Machine Learning Driven Threat Identification to Enhance FANET Security using Genetic Algorithm

Shikha Gupta
Department of Computer Science and Engineering, Engineering College, India
shikhagupta@ecajmer.ac.in

Neetu Sharma
Department of Computer Science and Engineering, Engineering College, India
drneetu@ecajmer.ac.in

Abstract: *Emerging as a highly promising technology, Flying Ad-hoc Networks represent self-organizing networks of Unmanned Aerial Vehicles (UAVs), garnering attention for their diverse applications spanning environmental monitoring, disaster management, precision agriculture, surveillance, and military operations. However, these networks face challenges to various security threats, including malicious node detection due to their deployment in dynamic environments. To address this issue, we present an improved novel security solution, Machine Learning-based Threat Identification for FANET using a Genetic Algorithm (ML-TIFGA) in this paper. The research includes the detection of abnormal behavior nodes using a basic genetic algorithm and dynamically adapting the changing network conditions by utilizing a reputation system. To enhance our security solution ML-TIFGA, we evaluated two key factors: cooperation and trustworthiness, which act as genetic elements within the chromosome of the flying node in our genetic population. Further, a mechanism is incorporated to reconfigure the trust, addressing the challenge of dynamically extracting threats through the updated weighted reputation system while considering past behavior monitoring. Significant improvements were found in the experimental results using actual sample values from the NSL-KDD dataset, which produced a remarkable 99.829% classification accuracy. Additionally, threat identification rates reached 98.36% for training and 98.86% for testing samples, with a remarkable improvement of 99.3% in network reliability through ML-TIFGA. When benchmarked against state-of-the-art approaches, performance metrics such as delay, throughput, and data delivery rate exhibited notable enhancements of 24.65%, 29.16%, and 31.73%, respectively.*

Keywords: *FANET, threats, genetic, trust, reputation, network reliability, machine learning.*

Received March 30, 2024; accepted July 17, 2024
https://doi.org/10.34028/iajit/21/4/12

1. Introduction

Flying Ad Hoc Networks (FANET) are characterized by their dynamic topology, mobility, and potential applications in various domains such as surveillance, search and rescue, environmental monitoring, and communication relays. These networks are often deployed where traditional communication infrastructure is unavailable, impractical, or costly to establish, ranging from disaster management and surveillance to precision agriculture and delivery services [26]. Figure 1 shows the basic FANET scenario where at least one flying node must directly contact the ground control unit. In contrast, others can communicate through ad hoc in their communication range to provide multiple services to end-users.

FANETs present several challenges, including security concerns related to data confidentiality, integrity, and authentication [17]. Ensuring the security of these networks against malicious and selfish nodes is paramount due to several critical reasons [22, 28]. These attacks can severely compromise the integrity, availability, and confidentiality by packet dropping, data injection, and routing disruptions leads, to the potential data breaches or service disruptions, including aerial surveillance, disaster management, and military operations, where the accuracy and reliability of information exchange are of utmost importance [12]. Therefore, robust security mechanisms must be in place to detect, isolate, and mitigate the impact of malicious and selfish nodes, ensuring the reliability and efficiency of ad hoc networks, including vehicular communication [3, 20, 21].

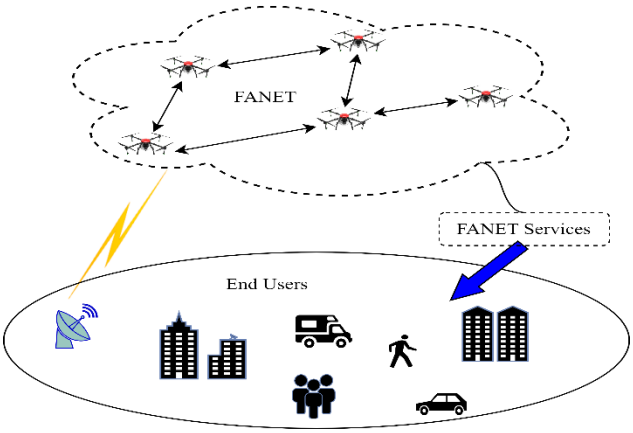

Figure 1. Essential FANET communication and service scenarios.

Application of Genetic Algorithms can excel in securing ad hoc networks by continuously optimizing security parameters and strategies in response to

changing network conditions and evolving threats [9, 14].

Machine learning significantly improves ad hoc network security, which offers vital capabilities for anomaly detection, threat detection, and adaptive defense mechanisms [25]. Because of the constantly shifting network topology and potentially malicious nodes, traditional security measures frequently fail in the dynamic and decentralized environment of ad hoc networks [2]. Large volumes of network data can be analyzed in real-time by machine learning algorithms, which can then be used to spot anomalies that might be signs of security threats and to identify patterns of typical behavior [15].

Integration of machine learning with a genetic approach to securing flying ad hoc networks enhances their resilience. It enables efficient and effective protection against evolving cyber threats in dynamic networking environments.

To grasp the significance of our work, we have summarized the key findings and challenges in solving the Machine Learning-based Threat Identification for FANET using a Genetic Algorithm (ML-TIFGA) problem and conducted a comparative study of existing technologies. The novelty of our approach lies in its design and implementation, a combination of co-evolutionary genetic algorithms and GA-based reputation systems, to enhance the security of FANETs. In this innovative strategy, Co-evolutionary GAs operate on two concurrent populations: one representing normal node behavior and the other capturing potentially malicious activity to adapt and evolve detection and mitigation strategies. Simultaneously, the GA-Based Reputation System evaluates node reputations based on observed behavior and interactions within the network. By integrating the outcomes of both approaches, nodes with suspicious behavior can be flagged. This combined approach with Machine Learning techniques enables FANETs to detect and mitigate threats posed by malicious nodes effectively. Further, we introduce a trust framework based on the updated reputation for classifying flying nodes into malicious, selfish, or normal nodes. Through comprehensive simulations, we demonstrate the significance of ML-TIFGA over the state-of-art, proving its effectiveness in improving the security of FANET.

The immediate requirement to improve the security environment of flying ad hoc networks is the motivation behind the proposal of the innovative ML-TIFGA technique, as many real-time services and national security aspects rely on this communication system. It provides unparalleled communication and data exchange abilities, but its dynamic nature makes it more vulnerable to malicious attacks, which could result in numerous significant issues.

The key contributions of the research article are:

1. Design and Development of a Genetic Model: we created a novel genetic model to simulate the behavior and interactions of flying nodes, enabling the accurate emulation of network dynamics and node communication patterns.
2. Development of a Genetic Algorithm: we developed and implemented adaptive genetic algorithms to detect and mitigate security threats in evolving network conditions.
3. Design of a Trust Model: we developed a trust model using dynamic reputation scores to classify nodes as malicious, selfish, or normal based on their behavior and interactions.
4. Development of the ML-TIFGA Interaction Model:. we developed a detailed mathematical model ML-TIFGA, detailing the interactions between genetic algorithms and machine learning techniques.
5. Implementation of ML-TIFGA: we used real-time datasets to implement the ML-TIFGA, combining machine learning for threat detection and mitigation.
6. Performance Evaluation: we evaluated and compared the ML-TIFGA with state-of-the-art protocols, demonstrating significant improvements in security and efficiency validating the effectiveness of our approach.

The rest of the paper is structured as follows: Section II reviews existing literature on security techniques in FANETs. Section III outlines the methodology used in our research. Section IV discusses our findings and experimental results, followed by a conclusion and future directions in Section V.

2. Literature Review

Methods proposed [10, 23, 30, 32] were based on trust models for security in UAV networks utilizing the genetic algorithms to optimize the weighting of parameters for assessing direct trust values, determining the overall trustworthiness of a node by combining direct trust with commendation and conducting risk analysis for uncertain nodes. A trust mechanism that integrates cluster and genetic approaches is also presented, with the Fish Swarm technique enhanced artificially, facilitating cluster leader selection. This mechanism uses a Bayesian network theory-based approach to compute direct and indirect trust and optimize secure route discovery, evaluated by metrics such as Packet Delivery Ratio (PDR), delay, throughput, and network overhead. Additionally, the architectural framework of FANETs focuses on machine learning-enhanced intrusion detection systems to distinguish between normal and abnormal data packets and improved network security with a novel cluster-based strategy and a fuzzy model that dynamically calculates node trust levels. Schemes utilizing optimization for FANET security through natured-inspired algorithms are proposed in [19], and the factors influencing FANET throughput are examined, resulting in a mathematical optimization model using genetic algorithms, fitness functions, and chromosome

replication to update UAV positions based on adjacency and correlation matrices. A hybrid approach combining genetic algorithms with the fruit fly optimization algorithm addresses energy consumption, employing a cluster-based and density-adaptive method to determine cluster membership and optimize routes. Threat identification and cryptanalysis in [6, 11, 18] through analyzing packet drop rates and message content, using a collective model to assess node trustworthiness and isolate malicious nodes. Efficient network paths and enhanced communication between flying nodes are achieved by integrating two-ray and shadow-fading models with the genetic firefly algorithm, improving security. Optimization results are compared for packet loss, end-to-end delay, and network performance. Additionally, a cryptanalysis of an existing scheme incorporates new algorithms to ensure message confidentiality and integrity. The importance of machine learning techniques [1, 31] in detecting irregularities within UAV groups suggests that the timely detection of anomalies should given priority. The development of IDS to secure FANET based on machine learning is explained in [7, 24] by creating a cognitive lightweight-LR method utilizing the UNSW-NB 15 dataset. An Internet of Things-based UAV network was subjected to machine learning to detect possible security risks. Furthermore, a recommended method investigates deploying UAV systems in wireless networks for agricultural data security, utilizing the Double Deep Q-Network (DDQN) algorithm based on geographic position data to identify intrusions and establish deployment locations. This technique simplifies the intricate calculations of channel state information, ensuring safe UAV deployment. A method using Reinforcement Learning to detect jamming attacks is outlined [8]. Minimizing gradient variance and ensuring safe training regions enhances accuracy and speeds up training. This approach effectively detects and mitigates jamming attacks. Moreover, Shitharth *et al*. [29] improves device-to-device applications by enhancing UAV transmission models between network nodes. This approach boosts device security during real-time data transmission, strengthening overall UAV network reliability and integrity.

Therefore, with the unique aspects of a combination of Machine Learning, Trust Model, and Genetic Algorithms, we proposed the novel technique ML-TIFGA in the next section to enhance FANET security.

3. Proposed Methodology

The Genetic Algorithms (GAs) apply evolutionary principles to optimize different aspects of network security, providing a potent method for addressing security concerns in FANET. The adaptive nature of GAs allows them to explore a wide range of potential solutions and adaptively converge towards optimal or near-optimal solutions, including applying multiple GA approaches parallel to solve the problem space. Considering these features of GAs, the combination of GA-Based Reputation Systems and Co-evolutionary Genetic Algorithms in the suggested approach offers a potent way to improve the security of FANET by early and effective identification and separation of malicious nodes from the network. In ML-TIFGA, we influence a refined combination of a Reputation System and a Co-Evolutionary Genetic Algorithm to fortify the security in FANET. Additionally, we introduce a Trust-based framework meticulously designed to distinguish and isolate malicious nodes from their regular counterparts. Central to this framework is using genetic reputation and computed reputation metrics for every flying node. These metrics serve as crucial inputs for our trust model, facilitating the continuous real-time updating and refinement of node reputations. By seamlessly integrating genetic algorithms with reputation-based trust mechanisms, the ML-TIFGA ensures heightened security levels. In the proposed ML-TIFGA model, we consider each flying node's cooperation and trustworthiness factors to enable the reputation criteria.

Figure 2 represents the workflow steps for ML-TIFGA and explains as follows.

- *Step* 1. Initializing all the FANET nodes based on the defined previous reputation system. Defining reputation (*R*) criteria for nodes in a FANET involves quantifying various aspects of node behavior and interactions within the network.
- *Step* 2. Upon allocating reputation values using the Genetic approach, the Co-Evolutionary process is employed to evolve two populations simultaneously.
- *Step* 3. The flying nodes are divided into groups based on a fitness function distinguishing malicious activity and normal node behavior.
- *Step* 4. The Trust-based framework is applied to target nodes that do not meet the criteria for fitness qualification within the designated group. Reputation updating of such nodes based on current cooperation and trustworthiness is implemented.
- *Step* 5. If the updated Reputation (R') of any node (N) from Step 4 is more than the previous value, such nodes are categorized within the group of nodes exhibiting normal behavior; otherwise, nodes are considered in the group of malicious behavior nodes.

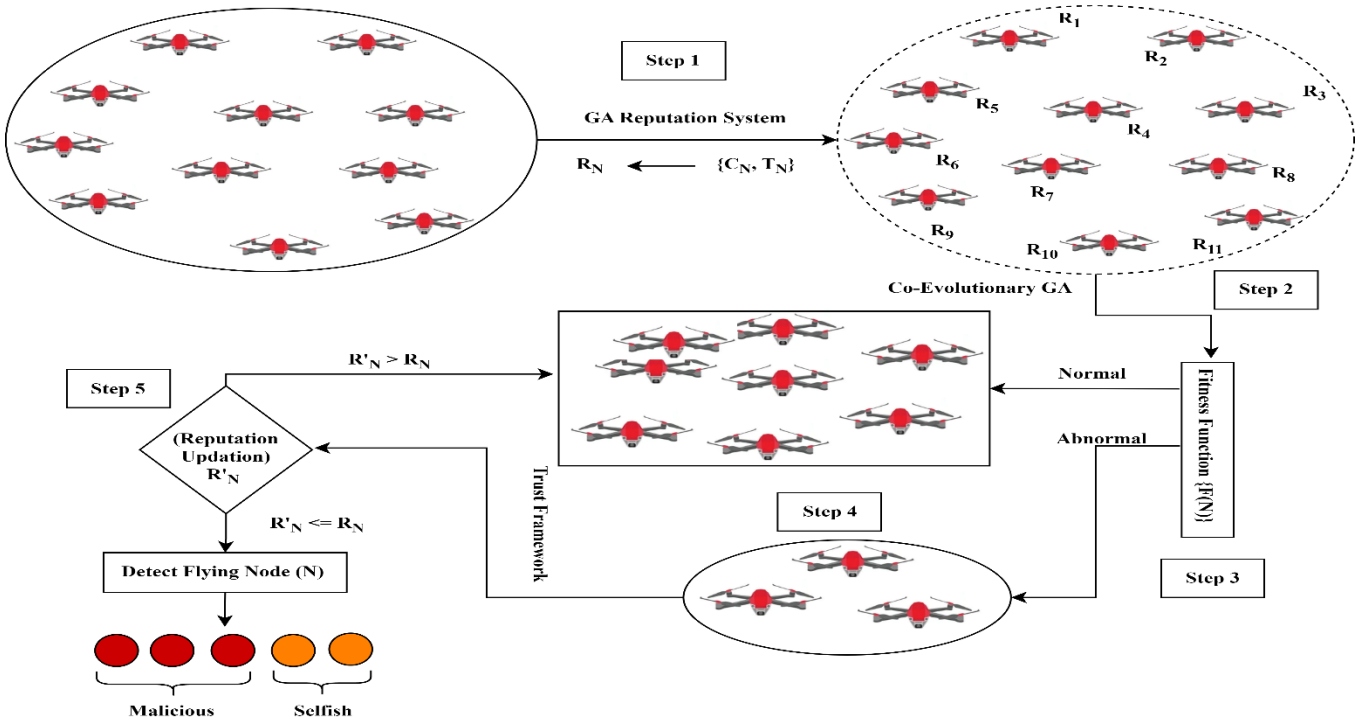

Figure 2. Workflow of the proposed approach.

3.1. GA-based Reputation Analysis

Employing a reputation system where nodes maintain reputations for their past behavior and use GA to evolve the criteria for calculating these reputations. Nodes with suspicious or malicious behavior can be flagged based on deviations from the evolved reputation criteria. In the ML-TIFGA model, each flying node's cooperation (C) and trustworthiness (T) factors are initially defined to enable the reputation criteria. Cooperation C_A for any node A is measured based on a node's willingness to participate in network tasks or share resources with other nodes as defined in (1).

$$C_A = {}^{X}/_{Y} \; where \; A \in [1 \; to \; N] \tag{1}$$

X is the number of times a node willingly cooperates with others, and Y is the number of opportunities for cooperation.

Similarly, multiple aspects of node behavior measure a node's trustworthiness in the network. They are calculated as a weighted sum of reliability (Q), cooperation computed previously (C), and consistency (K). Trustworthiness (T_A) of node A as given in (5) for each factor considered weight (w) for trust computation. Q is the measure of the trustworthiness of a flying node based on its past behavior and performance. It indicates how dependable a node is in consistently performing its tasks without failure or malicious intent. Its current measure includes Task Completion Rate (TCR), Error Rate (ER), and Resource utilization (RU). Reliability for any Node i is defined by (2) for respective weights w_1, w_2, and w_3 that sum to 1 representing the importance of each factor.

$$Q_i = w_1 * TCR_i + w_2 * (1 - ER_i) + w_3 * RU_i \tag{2}$$

K is the uniformity and predictability of a flying node's behavior over time and reflects how consistently the node adheres to expected operation patterns and maintains its performance standards. It is calculated based on variance or standard deviation over time of key performance metrics. K for any flying node (i) is given by (3), where σ_{ij} is the standard deviation of performance metric j for node i over a given period.

$$K_i = {}^{1}/_{(1 + \sigma_{ij})} \tag{3}$$

Initially, we assign equal weight to each factor, assuming a 50% probability of a node being malicious. Therefore, $w_R = 0.5$, $w_C = 0.5$, and $w_K = 0.5$. Subsequently, after each stage, the weights are adjusted based on feedback regarding the performance of the initial weights, with adjustments made according to the learning rate (α) and the partial derivative of the factors for Trustworthiness (T). The learning rate is based on the system's accuracy in identifying malicious nodes. Below in (4) is the updated weight (w_U) for the previous weight (w_P).

$$w_U = w_P + \alpha . \delta T / \delta w \tag{4}$$

$$T_A = \{w_R * Q + w_C * C + w_K * K\} \tag{5}$$

Considering the computation of C_A and T_A, the reputation (R_A) for A within the network for stages S is determined according to (6).

$$R_A = \left[\sum_{S=1}^{Stages} [C_{A_S} + T_{A_S}] * \left(\int_{S=1}^{stage-1} {}^{1}/_{R_{A_S}}\right)\right] \tag{6}$$

Figure 3 represents the computation of the reputation of the A^{th} node in the FANET network at the S stage.

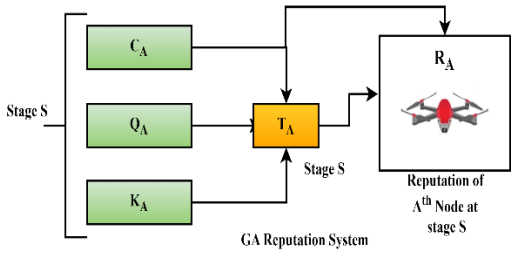

Figure 3. Reputation computation for a node.

3.2. Genetic Chromosome Method for Nodes

Encoding cooperation and trustworthiness as genes involves representing these attributes as binary within the chromosomes of individuals in the GA population. In binary encoding, *C* and *T* are defined as sequences of binary digits [0,1] within the chromosome. Each gene corresponds to a specific aspect of cooperation or trustworthiness, and its value indicates whether the node possesses that attribute. Further, the three aspects specifically for formulating Cooperation by a node in network activities are considered in ML-TIFGA. These include the willingness to share resources (*W*), the ability to forward messages (*V*), and participation in network tasks (*U*). Figure 4 shows the binary sequence of chromosomes for cooperation from genes {*W, V, U*}. Each gene of node *A* from the previous value $\{A1\}_{[W,\,V,\,U]}$ maps binary value and further these maps to following previous values $\{A2\}_{[W,\,V,\,U]}$ to generate the chromosome sequence through hidden layer *1* and *2* respectively which includes intermediate populations, genetic operators, or other forms of intermediate data structures that are not directly observable but play a crucial role in the evolutionary process of finding optimal solutions. Equations (7) to (9) represent the formation of chromosomes from cooperation genes. The role of gene function is to encode specific solution components, influence genetic variation through crossover and mutation, and guide the evaluation of fitness, ultimately shaping the evolutionary path toward optimal solutions in genetic algorithms.

$$Initialization\,;Genes \xleftarrow{[W,U,V]} C_A \tag{7}$$

$$Output_{Level1} \xleftarrow{Hidden\ Layer\ 1} Genes\ (W_{A1}, U_{A1}, V_{A1}) \tag{8}$$

$$Chromosome\ (A) \xleftarrow{Hidden\ Layer\ 2} Output_{Level1} \tag{9}$$

Algorithm 1: Chromosome Sequence Method

Step 1: Initialize Population P, t=0;

Step 2: I ← Individual (P);

Step 3: Evaluate fitness value (F) for each I;

Step 3: For all I ∈ P do
Parent Selection (I) [Highest F];
Reproduction (I)
Conduct Crossover;
Conduct Mutation of I;
Modify P;
t = t + 1;

Step 4: Return Chromosome sequence;

The Co-evolutionary approach generates the two population groups for nodes concurrently. One represents normal, and the other represents malicious behavior. The fitness of each population is evaluated based on its ability to outperform the other population. The design choices for the fitness function and genetic operations in the ML-TIFGA are grounded in established genetic algorithm principles and supported by extensive literature demonstrating their efficacy in dynamic and complex problem environments. The fitness function was selected to accurately assess each solution's effectiveness in enhancing security by detecting malicious nodes and minimizing false positives. This function ensures that the most optimal solutions are preferentially selected for reproduction. The genetic operations, including crossover and mutation, were chosen to balance exploration and exploitation. Crossover combines parent traits to find high-quality solutions, while mutation adds random variations to maintain diversity and prevent premature convergence. These operations ensure the robustness and adaptability of the ML-TIFGA.

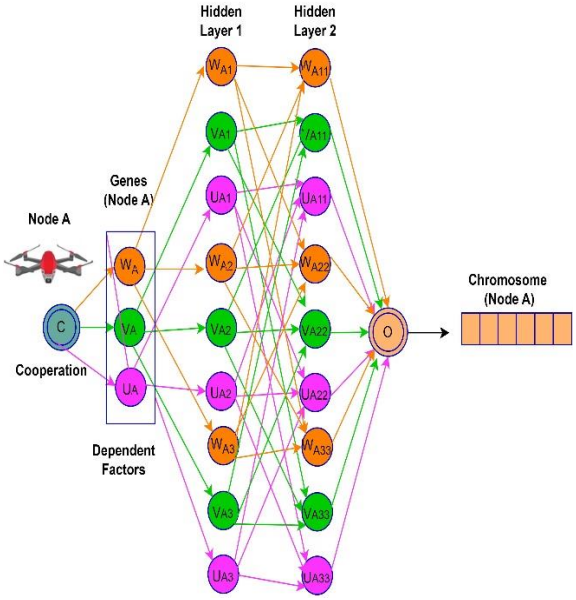

Figure 4. Genes to chromosome through node cooperation.

Upon allocating reputation values using the genetic approach, the co-evolutionary process is employed to simultaneously evolve the distinguishing groups based on a fitness function. Equation (10) defines the fitness function (*F*) based on reputation values (*R*) for *w* weight assigned to the reputation value. It can be adjusted based on the desired balance between reputation.

$$F_A(t) = \left[\left(\int_{time=0}^{t}\int_{w=0}^{w=1} w.R_A.(t)\right) * \frac{1}{R_A(t-1)}\right] \tag{10}$$

Algorithm (1) represents the basic steps involved in chromosome generation for the network of population *P*

based on their fitness value for reproduction through mutation and crossover. Chromosomes are chosen during development from several different combinations that the population has. The individual with the greater fitness value will likely be opted for repeatedly during growth. Each chromosome integrates its genes during the crossover phase to create a new population with both characteristics. To keep a group of chromosomes from becoming remarkably similar, mutations protect genetic variation from one generation to another. FTH is defined as employing the group for flying nodes for the fitness threshold. Further, in ML-TIFGA, we consider the selection of nodes to respective groups through the rank-based selection method as given in Algorithm (2).

3.3. Node Trust Model

This framework includes the computation of trust for each node from both the genetic population groups to enhance accuracy and efficiency. Before the calculation of trust, an updated model for a reputation system in a network involves determining how the reputations adjusted over time based on node interactions and observed behavior based on continuous monitoring of interactions between nodes in the network, such as message exchanges, resource sharing, or task participation even after group's formation. Feedback is collected from interactions to evaluate node behavior. This feedback includes successful message delivery, completion of tasks, or cooperation in resource sharing. Hence, an updated reputation is proposed to overrule the chance of error in node identification. The previous reputation R_A of node A is based on successful message exchanges and updates to R'_A using a weighted average approach. Equation (11) shows the message counts after the formation of groups and the updated reputation R'_A for weight p of feedback. $F_{Message}$ is the feedback value based on successful message exchange (12).

$$F_{Message}(A) = \frac{Successful_{Message}(A)}{Transmitted_{Message}(A)} \tag{11}$$

$$R'_A = \left[\int_{p=0}^{p=1} [p * F_{Message} + R_A(1-p)]\right] \tag{12}$$

Table 1 represents the assignment of group and trust organization representing the node behaviour for node A before and after applying the proposed genetic approach. Algorithm (3) defined the trust framework to detect untrusted nodes and to separate them into malicious and selfish nodes. Malicious nodes are driven by harmful intent and engage in deliberate attacks. In contrast, selfish nodes prioritize their interests but may not necessarily aim to cause direct harm to the network.

Algorithm 2: Genetic Method For Node Groups'

Input : Nodes (N), F_N, F_{TH}

Output : $Node_{Normal}[\ ]$, $Node_{Abnormal}[\ ]$

function Node Selection( )

Initialization

$Node_{Normal}[\ ] = NULL$

$Node_{Abnormal}[\ ] = NULL$

Cumulative Probability (CmP) = 0

Index = 0

Total Fitness and Normalization

$T_{Sum} = \sum_{i=1}^{N} F(i)$

forall k ϵ N do

$F_N[k] = F(k)/T_{Sum}$

Sort $F_N[k]$ in Descending Ranks

forall n ϵ range(N) do

Random number generation [0, 1]

Num_{Random} = Random([0, 1])

Node identification

while CmP <= Num_{Random} do

CmP = Cmp + F_N[Index]

if ($F_N > F_{TH}$) then

$Node_{Normal}[\ ] = Node_{Normal}[\ ] \cup F_N[Index]$

else

$Node_{Abnormal}[\ ] = Node_{Abnormal}[\ ] \cup F_N[Index]$

Index = Index + 1

Return($Node_{Normal}[\ ]$, $Node_{Abnormal}[\ ]$)

Table 1. Trust evaluation for population assignment.

Previous Reputation (R_A)	Classification (A)	Group	Genetic Assignment	
			Updated Reputation (R'A)	Updated Group
R_A < =0.4	Untrusted	Abnormal	$R'_A > R_A$	Normal
0.4<R_A < 0.7	Verification Required	Normal	$R'_A <= R_A$	Abnormal
R_A >= 0.7	Trusted			

4. Performance Analysis

The experimental and different simulation setups for FANET to implement the various proposed algorithms and methods are included in this section. Additionally, particular metrics and criteria have been developed to assess how well the suggested technique performs in a simulation setting designed to improve secure communication.

4.1. Experiment Setup

To evaluate the performance of the ML-TIFGA method, we conduct three experiments. Experiment 1 is carried out to measure the accuracy, precision, recall, and F1-score with the variation in chromosome sequences on the NSL-KDD dataset that derived from the previous Knowledge Discovery and Data Mining (KDD) Cup 99 dataset [33] by Network Security Laboratory (NSL), the NSL-KDD dataset addresses essential issues that previously affected intrusion detection accuracy due to many duplicate packets. The size of the NSL-KDD train dataset is manageable for full use without random

sampling, with 125,973 records, while the test dataset has 22,544 records. It has 41 attributes and 22 different training intrusion attack types, offering a solid foundation for consistent and comparable findings across numerous research projects [5]. We included the cleaned sample, size 133000, with a learning rate of 0.05 for the sigmoid activation function. The classification of the sample is shown in Table 2 with 03 class labels. Google Colab tool is utilized to analyse the results of the proposed approach.

Table 2. Working dataset and class labels.

Labels	Group	Sample Filtering	
		Training (80%)	Testing (20%)
Normal Nodes	Normal	57951	14507
Selfish Nodes	Abnormal	17534	4382
Malicious Nodes		30901	7725

Table 3. Class detection by ML-TIFGA for 80% of training data.

Labels	Accuracy	Precision	Recall	F1-Score
Normal Nodes	99.726	99.586	99.785	99.685
Selfish Nodes	99.831	99.627	99.814	99.721
Malicious Nodes	99.93	99.709	99.796	99.752
Mean (%)	99.829	99.641	99.798	99.719

Tables 3 and 4 represent the performance proposed for 80% of training and 20% of testing data, respectively, by improving accuracy, precision, recall, and F1-score in the mean by 99.829%, 99.641%, 99.798%, and 99.719% for training data. Similarly, 99.856%, 99.759%, 99.835%, and 99.797% for testing data. Figure 5 shows the comparison analysis for the mean detection rate for class labels for training and testing data.

Table 4. Class detection by ML-TIFGA for 20% testing data.

Labels	Accuracy	Precision	Recall	F1-Score
Normal Nodes	99.833	99.735	99.854	99.794
Selfish Nodes	99.852	99.816	99.864	99.839
Malicious Nodes	99.884	99.726	99.787	99.756
Mean (%)	99.856	99.759	99.835	99.797

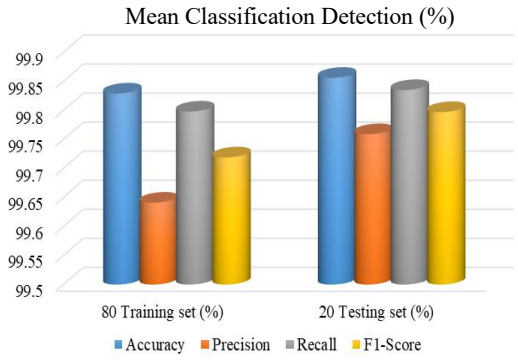

Figure 5. Classification detection for sample dataset.

Further, the comparison analysis of ML-TIFGA similar class labels with random forest [27], FA-ML [16], and Particle Swarm Optimization (PSO) with K-Nearest Neighbour (KNN) [34] techniques for performance metrics is shown in Tables 5 and 6, and their respective comparative analysis is depicted in Figures 6 and 7.

Table 5. Comparative analysis for 80% training data.

Labels	Accuracy	Precision	Recall	F1-Score
ML-TIFGA	98.36	98.01	97.86	97.934
Random Forest	95.15	93.47	93.22	93.344
FA-ML	97.08	95.86	94.37	95.109
PSO with KNN	97.86	96.24	95.79	96.014

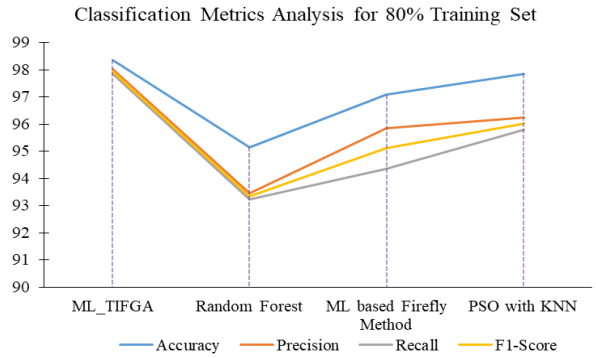

Figure 6. Classification metrics comparative analysis for training sample.

Table 6. Comparative analysis for 20% testing data.

Labels	Accuracy	Precision	Recall	F1-Score
ML-TIFGA	98.86	99.17	98.65	98.719
Random Forest	95.85	94.73	95.36	96.106
FA-ML	97.92	97.35	96.65	96.998
PSO with KNN	98.44	97.83	97.13	97.478

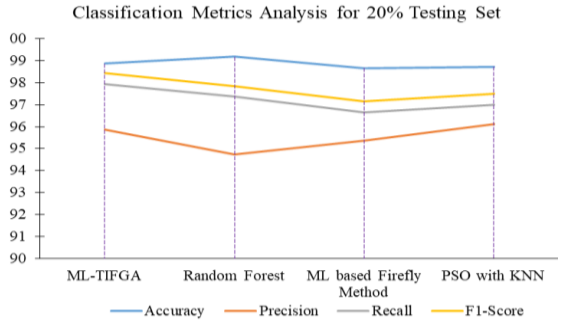

Figure 7. Classification metrics comparative analysis for testing sample.

Through Experiment 2, we monitor the reliability of the existing network of flying units, considering the success rate of node classification, along with time and population. For this, we consider the 5% and 20% injection of selfish and malicious nodes of population size. A maximum of 1000 population sizes (flying nodes) are considered for analysis in this scenario. Table 7 represents the various time and population instances considered to evaluate the network reliability by injecting the number of malicious and selfish units corresponding to total nodes. Based on the experiment, the improved network reliability by enhancing the identification of node types with the rise in time and node population through the ML-TIFGA is shown in Table 8. The same is depicted in Figure 8.

Table 7. Consider instances for network reliability.

Time (Seconds)	Total Nodes	Normal Nodes	Malicious Nodes	Selfish Nodes
20	50	38	10	2
40	200	150	40	10
60	400	300	80	20
80	600	450	120	30
100	800	600	160	40
120	1000	750	200	50

Table 8. Achieved Network reliability by ML-TIFGA.

Time (Secs)	Total Nodes	Normal Nodes	Malicious Nodes	Selfish Nodes	Achieved Network Reliability
20	50	37	9	2	0.96
40	200	146	39	9	0.97
60	400	294	77	19	0.975
80	600	448	119	28	0.991
100	800	597	157	38	0.99
120	1000	746	198	49	0.993

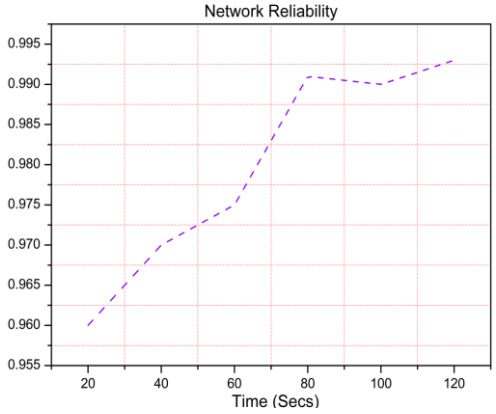

Figure 8. Network reliability through ML-TIFGA.

Further, the research is extended to analyze the performance of various network factors, including delay, throughput, and data delivery rate for assessment by proposed. a comparative study between the ml-tifga and two other state-of-the-art models, a Secure Energy Efficient Dynamic Routing Protocol (SEEDRP) [4], for energy efficient security and based on Fuzzy Trust Based Secure Routing (FTSR) [13] is performed through experiment 3 with the variation in both simulation time and abnormal nodes considering the simulation parameters and their respective values as shown in Table 9. to evaluate metrics for efficiency, a range of malicious and selfish nodes between 10 and 20 and 5 and 10, respectively, are considered in the network having 250 total nodes. results for the mentioned parameters through the conduction of the experiment are shown in Table 10, and the respective comparison graphs with state-of-art are shown in Figures 9 to 14.

Table 9. Simulation parameters.

Parameters	Values
Flying Nodes	250
Flying Region	[1000 X 1000] m^2
Time	300 seconds
Simulations	10
Node Speed	[20-80] mph
Transmission Range	250 meters
Message Size	2 KB
MAC Layer	802.11p

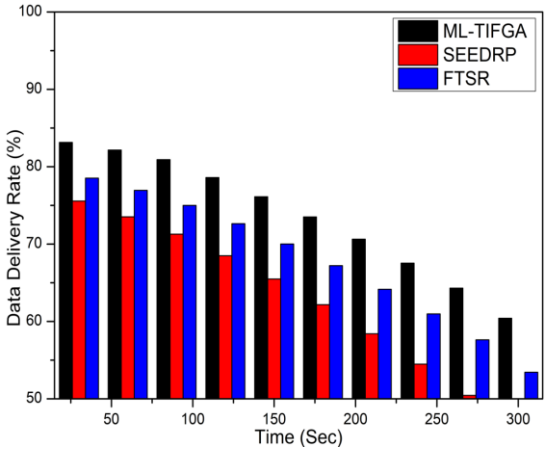

Figure 9. Comparative analysis for data delivery rate with time.

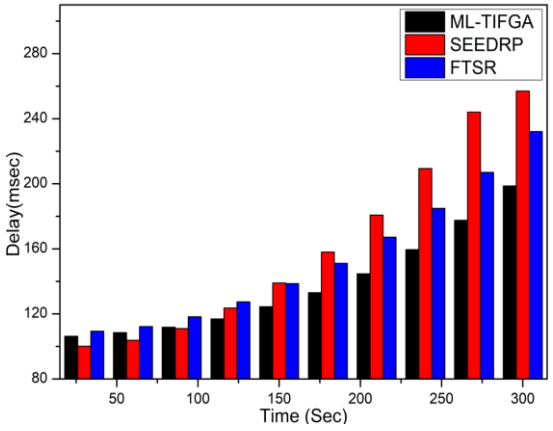

Figure 10. Comparative analysis for delay with time.

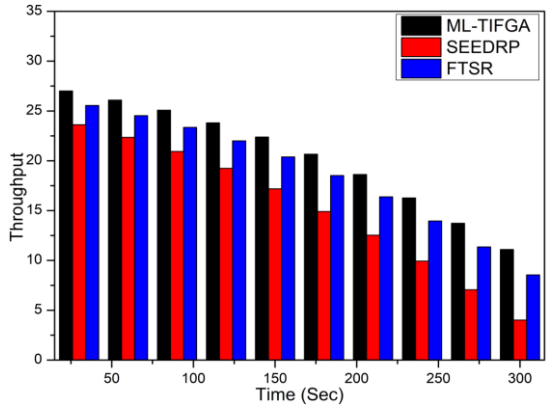

Figure 11. Comparative analysis for throughput with time.

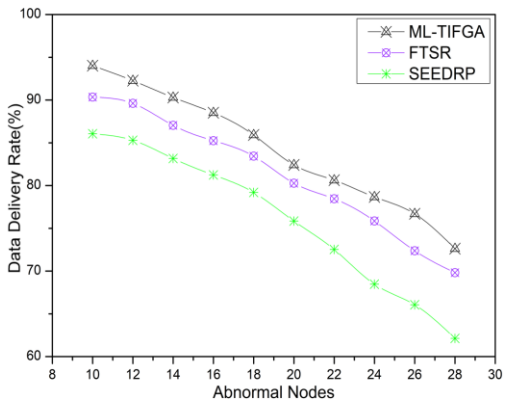

Figure 12. Comparative analysis for data delivery rate with threats.

Table 10. Experiment 3, outcomes for network factors.

Factors		Data Delivery Rate			Delay			Throughput		
Time (Secs)	30	83.146	75.554	78.532	106.246	100.181	109.345	27.006	23.627	25.554
	60	82.151	73.518	76.945	108.504	103.787	112.249	26.086	22.357	24.534
	90	80.916	71.281	75.006	111.812	111.038	118.283	25.076	20.937	23.354
	120	78.597	68.5	72.637	116.936	123.598	127.395	23.806	19.247	22.004
	150	76.115	65.486	70.021	124.441	139.038	138.636	22.386	17.187	20.394
	180	73.504	62.175	67.212	133.058	158.002	151.093	20.656	14.907	18.524
	210	70.631	58.403	64.157	144.625	180.744	167.211	18.626	12.537	16.384
	240	67.537	54.489	60.974	159.516	209.387	184.879	16.276	9.927	13.954
	270	64.302	50.437	57.627	177.553	244.068	207.03	13.736	7.067	11.354
	300	60.409	46.128	53.437	198.671	257.041	232.144	11.096	4.017	8.544
Abnormal Nodes (Count)	10	94.02	90.34	86.06	94.055	98.758	102.612	0.9847	0.9356	0.9117
	12	92.27	89.61	85.29	97.382	104.771	108.586	0.9631	0.8942	0.8769
	14	90.3	87.04	83.17	102.622	106.825	110.548	0.9411	0.8546	0.8407
	16	88.51	85.24	81.24	110.887	115.674	119.288	0.9157	0.8045	0.7764
	18	85.91	83.44	79.2	120.426	126.576	131.663	0.8927	0.7641	0.7529
	20	82.41	80.27	75.84	134.008	141.294	150.673	0.8207	0.7117	0.6704
	22	80.62	78.45	72.53	150.76	161.946	169.608	0.8004	0.6628	0.6318
	24	78.67	75.86	68.46	170.905	192.775	197.063	0.7629	0.614	0.5617
	26	76.7	72.37	66.06	192.717	215.741	221.894	0.7105	0.5818	0.5387
	28	72.62	69.81	62.13	219.509	244.367	256.866	0.6822	0.5416	0.4927

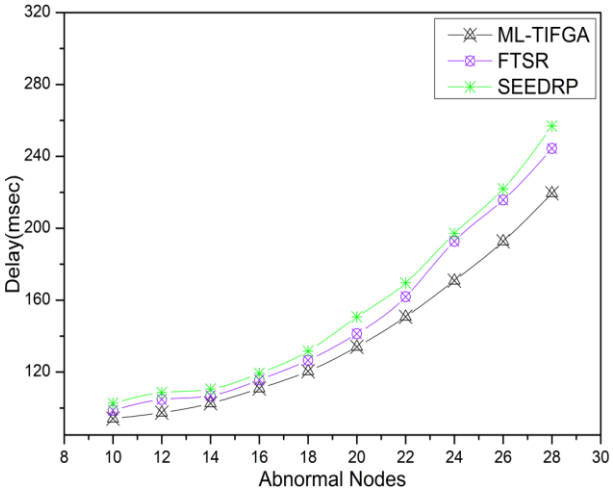

Figure 13. Comparative analysis for delay with threats.

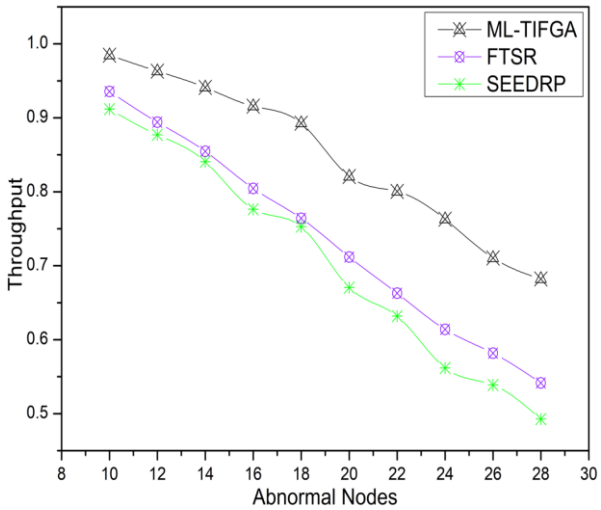

Figure 14. Comparative analysis for throughput with threats.

5. Conclusions and Future Work

Presented ML-TIFGA concluded a promising approach enhancing FANET security. Compared to random forest, FA-ML, and PSO with KNN techniques, the high accuracy precision, recall, and F1-score from experiment 1 demonstrate the robustness of ML-TIFGA in detecting various types of network intrusions. The ability to accurately classify different attacks and normal traffic implies that ML-TIFGA can effectively enhance the security of network environments by minimizing false positives and ensuring reliable intrusion detection. The approach is particularly significant for real-world applications where quick and accurate threat detection is crucial. Further, the results of experiment 2 showed that ML-TIFGA maintains a high success rate in classifying nodes even with significant injections of malicious and selfish nodes, indicating the method is resilient to adversarial conditions. For practical applications, this shows the deployment of ML-TIFGA is worthwhile in those environments where node behavior is unpredictable and potentially hostile for classification into normal, selfish, and malicious. With the outcomes from experiment 3, ML-TIFGA effectively minimizes network delay, optimizes throughput, and ensures a high data delivery rate even in the presence of abnormal nodes compared to SEEDRP and FTSR with the variation in both simulation time and abnormal nodes and is crucial for real-time data exchange applications, such as environmental monitoring or emergency response. Our extensive experimental analysis achieved an accuracy of 99.829% in node classification, while 98.36% and 98.86% in threat identification for 80% training and 20% testing samples. Further, the evaluation improves network reliability from 0.96 to 0.993 for a network size of 50 to 1000 nodes. Performance metrics delay, throughput, and data delivery rate are improved by 24.65%, 29.16%, and 31.73%, respectively, with the increase of abnormal nodes. In the future, the work of ML-TIFGA can be enhanced to incorporate nature-

inspired optimization algorithms with a large node density.

References

[1] Bangui H. and Buhnova B., "Recent Advances in Machine-Learning Driven Intrusion Detection in Transportation: Survey," *Procedia Computer Science*, vol. 184, pp. 877-886, 2021. https://doi.org/10.1016/j.procs.2021.04.014

[2] Basan E., Lapina M., Mudruk N., and Abramov E., "Intelligent Intrusion Detection System for a Group of UAVs," *in Proceedings of the 12th International Conference in Advances in Swarm Intelligence*, Qingdao, pp. 17-21, 2021. https://doi.org/10.1007/978-3-030-78811-7_22

[3] Bhardwaj V. and Kaur N., "SEEDRP: A Secure Energy Efficient Dynamic Routing Protocol in FANETs," *Wireless Personal Communications*, vol. 120, no. 2, pp. 1251-1277, 2021. https://doi.org/10.1007/s11277-021-08513-0

[4] Bhardwaj V., Kaur N., Vashisht S., Jain S., "Secrip: Secure and Reliable Intercluster Routing Protocol for Efficient Data Transmission in Flying Ad Hoc Networks," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 6, pp. 4068, 2021. https://doi.org/10.1002/ett.4068

[5] Choudhary S. and Kesswani N., "Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 Datasets Using Deep Learning in IoT," *Procedia Computer Science*, vol. 167, pp. 1561-1573, 2020. https://doi.org/10.1016/j.procs.2020.03.367

[6] Din N., Waheed A., Zareei M., and Alanazi F., "An Improved Identity-Based Generalized Signcryption Scheme for Secure Multi-Access Edge Computing Empowered Flying Ad Hoc Networks," *IEEE Access*, vol. 9, pp. 120704-120714, 2021. DOI:10.1109/ACCESS.2021.3108130

[7] Fu R., Ren X., Li Y., Wu Y., and Sun H., "Machine-Learning-Based UAV-assisted Agricultural Information Security Architecture and Intrusion Detection," *IEEE Internet of Things Journal*, vol. 10, no. 21, pp. 18589-18598, 2023. DOI:10.1109/JIOT.2023.3236322

[8] Ghelani J., Gharia P., and El-Ocla H., "Gradient Monitored Reinforcement Learning for Jamming Attack Detection in FANETs," *IEEE Access*, vol. 12, pp. 23081-23095, 2024. DOI:10.1109/ACCESS.2024.3361945

[9] Gupta D. and Rathi R., "RDVFF- Reliable Data Dissemination in Vehicular Ad Hoc Networks Based on Validation of Far to Farthest Zone," *Journal of Internet Technology*, vol. 25, no. 1, pp. 87-104, 2024. DOI:10.53106/160792642024012501008

[10] Gupta S. and Sharma N., "SCFS-Securing Flying Ad Hoc Network Using Cluster-Based Trusted Fuzzy Scheme," *Complex and Intelligent Systems*, vol. 10, no. 3, pp. 1-20, 2024. DOI:10.1007/s40747-024-01348-9

[11] Gupta S., Sharma N., Rathi R., and Gupta D., "Dual Detection Procedure to Secure Flying Ad Hoc Networks: A Trust-Based Framework," *in Proceedings of the Smart Technologies in Data Science and Communication Conference*, vol. 210, pp. 83-95, 2021. https://doi.org/10.1007/978-981-16-1773-7_7

[12] Gupta D., Rathi R., Gupta S., and Sharma N., "Multiple Relay Nodes Selection Scheme using Exit Time Variation for Efficient Data Dissemination in VANET," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, pp. 276-286, 2021. DOI:10.14569/IJACSA.2021.0120731

[13] Hosseinzadeh M., Mohammed A., Alenizi F., Malik M., and Yousefpoor E., "A Novel Fuzzy Trust-Based Secure Routing Scheme in Flying Ad Hoc Networks," *Vehicular Communications*, vol. 44, no. 100665, 2023. https://doi.org/10.1016/j.vehcom.2023.100665

[14] Kai-Yun T., Girdler T., and Vassilakis V., "A Survey of Cyber Security Threats and Solutions for UAV Communications and Flying Ad-Hoc Networks," *Ad Hoc Networks*, vol. 133, no. 102894, 2022. https://doi.org/10.1016/j.adhoc.2022.102894

[15] Karthiga B., Durairaj D., Nawaz N., Venkatasamy T., and Ramasamy G., "Intelligent Intrusion Detection System for VANET Using Machine Learning and Deep Learning Approaches," *Wireless Communications and Mobile Computing*, vol. 2022, 2022. https://doi.org/10.1155/2022/5069104

[16] Karthikeyan M., Manimegalai D., and RajaGopal K., "Firefly Algorithm Based WSN-IoT Security Enhancement with Machine Learning for Intrusion Detection," *Scientific Reports*, vol. 14, no. 1, pp. 231, 2024. https://doi.org/10.1038/s41598-023-50554-x

[17] Kaur M., Prashar D., Rashid M., Alshamrani S., and AlGhamdi A., "A Novel Approach for Securing Nodes Using Two-Ray Model and Shadow Effects in Flying Ad-Hoc Network," *Electronics*, vol. 10, no. 24, pp. 3164, 2021. https://doi.org/10.3390/electronics10243164

[18] Khan M., Ullah I., Nisar S., Noor F., and Qureshi I., "Multiaccess Edge Computing Empowered Flying Ad Hoc Networks with Secure Deployment Using Identity-Based Generalized Signcryption," *Mobile Information System*, vol. 2020, 2020. https://doi.org/10.1155/2020/8861947

[19] Liu J., Shuai H., and Yi W., "Throughput Optimization for Flying Ad Hoc Network Based On Position Control Using Genetic Algorithm,"

International Journal of Metrology and Quality Engineering, vol. 11, pp. 1-13, 2020. https://doi.org/10.1051/ijmqe/2020012

[20] Manoharan S., Sugumaran P., and Kumar K., "Multichannel Based IoT Malware Detection System Using System Calls and Opcode Sequences," *The International Arab Journal of Information Technology*, vol. 19, no. 2, pp. 261-271, 2022. https://doi.org/10.34028/iajit/19/2/13

[21] Namdev M., Goyal S., and Agarwal R., "An Optimized Communication Scheme for Energy Efficient and Secure Flying Ad-Hoc Network (FANET)," *Wireless Personal Communications*. vol. 120, no. 2, pp. 1291-1312, 2021. https://doi.org/10.1007/s11277-021-08515-y

[22] Oubbati O., Atiquzzaman M., Lorenz P., Tareque H., and Hossain S., "Routing in Flying Ad Hoc Networks: Survey, Constraints, and Future Challenge Perspectives," *IEEE Access*, vol. 7, pp. 81057-81105, 2019. Doi:10.1109/ACCESS.2019.2923840

[23] Priya S. and Mohanraj M., "Flying Ad-Hoc Networks Rely on Trust-Aware Route Selection for Efficient Packet Transmission," *Journal of Algebraic Statistics*, vol. 13, no. 3, pp. 682-692, 2022. https://www.publishoa.com/index.php/journal/article/view/675/563

[24] Rahman K., Aziz M., Usman N., Kiren T., and Cheema T., "Cognitive Lightweight Logistic Regression-Based IDS for IoT-Enabled FANET to Detect Cyberattacks," *Mobile Information Systems*, vol. 2023, no. 1, pp. 7690322, 2023. https://doi.org/10.1155/2023/7690322

[25] Rahman K., Aziz M., Kashif A., and Cheema T., *Big Data Analytics and Computational Intelligence for Cybersecurity*, Springer, 2022. https://doi.org/10.1007/978-3-031-05752-6_7

[26] Raj J., "A Novel Hybrid Secure Routing for Flying AD-HOC Networks," *Journal of Trends in Computer Science and Smart Technology*, vol. 2, no. 3, pp. 155-164, 2020. DOI:10.36548/jtcsst.2020.3.005

[27] Ramadan R., Emara A., Al-Sarem M., and Elhamahmy M., "Internet of Drones Intrusion Detection Using Deep Learning," *Electronics*, vol. 10, no. 21, pp. 2633, 2021. https://doi.org/10.3390/electronics10212633

[28] Safia L., Rizwan M., Hassan M., "Security Threats in Flying Ad Hoc Network (FANET)," *Computational Intelligence for Unmanned Aerial Vehicles Communication Networks*, vol. 1033, pp 73-96, 2022. https://doi.org/10.1007/978-3-030-97113-7_5

[29] Shitharth S., Yonbawi S., Manoharan H., Shankar A., and Maple C., "Secured Data Transmissions in Corporeal Unmanned Device to Device Using Machine Learning Algorithm," *Physical Communication*, vol. 59, pp. 102116, 2023. https://doi.org/10.1016/j.phycom.2023.102116

[30] Singh K. and Verma A., "A Trust Model for Effective Cooperation in Flying Ad Hoc Networks Using Genetic Algorithm," *in Proceedings of the International Conference on Communication and Signal Processing*, Chennai, pp. 0491-0495, 2018. DOI:10.1109/ICCSP.2018.8524558

[31] Sucharitha Y., Reddy P., and Suryanarayana G., *Drone Technology: Future Trends and Practical Applications*, Wiley 2023. https://doi.org/10.1002/9781394168002.ch15

[32] Tangade S., Kumaar R., Malavika S., Monisha S., and Azam F. "Detection of Malicious Nodes in Flying Ad-hoc Network with Supervised Machine Learning," *in Proceedings of the 3rd International Conference on Smart Technologies in Computing, Electrical and Electronics*, Bengaluru, pp. 1-5, 2022. DOI:10.1109/ICSTCEE56972.2022

[33] Tavallaee M., Bagheri E., Lu W., Ghorbani A., "A Detailed Analysis of the KDD CUP 99 Data Set," *in Proceedings of the 2nd IEEE Symposium on Computational Intelligence for Security and Defense Applications*, Ottawa, 2009.

[34] Vijay A., Patidar K., Yadav M., and Kushwah R., "An Efficient Intrusion Detection Mechanism Based on Particle Swarm Optimization and KNN," *ACCENTS Transactions on Information Security*, vol. 5, no. 20, pp. 36-41, 2020. DOI:10.19101/TIS.2020.517003

Shikha Gupta graduated with a Bachelor's (Hons.) and Master's (Hons.) in Computer Engineering from Rajasthan University, Jaipur, and Bhagwant University, Ajmer, India, in 2003 and 2012, respectively. She started working as an Assistant Professor full-time in the Engineering College Ajmer's Department of Computer Science and Engineering in 2006. She has presented several research papers at conferences and reputed journals, and her interests include security in communication with VANETs, FANET, and ad hoc networks on the Internet of Things.

Neetu Sharma obtained her Ph.D. from Gyan Vihar University, Jaipur, India, in 2011. She is working as a full-time Assistant Professor in the Department of Computer Engineering of Engineering College Ajmer, India. She has extensive research and teaching experience in this institute since 2008 and has published over 50 articles in reputed journals and conferences. Her research interests include wireless sensors and ad hoc networks.