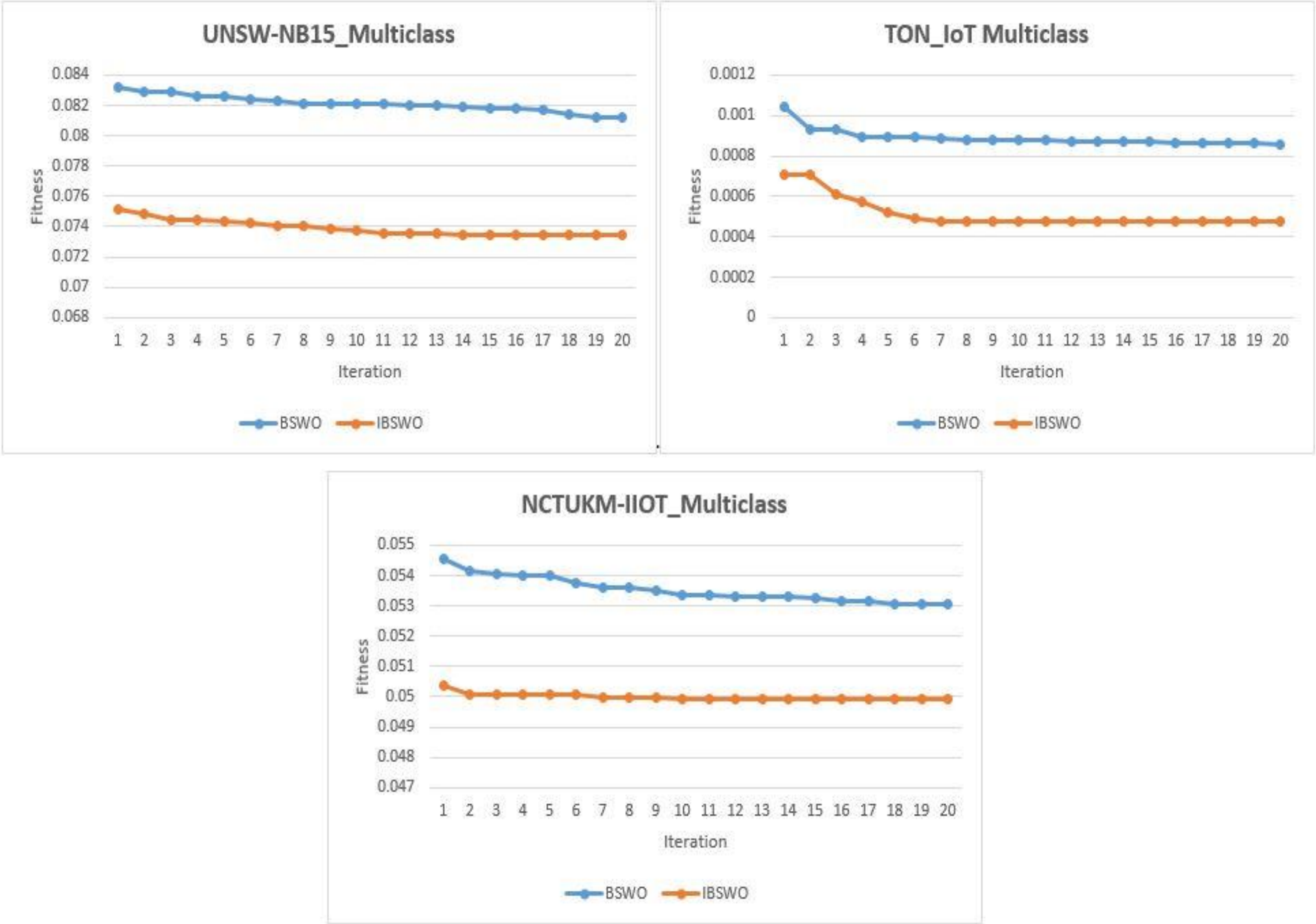

FIGURE 7. The BSWO and IBSWO multiclass convergence curves over the UNSW-NB15, TON_IoT, and NCTUKM-IIOT datasets.

TABLE 4. Performance comparison using UNSW-NB15 dataset

Citation	Year	Optimization technique	Performance			
			Accuracy	Precision	Recall	F1-score
[37]	2021	MOBGA-AOS	0.903	0.888	0.887	0.878
[36]	2022	IHHO	0.941	0.917	0.923	0.911
[35]	2022	SSA_FGWO	0.883	0.861	0.851	0.842
[34]	2021	SCHHO	0.916	0.905	0.867	0.865
[33]	2021	MPPSO	0.926	0.909	0.906	0.917
[32]	2021	ISBPSO	0.842	0.847	0.821	0.826
[31]	2022	RSA	0.979	0.953	0.983	0.939
[30]	2020	WOA_V_ET 53	0.942	0.927	0.929	0.926

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

[29]	2022	BERSOC 34	0.921	0.903	0.904	0.919
[45]	2022	bGWbPS 16	0.973	0.951	0.952	0.931
[46]	2023	BIWSO3	0.984	**0.987**	0.972	0.954
[26]	2023	OCNN-LSTM with GWO	0.97	0.96	0.72	0.74
Proposed	**2024**	**BSWO**	0.978	0.977	0.975	0.976
Proposed	**2024**	**IBSWO**	**0.987**	0.9863	**0.984**	**0.985**

TABLE 5. Performance comparison using TON_IoT dataset

Citation	Year	Optimization technique	Performance			
			Accuracy	Precision	Recall	F1-score
[1]	2023	Ensemble CNN IDS model	0.997	1	1	1
[38]	2021	Chi2-SMOTE with XGBoost	0.982	0.959	0.989	0.974
[28]	2022	DRaNN_PSO	0.996	0.996	0.996	0.996
[26]	2023	OCNN-LSTM with GWO	0.97	0.96	0.74	0.76
[25]	2023	EMSVM-CGWO	---	0.963	0.941	0.952
Proposed	**2024**	**BSWO**	**0.997**	**0.997**	**0.997**	**0.997**
Proposed	**2024**	**IBSWO**	**0.999**	**1**	**1**	**1**

TABLE 6. Performance comparison using NCTUKM-IIOT dataset

Citation	Year	Optimization technique	Performance			
			Accuracy	Precision	Recall	F1-score
[1]		Ensemble CNN IDS model	**0.980**	**0.980**	**0.980**	**0.980**
Proposed	**2024**	**BSWO**	**0.992**	**0.992**	**0.993**	**0.992**
Proposed	**2024**	**IBSWO**	**0.997**	**0.997**	**0.997**	**0.997**

TABLE 7. Parameter settings for the compared algorithms

Algorithm	Parameter setting
BIWSO3	Population size =30, # of iteration = 200. # of runs = 30, Dimension (D) = number of Features, lb = 0, ub = 1, fmin= 0.07, fmax = 0.75, pmin = 0.5, pmax = 1.5, tau = 4.125, a0 = 6.25, a1 = 100, a2 = 0.0005, α= 0.99, β= 0.01, crossover = 0.5 * dimension, 0.3 * dimension, and 0.7 * dimension, unified crossover = 0.5
MOBGA-AOS	# of runs = 30, maxFEs = 300,000, N = 100, Problem dimension (D) = number of Features, M = 2, Q = 5, LP = 5, Pc = 0.9, Pm = 1/D
IHHO	Population size = 30, # of dimensions = 30, Max iteration = 500
SSA-FGWO	Population size = 20, # of iterations = 1000, Coefficient (c1) = [2/e,2], convergence constant = [0,2]
SCHHO	Population size = 10, # of runs = 30, # of iterations = 100 Problem dimension = Feature count, a = 2
MPPSO	# of runs = 20, # of search individual = 20, K for cross validation = 10, k for k-NN = 5, k-NN distance metric = Euclidean, # of iterations = 100, Search dimension = Feature count, a = 0.99, and b = 0.01, C1 = C2 = 2, V max = 10 and 6, W = 1, Wmax = 0.9, Wmin = 0.4, Transfer function S, v Shaped,
ISBPSO	Swarm size = 30, # of generations = 100, Step parameter = 50, Number of runs = 250, c ¼ 0:5, and r ¼ 10, l ¼ 0:25, / ¼ 0:05, Inertia weight = 0.9, thetaMax ¼ 0:05p, thetaMin ¼ 0:0p, LB = 0, UB = 1, CSO(A) = [0,1], CSO(b) = [0,1]
RSA	# of crocodiles = 30, # of iterations = 100, a = 0.9, and b = 0.1, rnd is random number between [0,1]
WOA_V_ET	# of iterations = 50, # of runs = 30, KNN with K = 5
BERSOC	Population size = 30, # of iterations = 200, # of runs = 30, a = 0.99, and b = 0.01train an test samples = 0.8, 0.2
bGWbPS	# of wolves = 12, # of iterations = 20, Initial weight = 0.9, Final weight = 0.4, Weighting factor = 0.5, Uniformly distributed random number = 0-1 , Lower bound = 0, Upper bound = 1, # of runs = 10, 20, 30, 40, 50

IBSWO achieved the highest detection and classification accuracy across all datasets. Its exceptional performance is due to a robust preprocessing stage that thoroughly cleans and filters the input data. By eliminating noise, redundant information, and irrelevant features, the algorithm focuses on the most crucial aspects of the data, which boosts detection and classification accuracy. Furthermore, IBSWO uses advanced techniques for model selection and hyperparameter optimization. By systematically testing different model configurations and choosing the best ones, the algorithm enhances its performance, leading to superior detection and classification accuracy.

IBSWO is more effective than other algorithms at reliably detecting intrusions in a variety of network environments because it consistently achieves higher average precision values across all IDS datasets. The algorithm's capacity to reduce false positives and reliably identify intrusions is demonstrated by this improved precision performance. Its sophisticated search techniques, which carefully comb through the solution space to identify the best intrusion detection patterns and enable accurate network traffic classification, are important contributors to IBSWO's increased precision. Furthermore, to guarantee that only the most pertinent and instructive features are used during detection, IBSWO uses enhanced feature selection procedures. IBSWO achieves greater precision values by concentrating on crucial network characteristics, which enable it to differentiate between malicious and legitimate traffic patterns more accurately.

True Positive Rate (TPR), which indicates the proportion of correctly predicted intrusions, is also shown in the tables. Among the three datasets, IBSWO obtains the highest TPR scores. The improved TPR performance can be attributed to IBSWO's use of an efficient crossover operator. By streamlining the solution space search, part enables the algorithm to find high-quality solutions and improve them over time. By incorporating this powerful crossover operator, dubbed "Flat Crossover" the algorithm becomes more adaptive and has a higher potential for evolution, which in turn leads to the identification of better solutions. The effective application of the flat crossover technique highlights the value of IBSWO as a tool for improving network security by strengthening its predictive power of intrusions.

The F1-score is employed due to the presence of an unbalanced class distribution. The findings reveal that IBSWO achieves the highest positive prediction rate across all datasets. These results consistently highlight IBSWO's superiority over other algorithms in terms of positive prediction rate. Several key factors contribute to this outstanding performance. Firstly, IBSWO incorporates a robust training process that addresses the imbalanced data distribution. Through oversampling, the algorithm effectively handles the class imbalance issue, resulting in a higher positive prediction rate. This approach allows the algorithm to allocate more attention and resources to the minority class, effectively capturing instances of intrusion and reducing false negatives. The algorithm increases its capacity to discern between benign and harmful activity by pinpointing the most relevant and distinctive traits, which eventually results in higher positive prediction rates.

Table 7 presents the parameter settings of the algorithms as used in their original papers. Choosing the right parameter settings is crucial for the experimental design when comparing optimization algorithms. This choice depends on the FS problem for IDS and the characteristics of IBSWO. There are several approaches to consider: using default settings as a baseline, performing systematic parameter tuning with a validation dataset, or selecting fixed parameters based on prior knowledge of the problem or algorithm. Each approach has its advantages depending on the context. Using the parameter settings in Table 7 ensures a fair and consistent comparison and can enhance performance based on prior knowledge. Additionally, these settings are computationally efficient, allowing for more runs. It's important to note that parameter settings significantly impact results; some algorithms are more sensitive to these settings than others. To ensure robust and reliable results, we considered parameter settings and conducted sensitivity analyses.

Based on the comparisons carried out in these experiments, IBSWO demonstrates stability and accuracy across all IDS datasets in relation to fitness values. These values signify the selection of valuable features with the highest accuracy in predicting attacks.

The primary limitation of the SWO is the difficulty in determining the control parameters that maximize its performance [20]. Similarly, while the IBSWO demonstrates promising results, it faces challenges in handling high-dimensional data and identifying optimal solutions in complex, rugged search spaces.

VI. CONCLUSION

This paper presents IBSWO, an enhanced version of the SWO algorithm tailored specifically for Intrusion Detection Systems (IDS) applications in Industrial Internet of Things (IIoT) networks. Through meticulous evaluation using three distinct IIoT datasets, IBSWO's effectiveness in accurately identifying malicious activities within network traffic is demonstrated. The algorithm incorporates key modifications, including a Transfer Function for Binary Conversion and Evolution with Flat Crossover and Genetic Algorithm (GA), which significantly improve its performance in terms of stability, accuracy, and efficiency.

IBSWO consistently outperforms the original BSWO algorithm and a range of leading optimization and Machine Learning algorithms in terms of various performance metrics, including classification accuracy, precision, recall, and F1-score. The superior performance of IBSWO is attributed to its ability to effectively navigate the complex search space of intrusion detection, thanks to the integration of advanced techniques such as flat crossover and robust training processes that address imbalanced data distributions.

While IBSWO demonstrates promising results, challenges remain, particularly in handling high-dimensional data and identifying optimal solutions in rugged search spaces. To address these challenges, future research directions include exploring filter-based feature selection methods to optimize feature sets and enhance classification performance.

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

To further solidify IBSWO's potential for real-world application, future studies will focus on demonstrating its effectiveness across diverse network environments and potential attack scenarios. By conducting broader evaluations beyond the datasets used in this study, IBSWO's robustness and adaptability as a solution for IIoT intrusion detection can be further established.

Acknowledgement

This research was conducted at the Network and Communication Technology (NCT) lab, center for cyber security, Universiti Kebangsaan Malaysia (UKM), and funded by the Ministry of Higher Education under research grant scheme FRGS/1/2020/ICT03/UKM/02/6. Therefore, the research work truly acknowledges to the head of NCT Lab, FTSM, UKM Dr. Mohammad Kamrul Hasan for supervising continuously to the Ph.D. scholar Mousa’b Shtayat.

REFERENCES

[1] M. M. Shtayat, M. K. Hasan, R. Sulaiman, S. Islam, and A. U. R. Khan, “An Explainable Ensemble Deep Learning Approach for Intrusion Detection in Industrial Internet of Things,” *IEEE Access*, vol. 11, pp. 115047–115061, 2023.

[2] M. R. Hassan, P. Podder, T. Debnath, and M. O. Faruk, “An IoT Based Water Quality Testing Device: An Approach to Modelling a Geographical Map Based on Water Quality Data and Decision Support System,” *Int. J.*, vol. 10, no. 2, 2021.

[3] S. Ahmed and M. Khan, “Securing the Internet of Things (IoT): A comprehensive study on the intersection of cybersecurity, privacy, and connectivity in the IoT ecosystem,” *AI, IoT Fourth Ind. Revolut. Rev.*, vol. 13, no. 9, pp. 1–17, 2023.

[4] K. Scarfone and P. Mell, “Guide to intrusion detection and prevention systems (idps),” *NIST Spec. Publ.*, vol. 800, no. 2007, p. 94, 2007.

[5] O. Alkadi, N. Moustafa, and B. Turnbull, “A review of intrusion detection and blockchain applications in the cloud: approaches, challenges and solutions,” *IEEE Access*, vol. 8, pp. 104893–104917, 2020.

[6] S. Zaman and F. Karray, “Features selection for intrusion detection systems based on support vector machines,” in *2009 6th IEEE consumer communications and networking conference*, 2009, pp. 1–8.

[7] R. Guha, K. K. Ghosh, S. K. Bera, R. Sarkar, and S. Mirjalili, “Discrete equilibrium optimizer combined with simulated annealing for feature selection,” *J. Comput. Sci.*, vol. 67, p. 101942, 2023.

[8] X. Tang, Y. Dai, and Y. Xiang, “Feature selection based on feature interactions with application to text categorization,” *Expert Syst. Appl.*, vol. 120, pp. 207–216, 2019.

[9] T. El-Ghazali, “Metaheuristics: from design to implementation.” Wiley New Jersey, 2009.

[10] A. M. Vommi and T. K. Battula, “A hybrid filter-wrapper feature selection using Fuzzy KNN based on Bonferroni mean for medical datasets classification: A COVID-19 case study,” *Expert Syst. Appl.*, vol. 218, p. 119612, 2023.

[11] N. Karlupia and P. Abrol, “Wrapper-based optimized feature selection using nature-inspired algorithms,” *Neural Comput. Appl.*, vol. 35, no. 17, pp. 12675–12689, 2023.

[12] A. I. Hammouri, M. Mafarja, M. A. Al-Betar, M. A. Awadallah, and I. Abu-Doush, “An improved dragonfly algorithm for feature selection,” *Knowledge-based Syst.*, vol. 203, p. 106131, 2020.

[13] D. J. Kalita, V. P. Singh, and V. Kumar, “A novel adaptive optimization framework for SVM hyper-parameters tuning in non-stationary environment: A case study on intrusion detection system,” *Expert Syst. Appl.*, vol. 213, p. 119189, 2023.

[14] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey wolf optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014.

[15] N. F. Johari, A. M. Zain, M. H. Noorfa, and A. Udin, “Firefly algorithm for optimization problem,” *Appl. Mech. Mater.*, vol. 421, pp. 512–517, 2013.

[16] X. Yang and A. Hossein Gandomi, “Bat algorithm: a novel approach for global engineering optimization,” *Eng. Comput.*, vol. 29, no. 5, pp. 464–483, 2012.

[17] L. Abualigah, M. Shehab, M. Alshinwan, and H. Alabool, “Salp swarm algorithm: a comprehensive survey,” *Neural Comput. Appl.*, vol. 32, no. 15, pp. 11195–11215, 2020.

[18] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, “Particle swarm optimization: A comprehensive survey,” *Ieee Access*, vol. 10, pp. 10031–10061, 2022.

[19] F. S. Gharehchopogh and H. Gholizadeh, “A comprehensive survey: Whale Optimization Algorithm and its applications,” *Swarm Evol. Comput.*, vol. 48, pp. 1–24, 2019.

[20] M. Abdel-Basset, R. Mohamed, M. Jameel, and M. Abouhawwash, “Spider wasp optimizer: A novel meta-heuristic optimization algorithm,” *Artif. Intell. Rev.*, vol. 56, no. 10, pp. 11675–11738, 2023.

[21] S. Saber and S. Salem, “High-performance technique for estimating the unknown parameters of photovoltaic cells and modules based on improved spider wasp optimizer,” *SMIJ*, vol. 5, no. 2, p. 14, 2023.

[22] Z. Feng, D. Zhu, H. Guo, G. Sun, and C. Zhou, “A multi-strategy spider wasp optimizer based on grouping and dimensional symmetry method with a time-varying weight,” *Int. J. Mach. Learn. Cybern.*, pp. 1–35, 2024.

[23] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, “Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification,” *Neural Comput. Appl.*, vol. 32, pp. 12201–12220, 2020.

[24] I. Guyon and A. Elisseeff, “An introduction to variable and feature selection,” *J. Mach. Learn. Res.*, vol. 3, no. Mar, pp. 1157–1182, 2003.

[25] K. S. Shaik, N. S. K. Thumboor, S. P. Veluru, N. J. Bommagani, D. Sudarsa, and G. K. Muppagowni, “Enhanced SVM Model with Orthogonal Learning Chaotic Grey Wolf Optimization for Cybersecurity Intrusion Detection in Agriculture 4.0,” *Int. J. Saf. Secur. Eng.*, vol. 13, no. 3, pp. 509–517, 2023, doi: 10.18280/ijsse.130313.

[26] U. K. Lilhore *et al.*, “HIDM: Hybrid Intrusion Detection Model for Industry 4.0 Networks Using an Optimized CNN-LSTM with Transfer Learning,” *Sensors*, vol. 23, no. 18, 2023, doi: 10.3390/s23187856.

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

[27] A. R. Gad, A. A. Nashat, and T. M. Barkat, “Intrusion Detection System Using Machine Learning for Vehicular Ad Hoc Networks Based on ToN-IoT Dataset,” *IEEE Access*, vol. 9, no. October, pp. 142206–142217, 2021, doi: 10.1109/ACCESS.2021.3120626.

[28] J. Ahmad, S. A. Shah, S. Latif, F. Ahmed, Z. Zou, and N. Pitropakis, “DRaNN_PSO: A deep random neural network with particle swarm optimization for intrusion detection in the industrial internet of things,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 10. pp. 8112–8121, 2022. doi: 10.1016/j.jksuci.2022.07.023.

[29] M. A. Awadallah, M. A. Al-Betar, M. S. Braik, A. I. Hammouri, I. A. Doush, and R. A. Zitar, “An enhanced binary Rat Swarm Optimizer based on local-best concepts of PSO and collaborative crossover operators for feature selection,” *Comput. Biol. Med.*, vol. 147, p. 105675, 2022.

[30] M. Mafarja, A. A. Heidari, M. Habib, H. Faris, T. Thaher, and I. Aljarah, “Augmented whale feature selection for IoT attacks: Structure, analysis and applications,” *Futur. Gener. Comput. Syst.*, vol. 112, pp. 18–40, 2020.

[31] A. Dahou *et al.*, “Intrusion detection system for IoT based on deep learning and modified reptile search algorithm,” *Comput. Intell. Neurosci.*, vol. 2022, 2022.

[32] A.-D. Li, B. Xue, and M. Zhang, “Improved binary particle swarm optimization for feature selection with new initialization and search space reduction strategies,” *Appl. Soft Comput.*, vol. 106, p. 107302, 2021.

[33] F. Kılıç, Y. Kaya, and S. Yildirim, “A novel multi population based particle swarm optimization for feature selection,” *Knowledge-Based Syst.*, vol. 219, p. 106894, 2021.

[34] K. Hussain, N. Neggaz, W. Zhu, and E. H. Houssein, “An efficient hybrid sine-cosine Harris hawks optimization for low and high-dimensional feature selection,” *Expert Syst. Appl.*, vol. 176, p. 114778, 2021.

[35] M. Qaraad, S. Amjad, N. K. Hussein, and M. A. Elhosseini, “Large scale salp-based grey wolf optimization for feature selection and global optimization,” *Neural Comput. Appl.*, vol. 34, no. 11, pp. 8989–9014, 2022.

[36] A. G. Hussien and M. Amin, “A self-adaptive Harris Hawks optimization algorithm with opposition-based learning and chaotic local search strategy for global optimization and feature selection,” *Int. J. Mach. Learn. Cybern.*, vol. 13, no. 2, pp. 309–336, 2022.

[37] Y. Xue, H. Zhu, J. Liang, and A. Słowik, “Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification,” *Knowledge-Based Syst.*, vol. 227, p. 107218, 2021.

[38] A. R. Gad, A. A. Nashat, and T. M. Barkat, “Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset,” *IEEE Access*, vol. 9, pp. 142206–142217, 2021.

[39] A. H. Azizan *et al.*, “A machine learning approach for improving the performance of network intrusion detection systems,” *Ann. Emerg. Technol. Comput.*, vol. 5, no. 5, pp. 201–208, 2021.

[40] C. K. Starr, “Nesting biology and sex ratio in a Neotropical spider wasp, Priochilus captivum (Hymenoptera: Pompilidae),” *Trop. Zool.*, vol. 25, no. 2, pp. 62–66, 2012.

[41] M. Benamú, L. F. García, C. Viera, M. Lacava, and S. Korenko, “Koinobint life style of the spider wasp Minagenia (Hymenoptera, Pompilidae) and its consequences for host selection and sex allocation,” *Zoology*, vol. 140, p. 125797, 2020.

[42] B. H. Abed-Alguni, N. A. Alawad, M. A. Al-Betar, and D. Paul, “Opposition-based sine cosine optimizer utilizing refraction learning and variable neighborhood search for feature selection,” *Appl. Intell.*, vol. 53, no. 11, pp. 13224–13260, 2023.

[43] N. Moustafa and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in *2015 military communications and information systems conference (MilCIS)*, 2015, pp. 1–6.

[44] A. Alsaedi, N. Moustafa, Z. Tari, A. Mahmood, and A. Anwar, “TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems,” *Ieee Access*, vol. 8, pp. 165130–165150, 2020.

[45] Q. M. Alzubi, M. Anbar, Y. Sanjalawe, M. A. Al-Betar, and R. Abdullah, “Intrusion detection system based on hybridizing a modified binary grey wolf optimization and particle swarm optimization,” *Expert Syst. Appl.*, vol. 204, p. 117597, 2022.

[46] N. A. Alawad, B. H. Abed-alguni, M. A. Al-Betar, and A. Jaradat, “Binary improved white shark algorithm for intrusion detection systems,” *Neural Comput. Appl.*, vol. 35, no. 26, pp. 19427–19451, 2023.

MOUSA’B MOHAMMAD SHTAYAT doing PhD in UKM University- Malaysia in CYBERSECURITY. He completed his Master’s degree in Information Technology Management (ITM) from the University of Sunderland, the UK in July 2009. He also attained the Bachelor’s Degree in Electrical and Computer Engineering from Hashemite University, Jordan in June 2003.

He is working in Royal Commission for Jubail and Yanbu as a lecturer at Yanbu Industrial Collage) in Computer Science and Engineering Department since September 2012. He is teaching many courses like Computer Networks, Network Security, Digital Logic, and Electrical Circuits. In addition, He worked as a course development member for network courses and digital logic course materials and labs. He also a member of different committees such as Quality Management System, ISO Auditing as an Internal Auditor, and COOP Coordinator as a member of the Academic Support Committee.

MOHAMMAD KAMRUL HASAN (M'13–SM'17) is currently working as an Associate Professor and Head of Network and Communication Technology research lab at the Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM). He completed a Doctor of Philosophy (Ph.D.) degree in Electrical and Communication Engineering from the Faculty of Engineering, International Islamic University, Malaysia, in 2016. He is specialized in elements pertaining to cutting-edge information centric networks; Computer networks, Data

This article has been accepted for publication in IEEE Open Journal of the Communications Society. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/OJCOMS.2024.3421647

communication and security, Mobile Network and Privacy Protection, Cyber-physical systems, Industrial IoT, Transparent AI, and Electric Vehicles Networks. He has published more than 150 indexed papers in ranked journals and conference proceedings. Dr. Kamrul is a Senior Member of the Institute of Electrical and Electronics Engineers, a Member of Institution of Engineering and Technology, and a member of Internet Society. Dr. Kamrul is a certified professional technologist (P. Tech /Ts.), Malaysia. He also served the IEEE student branch as chair from 2014 to 2016. He has actively participated in many events/workshops/trainings for the IEEE and IEEE Humanity programs in Malaysia. He works as an editorial member in many prestigious high-impact journals Such as IEEE, IET, Elsevier, Frontier, and MDPI, and as general chair, co-chair, and speaker for conferences and workshops for the shake of society and academy knowledge building and sharing and learning. He has been contributing and working as a volunteer for underprivileged people for the welfare of society.

ROSSILAWATI SULAIMAN received a B.Sc. degree in computer science from Universiti Kebangsaan Malaysia in 2000, the M.Sc. degree in computer science from the University of Essex, U.K., in 2003, and the Ph.D. degree from the University of Canberra, in 2011. She is currently a Senior Lecturer at Universiti Kebangsaan Malaysia. Her work has been published in Journal Theoretical and Applied Information Technology, International Journal of Advanced Computer Science and Applications, and Journal of Computer Science. Her research interests include steganography and applied cryptography.

SHAYLA ISLAM (Member, IEEE) received the B.Sc. degree in computer science and engineering from International Islamic University Chittagong, Bangladesh, the M.Sc. degree from the Department of Electrical and Computer Engineering, International Islamic University Malaysia (IIUM), in 2012, and the Ph.D. degree in engineering from the Electrical and Computer Engineering (ECE) Department, IIUM, in 2016, under Malaysian International Scholarship (MIS). She is currently an Assistant Professor with UCSI University, Malaysia. She was awarded a Silver Medal for her research work with International Islamic University Malaysia. In consequence, she was also awarded a Young Scientist Award for the contribution of a research paper at the second International Conference on Green Computing and Engineering Technologies, 2016 (ICGCET'16), organized by the Department of Energy Technology, Aalborg University, Esbjerg, Denmark.

BISHWAJEET PANDEY is a Professor at Department of Intelligent System and Cyber Security, Astana IT University Kazakhstan. He is also a visiting professor at Eurasian National University, Astana, Kazakhstan (QS World Rank 355) and UCSI University, Kuala Lumpur, Malaysia (QS World Rank 300). He was the Research Head of the School of Computer Science and Engineering at Jain University (NIRF India Rank 68), Bangalore, India. He holds a Ph.D. in Computer Science and Engineering from Gran Sasso Science Institute, Italy. He completed master's from the Indian Institute of Information Technology (IIIT), Gwalior, India. With over 15 years of experience, he has worked as a Research Consultant and Associate Visiting Professor. He has also held positions as an Assistant Professor at various universities e.g. Chitkara University, Chandigarh, India (QS World Rank 1200-1400). Dr. Pandey has visited 47 countries and attended 98 conferences, receiving best paper awards in multiple countries. He has authored over 170 papers and published five books. With interests in Green Computing, High-Performance Computing, Cyber-Physical Systems, Machine Learning, and Cyber Security, he has several Indian patents and serves on the board of directors for student-founded startups. .

HUDA SALEH ABBAS has a Ph.D. degree in Network Engineering from RMIT University, Melbourne, Australia in 2017. She is currently an Assistant Professor in Computer science and Engineering collage, Taibah University. Her research interests include Computer Networks, Wireless Networks, Optical Networks, Edge Computing.

MAMOON MOHAMMED ALI SAEED is Deputy Dean of the College of Engineering and Information Technology, Director of the University Branch, and a Lecturer at the Department of Communication and Electronics Engineering, UMS University, Yemen. received his Bachelor's degree in Communication and Electronics Engineering from Sana'a University, Yemen 2005, and his M.S. degree from the Department of Computer Networks and Information Security in Yemen Academy for Graduate Studies Yemen 2013. Recently, a Ph.D. in Alzaiem Alazhari University, Faculty of Engineering, Electrical Engineering Department, Khartoum, Sudan 2021. His research areas include cyber security, communication security, and artificial intelligence.