A Crowding Distance-Driven Adaptive Genetic Algorithm with Multiobjective Evolutionary Optimization for Robust Attack Detection

D. Sudha
School of Computer Science and IT, JAIN (Deemed-to-be) University, Bangalore, India
sudhashinu@gmail.com (corresponding author)

D. Ganesh
School of Computer Science and IT, JAIN (Deemed-to-be) University, Bangalore, India
d.ganesh@jainuniversity.ac.in

Received: 21 September 2025 | Revised: 25 October 2025 and 15 November 2025 | Accepted: 17 November 2025

 DOI: https://doi.org/10.48084/etasr.14980

ABSTRACT

With the growing sophistication of cyber-attacks, traditional detection mechanisms struggle to maintain high accuracy while minimizing false positives and false negatives. The selection of optimal feature subsets and configurations significantly affects detection performance, necessitating advanced optimization strategies. Conventional Genetic Algorithms (GAs) lack dynamic adaptability and often converge prematurely due to low diversity. Moreover, single-objective optimization fails to balance conflicting objectives such as maximizing detection accuracy while minimizing false alarms. This study proposes an Adaptive Genetic Algorithm (AGA) integrated with Crowding Distance Selection (CDS) in a Multiobjective Evolutionary Algorithm (MOEA) framework to optimize binary-encoded feature subsets and system configurations. The proposed AGA dynamically adjusts mutation rates based on population diversity metrics calculated using the Hamming distance. Higher mutation rates are triggered when diversity is low to encourage exploration. The crowding distance is used to select diverse solutions in the multiobjective space defined by accuracy, false positives, and false negatives. A diverse initial population of feature subsets is evolved using adaptive mutation and crossover, guided by fitness scores from MOEA. Experimental evaluation on the NSL-KDD dataset reveals that the proposed AGA-MOEA method achieves a detection accuracy of 97.4%, a false positive rate of 2.1%, and a false negative rate of 3.8%.

Keywords-adaptive genetic algorithm; multiobjective optimization; crowding distance; feature selection; attack detection

I. INTRODUCTION

The rapid expansion of networked systems and the increasing complexity of cyber threats necessitate advanced Intrusion Detection Systems (IDS) to protect critical infrastructure. In particular, Distributed Denial of Service (DDoS) attacks pose a significant threat due to their ability to overwhelm network resources, causing service disruptions and financial losses [1-3]. Modern IDS must balance high detection accuracy with low false alarms and computational efficiency, especially in dynamic environments such as Software-Defined Networks (SDN) and Internet of Things (IoT) ecosystems. Despite advances, several challenges persist in the design of effective IDS. First, the high dimensionality of network traffic data, with thousands of features, often results in overfitting and increased computational cost [4-5]. Second, conventional machine learning models typically optimize for a single objective, such as accuracy, often neglecting other critical factors such as false positive and false negative rates [6]. Third, the evolving nature of cyber threats demands adaptive and scalable detection frameworks capable of handling large-scale heterogeneous data in real-time [7, 8].

A. Related Works

Feature selection and optimization for intrusion detection have garnered significant research attention in recent years. Hybrid approaches that combine evolutionary algorithms with machine learning classifiers have proven effective in handling high-dimensional network data [9-10]. For example, Support Vector Machine integrated with Genetic Algorithm (SVM-GA) has been widely studied for its ability to improve detection accuracy by selecting relevant features while tuning hyperparameters [11]. Similarly, Particle Swarm Optimization combined with Random Forest (PSO-RF) has shown promise in balancing accuracy and computational efficiency [12]. Differential Evolution with Neural Networks (DE-NN) leverages population-based mutation and crossover strategies to thoroughly explore feature spaces, yielding competitive

detection rates [13]. Ant Colony Optimization coupled with Artificial Neural Networks (ACO-ANN) exploits the pheromone-inspired heuristic to guide feature selection and training, demonstrating improvements in the reduction of false positives [14].

However, many of these methods optimize for a single objective or treat multiobjective criteria in a sequential or weighted-sum manner, which can lead to bias and suboptimal solutions [15]. Multiobjective Evolutionary Algorithms (MOEAs), such as NSGA-II and SPEA2, address this by searching for Pareto-optimal fronts that represent trade-offs between conflicting objectives such as accuracy, False Positive Rate (FPR), and False Negative Rate (FNR) [16]. Despite this, standard MOEAs often suffer from premature convergence and lack diversity maintenance, which hinders exploration of the solution space [17].

Adaptive mutation rates based on population diversity have recently gained attention as a mechanism to maintain genetic diversity and avoid local optima. Studies show that integrating diversity-aware mutation strategies into genetic algorithms can significantly enhance exploration capabilities [18]. Additionally, crowding distance-based selection has been employed to maintain diversity on the Pareto front, facilitating a spread of diverse high-quality solutions [19]. In [20], an accuracy of 0.90 was obtained by selecting 30 features, demonstrating that using an optimal number of features, maximum discriminatory power can be obtained along with minimizing redundancy. Feature selection can improve the ability of deep learning models to identify DDoS attacks. The results in [21] show a noticeable boost in detection performance, proving that the feature selection strategy used makes a significant contribution to SDN security.

Although these advances provide a solid foundation, there remains a gap in applying these adaptive multiobjective techniques explicitly to feature subset selection and system configuration tuning for DDoS detection in SDN and IoT environments. This research work fills that gap by proposing an Adaptive Genetic Algorithm (AGA) integrated with crowding distance within a Multiobjective Evolutionary Algorithm (MOEA) framework, offering improved accuracy, balanced false alarm rates, and efficient convergence. The primary problem addressed is the optimization of feature selection and system configurations to maximize detection accuracy while simultaneously minimizing FPR and FNR in a multiobjective framework [8]. Existing solutions struggle to maintain this balance efficiently, often leading to suboptimal trade-offs or excessive computational overhead. The proposed AGA is integrated with Crowding Distance Selection (CDS) within an MOEA framework to address these challenges. The objectives are to: (i) optimize feature subsets for improved detection performance, (ii) dynamically adjust mutation rates based on population diversity to maintain exploration, and (iii) leverage crowding distance to preserve solution diversity in the Pareto front. The novelty of this approach lies in its adaptive mutation mechanism informed by diversity metrics such as Hamming distance, which guides the evolutionary search more effectively than static parameter settings. Additionally, the integration of crowding distance-based selection fosters a diverse set of Pareto-optimal solutions, offering flexible trade-offs between competing objectives. The main contributions of this work include:

- A novel adaptive mutation strategy in genetic algorithms tailored for feature subset selection in IDS.
- Multiobjective optimization of accuracy, false positive rate, and false negative rate using crowding distance.
- Comprehensive evaluation against four state-of-the-art hybrid methods across multiple benchmark DDoS datasets.

II. PROPOSED METHOD

The proposed method, shown in Figure 1, dynamically balances exploration and exploitation in the search space by adjusting genetic parameters based on population diversity and uses MOEA with crowding distance to maintain Pareto diversity among solutions.

Fig. 1. Proposed framework.

This method can be described by the following steps:

1. Initialization: Generate an initial population of binary-encoded chromosomes, each representing a feature subset and system configuration.
2. Diversity Calculation: Compute the Hamming distance between chromosomes to measure population diversity.
3. Adaptive Mutation: If diversity is low, increase the mutation rate to promote exploration.
4. Fitness Evaluation: Use MOEA to evaluate chromosomes based on detection accuracy, false positives, and false negatives.

5. Crowding Distance Selection: Rank solutions based on nondomination and crowding distance to maintain diversity in the Pareto front.
6. Genetic Operations: Apply crossover and adaptive mutation to evolve new generations.
7. Convergence: Repeat until convergence criteria (e.g., max generations or stagnation) are met.

A. Initialization and Diversity Calculation

The optimization process begins with the random initialization of a population of binary-encoded chromosomes P, where each chromosome represents a potential feature subset and system configuration. If the total number of features is n, each chromosome $C_i \in P$ is a binary string of length n:

$$C_i = [f_1, f_2, \ldots, f_n],\ f_j \in \{0,1\}$$

where $f_j = 1$ indicates that feature j is selected, and 0 otherwise. As shown in Table I, each chromosome is a binary vector where 1 is used to mark selected features. The population is initialized randomly to maintain diversity at the start. To quantify diversity, the average pairwise Hamming distance among chromosomes is calculated. The Hamming distance between two chromosomes C_i and C_j is:

$$D_H(C_i, C_j) = \sum_{k=1}^{n} \delta\,(C_{ik}, C_{jk})$$

where:

$$\delta(a,b) = \begin{cases} 1, & \text{if } a \neq b \\ 0, & \text{otherwise} \end{cases}$$

$$AverageDiversity = \frac{2}{|P|(|P|-1)} \sum_{i=1}^{|P|} \sum_{j=i+1}^{|P|} D_H\,(C_i, C_j)$$

TABLE I. INITIAL CHROMOSOMES (FEATURE SUBSETS)

Chromosome ID	f_1	f_2	f_3	f_4	f_5	f_6	f_7
C_1	1	0	1	1	0	1	0
C_2	0	1	1	0	1	0	1
C_3	1	1	0	1	0	1	1

B. Adaptive Mutation Based on Diversity

To avoid premature convergence, the mutation rate μ is dynamically adapted based on the measured diversity D. A threshold θ is set for the minimum required diversity. If diversity falls below θ, mutation is increased to encourage exploration:

$$\mu = \begin{cases} \mu_{\max}, & \text{if } D < \theta \\ \mu_{\min} + \left(\frac{D-\theta}{D_{\max}-\theta}\right) \cdot (\mu_{\max} - \mu_{\min}), & \text{otherwise} \end{cases}$$

Let's assume $\mu_{min} = 0.01$, $\mu_{max} = 0.3$, $\theta = 2.5$, and a current average diversity $D = 1.8$ triggers max mutation.

As seen in Table II, the mutation rate increases as diversity falls, helping the algorithm escape local optima by generating more varied offspring. The mutation operator is applied bitwise to chromosomes with the probability μ. For each gene f_j in chromosome C_i, if a random number $r < \mu$, then:

$$f_j' = 1 - f_j$$

TABLE II. ADAPTIVE MUTATION RATE BASED ON DIVERSITY

Generation	Avg Hamming distance D	Mutation rate μ
Gen 1	3.1	0.05
Gen 5	2.7	0.10
Gen 10	1.8	0.30

Table III shows how bits are flipped due to mutation. This stochastic change allows exploration of new regions in the feature space.

TABLE III. MUTATION EFFECT ON A CHROMOSOME

Original chromosome C_4	1	0	1	0	0	1	1
After mutation ($\mu = 0.3$)	1	1	1	1	0	0	1

C. Fitness Evaluation

Each chromosome (solution) is evaluated using a multiobjective fitness function that considers three critical performance metrics: Detection Accuracy (Acc), False Positive Rate (FPR), and False Negative Rate (FNR). These metrics are computed from the confusion matrix as:

$$\text{Accuracy (Acc)} = \frac{TP+TN}{TP+TN+FP+FN}$$

$$\text{False Positive Rate (FPR)} = \frac{FP}{FP+TN}$$

$$\text{False Negative Rate (FNR)} = \frac{FN}{FN+TP}$$

In MOEA, each objective is treated independently.

As seen in Table IV, multiple chromosomes may have trade-offs; for example, C_3 has the best accuracy and FPR combination but not the lowest FNR, which sets the stage for Pareto-based ranking.

TABLE IV. FITNESS VALUES FOR SELECTED CHROMOSOMES

Chromosome ID	Accuracy (%)	FPR (%)	FNR (%)
C_1	96.3	3.5	4.1
C_2	95.7	4.2	3.8
C_3	97.4	2.1	3.8
C_4	94.9	5.1	5.3

D. Crowding Distance Selection

To maintain diversity in the Pareto front and avoid clustering of solutions, Crowding Distance (CD) is computed for each chromosome, estimating the density of solutions surrounding a particular point in the objective space. For each objective f_k, the crowding distance for solution i is:

$$CD_i^k = \frac{f_{i+1}^k - f_{i-1}^k}{f_{\max}^k - f_{\min}^k}$$

The total crowding distance is:

$$CD_i = \sum_{k=1}^{m} C\,D_i^k$$

where m is the number of objectives.

Solutions are sorted in nondominated fronts, and within each front, those with higher CD values are favored for

selection. Table V shows the normalized crowding distances. Chromosomes with higher CD values are less crowded and preferred for the next generation.

TABLE V. CROWDING DISTANCE CALCULATION FOR FRONT 1

Chromosome ID	Accuracy rank	FPR rank	FNR rank	Crowding Distance (CD)
C_3	2	1	2	0.95
C_1	3	2	3	0.87
C_2	4	3	1	0.89

E. Genetic Operations (Crossover and Mutation)

Selected chromosomes undergo genetic operators:

- Crossover: A single-point or two-point crossover is applied to parent chromosomes with probability P_c (typically 0.8), generating offspring by combining gene segments.

  $\text{Child}_1 = [P_1[1:k], P_2[k+1:n]]$

  $\text{Child}_2 = [P_2[1:k], P_1[k+1:n]]$

- Mutation: As previously explained, mutation is adaptive based on diversity. Each bit is flipped with a rate μ, which increases when population diversity decreases.

As shown in Table VI, crossover produces an offspring combining segments of both parents, and mutation further tweaks the chromosome for variability.

TABLE VI. GENETIC OPERATOR (CROSSOVER + MUTATION)

Parent A	1	0	1	1	0	1	0
Parent B	0	1	0	0	1	0	1
Crossover →	1	0	1	0	1	0	1
Mutation (μ=0.3) →	1	1	1	0	1	1	1

The evolutionary process continues until one of the following convergence criteria is met:

- Maximum number of generations reached (e.g., 200).
- Stagnation in Pareto front improvement for a set number of iterations.
- Minimum diversity threshold breached repeatedly.

During each generation, Pareto fronts and crowding distance ranks are updated, and elitism ensures the best solutions are retained. As shown in Table VII, solution quality and front diversity improve over time and then plateau, indicating convergence.

TABLE VII. CONVERGENCE PROGRESSION OVER GENERATIONS

Generation	Best accuracy (%)	Avg CD	Non-dominated solutions
Gen 1	93.8	0.42	4
Gen 50	96.2	0.66	12
Gen 150	97.4	0.89	22
Gen 200	97.4	0.91	23

III. RESULTS AND DISCUSSION

MATLAB R2023b with the MOEA framework plugin was used for simulations on a PC with Intel Core i9-13900K CPU, 64 GB RAM, and NVIDIA RTX 4090 GPU, with the NSL-KDD benchmark [22].

The proposed method was compared with existing methods, including SVM-GA, PSO-RF, DE-NN, and ACO-ANN. The proposed model was evaluated using 10-fold cross-validation for reliability, with metrics averaged over 5 independent runs. Table VIII summarizes the parameters used in this study.

TABLE VIII. PARAMETER SETTINGS

Parameter	Value
Population Size	100
Number of generations	200
Crossover probability	0.8
Mutation rate range	0.01 – 0.3 (adaptive)
Diversity metric	Hamming Distance
Selection strategy	Crowding Distance-Based NSGA-II
Encoding scheme	Binary (feature selection)
Dataset split	70% Training / 30% Testing

Accuracy, False Positive Rate (FPR), False Negative Rate (FNR), Detection Time (avg), and the number of selected features were used to evaluate the proposed method. Table IX shows a clear performance advantage of the proposed AGA-MOEA model over four widely used hybrid methods across all generations. Initially, in generation 20, AGA-MOEA starts with the highest accuracy (91.5%), exceeding ACO-ANN by over 1%. As the number of generations increases, the gap in performance becomes more pronounced. By generation 100, AGA-MOEA achieves 96.7%, surpassing ACO-ANN (93.0%) and significantly outperforming SVM-GA (91.4%) and DE-NN (91.3%). The superior accuracy trend of AGA-MOEA continues and saturates around generation 160 at 97.4%, maintaining this peak through generation 200. This plateau indicates successful convergence and stability of the model. In contrast, the other methods show slower improvements and earlier stagnation.

TABLE IX. ACCURACY COMPARISON OVER GENERATIONS

Generation	SVM-GA	PSO-RF	DE-NN	ACO-ANN	AGA-MOEA (Proposed)
20	88.4	89.1	87.6	90.3	91.5
40	89.2	90.4	88.7	91.1	93.2
60	90.1	91.5	89.9	91.8	94.6
80	91.0	92.2	90.7	92.6	95.8
100	91.4	92.8	91.3	93.0	96.7
120	91.9	93.0	91.9	93.4	97.1
140	92.1	93.2	92.3	93.7	97.3
160	92.3	93.4	92.6	93.9	97.4
180	92.5	93.5	92.8	94.0	97.4
200	92.5	93.5	92.9	94.1	97.4

The performance gain can be attributed to adaptive mutation, crowding distance preservation, and multiobjective evaluation in AGA-MOEA, which ensure continuous exploration and effective exploitation. Additionally, the algorithm's ability to balance false positives and false negatives

during fitness evaluation contributes to its superior generalization. Thus, AGA-MOEA not only accelerates learning but also achieves higher accuracy and robustness than its hybrid counterparts, making it a strong candidate for real-world attack detection applications.

AGA-MOEA consistently outperforms existing hybrid approaches in minimizing the FPR throughout the evolutionary process. In generation 20, it already achieves a lower FPR (5.5%) than all competitors, with SVM-GA and DE-NN showing particularly higher false alarms at 7.4% and 8.1%, respectively, as shown in Table X.

TABLE X. FPR% COMPARISON OVER GENERATIONS

Generation	SVM-GA	PSO-RF	DE-NN	ACO-ANN	AGA-MOEA (Proposed)
20	7.4	6.9	8.1	6.1	5.5
40	6.9	6.5	7.7	5.7	4.9
60	6.4	6.2	7.2	5.4	4.2
80	6.1	5.9	6.7	5.1	3.6
100	5.9	5.7	6.3	4.8	3.1
120	5.7	5.5	6.0	4.5	2.7
140	5.5	5.3	5.8	4.3	2.3
160	5.4	5.2	5.6	4.2	2.1
180	5.3	5.2	5.5	4.1	2.1
200	5.3	5.2	5.4	4.0	2.1

As evolution progresses, the adaptive nature of AGA-MOEA, driven by diversity-aware mutation and crowding distance selection, allows it to sharply reduce FPR. By generation 100, AGA-MOEA reaches 3.1%, substantially lower than PSO-RF (5.7%) and ACO-ANN (4.8%). The performance gains are sustained and refined, stabilizing around 2.1% by generation 160, whereas other methods either plateau higher or reduce more slowly. This improvement in FPR is crucial for IDS, where false alarms can overwhelm administrators or trigger unnecessary countermeasures. AGA-MOEA's multiobjective optimization ensures that it does not improve accuracy at the expense of higher FPR, which is a common flaw in many classifiers. Thus, the proposed method shows superior control over FPR due to its robust evolutionary design, making it a more reliable choice for real-world attack detection tasks where minimizing false alerts is vital.

TABLE XI. AVERAGE DETECTION TIME COMPARISON (MS)

Generation	SVM-GA	PSO-RF	DE-NN	ACO-ANN	AGA-MOEA (Proposed)
20	11.6	9.7	13.3	8.9	7.5
40	11.2	9.3	12.6	8.4	6.9
60	10.8	8.9	12.2	8.1	6.4
80	10.5	8.6	11.8	7.8	6.0
100	10.2	8.3	11.5	7.6	5.8
120	10.0	8.1	11.2	7.4	5.6
140	9.9	8.0	10.9	7.3	5.5
160	9.8	8.0	10.7	7.3	5.4
180	9.8	8.0	10.6	7.2	5.4
200	9.8	8.0	10.6	7.2	5.4

The proposed AGA-MOEA shows clear superiority in reducing the average detection time, making it not only accurate but also efficient. At generation 20, AGA-MOEA detects attack patterns in an average of 7.5 ms, significantly outperforming DE-NN (13.3 ms) and even advanced hybrids such as ACO-ANN (8.9 ms). Through its binary-encoded optimization and feature subset minimization, AGA-MOEA increasingly reduces computational overhead. By generation 100, it averages 5.8 ms, whereas the next best method, ACO-ANN, still lags at 7.6 ms. The detection time stabilizes at 5.4 ms by generation 160, showing convergence and runtime consistency. This performance gain is due to the adaptive mutation strategy and diversity preservation, which eliminate redundant or low-contribution features during evolution. As a result, the model operates on more compact and relevant feature subsets, reducing the decision-making load.

Fast detection is essential in real-time systems where delays can lead to undetected threats or slow mitigation. Hence, the efficiency of the proposed method alongside high accuracy and low false positives makes it a well-rounded solution for intelligent, real-time intrusion detection systems in software-defined or cloud-based environments.

The AGA-MOEA consistently selects fewer, yet more relevant, features compared to existing methods across all generations. Initially, it identified 21 key features from the dataset, fewer than any of the baselines, and continued refining this subset to 15 stable features by generation 100, maintaining this compact subset until generation 200, as shown in Table XII.

TABLE XII. NUMBER OF SELECTED FEATURES OVER GENERATIONS

Generation	SVM-GA	PSO-RF	DE-NN	ACO-ANN	AGA-MOEA (Proposed)
20	28	26	30	24	21
40	26	25	29	23	19
60	25	23	27	22	17
80	24	22	26	21	16
100	24	22	25	21	15
120	23	21	25	21	15
140	23	21	24	21	15
160	23	21	24	21	15
180	23	21	24	21	15
200	23	21	24	21	15

In contrast, methods such as DE-NN and SVM-GA select a higher number of features (25–30), indicating less effective feature optimization. Although ACO-ANN and PSO-RF show moderate feature reduction, they still cannot match AGA-MOEA's minimization capability. This improvement is due to AGA-MOEA's binary chromosome representation and multiobjective optimization, which not only optimize for accuracy but also penalize overly complex solutions. The adaptive mutation strategy, triggered by population diversity levels, helps escape local minima and find feature subsets that strike a balance between performance and efficiency. Reducing the number of features not only minimizes computational costs and model complexity but also improves generalization and detection speed. Therefore, AGA-MOEA offers a more lightweight, faster, and robust model, making it ideal for real-time deployment in high-throughput environments such as SDN-based intrusion detection systems or cloud security platforms [2].

As shown in Table XIII, the proposed AGA-MOEA shows significant improvement in minimizing FNR across all generations compared to existing hybrid methods. At generation 20, AGA-MOEA achieves an FNR of 7.1%, outperforming SVM-GA (9.6%) and DE-NN (10.4%) by a wide margin. This early advantage suggests the model's capacity to detect a greater number of actual attack packets from the onset.

TABLE XIII. FNR% COMPARISON OVER GENERATIONS

Generation	SVM-GA	PSO-RF	DE-NN	ACO-ANN	AGA-MOEA (Proposed)
20	9.6	8.9	10.4	8.3	7.1
40	8.8	8.2	9.7	7.5	6.3
60	8.1	7.6	9.1	6.9	5.5
80	7.5	7.1	8.6	6.4	4.9
100	7.2	6.8	8.2	6.1	4.5
120	7.0	6.6	8.0	5.9	4.2
140	6.9	6.5	7.8	5.8	4.1
160	6.9	6.5	7.7	5.7	4.0
180	6.9	6.5	7.7	5.7	4.0
200	6.9	6.5	7.7	5.7	4.0

By generation 100, AGA-MOEA reduces its FNR to 4.5%, whereas ACO-ANN (6.1%) and PSO-RF (6.8%) show slower improvements. The FNR plateaus at 4.0% by generation 160, demonstrating stability and convergence in correctly identifying malicious activities with minimal undetected instances. This superior performance results from the multiobjective evolutionary strategy, which directly incorporates FNR as a fitness criterion alongside accuracy and FPR. Furthermore, the adaptive genetic operations, guided by population diversity and crowding distance, help avoid overfitting and ensure robust detection. Low FNR is critical in cybersecurity applications where undetected threats can cause severe damage. The ability of AGA-MOEA to consistently reduce FNR while maintaining high accuracy and low FPR makes it a dependable model for IDS in sensitive and dynamic network environments.

Table XIV summarizes the experimental evaluations of different machine learning and optimization-based methods applied to four network intrusion detection datasets: NSL-KDD [22], CIC-IDS2017 [23], UNSW-NB15 [24], and TON_IoT [25]. All baseline methods (SVM-GA, PSO-RF, DE-NN, and ACO-ANN) were implemented based on the algorithmic specification as described in the literature [11, 26, 27]. All methods were subjected to the same preprocessing pipeline, identical stratified train validation test splits, the same computational environment, and evaluated using consistent metrics calculated from confusion matrices.

TABLE XIV. ACCURACY COMPARISON ACROSS DATASETS (TRAIN/TEST/VALIDATION IN %)

Method	NSL-KDD	CIC-IDS2017	UNSW-NB15	TON_IoT
SVM-GA	95.2/92.4/91.0	94.5/91.8/90.2	93.1/89.7/88.3	91.2/88.6/86.5
PSO-RF	96.0/93.1/91.8	95.3/92.5/90.8	93.9/90.4/89.0	92.1/89.4/87.3
DE-NN	94.7/91.3/89.5	93.8/90.2/88.6	92.4/88.9/87.0	90.5/87.1/85.2
ACO-ANN	96.5/94.0/92.6	95.7/93.4/91.2	94.6/91.2/89.6	93.0/90.3/88.4
AGA-MOEA	98.1/96.4/95.1	97.4/95.0/93.6	96.9/94.3/92.7	95.2/93.0/91.1

The proposed AGA-MOEA significantly outperformed existing hybrid methods in training, testing, and validation accuracy across all four DDoS datasets. On the NSL-KDD dataset, AGA-MOEA achieved 98.1% training, 96.4% testing, and 95.1% validation accuracy, clearly surpassing the best existing method (ACO-ANN at 96.5/94.0/92.6%). Similar trends were observed for the CIC-IDS2017 dataset, where AGA-MOEA maintained more than 95.0% testing accuracy and the highest generalization (93.6% validation accuracy), indicating robustness and minimal overfitting. For more challenging datasets, such as UNSW-NB15 and TON_IoT, which contain complex and noisy traffic patterns, AGA-MOEA still delivered superior performance, with testing accuracies of 94.3% and 93.0% respectively—about 2–3% higher than the next best models.

These results stem from AGA-MOEA's ability to optimally select discriminative features, reduce noise via adaptive mutation, and maintain a diverse yet convergent population of candidate solutions through crowding distance-based multiobjective selection. The model's structure allows it to generalize better across different network scenarios and adapt to various traffic types. Therefore, AGA-MOEA is proven to be a versatile and accurate model for DDoS detection in real-world SDN and IoT-driven environments, consistently achieving high reliability across multiple datasets.

IV. CONCLUSION

This study presents a novel AGA integrated with CDS within an MOEA framework to effectively optimize feature subsets and system configurations for DDoS attack detection. By dynamically adjusting mutation rates based on population diversity and employing crowding distance to maintain solution diversity, the proposed method achieves superior trade-offs among accuracy, FPR, and FNR. Extensive experiments on multiple benchmark datasets show the superiority of the proposed algorithm over existing hybrid methods, delivering higher detection accuracy, lower false alarm rates, and reduced computational times. The proposed approach's ability to converge to diverse Pareto-optimal solutions makes it highly adaptable to varying network conditions and attack scenarios. These results indicate that the proposed AGA-MOEA is a promising and scalable solution for real-time intrusion detection in modern, heterogeneous network environments, such as SDN and IoT. Future work will focus on extending the framework to address evolving attack patterns through online learning and deploying it in large-scale operational networks.

REFERENCES

[1] S. Wang, J. Liu, and Y. Jin, "A Computationally Efficient Evolutionary Algorithm for Multiobjective Network Robustness Optimization," *IEEE Transactions on Evolutionary Computation*, vol. 25, no. 3, pp. 419–432, June 2021, https://doi.org/10.1109/TEVC.2020.3048174.

[2] S. Wang, Y. Jin, and M. Cai, "Enhancing the Robustness of Networks Against Multiple Damage Models Using a Multifactorial Evolutionary Algorithm," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 7, pp. 4176–4188, July 2023, https://doi.org/10.1109/TSMC.2023.3241621.

[3] S. Zhou, M. Huang, Y. Sun, and K. Li, "Evolutionary Multi-objective Optimization for Contextual Adversarial Example Generation," *Proceedings of the ACM Software Engineering*, vol. 1, no. FSE, Apr. 2024, Art. no. 101, https://doi.org/10.1145/3660808.

[4] Y. Chen, Q. Lin, W. Wei, J. Ji, K. C. Wong, and C. A. C. Coello, "Intrusion detection using multi-objective evolutionary convolutional neural network for Internet of Things in Fog computing," *Knowledge-Based Systems*, vol. 244, May 2022, Art. no. 108505, https://doi.org/10.1016/j.knosys.2022.108505.

[5] N. Nissar, N. Naja, and A. Jamali, "Securing VANETs: Multi-Objective Intrusion Detection With Variational Autoencoders," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 3867–3874, Oct. 2024, https://doi.org/10.1109/TCE.2024.3372691.

[6] J. Wang, Z. Yin, J. Jiang, and Y. Du, "Attention-guided black-box adversarial attacks with large-scale multiobjective evolutionary optimization," *International Journal of Intelligent Systems*, vol. 37, no. 10, pp. 7526–7547, 2022, https://doi.org/10.1002/int.22892.

[7] J. Zhang, B. Gong, M. Waqas, S. Tu, and S. Chen, "Many-Objective Optimization Based Intrusion Detection for in-Vehicle Network Security," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 12, pp. 15051–15065, Sept. 2023, https://doi.org/10.1109/TITS.2023.3296002.

[8] S. Shao, Y. Tian, L. Zhang, K. C. Tan, and X. Zhang, "An Evolutionary Algorithm for Solving Large-Scale Robust Multi-Objective Optimization Problems," *IEEE Transactions on Evolutionary Computation*, 2024, https://doi.org/10.1109/TEVC.2024.3435006.

[9] A. Muneer, S. M. Taib, S. M. Fati, A. O. Balogun, and I. A. Aziz, "A Hybrid Deep Learning-Based Unsupervised Anomaly Detection in High Dimensional Data," *Computers, Materials & Continua*, vol. 70, no. 3, pp. 5363–5381, 2022, https://doi.org/10.32604/cmc.2022.021113.

[10] Y. K. Saheed, A. I. Abiodun, S. Misra, M. K. Holone, and R. Colomo-Palacios, "A machine learning-based intrusion detection for detecting internet of things network attacks," *Alexandria Engineering Journal*, vol. 61, no. 12, pp. 9395–9409, Dec. 2022, https://doi.org/10.1016/j.aej.2022.02.063.

[11] M. T. Tally and H. Amintoosi, "A hybrid method of genetic algorithm and support vector machine for intrusion detection," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 1, Feb. 2021, Art. no. 900, https://doi.org/10.11591/ijece.v11i1.pp900-908.

[12] M. Ajdani and H. Ghaffary, "Introduced a new method for enhancement of intrusion detection with random forest and PSO algorithm," *Security and Privacy*, vol. 4, no. 2, 2021, Art. no. e147, https://doi.org/10.1002/spy2.147.

[13] J. C. Huang, G. Q. Zeng, G. G. Geng, J. Weng, K. D. Lu, and Y. Zhang, "Differential evolution-based convolutional neural networks: An automatic architecture design method for intrusion detection in industrial control systems," *Computers & Security*, vol. 132, Sept. 2023, Art. no. 103310, https://doi.org/10.1016/j.cose.2023.103310.

[14] K. P. Chandrasekaran and V. M. Chidambaram, "Integrating Novel Mechanisms for Threat Detection in Enhanced Data Classification using Ant Colony Optimization with Recurrent Neural Network," *Journal of Cybersecurity and Information Management*, vol. 14, no. 2, pp. 132–147, 2024, https://doi.org/10.54216/JCIM.140209.

[15] R. R. N. A. Ogaili *et al.*, "AntDroidNet Cybersecurity Model: A Hybrid Integration of Ant Colony Optimization and Deep Neural Networks for Android Malware Detection," *Mesopotamian Journal of CyberSecurity*, vol. 5, no. 1, pp. 104–120, Feb. 2025, https://doi.org/10.58496/MJCS/2025/008.

[16] T. Chen and M. Li, "The Weights Can Be Harmful: Pareto Search versus Weighted Search in Multi-objective Search-based Software Engineering," *ACM Transactions on Software Engineering Methodology*, vol. 32, no. 1, Oct. 2023, Art. no. 5, https://doi.org/10.1145/3514233.

[17] I. Rahimi, A. H. Gandomi, M. R. Nikoo, and F. Chen, "A comparative study on evolutionary multi-objective algorithms for next release problem," *Applied Soft Computing*, vol. 144, Sept. 2023, Art. no. 110472, https://doi.org/10.1016/j.asoc.2023.110472.

[18] X. Xu, C. Xie, Z. Luo, C. Zhang, and T. Zhang, "A multi-objective evolutionary algorithm based on dimension exploration and discrepancy evolution for UAV path planning problem," *Information Sciences*, vol. 657, Feb. 2024, Art. no. 119977, https://doi.org/10.1016/j.ins.2023.119977.

[19] T. Yang and Y. Zhou, "Analysis of Multiobjective Evolutionary Algorithms on Fitness Function With Time-Linkage Property," *IEEE Transactions on Evolutionary Computation*, vol. 28, no. 3, pp. 837–843, June 2024, https://doi.org/10.1109/TEVC.2024.3371519.

[20] R. Basfar, M. Y. Dahab, A. M. Ali, F. Eassa, and K. Bajunaied, "Enhanced Intrusion Detection in Software-Defined Networking using Advanced Feature Selection: The EMRMR Approach," *Engineering, Technology & Applied Science Research*, vol. 14, no. 6, pp. 19001–19008, Dec. 2024, https://doi.org/10.48084/etasr.9256.

[21] D. K. Singh and M. Shrivastava, "Evolutionary Algorithm-based Feature Selection for an Intrusion Detection System," *Engineering, Technology & Applied Science Research*, vol. 11, no. 3, pp. 7130–7134, June 2021, https://doi.org/10.48084/etasr.4149.

[22] "Intrusion Detection System [NSL-KDD]." Kaggle, [Online]. Available: https://kaggle.com/code/eneskosar19/intrusion-detection-system-nsl-kdd.

[23] "IDS 2017." Canadian Institute for Cybersecurity, [Online]. Available: https://www.unb.ca/cic/datasets/ids-2017.html.

[24] "UNSW-NB15." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/dhoogla/unswnb15.

[25] "Ton-IoT." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/amaniabourida/ton-iot.

[26] A. K. Shukla and A. Sharma, "Reduce false intrusion alerts by using PSO feature selection in NSL-KDD dataset," *IET Conference Proceedings*, vol. 2023, no. 5, pp. 226–231, July 2023, https://doi.org/10.1049/icp.2023.1495.

[27] K. Rithesh, A. V. Gautham, and K. Chandra Sekaran, "Network Anomaly Detection Using Artificial Neural Networks Optimised with PSO-DE Hybrid," in *Security in Computing and Communications*, vol. 969, S. M. Thampi, S. Madria, G. Wang, D. B. Rawat, and J. M. Alcaraz Calero, Eds. Springer Singapore, 2019, pp. 257–270.