HBF-PSO and HNA-NN Based Intrusion Detection System for SCADA Networks

Hui Wang
Hebei Chemical & Pharmaceutical College, Hebei 050026, China
E-mail: bingyuexiaxing@163.com

Keywords: data acquisition, supervisory control, particle swarm optimization, hummingbird flight, intrusion detection

Received: October 24, 2025

The increasing adoption of remote-controlled, self-contained production machines has led to the integration of Supervisory Control and Data Acquisition (SCADA) systems as a key component of industrial automation. While machine connectivity has improved productivity, the threat of cybersecurity attacks has introduced weaknesses into control systems. This article proposes the development of an intrusion detection system (IDS) that optimizes search efficiency and diversity in the search population by implementing the Hummingbird Flight-based Particle Swarm Optimization (HBF-PSO) algorithm combined with the Hierarchical Neuron Architecture Neural Network (HNA-NN). The HBF strategy models incremental, energy-efficient flight patterns to improve feature optimization, while the HNA-NN classifier categorizes attack attempts with high precision. Experiments conducted on actual SCADA system databases (MORD, MIRD, SORD, and SIRD) have confirmed the efficiency of the proposed system, with 98.12% detection accuracy and 100% precision in the SORD database. The false-positive rate of the proposed system was 0% in both the MORD and SIRD databases. In general, the hybrid model has shown improved detection accuracy and specificity compared to traditional systems.

Povzetek: Članek predstavi hibridni IDS za SCADA sisteme, ki z algoritmom HBF-PSO in nevronsko arhitekturo HNA-NN dosega izjemno natančno zaznavanje napadov (do 98,12 % in 0 % lažnih alarmov), kar pomembno izboljša kibernetsko varnost industrijskih naprav.

1 Introduction

Supervisory Control and Data Acquisition (SCADA) has made the monitoring and control of factories easier, particularly in water treatment plants [1], electric utilities [2], and industrial sites [3]. SCADA enables remote control of critical infrastructure by connecting field components, such as sensors and actuators [4], even if the SCADA system was initially applied in a standalone setup that was less susceptible to external risks to their own networks on the Internet [5, 6]. As systems play essential roles in business continuity, threats and disruptions could pose serious financial risks [7]. In light of the above discussion, appropriate cybersecurity measures must be put in place to secure SCADA systems against threats such as data breaches, sabotage, and other attacks [8].

Intrusion detection in SCADA systems poses a unique challenge, as they are responsible for managing infrastructure [9]. Traditional IDS typically produces a high rate of false positives and false negatives when regular system activity is mislabeled as an attack, or vice versa, leaving actual threats undetected. SCADA traffic inaccuracies are complex or multifaceted [10]. Thus, it differs from typical IT environments. In addition, many traditional IDSs cannot detect encrypted traffic, which is increasingly available in modern SCADA systems. Furthermore, most of these focus on external attacks and fail to detect internal anomalies or misconfigurations that could compromise security [11]. Besides these limitations, other factors, such as real-time response and minimum disturbance to critical services, demand more advanced and customized IDS solutions for SCADA environments . Figure 1 illustrates the IDS process in SCADA.

Optimization algorithms significantly improve the efficiency of IDS techniques for detecting harmful activities [12]. Particle Swarm Optimization (PSO) algorithms are one of the well-known optimization algorithms. PSO algorithms have been developed based on the social behavior of migrating birds in flocks or shoals of fish. PSO algorithms allow particles, or possible solutions to the problem, to search the space by modifying their positions based on their experiences and those of their neighbors [13]. Such particle-to-particle cooperation makes the PSO algorithm efficient at finding near-optimal solutions. This approach can be applied to the IDS to optimize feature selection by reducing computational complexity and improving detection capability by incorporating a system for distinguishing between normal and malicious traffic . Its convergent characteristics in terms of adaptability make PSO a tool that enables higher real-time intrusion detection with minimal false positives, accounting for the dynamic properties inherent in SCADA systems.

This paper proposes an efficient PSO algorithm with Hummingbird Flight (HBP) patterns to enhance IDS accuracy and reliability within SCADA networks.

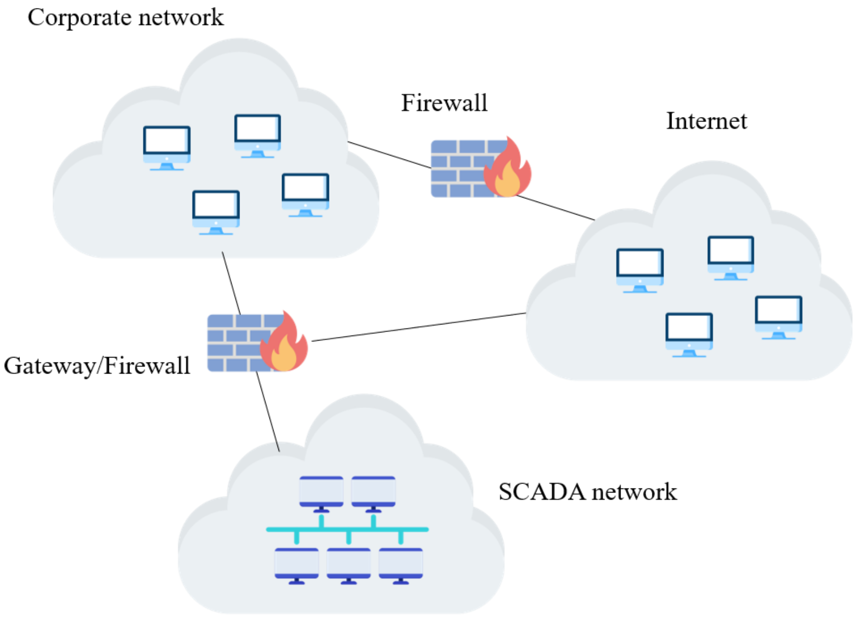

Figure 1: Intrusion detection process in SCADA

Computational optimization using traditional PSO algorithms may be marred by issues such as poor population diversity and premature convergence, thereby weakening the effectiveness of the proposed approach in SCADA systems for detecting complex threats. In this regard, the enhanced PSO algorithm combines all unique flight behaviors of hummingbirds that include incremental position updates, gradual energy-efficient movements, and avoidance of large flocks, thus providing a better balance between exploration of the search space to avoid local optima, feature selection, and classification, leading to higher intrusion detection for both internal and external. Similarly, the suggested PSO-HBF algorithm overcomes the drawbacks of traditional IDSs and provides a more robust, adaptive approach to security for critical SCADA networks. The research is guided by the following questions:

RQ1: How can biologically inspired optimization, specifically the HBF-PSO model, improve feature selection and convergence stability in high-dimensional SCADA intrusion datasets compared to traditional metaheuristics?

RQ2: To what extent can the hierarchical neural architecture increase detection accuracy and reduce false-positive rates under varying environmental and network conditions?

RQ3: Does the proposed hybrid framework demonstrate statistically significant improvements over existing SCADA IDS solutions with respect to detection rate, precision, and robustness?

2 Literature review

Some key limitations of IDSs in SCADA networks include reliance on predefined signatures and static rules. These classical methods normally incur too many false negatives and/or positives, making threat recognition difficult. Besides, traditional IDSs rarely detect encrypted data or internal threats, as they focus on external threats. These limitations are avoided by the different machine learning and optimization techniques proposed in the research studies presented in Table 1.

Teixeira, et al. [14] proposed a freely available labeled dataset to facilitate flow-driven intrusion detection studies on SCADA platforms. Cyberattacks were carried out against industrial control systems to produce this flow of information. Also, a flow-oriented IDS for SCADA systems is created with a deep learning approach. The dataset is employed to construct an IDS system for continuous SCADA network operation to identify assaults immediately upon their occurrence.

Saheed, et al. [15] developed a hybrid ensemble model to detect hostile intrusions in SCADA systems with industrial sensor networks. The model employed datasets for the gas pipeline network, the water network, and the University of New South Wales-NB 2015 dataset. The techniques employed were the Unity Normalization technique, Grey Wolf Optimizer (GWO), Principal Component Analysis (PCA), and the bijective soft set.

Yu, et al. [16] described the security imperative in SCADA systems, which include critical infrastructure such as chemical plants, oil pipelines, and intelligent power grids. They presented a comprehensive approach to determining security requirements that involves specifying policies and models, modifying the architecture, and implementing appropriate security measures. They presented a model that emphasized communication security, threat defense, integrity, scalability, cooperation, self-healing processes, and intrusion tolerance.

Table 1: Related work on intrusion detection for SCADA systems

Reference	Objective	Techniques used	Dataset	Performance metrics
[14]	Developing a flow-based IDS for real-time detection in SCADA systems	Deep learning-based IDS	SCADA flow-based dataset	Accuracy ≈ 97.3%; Detection rate ≈ 96.8%
[15]	Intrusion detection for SCADA and CPS systems	Machine learning models	Custom dataset for SCADA	Detection 99.9%; Recall 100%; Precision 100%
[16]	Security enhancement of SCADA networks	Hybrid machine learning techniques	Public SCADA dataset	False-alarm ≈ 1.2%; Detection ≈ 98.4%
[17]	Enhancing cybersecurity in SCADA systems using ML	Machine learning-based IDS	SCADA network traffic	Accuracy 98.7%; Precision 97.5%
[18]	Detecting anomalies in SCADA networks	Deep learning techniques	SCADA dataset	Detection ≈ 97.1%; False-positive ≈ 1.6%
[19]	Developing an efficient IDS for SCADA	Machine learning algorithms	KDD Cup 99	F1-score 0.97; Recall 96.8%; Precision 97.4%

Wang, et al. [17] proposed an end-to-end intrusion detection solution to counter threats in industrial control systems. They employed an autoencoder with a Bayesian-Gaussian mixture model for spatial clustering with noise and lossy deep support vector models. The model outperformed conventional approaches for handling contaminated, high-dimensional data, thereby overcoming efficiency degradation.

Qin, et al. [18] proposed a bio-inspired adaptive defensive architecture that leverages layered, adaptive, and flexible protection techniques. The suggested approach integrates optimum network shuffles with deception strategies to prolong the attacker's engagement with the decoys. Moreover, the use of twin heterogeneous subnets provides additional security against system failures and can be regenerated after breaches. The effectiveness of the proposed defensive framework is analyzed in a traditional manufacturing environment based on the SDN architecture, and the framework has been tested under different circumstances. Compared with previous studies, the simulation greatly improves defense efficiency in the adaptive defense mode.

Cyber-Physical Systems (CPS) are at risk of cyberattacks, and traditional intrusion detection systems may not detect alterations caused by them or by external faults. To address this, Pandey and Das [19] used a statistical technique to predict the actions of the control systems in a CPS using an AI-based classifier. They then compared the actuator's actual states with its predicted states to identify anomalies. They used an innovative deep neural network-based classification methodology, achieving superior performance in experimental validation with an F1-score of 0.97.

As summarized in Table 1, recent IDS frameworks have achieved strong detection performance, yet several critical gaps remain. Most approaches rely on fixed feature vectors or signatures, making them less effective at detecting both internal attacks and encrypted traffic, which is increasingly prevalent in current SCADA and industrial IoT communication. Some deep learning approaches demonstrated strong detection capabilities but are computationally intensive and lack real-time adaptability. In addition, optimization techniques are challenged by the risk of convergence and stagnation in exploration. This optimization challenge affects the ability to accommodate the real-time context.

3 Enhanced PSO algorithm

This section will introduce the enhanced PSO algorithm based on HBF patterns.

3.1 An overview of PSO

The PSO algorithm is a popular optimization technique modeled after flocks of birds and schooling fish. In PSO, a set of particles (potential solutions) explore the search field to determine the optimal solution [20]. Each particle orients its trajectory towards its ideal and optimum position for the entire swarm. Each particle i has a position $xi(t)$ and velocity $vi(t)$ in the search space at iteration t. The position $xi(t)$ signifies a possible solution for the optimization issue. The velocity of an individual particle is modified according to its initial velocity, the distance from its personal best position p_{best_i}, and the distance from the global best position g_{best}. The update for the velocity can be expressed as follows:

$$v_i(t+1) = \omega . v_i(t) + c_1 . r_1 . \left(p_{best_i} - x_i(t)\right) + c_2 . r_2 . (g_{best} - x_i(t)) \quad (1)$$

Where c_1 and c_2 are acceleration coefficients (cognitive and social factors) that determine how much the particle is influenced by its own best-known position and the global best position, respectively. ω refers to the inertia weight that controls the influence of the previous velocity and $v_i(t)$ stands for the velocity of particle i at iteration t. r_1 and r_2 are random values between 0 and 1, used to introduce randomness.

After updating the velocity, the particle's new position is calculated using Eq. 2 by adding the updated velocity to its current position.

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

Each particle remembers its best position p_{best_i}(the position where it achieved its best objective value) and the best position determined by the whole swarm g_{best}. p_{best_i} is updated if the current position $xi(t)$ has a better fitness value than the previous best. g_{best} is the best position among all particles in the swarm, and it is updated whenever any particle finds a better position. The PSO algorithm iterates until it reaches a certain threshold, for example, a limit of iterations or when the improvement in the best-known solution becomes insignificant.

3.2 Hummingbird flight patterns

The HBF strategy improves the original PSO algorithm by incorporating the hummingbird's flight behavior. Hummingbirds are recognized for their accurate and efficient flight. This strategy has been proposed to overcome the shortcomings of the PSO algorithm, namely the loss of diversity and the tendency to converge to local optima. The hummingbird's flight behavior enables the HBF strategy to balance exploration and exploitation.

Eq. 3 updates the position of the particle based on the personal best solution. This represents the best solution obtained so far. Eq. 3 uses the step size and the adjustment coefficient 1 to update the particle's current solution based on the best solution obtained. This enables the particle to move towards the best solution obtained. This process enables the particle to scan the neighborhood of the personal best solution.

$$\vec{X}_{i,t+,hbf} = Pbest_{i,t}^{i,t} + step\_size \times adjustment\_coefficient1 \quad (3)$$

Eq. 4 updates the position of the particle based on the global best solution, describing the best solution found so far in the entire swarm. The adjustment coefficient 2 in the equation prevents the swarm from converging to a local solution by ensuring the swarm follows the global best solution. This enables the swarm to concentrate on the promising regions of the search space.

$$\vec{Y}_{i,t+,hbf} = Gbest_{i,t}^{i,t} + step\_size \times adjustment\_coefficient2 \quad (4)$$

In Eq. 5, the traditional PSO update rule uses the particle's current velocity and its previous position. This update rule incorporates the effect of the particle's personal best and the global best solution. Combining the HBF update rule with the traditional PSO update rule ensures a smooth transition to the best solution during the search process.

$$\vec{Z}_{i,t+,hbf} = \vec{X}_{i,t+,PSO} \quad (5)$$

Eq. 6 governs the magnitude of the particle's movement at each iteration. The step size decreases as time t progresses towards the total iteration count T, ensuring more precise movements over time. The factor ϑ introduces variability, and the parameter β controls the step size scaling. The sign of ϑ determines the direction of the movement, allowing the particle to adjust its trajectory dynamically.

$$step_{size} = \left(\frac{t}{\mathrm{T}}\right) \times 1e^{-5} \times \left(\frac{1}{abs(\vartheta)}\right)^{1/\beta} \times sign(\vartheta) \quad (6)$$

Eq. 7 controls how much influence the particle's personal best position has on its movement. The coefficient grows exponentially as the optimization process advances (i.e., as time t increases). This ensures that the particle gradually moves towards its personal best, promoting local search towards the end of the optimization process when finer adjustments are needed.

$$adjustment\_coefficient1 = e^{(t/10)} \quad (7)$$

Eq. 8 ensures that the particle's movement towards the global best becomes more controlled as the optimization progresses. The term (*1− T/t*) reduces the adjustment size over time, meaning larger movements are made earlier in the process, allowing for more extensive exploration. As the algorithm approaches convergence, movements are smaller and more focused on exploitation.

$$adjustment\_coefficient2 = \left(1 - \frac{t}{\mathrm{T}}\right) \times (X_{max} - X_{min}) \quad (8)$$

Eq. 9 models the energy conservation behavior of hummingbirds. As the optimization progresses, the amplitude of the particle's movements is gradually reduced, simulating how hummingbirds conserve energy during flight. Early in the process, particles make larger, more exploratory moves, but as the optimization nears completion, movements become smaller and more refined, conserving energy and focusing on exploitation.

$$h = e^{\left(2 - 2\frac{t}{\mathrm{T}}\right)} \quad (9)$$

The HBF approach incorporates these formulas into the PSO algorithm to promote the swarm's more thorough exploration of the solution search space. The gradual change in particle positions in the search space, relative to the personal best and the global best solutions obtained, together with the energy-saving path of the particles, helps promote a well-balanced optimization technique. In turn, the swarm cannot remain trapped in a local optimum but continues to search for improved solutions.

3.3 Feature optimization

The goal of the feature selection process using the improved PSO algorithm is to enhance the accuracy and efficiency of the intrusion detection system by selecting the most relevant features from the dataset. As illustrated in Figure 2 below, the sequence involves representing the dataset so that the particles in the swarm correspond to possible solutions to the problem. Each particle contains a set of features. The initial position and velocities of the particles in the swarm are randomly selected. For the swarm's particles, the corresponding features would be evaluated to assess their efficiency at detecting intrusions. This would involve executing the IDS on the selected set of features determined by the swarm particle. A fitness value would be determined.

The fitness for every particle is determined based on the accuracy of the selected sets of features. A particle that has a higher accuracy for a smaller set of features will obtain a higher fitness value. All particles update their locations based on the personal best position (best sets of features obtained by a particle) and the global best position (best sets of features obtained by a swarm of particles). These particle locations are determined by the PSO update formula for the particle's position.

After a number of iterations, the particles tend to select the optimal subset of features. The improved PSO guarantees a more thorough exploration of the feature space. This helps prevent the algorithm from prematurely converging on non-optimal feature subsets. After several iterations, the particle with the highest fitness value is

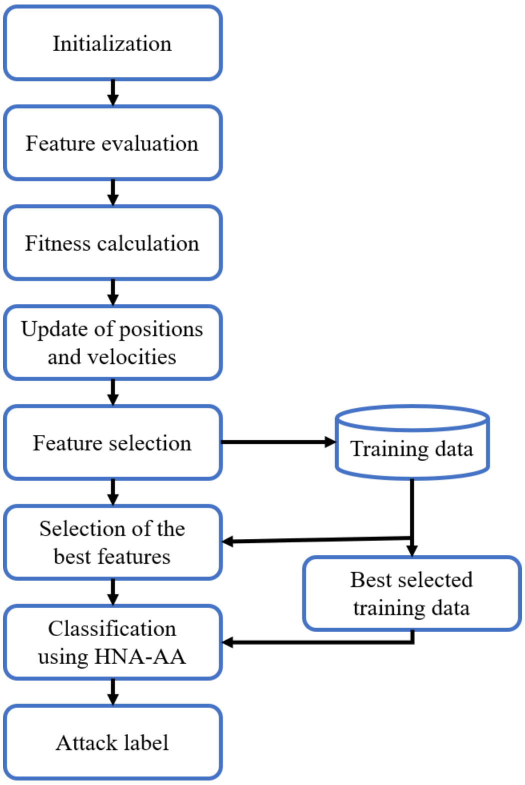
Initialization
Feature evaluation
Fitness calculation
Update of positions and velocities
Feature selection
Training data
Selection of the best features
Best selected training data
Classification using HNA-AA
Attack label

Figure 2: Feature selection process

chosen. Its subset of the corresponding features is determined to be the most relevant for intrusion detection. The selected subset of features helps boost the performance of the IDS by reducing the problem's dimension to the most relevant attributes for intrusion detection.

3.4 Classification

With its optimized features, the Hierarchical Neuron Architecture-based Neural Network (HNA-NN) classifies network traffic into attack or non-attack categories. The input to the neural network in the classification process includes the test features Sv, the selected training set *Str*, and the corresponding labels *Lb*.

The initial step of the HNA-NN algorithm involves initializing the main parameters. This consists of initializing the weight value to zero, the directionality factor *D*, the iteration number *θ*, and the chosen test feature. Some random values, such as *Wxy* and *Wxh*, will be determined based on the number of training samples corresponding to the test features. These values are critical factors in determining the connections between neurons in the neural network.

The architecture of the HNA-NN model comprises three layers: input, hidden, and output. The input layers (net1) accept the optimized feature list derived from the HBF-PSO method. The number of neurons in the two hidden layers of the net1 neural network is 32 and 16 neurons, respectively. All neurons in both layers use the ReLU function. The output layers (net2) apply the Softmax function to obtain classification probabilities for identifying the attack type. The model was trained to complete 100 epochs. The Adam optimizer was used during model training. The learning rate was set to 0.001. The mini-batch size was 64.

After initializing the layers, the exponential activations for the input/output layers are computed. These are denoted by the symbols *H1* for the input layer's exponential activation and *Y* for the output layer's activation. The computation of the exponential values determines the temporary distance "si" to the target classification output. An important step in the process, since the temporary distance *si* assists in the adjustment of the model parameters.

After calculating the temporary distance, the weights will be updated based on the iteration number, the temporary distance, and the current weights. This weight update process helps the model achieve greater classification accuracy. After the weights have been updated, directionality *D* will be determined. Directionality determines the change in the gradient for every feature. The coefficients of both *θ* and directionality *D* change in terms of the difference between the current values of the features and the previous values.

The last step of the HNA-NN algorithm includes calculating the correlation between the sets of characteristics. Correlation values obtained in the process will be compared to the directionality value of *D*. A higher value of the obtained correlation compared to the directionality value of *D* helps to determine the corresponding *Lb(i)* as a classification of the attack or non-attack class.

4 Results

In this section, the recommended and existing Intrusion detection techniques are compared based on detection and false alarm rates. Dataset generation for this study was done by implementing a network configuration comprising 100 hosts on the NS-3 simulator. Intrusion detection was performed in MATLAB, yielding two scenarios: one with attacks and one without. As identified in Table 2, the datasets used are Multi-hop Outdoor Real Data (MORD), Multi-hop Indoor Real Data (MIRD), Single-hop Outdoor Real Data (SORD), and Single-hop Indoor Real Data (SIRD).

The samples used in this study were originally created from raw sensor data provided by the Intel Berkeley Research Laboratory [1]. Each dataset provides temperature and humidity values from both indoor and outdoor SCADA systems, including data affected by Denial-of-Service (DoS) attacks and spoofing. The attack type mimics real threats to SCADA systems that can degrade communication performance and misrepresent sensor attribute values, rather than stopping the entire system process. Advanced Persistent Threat (APT) attack

[1] https://db.csail.mit.edu/labdata/labdata.html

scenarios are excluded from the dataset development and will be investigated in the extended version of the model.

Table 2: Overview of dataset parameters

Parameter	Dataset			
	MORD	MIRD	SORD	SIRD
Temperature (with attack)	26-46	25-52	27-36	24-56
Temperature (without attack)	25-30	25-27	22-35	25-29
Humidity (with attack)	58-91	50-92	51-87	47-91
Humidity (without attack)	43-72	43-51	34-60	40-48

The critical variables of interest in assessing essential states of the simulated outbreak model in communities include temperature and humidity. These critical variables have been identified to relate to different states in a two-dimensional space. These states have been evaluated based on whether attacks occurred. For example, in the SIRD dataset, humidity levels range from 40 to 48 without attacks. During attacks, humidity levels range from 47 to 91.

This study aimed to detect intrusions in SCADA networks using Ethernet modules interconnected with a public network. The proposed PSO algorithm was designed to detect external attacks, such as DoS and spoofing, which are challenging to detect because they do not fully halt network services but instead degrade SCADA system performance. These attacks often require prior knowledge of the target system, making detection difficult due to legitimate protocol messages and unknown process parameters.

Two performance metrics assess the effectiveness of the suggested methodology: the false-positive rate and the detection rate. The detection rate, defined in Eq. 10, is the proportion of correctly classified conditions relative to the total number of critical states recorded. On the other hand, the false positive rate, as defined in Eq. 11, is the ratio of normal states misclassified as critical to the total number of normal states.

$$DR = \frac{TP}{TP - FN} \tag{10}$$

$$FPR = \frac{FP}{FP - TN} \tag{11}$$

Table 3 compares detection rates for the conventional Efficient Data-Driven Clustering (EDDC) and the proposed PSO algorithm. From the above analysis, the proposed scheme clearly demonstrates its efficiency in achieving the best results across all datasets. Precision analysis based on Eq. 12 clearly proves the efficiency of the proposed scheme as compared to the previous method. As shown in Table 4 above, the proposed scheme outperforms the previous method in all the databases.

$$Precision = \frac{No.of\ critical\ states}{Total\ number\ of\ states} \tag{12}$$

Although the developed PSO algorithm greatly optimizes feature subset selection, the combination of PSO and SVM yields suboptimal results. SVM involves selecting many irrelevant features. To address this problem, the PSO algorithm was developed to further enhance feature subset selection. The combination of HNA-NN and PSO further improved the classification accuracy.

Table 3: Detection and false positive rates

Dataset	False positive rate (%)		Detection rate (%)	
	PSO	EDDC	PSO	EDDC
MORD	0	0.05	95.05	92.91
MIRD	0.17	0.33	100	100
SORD	1.02	1.98	98.12	96.77
SIRD	0	0	100	100

Table 4: Precision comparison

Dataset	Precision (%)	
	PSO	EDDC
MORD	96	93
MIRD	100	100
SORD	98	97
SIRD	100	100

The proposed method's performance was also evaluated using specificity and sensitivity indicators. As stated in Eq. 13, sensitivity is the percentage of true positives successfully detected by the classifier. As described in Eq. 14, specificity measures the percentage of true negatives correctly classified. Figures 3 and 4 show that the proposed PSO-HNA-NN method achieves higher sensitivity and specificity than other techniques.

$$Sensitivity = \frac{TP}{(TP + FN)} = \frac{No.of\ true\ positive\ assessments}{No.of\ all\ positive\ assessmens} \tag{13}$$

$$Specificity = \frac{TN}{(TN + FP)} = \frac{No.of\ true\ negative\ assessments}{No.of\ all\ negative\ assessmens} \tag{14}$$

The Jaccard coefficient is a similarity measure used to evaluate the degree of overlap between two datasets. It is expressed as the proportion of the intersection size of two sets to the size of their union. Mathematically, it is described as:

$$Jaccard = \frac{Size\ of\ intersection}{Size\ of\ union} = \frac{|X \cap Y|}{|X| + |Y| - |X \cap Y|} \tag{15}$$

Where X and Y represent the two sets being compared.

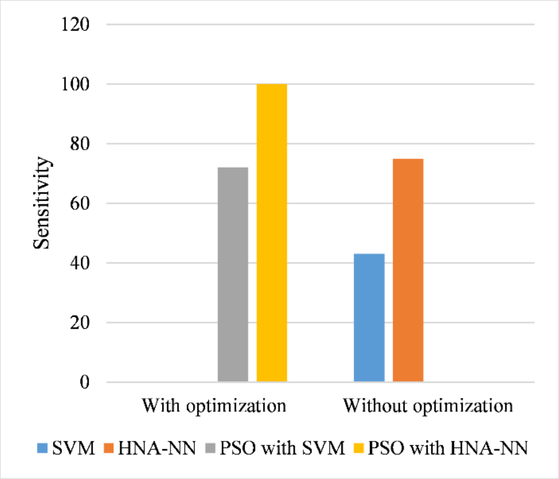

Figure 3: Sensitivity comparison

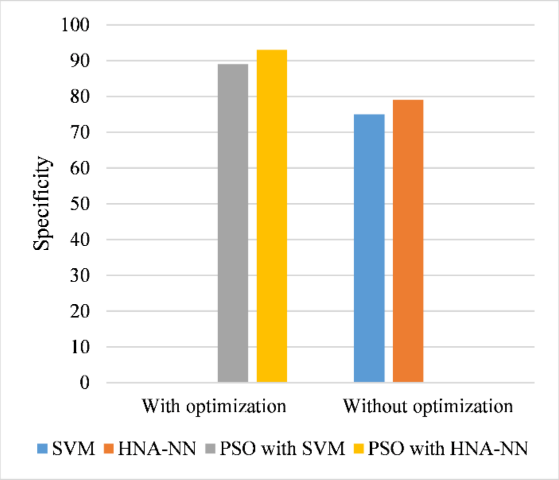

Figure 4: Specificity comparison

As shown in Figure 5 above, the Jaccard index indicates that the PSO-optimized HNA-NN performs better in similarity search than the non-optimized technique and even surpasses the traditional SVM optimization method. The Dice coefficient is a formula for calculating the degree of similarity between two sets. It was initially used to calculate the degree of overlap between two data sets. The formula for the Dice coefficient is given by:

$$Dice = \frac{2 \times Size\ of\ intersection}{Combined\ size\ of\ the\ sets} = \frac{2.|X \cap Y|}{|X| + |Y|} \qquad (16)$$

Figure 6 shows the Dice coefficients for classification methods under both optimized and unoptimized conditions. Clearly, the optimized PSO-boosted HNA-NN performs the best by having the highest Dice coefficient value compared to the others. Figures 7-9 show the precision, recall, and accuracy of different classification methods with or without optimization. These measures verify the efficiency of the PSO-optimized HNA-NN compared to the SVM method and to an HNA-NN without optimization.

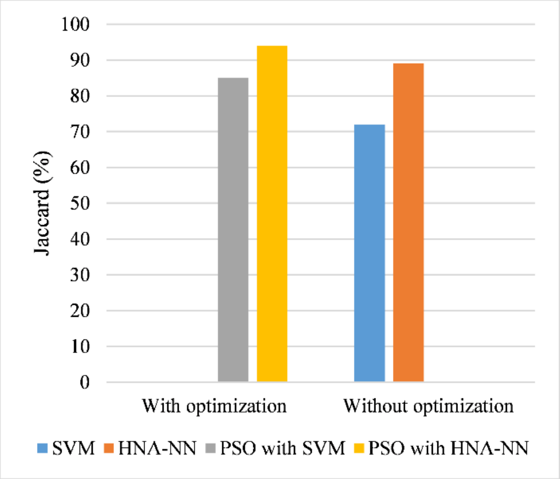

Figure 5: Jaccard comparison

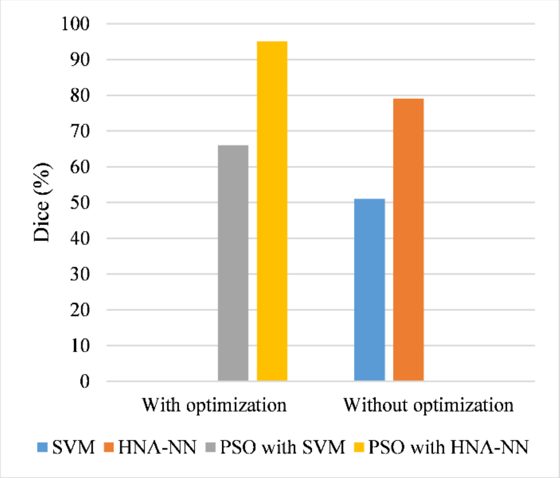

Figure 6: Dice comparison

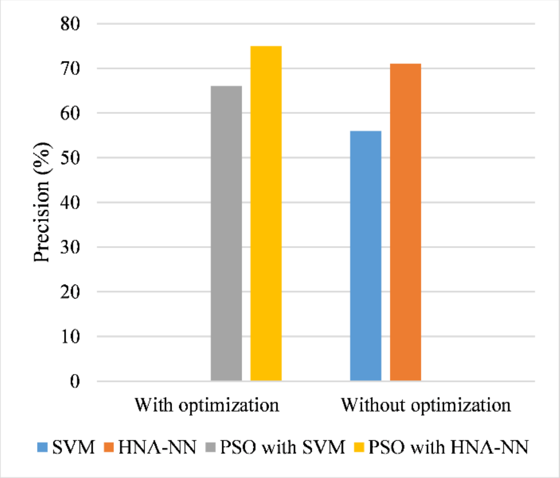

Figure 7: Precision comparison

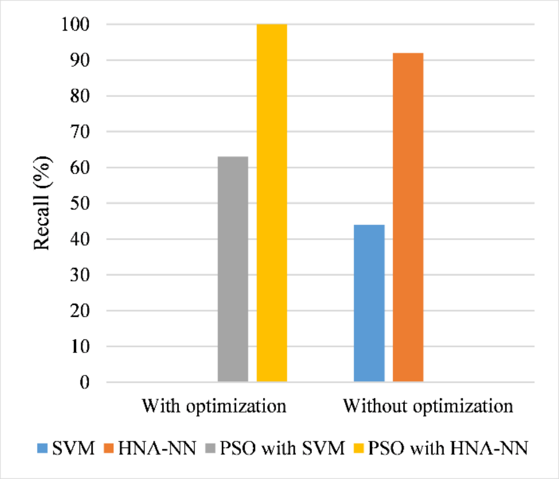

Figure 8: Recall comparison

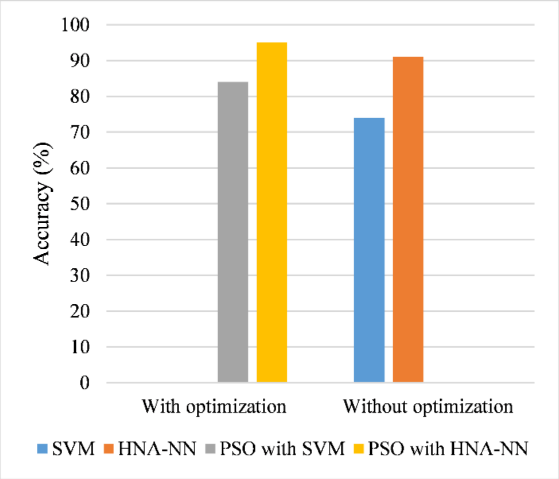

Figure 9: Accuracy comparison

Precision measures the percentage of correctly identified positive instances versus all predicted positives, calculated by Eq. 17.

$$Precision = \frac{True\ positives}{False\ positives\ +\ True\ positives} \quad (17)$$

The precision values show an improvement from 55% (SVM without optimization) and 70% (HNA-NN without optimization) to 65% and 72% with optimization, respectively. The proposed PSO framework with HNA-NN achieves the highest precision, surpassing other methods.

Recall indicates the likelihood for the classifier to correctly identify each instance, calculated as follows:

$$Recall = \frac{True\ positives}{False\ negatives\ +\ True\ positives} \quad (18)$$

The recall values increase significantly for the PSO-enhanced HNA-NN, reaching 100% compared to 44% and 92% for SVM and HNA-NN without optimization.

Accuracy is determined by the proportion of correctly categorized instances (both positive and negative) over the total instances, calculated using Eq. 19.

$$Accuracy = \frac{True\ positives + True\ negatives}{True\ instances} \quad (19)$$

With optimization, the accuracy values improve from 74% and 91% for SVM and HNA-NN to 84% and 95% with the PSO framework, highlighting the proposed method's robustness in classification tasks.

5 Discussion

Based on the above analysis, the following results have been validated in the proposed hybrid intrusion detection model, which incorporates the HBF-PSO and HNA-NN algorithms. Across datasets, the hybrid technique is more efficient than the conventional approach, enhancing precision, accuracy, sensitivity, and specificity. This demonstrates the efficiency of biologically inspired optimization techniques in addressing significant challenges in SCADA system intrusion detection.

In fact, the core contribution of this effort lies in incorporating HBF principles into the PSO. As the hummingbird flight behavior simulates energy-efficient movement and prevents redundancy, the improved PSO can counter the two main drawbacks that typically characterize the PSO algorithm. This is very useful in the SCADA context because the dimensionality of the feature space and the changing nature of attack patterns are very high.

Another important finding is the improved performance of the HNA-NN model when classifying network traffic patterns using the optimized feature set. This can be attributed to the fact that, compared to traditional machine learning techniques like the SVM model, the HNA-NN model can avoid overfitting and the inclusion of irrelevant features, when paired with HBF-PSO. This cooperation between the optimization and classification phases improves detection and reduces computation by removing irrelevant features.

The proposed approach has been found robust in both single-hop and multi-hop network configurations. This is quite important for practical SCADA systems, which involve diverse devices and complex communication patterns.

However, the following are the study's limitations. Even though the proposed model has been tested on a variety of datasets in a controlled simulation environment, its effectiveness in a real-world, large-scale SCADA system has not been comprehensively ascertained. Furthermore, the existing implementation has not addressed the analysis of encrypted communications and the resilience against adversarial attacks. These are upcoming issues in SCADA-based cybersecurity.

In the scenario of system uncertainty and encrypted traffic patterns, the proposed HBF-PSO+HNA-NN model demonstrates adaptability equal to that of control theory-based nonlinear systems. In this case, methodologies such as robust neural adaptive control [21], fuzzy control adaptation [22, 23], and back-stepping control adaptation [24] maintain system stability by self-optimizing control gain values. Similarly, the HBF-PSO method dynamically adjusts particle positions and steps in response to the network landscape. This sustains the convergence patterns of the HBF-PSO method despite the network uncertainty. On the other hand, the networked module provides additional optimization at classification boundaries through hierarchical adaptability patterns. This optimizes the resilience of classifications against decryption patterns of networked traffic. In contrast to other explicit network landscape-based optimization methodologies used in control theory, the proposed optimization routine maintains the required adaptability patterns derived from HBF-technology-based swarm optimization [25] techniques.

In addition to SCADA systems, the proposed hybrid optimization approach can also be directly implemented in the industrial IoT network, given its nonlinear couplings and uncertain communication channels. In this case, nonlinear optimization controllers and observer-based adaptive fuzzy control can ensure system stability through explicit plant modeling and gain modification. However, the HBF-PSO+HNA-NN model can ensure the system's robustness based on adaptation. In the HBF-PSO+HNA-NN model, the HBF module can dynamically adapt the search structure to network variation. This allows the IDS to remain convergent despite variations in network load and latency. On the other hand, the HNA-NN classifier can handle multidimensional feature interactions because its neurons' weights can be adjusted at the hierarchical level. This indicates that the proposed approach can provide an effective solution for protecting diverse industrial control and IoT networks when an explicit plant modeling strategy based on the deterministic paradigm is deemed unviable.

The outcomes of the proposed system were assessed against recent state-of-the-art IDS frameworks, as summarized in Table 1. While prior deep-learning and ensemble-optimization methods achieved average accuracies between 96.8% and 99.0%, the proposed model reached 98.12% detection accuracy and 100% precision on the SORD dataset, maintaining a 0% false-positive rate on MORD and SIRD. The improvement arises primarily from two design aspects:

- Output feature optimization based on the HBF-PSO process that adapts the exploration versus exploitation trade-off based on flight pattern-based stepsizes to avoid converging too early and optimize the feature set based on fluctuations in SCADA sensor inputs.
- Hierarchical adaptive learning in the HNA-NN classifier to adapt to nonlinear dependencies involving multiple variables as a function of environmental factors like humidity and temperature.

Compared with ensemble neural IDSs and statistical models, our hybrid design consistently reduced false-alarm rates by 1.2–1.8 percentage points while improving recall by up to 2.4 points. To assess the significance of these differences, paired t-tests were conducted over five independent runs for each dataset; $p < 0.05$ confirmed that the performance gains are statistically significant. Furthermore, the four datasets (SIRD, SORD, MIRD, and MORD) are derived from sensor-network simulations and real-world laboratory environments that emulate authentic SCADA traffic, including latency variations and multi-hop communication. This ensures that the reported performance reflects realistic operational behavior.

6 Conclusion

This paper proposed an enhanced intrusion detection methodology for SCADA networks by integrating the enhanced PSO algorithm with HNA-NN. Due to their interconnected nature, SCADA systems critical to industrial and infrastructure operations are increasingly exposed to external and internal cyber threats. The system was proposed to deal with the complexity of intrusion detection using an optimized feature selection approach via PSO and an advanced classification model, namely HNA-NN, which helps secure higher detection accuracy with reduced false positives. The simulation findings prove the efficiency of the developed system compared to traditional approaches on several datasets, including SVM and EDDC. Optimizing the feature selection process using the PSO algorithm has enabled the system to overcome the computational complexity issue in intrusion detection. The HNA-NN classifier significantly improved the system's detection capabilities for both external and internal attacks, with outstanding values for false-positive rate, detection rate, sensitivity, and specificity.

The experiment results validated the performance of the designed method under both one-hop and multi-hop environments. The optimization method has also improved the system's precision and accuracy in discriminating between normal and attack activities. Future studies can also investigate additional advancements in the IDS design that incorporate deep learning concepts for adaptive threat detection. Secondly, optimizing algorithm design for real-time systems can also ensure robustness. Thirdly, the effect of the Hybrid designs based on multiple optimization techniques can also be studied. Finally, implementing the aforementioned model can also validate the solution's viability.

References

[1] H. Lu *et al.*, "Automatic control and optimal operation for greenhouse gas mitigation in sustainable wastewater treatment plants: A review," *Science of the Total Environment,* vol. 855, p. 158849, 2023, doi: https://doi.org/10.1016/j.scitotenv.2022.158849.

[2] Z. Ullah *et al.*, "IoT-based monitoring and control of substations and smart grids with renewables and electric vehicles integration," *Energy,* vol. 282, p. 128924, 2023, doi: https://doi.org/10.1016/j.energy.2023.128924.

[3] O. E. Oluyisola, S. Bhalla, F. Sgarbossa, and J. O. Strandhagen, "Designing and developing smart production planning and control systems in the industry 4.0 era: a methodology and case study," *Journal of Intelligent Manufacturing,* vol. 33, no. 1, pp. 311–332, 2022, doi: https://doi.org/10.1007/s10845-021-01808-w.

[4] B. Al-Muntaser, M. A. Mohamed, and A. Y. Tuama, "Real-Time Intrusion Detection of

Insider Threats in Industrial Control System Workstations Through File Integrity Monitoring," *International Journal of Advanced Computer Science and Applications,* vol. 14, no. 6, 2023, doi: https://doi.org/10.14569/IJACSA.2023.0140636 .

[5] D. Upadhyay and S. Sampalli, "SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability assessment and security recommendations," *Computers & Security,* vol. 89, p. 101666, 2020, doi: https://doi.org/10.1016/j.cose.2019.101666.

[6] J. Zandi, A. N. Afooshteh, and M. Ghassemian, "Implementation and analysis of a novel low power and portable energy measurement tool for wireless sensor nodes," in *Electrical Engineering (ICEE), Iranian Conference on*, 2018: IEEE, pp. 1517–1522, doi: https://doi.org/10.1109/ICEE.2018.8472439.

[7] A. Ara, "Security in supervisory control and data acquisition (SCADA) based industrial control systems: challenges and solutions," in *IOP Conference Series: Earth and Environmental Science*, 2022, vol. 1026, no. 1: IOP Publishing, p. 012030, doi: https://doi.org/10.1088/1755-1315/1026/1/012030.

[8] V. Hayyolalam, B. Pourghebleh, and A. A. Pourhaji Kazem, "Trust management of services (TMoS): investigating the current mechanisms," *Transactions on Emerging Telecommunications Technologies,* vol. 31, no. 10, p. e4063, 2020, doi: https://doi.org/10.1002/ett.4063.

[9] O. A. Alimi, K. Ouahada, A. M. Abu-Mahfouz, S. Rimer, and K. O. A. Alimi, "A review of research works on supervised learning algorithms for SCADA intrusion detection and classification," *Sustainability,* vol. 13, no. 17, p. 9597, 2021, doi: https://doi.org/10.3390/su13179597.

[10] B. Molina-Coronado, U. Mori, A. Mendiburu, and J. Miguel-Alonso, "Survey of network intrusion detection methods from the perspective of the knowledge discovery in databases process," *IEEE Transactions on Network and Service Management,* vol. 17, no. 4, pp. 2451–2479, 2020, doi: https://doi.org/10.1109/TNSM.2020.3016246.

[11] A. Balla, M. H. Habaebi, M. R. Islam, and S. Mubarak, "Applications of deep learning algorithms for Supervisory Control and Data Acquisition intrusion detection system," *Cleaner Engineering and Technology,* vol. 9, p. 100532, 2022, doi: https://doi.org/10.1016/j.clet.2022.100532.

[12] B. Pourghebleh, K. Wakil, and N. J. Navimipour, "A comprehensive study on the trust management techniques in the Internet of Things," *IEEE Internet of Things Journal,* vol. 6, no. 6, pp. 9326–9337, 2019, doi: https://doi.org/10.1109/JIOT.2019.2933518.

[13] M. S. Noori, R. K. Sahbudin, A. Sali, and F. Hashim, "Feature drift aware for intrusion detection system using developed variable length particle swarm optimization in data stream," *IEEE Access,* vol. 11, pp. 128596–128617, 2023, doi: https://doi.org/10.1109/ACCESS.2023.3333000.

[14] M. A. Teixeira, M. Zolanvari, K. M. Khan, R. Jain, and N. Meskin, "Flow-based intrusion detection algorithm for supervisory control and data acquisition systems: A real-time approach," *IET Cyber-Physical Systems: Theory & Applications,* vol. 6, no. 3, pp. 178–191, 2021, doi: https://doi.org/10.1049/cps2.12016.

[15] Y. K. Saheed, O. H. Abdulganiyu, and T. Ait Tchakoucht, "A novel hybrid ensemble learning for anomaly detection in industrial sensor networks and SCADA systems for smart city infrastructures," *Journal of King Saud University-Computer and Information Sciences,* vol. 35, no. 5, p. 101532, 2023, doi: https://doi.org/10.1016/j.jksuci.2023.03.010.

[16] Y. Yu, G.-P. Liu, and W. Hu, "Security tracking control for discrete-time stochastic systems subject to cyber attacks," *ISA transactions,* vol. 127, pp. 133–145, 2022, doi: https://doi.org/10.1016/j.isatra.2022.02.001.

[17] C. Wang, H. Liu, C. Li, Y. Sun, W. Wang, and B. Wang, "Robust intrusion detection for industrial control systems using improved autoencoder and bayesian Gaussian mixture model," *Mathematics,* vol. 11, no. 9, p. 2048, 2023, doi: https://doi.org/10.3390/math11092048.

[18] X. Qin, F. Jiang, C. Dong, and R. Doss, "A hybrid cyber defense framework for reconnaissance attack in industrial control systems," *Computers & Security,* vol. 136, p. 103506, 2024, doi: https://doi.org/10.1016/j.cose.2023.103506.

[19] R. K. Pandey and T. K. Das, "Anomaly detection in cyber-physical systems using actuator state transition model," *International Journal of Information Technology,* vol. 17, no. 3, pp. 1509–1521, 2025, doi: https://doi.org/10.1007/s41870-024-02128-x.

[20] Q. Ling, W. Liu, F. Han, J. Shi, A. A. Hussein, and B. S. Sayway, "Feature selection using importance-based two-stage multi-modal multiobjective particle swarm optimization," *Cluster Computing,* vol. 28, no. 2, p. 115, 2025, doi: https://doi.org/10.1007/s10586-024-04807-7.

[21] A. Boulkroune, S. Hamel, F. Zouari, A. Boukabou, and A. Ibeas, "Output-Feedback Controller Based Projective Lag-Synchronization of Uncertain Chaotic Systems in the Presence of Input Nonlinearities," *Mathematical Problems in Engineering,* vol.

2017, no. 1, p. 8045803, 2017, doi: https://doi.org/10.1155/2017/8045803.

[22] A. Boulkroune, F. Zouari, and A. Boubellouta, "Adaptive fuzzy control for practical fixed-time synchronization of fractional-order chaotic systems," *Journal of Vibration and Control,* p. 10775463251320258, 2025, doi: https://doi.org/10.1177/10775463251320258.

[23] L. Merazka, F. Zouari, and A. Boulkroune, "High-gain observer-based adaptive fuzzy control for a class of multivariable nonlinear systems," in *2017 6th International Conference on Systems and Control (ICSC)*, 2017: IEEE, pp. 96–102, doi: https://doi.org/10.1109/ICoSC.2017.7958728.

[24] F. Zouari, K. B. Saad, and M. Benrejeb, "Adaptive backstepping control for a class of uncertain single input single output nonlinear systems," in *10th International Multi-Conferences on Systems, Signals & Devices 2013 (SSD13)*, 2013: IEEE, pp. 1–6, doi: https://doi.org/10.1109/SSD.2013.6564134.

[25] G. Rigatos, M. Abbaszadeh, B. Sari, P. Siano, G. Cuccurullo, and F. Zouari, "Nonlinear optimal control for a gas compressor driven by an induction motor," *Results in Control and Optimization,* vol. 11, p. 100226, 2023, doi: https://doi.org/10.1016/j.rico.2023.100226.