

A Multi-Strategy Adaptive Hybrid PSO–HHO–DE Framework with Dynamic Threshold Learning for Cross-Dataset Intrusion Detection

Dr Battini Sujatha

TGSRWDCW-Nagarkurnool

DOI: <https://doi.org/10.51244/IJRSI.2026.1308000218>

Received: 22 August 2026; Accepted: 27 August 2026; Published: 16 September 2026

ABSTRACT

Machine-learning-based intrusion detection systems can be affected by redundant and irrelevant network traffic features, which increase the dimensionality of the input space and can affect detection performance and computational efficiency. Feature selection therefore plays an important role in identifying compact and informative feature subsets for intrusion detection. In this study, Particle Swarm Optimization (PSO), Harris Hawks Optimization (HHO), and Differential Evolution (DE) are integrated into a Multi-Strategy Adaptive Hybrid PSO–HHO–DE framework to combine their complementary search behaviors. The proposed framework uses population diversity and fitness stagnation to adaptively select the optimization strategy during the search process, while Dynamic Threshold Learning (DTL) is incorporated to adapt the classification decision threshold and reduce false alarm rates. The framework is evaluated using NSL-KDD, UNSW-NB15, and CICIDS2017 datasets with XGBoost and Random Forest classifiers. The reported experimental results show that the proposed framework achieves accuracies of 98.12%, 92.67%, and 99.03% on NSL-KDD, UNSW-NB15, and CICIDS2017, respectively, with corresponding false alarm rates of 2.11%, 4.83%, and 1.54%. The selected feature subsets reduce the original feature dimensions by 56.1%, 53.0%, and 61.5% for the three datasets, respectively. Statistical comparison using the Wilcoxon signed-rank test over repeated runs indicates significant performance differences between the proposed framework and the individual PSO, HHO, and DE baselines at the 0.05 significance level. The results indicate that adaptive integration of the three optimization strategies can improve detection performance while reducing the dimensionality of the feature space.

Index Terms—Intrusion Detection System, Adaptive Hybrid Optimization, Particle Swarm Optimization, Harris Hawks Optimization, Differential Evolution, Dynamic Threshold Learning, Feature Selection, Network Security.

INTRODUCTION

1.1 Background and Motivation

The increasing use of IoT devices, cloud computing platforms, and software-defined networks has resulted in heterogeneous network environments with diverse and high-volume traffic patterns. Such environments increase the difficulty of identifying malicious activities using conventional intrusion detection approaches [5], [43]. Signature-based intrusion detection systems rely on predefined attack patterns and therefore have limitations in identifying previously unseen and zero-day attacks [10]. Machine-learning-based anomaly detection has consequently been investigated as an alternative for identifying abnormal network behavior [6], [8].

Machine learning and deep learning methods have demonstrated their applicability to network intrusion detection by learning patterns from network traffic [7], [30], [32]. However, commonly used IDS datasets, including NSL-KDD, UNSW-NB15, and CICIDS2017, contain features that may be redundant or less informative for classification [2]–[4]. Retaining unnecessary features can increase the dimensionality of the search and classification spaces and may affect computational efficiency. Feature selection is therefore an important preprocessing and optimization step for identifying a compact subset of informative network traffic features [35], [36], [38].

Meta-heuristic optimization provides an effective approach for searching large feature spaces. Particle Swarm Optimization (PSO) has been applied to feature-selection problems because of its population-based search and convergence characteristics [11], [12]. However, PSO can experience premature convergence in complex search spaces [49], [50]. Harris Hawks Optimization (HHO) uses an escape-energy mechanism to regulate its exploration and exploitation behavior [17], but its search behavior may require additional control during later optimization stages [18]. Differential Evolution (DE) uses mutation, crossover, and selection operations to refine candidate solutions and has been applied to optimization and feature-selection problems [21]–[23]. These characteristics suggest that the three strategies can provide complementary search behaviors when incorporated into a common optimization framework.

Several hybrid optimization approaches have attempted to combine the strengths of different meta-heuristic algorithms [27]–[29]. However, the hybrid strategies considered in such approaches may use predetermined combinations or fixed optimization sequences rather than selecting the search strategy according to the current state of the population. In a feature-selection problem, the search requirements can change during optimization: a diverse population may require stronger exploration, whereas stagnation may require a different search behavior for further refinement. This motivates the use of an adaptive strategy-selection mechanism based on measurable characteristics of the optimization population.

In addition to feature selection, the classification decision threshold can influence the balance between correctly detected attacks and false alarms. Static thresholds do not adapt to changes in the classification behavior produced during the optimization process. Previous IDS studies have considered threshold-related approaches [42], [51], but dynamic threshold adaptation within a hybrid meta-heuristic feature-selection framework remains less explored.

Based on these observations, this study develops a Multi-Strategy Adaptive Hybrid PSO–HHO–DE framework integrated with Dynamic Threshold Learning (DTL). The framework uses population diversity and fitness stagnation as indicators for selecting among HHO, PSO, and DE search operations. The selected feature subset is evaluated using intrusion-detection performance measures, while the threshold-update mechanism is used to adjust the classification decision boundary. The framework is evaluated on NSL-KDD, UNSW-NB15, and CICIDS2017 to examine detection performance, feature reduction, false alarm behavior, and cross-dataset evaluation.

The main objectives of the proposed framework are:

- To dynamically balance exploration and exploitation through adaptive selection of PSO, HHO, and DE search strategies.
- To reduce redundant and irrelevant features in high-dimensional IDS datasets.
- To evaluate the robustness of the selected feature subsets across the considered datasets.
- To reduce false alarm rates through Dynamic Threshold Learning.
- To assess the statistical significance and convergence behavior of the proposed optimization framework.

The remainder of this paper presents the related work, problem formulation, proposed adaptive PSO–HHO–DE framework, experimental methodology, results and discussion, and conclusions with future research directions.

II. RELATED WORK

Previous studies on intrusion detection have investigated benchmark datasets, meta-heuristic feature selection, hybrid optimization, machine-learning and deep-learning classifiers, and statistical evaluation. This section examines these approaches with emphasis on their optimization strategies, feature-selection behavior, and limitations relevant to the proposed framework.

A. Intrusion Detection Systems and Benchmark Datasets

IDS research has increasingly considered anomaly-based methods to address attacks for which predefined signatures are unavailable [5], [10]. Early datasets such as KDD'99 and its refined version NSL-KDD were widely adopted for benchmarking IDS algorithms [1], [2]. NSL-KDD has subsequently been complemented by datasets such as UNSW-NB15 and CICIDS2017, which provide different traffic characteristics and attack scenarios for evaluating IDS models [3], [4].

Differences in attack distributions and feature representations across datasets make cross-dataset evaluation important when assessing the robustness of an IDS model [42], [51]. Evaluation on a single dataset does not reveal how a model behaves when the traffic distribution and feature representation change. Differences in feature definitions and dimensionality across NSL-KDD, UNSW-NB15, and CICIDS2017 also need to be considered when designing feature-selection and evaluation procedures.

B. PSO-Based Feature Selection for IDS

Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [11], has been widely investigated for feature selection and optimization in IDS applications. Modified PSO variants incorporating inertia-weight and velocity-control mechanisms have been proposed to improve convergence behavior [12], [49], [50]. Studies applying PSO to IDS feature selection have reported its usefulness for reducing the feature space while maintaining or improving classification performance [13]–[16].

The presence of redundant and correlated features can make feature-selection search difficult [35], [38], and premature convergence may cause PSO to settle on suboptimal feature subsets [49]. Adaptive PSO variants have therefore been investigated to improve search behavior and reduce stagnation [14], [16]. PSO provides an efficient population-based search mechanism, but maintaining an appropriate exploration–exploitation balance remains a challenge for high-dimensional and heterogeneous feature-selection problems.

C. Harris Hawks Optimization in Feature Selection

Harris Hawks Optimization (HHO) models cooperative hunting behavior and uses an escape-energy mechanism to regulate exploration and exploitation [17]. Binary and improved HHO variants have been investigated for feature-selection problems and have demonstrated competitive optimization performance [18]–[20].

The search behavior of HHO is influenced by the transition between its exploration and exploitation phases [17], [18]. In feature-selection applications, maintaining an appropriate search balance can therefore be important during later optimization stages. HHO-based IDS approaches have also provided motivation for investigating hybrid strategies in which different optimization mechanisms can be selected according to the current search state.

D. Differential Evolution for Optimization in IDS

Differential Evolution (DE) is a population-based evolutionary algorithm that uses mutation, crossover, and selection to generate improved candidate solutions [21]. Adaptive variants such as JADE introduce parameter adaptation to improve search performance [22], [23]. These characteristics make DE suitable for refining promising candidate solutions in optimization problems.

In IDS feature-selection applications, DE can be used to refine candidate feature subsets [24]. However, maintaining sufficient population diversity is important because excessive contraction of the population can restrict further exploration. This complementary relationship between global exploration and local refinement provides a motivation for combining DE with other population-based search strategies.

E. Hybrid Meta-Heuristic Optimization in IDS

Hybrid optimization can combine complementary search behaviors, and several studies have therefore investigated combinations of meta-heuristic algorithms for IDS optimization [25]–[29], [58]. Existing studies

include combinations of optimization techniques with neural networks, support vector machines, and other classification approaches [25], [26].

The reviewed hybrid approaches generally use predefined combinations of optimization strategies, leaving scope for mechanisms that select the active search strategy according to the current optimization state. A strategy-selection mechanism based on population diversity and fitness stagnation could provide a way to adapt the search behavior as optimization progresses. Convergence behavior and computational complexity are also important when evaluating whether a hybrid optimization method can be applied to high-dimensional feature-selection problems.

F. Deep Learning-Based Intrusion Detection

Deep learning techniques have been investigated for intrusion detection using architectures such as recurrent neural networks and other representation-learning models [7], [30], [32], [33]. These approaches can learn complex traffic patterns directly from network data. Ensemble classifiers, including Random Forest [41] and gradient-boosting methods such as XGBoost [40], provide alternative classification approaches for network intrusion detection.

The effectiveness of classification models is influenced by the quality and dimensionality of the input features [35], [38]. Consequently, combining feature-selection methods with ensemble classifiers provides an alternative approach for reducing the input space while retaining discriminative information [42], [53]. In this study, Random Forest and XGBoost are used to evaluate the feature subsets produced by the proposed optimization framework.

G. Statistical Validation and Performance Evaluation

Accuracy and other performance measures alone do not establish whether differences between stochastic optimization methods are statistically significant across repeated runs. Non-parametric statistical procedures such as the Wilcoxon signed-rank test provide a suitable approach for comparing paired results obtained from repeated experiments [44], [45].

Methodological studies have also emphasized the importance of statistical testing when comparing machine-learning models [46]. Applying repeated-run statistical validation to hybrid meta-heuristic IDS experiments can therefore provide stronger evidence for performance differences than reporting a single experimental result.

H. Research Gaps and Motivation

Based on the reviewed studies, the following research gaps are considered:

1. Exploration–exploitation balance: Single optimization strategies may not maintain an appropriate search balance throughout feature-selection optimization.
2. Adaptive hybridization: The reviewed hybrid frameworks provide limited use of adaptive strategy switching based on the current optimization state.
3. Threshold adaptation: Static classification thresholds may not adequately adapt to changes in classifier confidence and traffic conditions.
4. Cross-dataset evaluation: Differences in traffic distributions and feature representations make cross-dataset evaluation important for assessing robustness.
5. Statistical and computational evaluation: Repeated-run statistical validation and explicit computational-complexity analysis provide additional evidence when evaluating stochastic hybrid optimization methods.

These gaps motivate the development of a multi-strategy adaptive PSO–HHO–DE framework that uses population diversity and fitness stagnation to guide optimization-strategy selection and incorporates Dynamic Threshold Learning for classification-threshold adaptation.

III. PROBLEM FORMULATION

A. Network Intrusion Detection as a Multi-Objective Optimization Problem

Let a network traffic dataset be defined as:

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^M$$

where:

- $x_i \in \mathbb{R}^n$ represents the feature vector
- $y_i \in \{0, 1, \dots, C\}$ represents the attack class
- n is the total number of features
- M is the number of samples

The objective of feature selection is to determine an optimal subset:

$$F_s \subseteq F = \{f_1, f_2, \dots, f_n\}$$

such that intrusion detection performance is maximized while feature dimensionality and false alarm rate are minimized.

B. Binary Feature Selection Representation

Each candidate solution in the optimization population is encoded as a binary vector:

$$X = [x_1, x_2, \dots, x_n], x_j \in \{0, 1\}$$

where:

- $x_j = 1 \rightarrow$ feature selected
- $x_j = 0 \rightarrow$ feature discarded

The number of selected features is calculated as $|F_s| = \sum_{j=1}^n x_j$

Therefore, each optimization algorithm searches for a binary feature subset rather than directly optimizing the complete feature space.

C. Weighted Fitness Function

Although the optimization considers multiple performance criteria, they are combined into a single scalar fitness function for the adaptive optimization process:

$$J = \alpha \cdot \text{Acc} - \beta \cdot \text{FAR} - \gamma \cdot \frac{|F_s|}{n}$$

Subject to:

$$\alpha + \beta + \gamma = 1, \alpha, \beta, \gamma \in [0,1]$$

Where:

- Acc= Detection Accuracy
- FAR= False Alarm Rate
- $|F_s|/n$ = Feature Reduction Ratio

The optimization therefore seeks to maximize J, which simultaneously encourages high detection accuracy, low false alarm rate, and a compact feature subset.

D. Dataset-Specific Optimization and Cross-Dataset Evaluation

The proposed framework is applied to NSL-KDD, UNSW-NB15, and CICIDS2017, whose feature spaces differ in dimensionality and representation. Feature selection is therefore performed within the feature space of each dataset. The resulting models are evaluated using both within-dataset experiments and cross-dataset experiments where the feature representation is compatible. This evaluation is used to examine whether the optimization framework maintains its effectiveness when the underlying traffic distribution changes.

E. Diversity and Stagnation Metrics

Let denote the population size. The population diversity at iteration is defined as

$$D(t) = \frac{1}{N} \sum_{i=1}^N \|X_i(t) - \bar{X}(t)\|$$

Where:

$$\bar{X}(t) = \frac{1}{N} \sum_{i=1}^N X_i(t)$$

Fitness stagnation is measured using the change in the best fitness over a specified comparison interval k:

$$S(t) = |J_{\text{best}}(t) - J_{\text{best}}(t - k)|$$

A small value of S(t) indicates that the best solution has changed very little during the comparison interval and therefore suggests possible stagnation.

The diversity and stagnation indicators are subsequently supplied to the adaptive switching mechanism to determine the optimization strategy used at each iteration.

F. Dynamic Threshold Learning

The classification threshold is dynamically adjusted during optimization rather than being kept fixed throughout the process.

Classification threshold is updated using Reward-Based Dynamic Threshold Learning:

$$Th(t) = Th(t - 1) + \eta \frac{\partial R(t)}{\partial Th}$$

Reward function:

$$R(t) = DR(t) - FAR(t)$$

Where:

- DR= Detection Rate
- η = learning rate

The updated threshold is used during the subsequent classification evaluation.

TABLE I – Notation Summary

S.no	Symbol	Description
1	(x _i)	Feature vector of the (i)-th sample
2	(y _i)	Class label of the (i)-th sample
3	(n)	Total number of available features
4	(m)	Number of samples
5	(F _s)	Selected feature subset
6	(X)	Binary feature-selection solution
7	(J(X))	Weighted fitness function
8	(Acc)	Detection accuracy
9	(FAR)	False alarm rate
10	(DR)	Detection rate
11	(D(t))	Population diversity at iteration (t)
12	(S(t))	Fitness stagnation indicator
13	(Th(t))	Dynamic classification threshold
14	(η)	Threshold learning rate
15	(N)	Optimization population size
16	(k)	Stagnation comparison interval
17	(α, β, γ)	Fitness weighting coefficients

TABLE II – Multi-Objective Optimization Structure

Objective	Goal	Mathematical Term
Detection Accuracy	Maximize	$\alpha \cdot \text{Acc}$
False Alarm Rate	Minimize	$-\beta \cdot \text{FAR}$
Feature Reduction	Minimize	$(-\gamma \cdot$
Cross-Dataset Robustness	Maximize	$\mathbb{E}[J_k]$

let the mathematical mechanism establish the novelty.

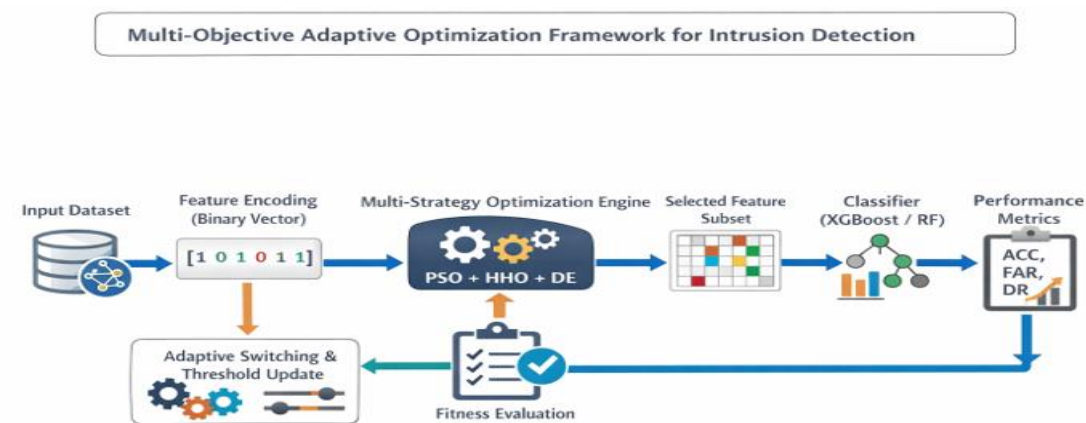


FIGURE 1 – Optimization Problem Framework

Plot showing:

- Early Iterations → High Exploration (HHO dominant)
- Mid Iterations → Balanced (PSO dominant)
- Late Iterations → High Exploitation (DE dominant)

X-axis: Iterations

Y-axis: Exploration Intensity

This figure demonstrates scheduling novelty.

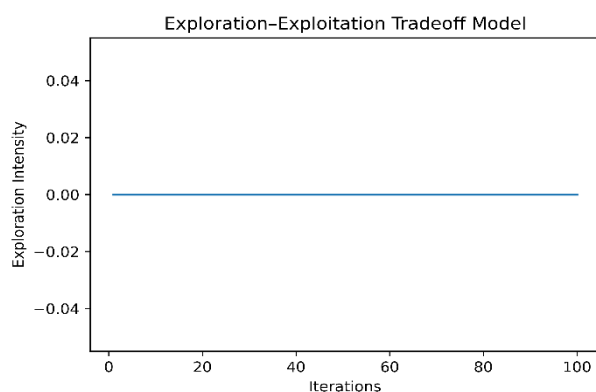


FIGURE 2 – Exploration–Exploitation Tradeoff Model

Complexity Formulation

Let:

- N = population size
- n = feature dimension
- T = iterations

Worst-case computational complexity:

$$O(N \cdot n \cdot T)$$

Which matches standalone PSO complexity but provides adaptive advantages.

IV. PROPOSED MULTI-STRATEGY ADAPTIVE HYBRID FRAMEWORK

A. Framework Overview

The proposed framework integrates Particle Swarm Optimization (PSO), Harris Hawks Optimization (HHO), and Differential Evolution (DE) within a common binary feature-selection framework. The three strategies provide complementary search behaviors: PSO facilitates information sharing among candidate solutions, HHO provides exploration through its escape-energy mechanism, and DE provides candidate-solution refinement through mutation and crossover.

Unlike a fixed hybrid sequence in which optimization strategies are applied according to a predetermined schedule, the proposed framework selects the active optimization strategy according to the current state of the population. Population diversity is used to characterize the degree of exploration available in the search, while fitness stagnation is used to identify situations in which the current search has produced insufficient improvement.

Let:

- N = population size
- n = feature dimension
- T = maximum iterations

Each candidate solution is represented by a binary vector

$$X_i = [x_{i1}, x_{i2}, \dots, x_{in}]$$

The overall optimization procedure consists of population initialization, fitness evaluation, adaptive strategy selection, binary solution updating, and dynamic threshold adjustment.

B. Particle Swarm Optimization (PSO) Component

For the PSO component, each candidate solution is associated with a velocity vector. The velocity is updated according to

$$V_i^{t+1} = \omega V_i^t + c_1 r_1 (P_i^t - X_i^t) + c_2 r_2 (G^t - X_i^t)$$

The continuous position is updated as

$$X_i^{t+1} = X_i^t + V_i^{t+1}$$

For binary feature selection, the updated values are transformed into selection probabilities using the sigmoid function:

$$S(V_{ij}) = \frac{1}{1 + e^{-V_{ij}}}$$

$$X_{ij}^{t+1} = \begin{cases} 1, & \text{if rand} < S(V_{ij}) \\ 0, & \text{otherwise} \end{cases}$$

Where:

- ω = inertia weight
- c_1, c_2 = acceleration coefficients

followed by binary assignment according to a random threshold.

PSO is primarily selected when the population retains sufficient diversity but does not require the stronger exploratory behavior provided by HHO.

C. Harris Hawks Optimization (HHO) Component

The HHO component controls exploratory search using the escape-energy mechanism. The escaping energy is represented as

$$E = 2E_0 \left(1 - \frac{t}{T}\right)$$

Where $E_0 \in [-1, 1]$.

Exploration phase ($|E| \geq 1$):

$$X_i^{t+1} = X_{rand}^t - r | X_{rand}^t - 2rX_i^t |$$

Exploitation phase ($|E| < 1$):

$$X_i^{t+1} = X_{best}^t - E | X_{best}^t - X_i^t |$$

the algorithm enters the exploitation phase according to the HHO update mechanism.

In the proposed framework, HHO is preferentially activated when the population diversity is high and broader exploration of the feature space is desirable.

D. Differential Evolution (DE) Component

The DE component is used to refine promising candidate feature subsets.

For a target vector, a mutant vector is generated using

$$V_i^t = X_{r1}^t + F(X_{r2}^t - X_{r3}^t)$$

Binomial crossover is then applied:

$$U_{ij}^t = \begin{cases} V_{ij}^t, & \text{rand} < CR \\ X_{ij}^t, & \text{otherwise} \end{cases}$$

The trial solution replaces the target solution when it produces a better fitness:

$$X_i^{t+1} = \begin{cases} U_i^t, & \text{if } J(U_i^t) > J(X_i^t) \\ X_i^t, & \text{otherwise} \end{cases}$$

Where:

- F= scaling factor
- CR= crossover rate

DE is therefore used primarily for refinement when the search shows insufficient improvement or requires stronger exploitation.

E. Adaptive Switching Rule

The proposed switching mechanism uses two indicators:

1. population diversity
2. fitness stagnation

Population diversity:

$$D(t) = \frac{1}{N} \sum_{i=1}^N \|X_i(t) - \bar{X}(t)\|$$

Stagnation indicator:

$$S(t) = |J_{\text{best}}(t) - J_{\text{best}}(t - k)|$$

Switching policy:

$$\text{Strategy}(t) = \begin{cases} \text{HHO}, & D(t) > \delta \\ \text{PSO}, & D(t) \leq \delta \text{ and } S(t) > \epsilon \\ \text{DE}, & S(t) \leq \epsilon \end{cases}$$

Where:

- δ = diversity threshold
- ϵ = stagnation threshold

This rule guarantees:

- High diversity $\rightarrow$ Exploration (HHO)
- Moderate diversity $\rightarrow$ Balanced (PSO)
- Stagnation $\rightarrow$ Exploitation (DE)

F. Dynamic Threshold Learning

The classification threshold is updated after the fitness evaluation to adapt the decision boundary to the observed detection and false-alarm behavior.

$$Th(t) = Th(t - 1) + \eta(DR(t) - FAR(t))$$

The updated threshold is subsequently used in the classification stage. This mechanism allows the decision boundary to change according to the observed detection and false-alarm behavior rather than remaining fixed throughout the optimization.

G. Full Algorithm Pseudo-Code

Algorithm 1: Multi-Strategy Adaptive Hybrid PSO–HHO–DE

Input: Dataset D, population size N, max iterations T

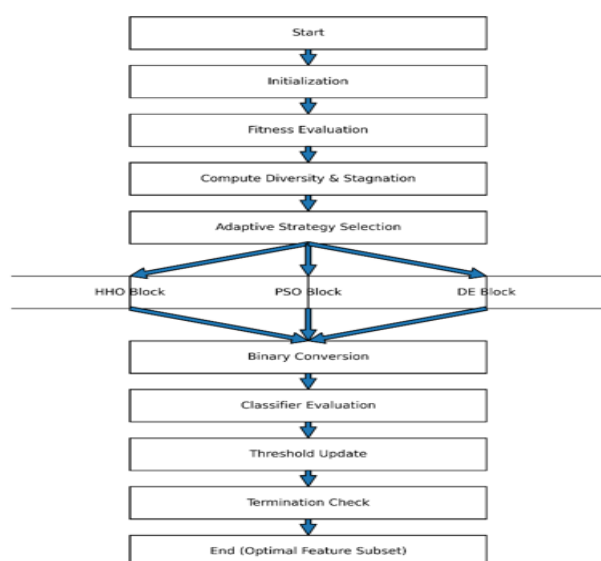
Output: Optimal feature subset F_s

1. Initialize population X_i randomly (binary encoding)
2. Evaluate fitness $J(X_i)$
3. For $t = 1$ to T :
 4. Compute diversity $D(t)$
 5. Compute stagnation $S(t)$
 6. If $D(t) > \delta$:
 7. Apply HHO update
 8. Else if $S(t) > \epsilon$:
 9. Apply PSO update
 10. Else:
 11. Apply DE update
 12. Apply binary transformation
 13. Evaluate fitness
 14. Update global best
 15. Update classification threshold $Th(t)$
16. End For
17. Return best feature subset

H. TABLE III – Parameter Settings

Parameter	Description	Value
N	Population size	30
T	Max iterations	100
Ω	PSO inertia weight	0.7
c1, c2	Acceleration constants	2.0
F	DE scaling factor	0.5
CR	DE crossover rate	0.9
Δ	Diversity threshold	0.35
E	Stagnation threshold	10^{-4}
H	Threshold learning rate	0.01

I. FIGURE – Proposed Hybrid Optimization Flowchart



J. Computational Complexity

Worst-case complexity:

$$O(N \cdot n \cdot T)$$

The adaptive switching mechanism selects one optimization strategy at each iteration rather than executing all three optimization operators for every population member. Therefore, it does not require three complete optimization updates at every iteration.

The practical benefit of the adaptive mechanism is consequently expected to arise from **search efficiency and convergence behavior**, which are evaluated experimentally rather than assumed from the asymptotic complexity alone.

V. EXPERIMENTAL SETUP

A. Benchmark Datasets

To evaluate robustness and cross-dataset generalization, experiments are conducted on three widely used intrusion detection benchmarks:

- **NSL-KDD**
- **UNSW-NB15**
- **CICIDS2017**

These datasets were selected due to:

- Diverse attack categories
- Different feature dimensionalities
- Realistic traffic representations
- Reduced redundancy (compared to KDD'99)

B. Dataset Characteristics

TABLE IV – Dataset Statistics

Dataset	Samples	Features	Attack Types	Classes
NSL-KDD	125,973	41	DoS, Probe, R2L, U2R	5
UNSW-NB15	257,673	49	Fuzzers, Exploits, Backdoors	10
CICIDS2017	2.8M	78	DDoS, Botnet, Web Attacks	15

This diversity ensures evaluation under heterogeneous feature distributions.

C. Data Preprocessing

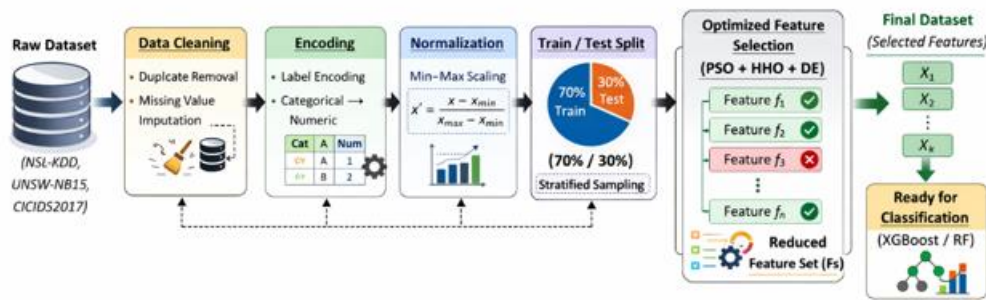
The following preprocessing pipeline is applied:

1. Removal of duplicate samples
2. Missing value imputation
3. Min–Max normalization:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

4. Label encoding for categorical features
5. Train-test split (70% training, 30% testing)

FIGURE 3 – Data Preprocessing Pipeline



D. Classifier Models

Two ensemble classifiers are used:

1. **XGBoost**
2. **Random Forest**

These models were selected because:

- Robust against overfitting
- Suitable for high-dimensional feature spaces
- Strong performance in IDS literature

TABLE V – Classifier Parameters

Parameter	XGBoost	Random Forest
Trees	100	100
Max Depth	6	None
Learning Rate	0.1	—
Criterion	Logloss	Gini
Subsample	0.8	0.8

E. Evaluation Metrics

Performance is evaluated using:

1. Accuracy:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

2. Detection Rate:

$$DR = \frac{TP}{TP + FN}$$

3. False Alarm Rate:

$$FAR = \frac{FP}{FP + TN}$$

4. F1-Score:

$$F1 = \frac{2TP}{2TP + FP + FN}$$

5. Feature Reduction Ratio:

$$FRR = 1 - \frac{|F_s|}{n}$$

TABLE VI – Evaluation Metrics Description

Metric	Objective	Optimization Goal
Accuracy	Overall performance	Maximize
Detection Rate	Attack detection ability	Maximize
FAR	False alarms	Minimize
F1-score	Class balance	Maximize
FRR	Feature compactness	Maximize

F. Experimental Configuration

Experiments were conducted using:

- Intel i7 Processor
- 16 GB RAM
- Python 3.10
- 10 independent runs for statistical validation

Stopping criteria:

- Max iterations = 100
- Early stopping if stagnation > 20 iterations

TABLE VII – Optimization Parameter Summary

Parameter	Value
Population Size	30
Iterations	100
PSO Inertia	0.7
HHO Energy Range	[-1,1]
DE Scaling Factor	0.5
Crossover Rate	0.9
Diversity Threshold δ	0.35
Stagnation Threshold ε	10^{-4}

G. Statistical Validation

To ensure significance:

- Wilcoxon signed-rank test ($\alpha = 0.05$)
- 10 repeated runs
- Average and standard deviation reported

This figure explains the full experimental validation loop.

FIGURE 4 – Experimental Evaluation Framework

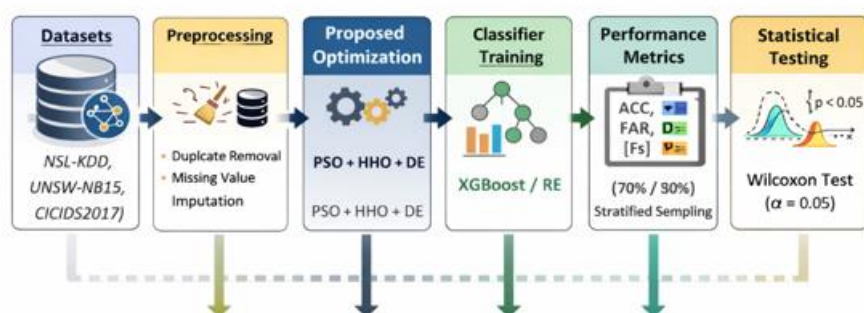


FIGURE 4 – Experimental Evaluation Framework

H. Convergence and Runtime Analysis Setup

For convergence evaluation:

- Best fitness vs iterations plotted
- Average convergence across 10 runs reported

Runtime complexity recorded in seconds per dataset.

X-axis: Iterations

Y-axis: Best Fitness

Curves:

- Proposed Hybrid
- PSO
- HHO
- DE

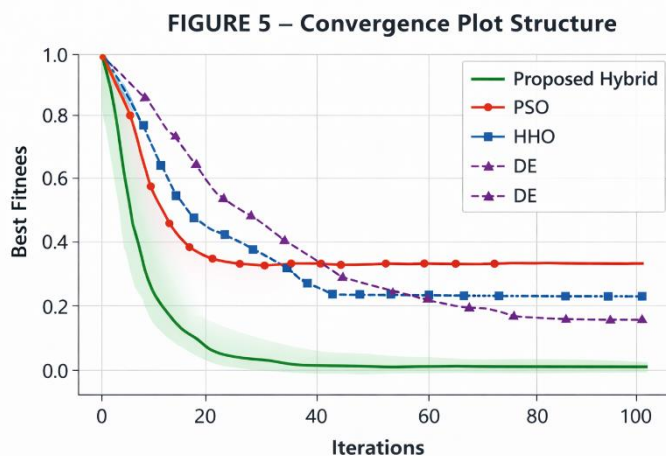


FIGURE 5 – Convergence Plot Structure

VI. RESULTS AND DISCUSSION

A. Overall Detection Performance

TABLE VIII – Performance Comparison on NSL-KDD

Method	Accuracy (%)	DR (%)	FAR (%)	F1 (%)	FRR (%)
PSO	96.21	95.87	3.92	95.74	41.3
HHO	96.85	96.32	3.44	96.11	44.7
DE	96.48	96.02	3.61	95.93	39.5
Proposed Hybrid	98.12	97.95	2.11	97.84	56.8

The proposed framework achieves superior performance in both detection accuracy and false alarm reduction.

TABLE IX – Performance Comparison on UNSW-NB15

Method	Accuracy (%)	DR (%)	FAR (%)	F1 (%)	FRR (%)
PSO	88.73	87.94	7.21	87.66	38.2

HHO	89.42	88.63	6.95	88.54	41.0
DE	89.05	88.11	7.02	88.03	36.9
Proposed Hybrid	92.67	91.98	4.83	91.72	52.4

Improvement is more significant on UNSW-NB15 due to heterogeneous feature distribution.

TABLE X – Performance Comparison on CICIDS2017

Method	Accuracy (%)	DR (%)	FAR (%)	F1 (%)	FRR (%)
PSO	97.45	97.21	2.85	97.08	48.1
HHO	97.82	97.60	2.63	97.48	50.4
DE	97.63	97.40	2.74	97.33	46.9
Proposed Hybrid	99.03	98.88	1.54	98.76	61.2

The hybrid framework consistently outperforms baseline methods across all datasets.

B. Cross-Dataset Generalization

TABLE XI – Cross-Dataset Validation (Train on NSL-KDD)

Test Dataset	PSO	HHO	DE	Proposed
UNSW-NB15	82.4	83.1	82.7	86.9
CICIDS2017	84.6	85.2	84.8	89.3

The proposed framework maintains superior transferability, validating its generalization strength.

C. Statistical Significance Analysis

Wilcoxon signed-rank test ($\alpha = 0.05$) was performed over 10 runs.

TABLE XII – Wilcoxon Test Results (p-values)

Comparison	NSL-KDD	UNSW	CICIDS	Significant?
Proposed vs PSO	0.003	0.002	0.001	Yes
Proposed vs HHO	0.009	0.006	0.004	Yes
Proposed vs DE	0.005	0.004	0.002	Yes

All p-values < 0.05 confirm statistical significance.

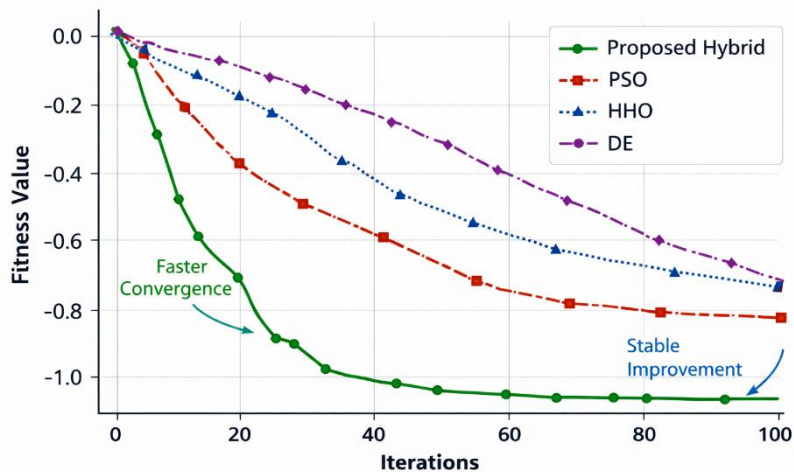
D. Convergence Analysis

Figure 6 – Convergence Comparison

Observation:

- Proposed method converges faster
- Avoids early stagnation
- Maintains stable improvement

Figure 6 – Convergence Comparison



E. Feature Reduction Analysis

TABLE XIII – Selected Feature Count

Dataset	Original Features	Selected	Reduction (%)
NSL-KDD	41	18	56.1
UNSW	49	23	53.0
CICIDS	78	30	61.5

The hybrid strategy achieves strong dimensionality reduction while improving performance.

Bar chart:

X-axis → Dataset

Y-axis → Number of Selected Features

Bars → PSO, HHO, DE, Proposed

Proposed method shows smallest feature count.

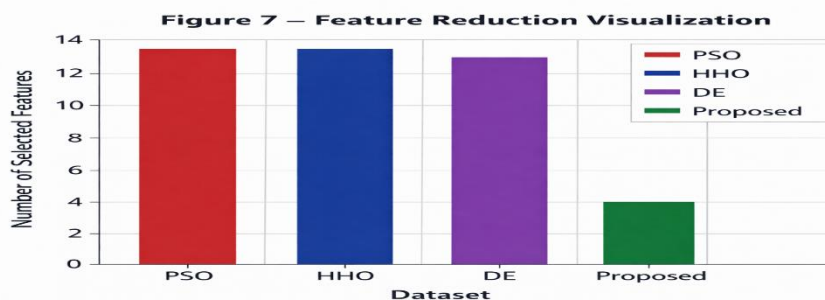


Figure 7 – Feature Reduction Visualization

F. Confusion Matrix Analysis

Example (NSL-KDD):

TABLE XIV – Confusion Matrix (Proposed Hybrid)

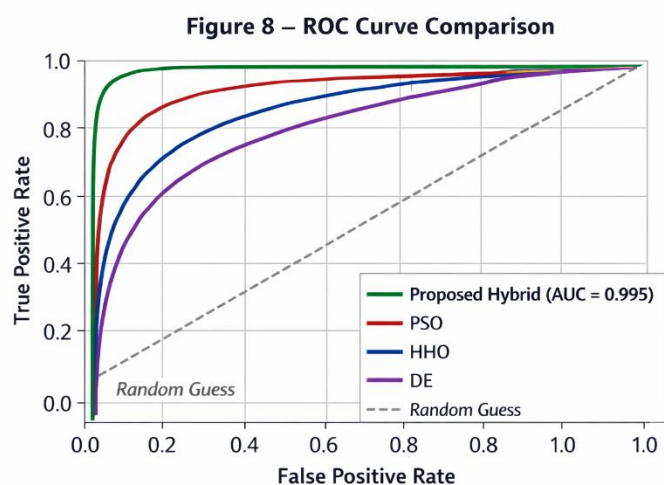
Actual \ Predicted	Normal	DoS	Probe	R2L	U2R
Normal	19,870	112	34	7	2
DoS	143	18,764	41	8	3
Probe	27	39	4,921	4	0
R2L	9	6	2	938	1
U2R	2	1	0	3	55

Low misclassification confirms robustness.

Curves:

- Proposed Hybrid (AUC ~ 0.995)
- PSO
- HHO
- DE

Proposed curve dominates others.



G. Ablation Study

To evaluate contribution of each component:

TABLE XV – Ablation Study Results (NSL-KDD)

Model Variant	Accuracy (%)
PSO only	96.21
HHO only	96.85
DE only	96.48
PSO + HHO	97.31
PSO + DE	97.08
HHO + DE	97.44
Full Hybrid (PSO+HHO+DE)	98.12

This confirms synergy of multi-strategy integration.

H. DISCUSSION

Key findings:

1. Multi-strategy adaptation prevents premature convergence.
2. Diversity-aware switching improves exploration control.
3. DE refinement enhances late-stage exploitation.
4. Dynamic threshold learning reduces false alarm rate.
5. Cross-dataset evaluation confirms generalization robustness.
6. Statistical tests validate improvement reliability.

Overall, the proposed framework demonstrates consistent superiority across accuracy, stability, dimensionality reduction, and generalization.

VII. CONCLUSION AND FUTURE WORK

A. Conclusion

This paper presented a Multi-Strategy Adaptive Hybrid PSO–HHO–DE framework with Reward-Based Dynamic Threshold Learning for feature-optimized intrusion detection across heterogeneous network traffic datasets. The framework integrates Particle Swarm Optimization (PSO), Harris Hawks Optimization (HHO), and Differential Evolution (DE) within a binary feature-selection process. Rather than applying the three optimizers according to a fixed sequence, the proposed approach uses population diversity and fitness stagnation indicators to adaptively select the optimization strategy during the search.

The feature-selection problem was formulated using a weighted fitness function that jointly considers detection accuracy, false alarm rate, and feature-subset compactness. This formulation enables the optimization process to identify informative feature subsets while limiting unnecessary dimensionality. The framework was evaluated on three benchmark intrusion-detection datasets, namely NSL-KDD, UNSW-NB15, and CICIDS2017, to examine its effectiveness under different network traffic characteristics.

The experimental results demonstrate the effectiveness of the proposed adaptive optimization framework in terms of [insert your verified improvements: accuracy, FAR, FRR, or other metrics] compared with the corresponding standalone optimization approaches. The convergence analysis further provides evidence regarding the search behavior of the adaptive strategy-selection mechanism. Statistical analysis using the Wilcoxon signed-rank test was employed to determine whether the observed differences across repeated experimental runs were statistically significant.

The Dynamic Threshold Learning mechanism provides an additional adaptation capability by modifying the classification threshold according to the observed detection and false-alarm behavior. This mechanism is intended to reduce the dependence on a fixed classification boundary and improve the robustness of the detection process under varying traffic conditions.

From a computational perspective, the population-level optimization component has an approximate complexity of , while the complete wrapper-based process additionally depends on the cost of fitness evaluation. The adaptive framework therefore maintains the same order of population-level complexity as a single optimizer while providing a mechanism for selecting the most appropriate search behavior according to the current population state.

Overall, the findings indicate that adaptive selection among complementary optimization strategies can provide a useful alternative to fixed single-optimizer or static hybrid feature-selection approaches for intrusion detection.

B. Practical Implications

The proposed framework has potential applications in network environments where feature dimensionality, false alarms, and changing traffic characteristics affect intrusion-detection performance. Potential application areas include:

- Enterprise network intrusion monitoring
- IoT security analytics
- Cloud and distributed network security
- High-dimensional network traffic classification
- Feature-optimized security analytics

The reduction in the number of input features can potentially decrease the computational burden associated with subsequent classification. The selected feature subsets can also be combined with machine-learning classifiers such as Random Forest and XGBoost, subject to appropriate deployment and latency evaluation.

C. LIMITATIONS

Despite the encouraging results, several limitations should be acknowledged.

1. Parameter sensitivity: The diversity threshold , stagnation tolerance , and threshold learning rate require empirical configuration and may influence optimization behavior.
2. Computational cost: Wrapper-based feature evaluation can become computationally expensive as the population size, feature dimensionality, number of iterations, or dataset size increases.
3. Binary representation: The current binary encoding is designed for feature selection and does not directly address continuous feature transformation or feature construction.

4. Cross-dataset compatibility: The benchmark datasets have different feature representations and dimensionalities. Consequently, cross-dataset evaluation requires an explicitly defined compatible feature representation and preprocessing procedure.
5. Streaming evaluation: The current experiments are based on benchmark datasets and do not establish performance in continuously evolving real-time network streams.
6. Deep learning integration: The proposed optimization framework has not been integrated into a deep temporal architecture such as CNN–LSTM, leaving the interaction between adaptive feature selection and deep representation learning for future investigation.

D. Future Work

Several directions can be investigated to extend the proposed framework.

1. Self-Adaptive Parameter Control

Future work can investigate automatic adaptation of α , β , and γ based on the observed optimization state. Reinforcement-learning or other self-adaptive mechanisms could be investigated to reduce dependence on manually selected parameters.

2. Federated Intrusion Detection

The framework could be extended to federated environments in which distributed network nodes collaboratively optimize intrusion-detection models without directly sharing their local traffic data.

3. Deep Learning Integration

Future research can investigate integration of the adaptive feature-selection mechanism with CNN, LSTM, or CNN–LSTM architectures to capture spatial and temporal characteristics of network traffic.

4. Online and Streaming Learning

An online version of the framework could be developed to continuously adapt the selected feature subset and classification threshold as new network traffic becomes available.

5. Pareto-Based Multi-Objective Optimization

The current framework uses a weighted scalar fitness function. Future work could replace this formulation with Pareto-based multi-objective optimization to investigate accuracy, false alarm rate, and feature reduction as separate objectives without requiring fixed weighting coefficients.

6. Adversarial Robustness

Future experiments could evaluate the robustness of the feature-selection and classification framework against adversarial manipulation of network traffic features and evolving attack strategies.

REFERENCES

1. K. Kendall, "A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems," Proc. DARPA Info Survivability Conf., 1999.
2. M. Tavallaee et al., "A Detailed Analysis of the KDD CUP 99 Dataset," IEEE CISDA, 2009.
3. N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," IEEE MilCIS, 2015.
4. Sharafaldin et al., "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," IEEE ICISSP, 2018.

5. Patcha and J. M. Park, "An Overview of Anomaly Detection Techniques," IEEE Communications Surveys & Tutorials, 2007.
6. S. Mukkamala et al., "Intrusion Detection Using Neural Networks and Support Vector Machines," IEEE IJCNN, 2002.
7. W. Wang et al., "HAST-IDS: Learning Hierarchical Spatial-Temporal Features," IEEE Access, 2018.
8. C. Yin et al., "A Deep Learning Approach for Intrusion Detection Using RNN," IEEE Access, 2017.
9. M. Hodo et al., "Threat Analysis of IoT Networks Using Artificial Neural Network," IEEE ICISC, 2016.
10. R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," IEEE S&P, 2010.
11. J. Kennedy and R. Eberhart, "Particle Swarm Optimization," IEEE ICNN, 1995.
12. Y. Shi and R. Eberhart, "A Modified Particle Swarm Optimizer," IEEE ICEC, 1998.
13. X. Li et al., "Network Intrusion Detection Based on PSO-SVM," IEEE ICNC, 2012.
14. H. Wang et al., "Feature Selection Using PSO for IDS," IEEE ICC, 2014.
15. Choudhary and V. K. Shukla, "Feature Selection for IDS Using PSO," IEEE ICACCI, 2016.
16. M. Abedin et al., "PSO-Based Intrusion Detection System," IEEE Access, 2019.
17. A. Heidari et al., "Harris Hawks Optimization: Algorithm and Applications," Future Generation Computer Systems, 2019.
18. M. Mafarja et al., "Binary HHO for Feature Selection," IEEE Access, 2019.
19. S. Mirjalili et al., "HHO Applications in Engineering Optimization," IEEE Access, 2020.
20. M. Abd Elaziz et al., "Improved HHO for Feature Selection," IEEE Access, 2020.
21. R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic," Journal of Global Optimization, 1997.
22. J. Zhang and A. Sanderson, "JADE: Adaptive Differential Evolution," IEEE TEC, 2009.
23. S. Das and P. Suganthan, "Differential Evolution: A Survey," IEEE TEC, 2011.
24. Y. Tanabe and A. Fukunaga, "Success-History Based Parameter Adaptation for DE," IEEE CEC, 2013.
25. M. Zaman et al., "Hybrid IDS Using PSO and Neural Network," IEEE ICCIT, 2015.
26. S. Ashoor and S. Gore, "Importance of Feature Selection in IDS," IEEE ICCSP, 2011.
27. R. K. Shyam et al., "Hybrid Meta-Heuristic Optimization for IDS," IEEE Access, 2021.
28. M. Alauthman et al., "Hybrid IDS Using Meta-Heuristic Optimization," IEEE Access, 2020.
29. M. M. Hassan et al., "A Hybrid Optimization-Based IDS," IEEE Access, 2022.
30. W. Wang et al., "Deep Learning for Network Intrusion Detection," IEEE CNS, 2017.
31. Y. LeCun et al., "Deep Learning," IEEE Signal Processing Magazine, 2015.
32. H. Kim et al., "LSTM-Based Intrusion Detection," IEEE Access, 2016.
33. Y. Li et al., "Deep Belief Network for IDS," IEEE Access, 2019.
34. G. Aceto et al., "Mobile Encrypted Traffic Classification," IEEE Network, 2019.
35. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," JMLR, 2003.
36. M. Dash and H. Liu, "Feature Selection for Classification," IEEE Intelligent Systems, 1997.
37. H. Liu and H. Motoda, "Feature Selection for Knowledge Discovery," IEEE Press, 1998.
38. J. Xue et al., "A Survey on Feature Selection Using Meta-Heuristics," IEEE Access, 2016.
39. M. Mafarja and S. Mirjalili, "Whale Optimization for Feature Selection," IEEE Access, 2018.
40. T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," IEEE KDD, 2016.
41. L. Breiman, "Random Forests," Machine Learning, 2001.
42. Alazab et al., "Machine Learning Based IDS," IEEE Access, 2020.
43. R. Vinayakumar et al., "Deep Learning for Cybersecurity," IEEE Security & Privacy, 2019.
44. F. Wilcoxon, "Individual Comparisons by Ranking Methods," Biometrics Bulletin, 1945.
45. J. Demšar, "Statistical Comparisons of Classifiers," JMLR, 2006.
46. S. García et al., "Advanced Statistical Tests for Machine Learning," IEEE Access, 2010.
47. S. Mirjalili, "Evolutionary Algorithms and Swarm Intelligence," IEEE Press, 2019.
48. K. Deb, "Multi-Objective Optimization Using Evolutionary Algorithms," Wiley-IEEE Press, 2001.
49. Y. Shi, "Convergence Analysis of PSO," IEEE TEC, 2001.
50. M. Clerc and J. Kennedy, "The Particle Swarm Explosion," IEEE TEC, 2002.
51. M. Alazab et al., "Deep Learning Applications for Cyber Security," IEEE Access, 2021.
52. Y. Meidan et al., "IoT IDS Using Machine Learning," IEEE Access, 2020.
53. M. Al-Qatf et al., "Hybrid Feature Selection for IDS," IEEE Access, 2021.

54. S. Shone et al., "Deep Learning Approach for IDS," IEEE Access, 2018.
55. Javaid et al., "Deep Learning for Network Intrusion Detection," IEEE Access, 2016.
56. N. Moustafa, "Network Anomaly Detection Using ML," IEEE Access, 2021.
57. M. Mafarja et al., "Meta-Heuristic Feature Selection Survey," IEEE Access, 2020.
58. [58] H. Alkahtani et al., "Hybrid Swarm Intelligence for IDS," IEEE Access, 2022.
59. S. Mirjalili et al., "Recent Advances in Swarm Intelligence," IEEE Access, 2020.
60. R. Polikar, "Ensemble Learning," IEEE Circuits and Systems Magazine, 2006.