Abou Bekr Belkaid University
Tlemcen Algeria
جامعة أبي بكر بلقايد
تلمسان الجزائر
PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
الجمهورية الجزائرية الديمقراطية الشعبية
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
وزارة التعليم العالي و البحث العلمي
FACULTY OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCES

A Thesis

Presented for obtaining the Degree of:

DOCTORATE

In: Computer Science
Specialty: Distributed Computing and Networking

By:

Sihem BENFRIHA

Theme

Secure and Reliable Communications in Flying Ad hoc Networks (FANETs)

Thesis defended on , 2024, at Tlemcen in Front of the Jury Composed of:

Mr Abdelkrim BENAMAR	Full Professor	University of Tlemcen	President
Mme Nabila LABRAOUI	Full Professor	University of Tlemcen	Supervisor
Mr Bouabdellah KECHAR	Full Professor	University of Oran 1	Examiner
Mr Sofiane BOUKLI-HACENE	Full Professor	University of Sidi Belabes	Examiner
Mr Mohammed MANA	Associate Professor	University of Tlemcen	Examiner

Academic Year 2023/2024

«« قَال رَسُول اللَّهِ صلى الله عليه وسلم : منْ سَلَكَ طَريقًا يَبْتَغِي فِيهِ علْمًا سهَّل اللَّه لَه طَريقًا إِلَى الجنةِ، وَإِنَّ الملائِكَةَ لَتَضَعُ أَجْنِحَتَهَا لِطالب الْعِلْمِ رِضًا بِما يَصْنَعُ، وَإِنَّ الْعالِمِ لَيَسْتَغْفِرُ لَهُ منْ في السَّمَواتِ ومنْ فِي الأَرْضِ حتَّى الحِيتانُ في الماءِ، وفَضْلُ الْعَالِمِ عَلَى الْعابِدِ كَفَضْلِ الْقَمر عَلى سَائِرِ الْكَوَاكِبِ، وإِنَّ الْعُلَماءَ وَرَثَةُ الأَنْبِياءِ وإِنَّ الأَنْبِياءَ لَمْ يُورِّثُوا دِينَارًا وَلا دِرْهَمًا وإِنَّما ورَّثُوا الْعِلْمَ، فَمنْ أَخَذَهُ أَخَذَ بِحظٍّ وَافِرٍ. »» رواهُ أَبُو داود والترمذيُّ.

Dedication

I proudly dedicate this thesis to:

- My dear father, you have been my guiding light from the beginning. Your unwavering support, wisdom, and strength have been my rock through every twist and turn. You have taught me the importance of perseverance, integrity, and the power of love. Your sacrifices and unwavering belief in me have shaped the person I am today. Thank you for being my superhero, role model, and source of inspiration.

- My precious mother, you are the epitome of love and selflessness. Your nurturing nature and boundless compassion have filled my life with warmth and tenderness. You have shown me the true meaning of unconditional love, and I am forever grateful for your guidance, patience, and endless sacrifices. Your unwavering faith in me has given me the confidence to chase my dreams and embrace every challenge that comes my way. Thank you for being my confidante and my mentor.

- My beloved husband, you are the anchor that keeps me grounded. Your unwavering support, love, and understanding have been my greatest strength. Your presence has brought me immeasurable joy, laughter, and comfort. Thank you for being my pillar of strength and my best friend.

- My daughter and my cherished sisters, thank you for your support, encouragement, and understanding. Your belief in me has served as a constant reminder of the importance of community and the power of connection. Your words of encouragement and gestures of kindness have made all the difference, and I am deeply grateful for your presence in my life.

– Sihem Benfriha

Acknowledgements

All praise and thanks go to Allah, the Lord of all worlds.We thank Allah for the gift of life, the ability to see, hear, and feel, and the opportunity to study and finish this thesis.

To my supervisor, Prof Nabila LABRAOUI, I am thankful for the countless hours you dedicated to our discussions, the patience you demonstrated in answering my questions, and the genuine interest you showed in my progress. Your mentorship has shaped my academic journey and contributed significantly to my personal and professional development.

I express my sincere gratitude and heartfelt thanks to the esteemed jury members who graciously accepted the invitation to evaluate and assess my Ph.D. thesis. Mr Abdelkrim BENAMAR i am deeply grateful for your valuable contribution. Mr Bouabdellah KECHAR Your commitment to upholding academic standards, rigor, and integrity is commendable. Mr Sofiane BOUKLI-HACENE Your dedication to the pursuit of knowledge and your willingness to contribute to the academic community through your evaluation of doctoral thesis is a testament to your passion and professionalism. Thank you Mr Mohammed MANA i am deeply grateful for your valuable contribution. I want to express my sincere appreciation to Mr. Mohamed LEHSAINI, the head of the STIC lab and our PhD Project Manager, for granting me the privilege to join their dynamic group of members in STIC.

Special thanks to Mr Haythem BANY SALAMEH Professor of Electrical and Computer Engineering,he was one of the most vital and active researchers who had a big impact in my last research papers, he also guided and gave me a lot of advises to foster my PhD career.

I am grateful to the teachers who instilled the patience to study computer science and fostered a love for this field. Special thanks to BRIKCI NIGASA for his focus on object-oriented programming, Prof Badr BENMAMMAR for their expertise in Java, and Prof Nabila LABRAOUI for her network guidance.

We acknowledge that Allah is the All-Wise. We praise Him for the trials and challenges He places in our lives, as they are opportunities for growth, purification, and faith strengthening.

Abstract

FANETs, or Flying Ad-Hoc Networks, are wireless communication networks comprising autonomous UAVs collaborating to fulfill various missions. FANETs are susceptible to numerous security threats. In light of this, the thesis focuses on addressing security and data privacy concerns, specifically emphasizing insider attack detection, considering drones' unique behavior and characteristics. While numerous techniques exist to address these issues, this research delves into two main areas. First, leveraging fuzzy logic, we introduce FUBA, a robust drone behavior analytics system, to enhance trust management in FANETs. Additionally, we provide a comprehensive survey of existing techniques in this domain. Second, we propose FLID, an intelligent Intrusion Detection System (IDS) tailored for FANETs, which integrates deep learning and federated learning to detect and prevent network attacks effectively. Moreover, we enhance FLID by employing reinforcement learning for drone-client selection, thereby strengthening network security and data privacy. Our findings demonstrate that insider attack detection can be achieved without compromising data privacy, offering tangible benefits across domains such as surveillance and disaster management.

keywords: Deep learning, FANET, federated learning, IDS, privacy, reinforcement learning, security, trust management, UAV.

Résumé

Les FANETs, ou réseaux ad hoc volants, sont des réseaux de communication sans fil constitués de drones autonomes collaborant pour remplir diverses missions. Les FANETs sont sensibles à de nombreuses menaces de sécurité. À la lumière de cela, la thèse se concentre sur la résolution des problèmes de sécurité et de confidentialité des données, en mettant spécifiquement l'accent sur la détection des attaques internes, en tenant compte du comportement et des caractéristiques uniques des drones. Bien qu'il existe de nombreuses techniques pour résoudre ces problèmes, cette recherche se concentre sur deux domaines principaux. Tout d'abord, en tirant parti de la logique floue, nous introduisons FUBA, un système robuste d'analyse du comportement des drones, pour améliorer la gestion de la confiance dans les FANETs. En outre, nous fournissons une étude complète des techniques existantes dans ce domaine. Deuxièmement, nous proposons FLID, un système intelligent de détection d'intrusion (IDS) adapté aux FANETs, qui intègre l'apprentissage profond et l'apprentissage fédéré pour détecter et prévenir efficacement les attaques réseau. De plus, nous améliorons le FLID en utilisant l'apprentissage par renforcement pour la sélection du drone-client en renforçant ainsi la sécurité du réseau et la confidentialité des données. Nos résultats démontrent que la détection d'attaques internes peut être réalisée sans compromettre la confidentialité des données, offrant des avantages tangibles dans des domaines tels que la surveillance et la gestion des catastrophes.

mots-clés: Apprentissage profond, FANET, apprentissage fédéré, système de détection d'intrusion, confidentialité, sécurité, apprentissage par renforcement, gestion de la confiance, véhicule aérien sans pilote.

مُلخّص

FANETs ، أو الشبكات الجوية المخصصة، هي شبكات اتصالات بدون طيار مكونة من طائرات بدون طيار مستقلة متعاونة للقيام بمهام متنوعة. في ضوء ذلك، تتناول هذه الأطروحة مخاوف الأمن وخصوصية البيانات، مع التركيز بشكل خاص على اكتشاف الهجوم من الداخل، مع الأخذ في الاعتبار السلوك والخصائص الفريدة للطائرات بدون طيار. في حين أن هناك العديد من التقنيات لمعالجة هذه القضايا، فإن هذا البحث يتعمق في مجالين رئيسيين. أولاً، الاستفادة من المنطق الغامض، نقدم FUBA ، وهو نظام قوي لتحليلات سلوك الطائرات بدون طيار، لتعزيز إدارة الثقة في FANETs . بالإضافة إلى ذلك، نقدم مسحًا شاملاً للتقنيات الموجودة في هذا المجال. ثانيًا، نقترح FLID ، وهو نظام ذكي للكشف عن الاقتحام (IDS) مصمم خصيصًا لـ FANETs ، والذي يدمج التعلم العميق والتعلم الاتحادي لاكتشاف ومنع هجمات الشبكة بشكل فعال. علاوة على ذلك، فإننا نعزز FLID من خلال توظيف التعلم المعزز لاختيار عملاء الطائرات بدون طيار، وبالتالي تعزيز أمن الشبكة وخصوصية البيانات. توضح النتائج التي توصلنا إليها أنه يمكن تحقيق اكتشاف الهجوم من الداخل دون المساس بخصوصية البيانات، مما يوفر فوائد ملموسة عبر مجالات مثل المراقبة وإدارة الكوارث.

الكلمات المفتاحيّة: التعلم العميق، شبكة الطيران المخصصة، التعلم الاتحادي، نظام الكشف عن الاقتحام، الخصوصية، الأمن، التعلم المعزز، إدارة الثقة، الطائرات بدون طيار.

List of Publications

Journal Publications

1) Sihem Benfriha, Nabila Labraoui, Radjaa Bensaid, Haythem Bany Salameh, and Hafida Saidi. "FUBA: A fuzzy-based unmanned aerial vehicle behaviour analytics for trust management in flying ad-hoc networks." IET Networks (2023).

https://ietresearch.onlinelibrary.wiley.com/doi/full/10.1049/ntw2.12108

2) Bensaid, Radjaa, Nabila Labraoui, Ado Adamou Abba Ari, Leandros Maglaras, Hafida Saidi, Ahmed Mahmoud Abdu Lwahhab, and Sihem Benfriha. "Toward a real-time tcp syn flood ddos mitigation using adaptive neuro-fuzzy classifier and sdn assistance in fog computing." Security and Communication Networks 2024 (2023). https://www.hindawi.com/journals/scn/2024/6651584/

3) Sihem Benfriha, Nabila Labraoui, Haythem Bany Salameh, and Radjaa Bensaid. " Reinforcement Learning-based Drone Client Selection for Efficient Federated Learning-based Intrusion Detection in FANETs."IEEE Transactions on Network Science and Engineering(Under Review)

Conference Communications

1) Sihem Benfriha and Nabila Labraoui, "Insiders Detection in the Uncertain IoD using Fuzzy Logic," 2022 International Arab Conference on Information Technology (ACIT), Abu Dhabi, United Arab Emirates, 2022, pp. 1-6, doi: 10.1109/ACIT57182.2022.9994119

2) Sihem. Benfriha, Nabila. Labraoui, Haythem Bany Salameh and Hafida Saidi, "A Survey on Trust Management in Flying Ad Hoc Networks: Challenges, Classifications, and Analysis," 2023 Tenth International Conference on Software Defined Systems (SDS), San Antonio, TX, USA, 2023, pp. 107-114, doi: 10.1109/SDS59856.2023.10329156.

3) Saidi, Hafida, Nabila Labraoui, Sihem Benfriha, and Zina Houhamdi. "DVD: Decentralized Verification of Diplomas in Smart Universities." In 2023 24th International Arab Conference on Information Technology (ACIT), pp. 1-7. IEEE, 2023.

Table of Contents

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

A-DSDV Adaptive Destination-Sequenced Distance Vector.

ACL Access Control List.

ADS-B Automatic Dependent Surveillance–Broadcast.

AI Artificial Intelligence.

AODV Ad hoc On-Demand Distance Vector.

AUC Area Under the Curve.

AVENS Aerial Vehicle Network Simulator.

CA Certificate Authority.

CH Charging Station.

CNN Convolutional Neural Network.

COG Center Of Gravity COG.

CRL Certificate Revocation List.

DBN Deep Belief Network.

DDoS Distributed Denial of Service.

DL Deep Learning.

DNN Deep Neural Network.

DoS Denial of Service.

DSDV Destination-Sequenced Distance Vector.

DSR Dynamic Source Routing.

DT Decision Tree.

FANET Flying Ad hoc Networks.

FDMA Frequency Division Multiple Access.

FL Federated Learning.

FRL Federated Reinforcement Learning.

GAN Generative Adversarial Network.

GCS Ground Control Station.

IDS Intrusion Detection System.

IEEE Institute of Electrical and Electronics Engineers.

IoD Internet of Drone.

IoT Internet of Things.

IoV Internet of Vehicle.

LiR Linear Regression.

LoR Logistique Regression.

LSTM Long Short-Term Memory.

MANET Mobile Ad hoc Network.

MAVLink Micro Air Vehicle Link,.

MDP Markov Decision Process.

MEC Multi-access Edge Computing.

MitM Man-in-the-Middle.

ML Machine Learning.

MLP Multi-layer Perceptron.

OCSP Online Certificate Status Protocol.

OLSR Optimized Link State Routing.

OMNeT Objective Modular Network.

PDR Packet Delivery Ratio.

RBAC Role-Based Access Control.

RF Random Forest.

RL Reinforcement Learning.

RNN Recurrent Neural Network.

ROC Receiver Operating Characteristic.

RSSD Received Signal Strength Difference.

RSSI Received Signal Strength Indicator.

SGD Stochastic Gradient Descent.

SMC Secure Multiparty Computation.

SSO Sparrow Search Optimization.

STFA Sea Turtle Foraging Algorithm.

SVM Support Vector Machine.

TD Transmission Delay.

TDoA Time Difference of Arrival.

UAV Unmanned Aerial Vehicle.

VANET Vehicular Ad-hoc Network.

WLAN Wireless Local Area Networking.

ZRP Zone Routing Protocol.

General Introduction

1 Introduction

The burst of Artificial intelligence (AI) marked a significant turning point in our relationship with technology, leading to profound changes in our societies and daily lives. AI is a rapidly evolving field focusing on developing intelligent systems capable of simulating human intelligence. One of the key drivers of this transformation is the development of robotics, such as autonomous vehicles and drones, which has opened up a new era of personal technology and connectivity. Over the past few decades, drone technology developments have revolutionized industries, enhanced data collection capabilities, and provided innovative solutions in various sectors as drone technology continues to evolve and play a significant role in areas such as logistics, infrastructure maintenance, environmental monitoring, and public safety, contributing to increased efficiency, improved security, and expanded applications. A drone ad hoc network, Unmanned Aerial Vehicle (UAV) ad hoc network, or generally Flying Ad Hoc network (FANET) refers to a wireless communication network consisting of autonomous drones that establish a temporary network infrastructure in the air. These drones work together collectively, collaborate, and cooperate to accomplish several missions in various fields such as agriculture,environmental monitoring,surveillance,delivery services and rescue operations.

Compared to traditional networks with fixed infrastructure, FANETs are dynamic and self-organizing networks where drones act as communication nodes and data relays.

With the increasing popularity and adoption of drones in various industries, the vulnerability of FANETs to cyberattacks has become a significant concern. However, ensuring secure and reliable communication in FANET is very important and remains problematic.

2 Problem Statement and Motivation

Since 2007, there has been a notable surge in cyberattacks targeting drones, posing significant risks with potentially catastrophic consequences ranging from data breaches and service disruptions to even physical harm or loss of lives and assets. This underscores the urgent need to fortify FANETs against internal and external threats. The fluid nature of drone participation presents a prime opportunity for malicious actors to compromise drones, assuming false identities and perpetrating insider attacks. These insiders exploit their privileged access to execute illicit activities, including inserting malicious hardware or software into the drone, compromising its functionality, or allowing unauthorized control. On the other hand, data privacy is paramount in FANET. Indeed, drones can gather sensitive information from the environment through various sensors and technologies. Equipped with cameras, they capture visual data revealing layouts of buildings and critical infrastructure. Additionally, drones with GPS can collect precise geolocation data, potentially exposing sensitive locations or tracking movements. They can also record audio data, intercept wireless signals, and analyze biometric information. Therefore, Curious actors and hackers can extract sensitive information and disrupt network privacy.

Consequently, cryptographic methods, securing communication protocols, and certificate revocation are promising strategies for protecting FANETs from potential attacks. However, the existing solutions have their main limitations which can obstruct their effectiveness in addressing specific challenges.

Moreover, the severity of attacks is amplified in FANETs due to their unique characteristics. The mobility of drones, the dynamic nature of the network, and the limited computational capabilities of the drones make it challenging to detect and respond to attacks effectively. Consequently, ensuring the security and data privacy within FANETs remains a paramount challenge in cybersecurity, warranting focused attention and innovative strategies. For this purpose, we have employed robust approaches for detecting insiders within FANETs, including trust management, intrusion detection systems (IDSs), and machine learning-based solutions.

3 Thesis Contributions and Outlines

In light of these challenges, this thesis presents three contributions focused on security and data privacy within FANETs.

- The first contribution offers a mechanism to evaluate the behavior of a drone using Fuzzy logic that relies on direct and indirect information and offers a more robust and adaptive approach to drone assessment. Unlike prior models, the proposed approach enhances network trustworthiness even in bad weather conditions and under poor signal strength.

- The second contribution offers an IDS-based on federated learning (FL), this mechanism detects the three frequent attacks: blackhole, flooding, and selective forwarding.The innovative approach allows drones within the network to engage in localized model training on their data while safeguarding sensitive information privacy and harnessing each drone's unique capabilities. It achieves efficient and effective IDS without compromising network performance.
- The third contribution combines FL and reinforcement learning (RL) to improve the second contribution. Fusing FL and RL algorithms aims to meet the network's security requirements, encompassing privacy protection, effectiveness, robustness, and context awareness.

Thi thesis is divided into two parts:a LITERATURE REVIEW section and SCIENTIFIC CONTRIBUTIONS section.

The LITERATURE REVIEW section is structured as follows:

- **Chapter 1** provides essential background information on FANETs. It outlines FANET architecture, characteristics, design considerations, applications, communication models, and routing protocols. Then, it summarizes the different security challenges in FANETs and discusses existing countermeasures.
- **Chapter 2** describes the concept of trust management in FANET. It provides a detailed review of related works and a novel classification for trust management techniques. Finally, it offers a comparative analysis of selected trust management schemes.

The SCIENTIFIC CONTRIBUTIONS is structured as follows:

- **Chapter 3** provides the first contribution to FANET security by addressing the limitations of existing trust management mechanisms and leveraging fuzzy logic-based drone behavior analytics for trust management.
- **Chapter 4** introduces the second contribution, which is a novel FL model tailored explicitly for IDS within FANETs.
- **Chapter 5** presents the third contribution while proposing an RL-based drone client selection for efficient FL-based IDS within FANETs.

In the conclusion, we offer a comprehensive thesis summary, delve into the security issue, and explore the insights that will guide future research endeavors.

PART ONE: LITERATURE REVIEW

Chapter I: Security Issues in Flying Ad hoc Networks Concepts and Challenges

"Threat is a mirror of security gaps. Cyber-threat is mainly a reflection of our weaknesses. An accurate vision of digital and behavioral gaps is crucial for a consistent cyber-resilience"

– Stephane Nappo

1 Introduction

In the dynamic realm of aerial technology, the convergence of ad hoc networking and unmanned aerial vehicles has ushered in a new era of possibilities and challenges. As we begin our exploration of the complex domain of flying ad hoc networks, this chapter seeks to illuminate the critical security dimensions within this fascinating domain. From soaring possibilities to inherent challenges, exploring security issues in FANETs is a compelling quest that demands attention. In this chapter, we provide essential background information on FANETs. In Section 2, we outline FANET architecture, characteristics, design considerations, applications, communication models, and routing protocols. In Section 3, we summarize the different security challenges in FANETs. In Section 4, we discuss existing countermeasures in FANET, and finally, we conclude the chapter in Section 5.

2 FANET: An Overview

FANET is a flying ad-hoc network that consists of a group of unmanned aerial vehicles (UAVs) or drones that communicate with each other wirelessly. FANET is a subclass of a mobile ad-hoc network (MANET) where drones autonomously establish and maintain a dynamic, decentralized network. Still, it has unique characteristics and challenges, such as high mobility, long-range, apparent line-of-sight propagation, and environment resilience [1]. FANET enables cooperative and self-organizing behavior among drones, facilitating communication, data sharing, and coordination in scenarios where traditional infrastructure-based communication may be impractical or unavailable [2]. A FANET requires highly accurate localization data and collision avoidance mechanisms; it can also interact with other networks, such as vehicular ad-hoc networks (VANETs), to provide more information and services to the users. Figure 1 represents the relation between FANET and the other network types.

2.1 FANET Architecture

In FANET, drones can fly autonomously and operate remotely without carrying any human personnel. These drones collaborate in real time, forming an ad-hoc communication network without relying on fixed infrastructure links as shown in Figure 2.

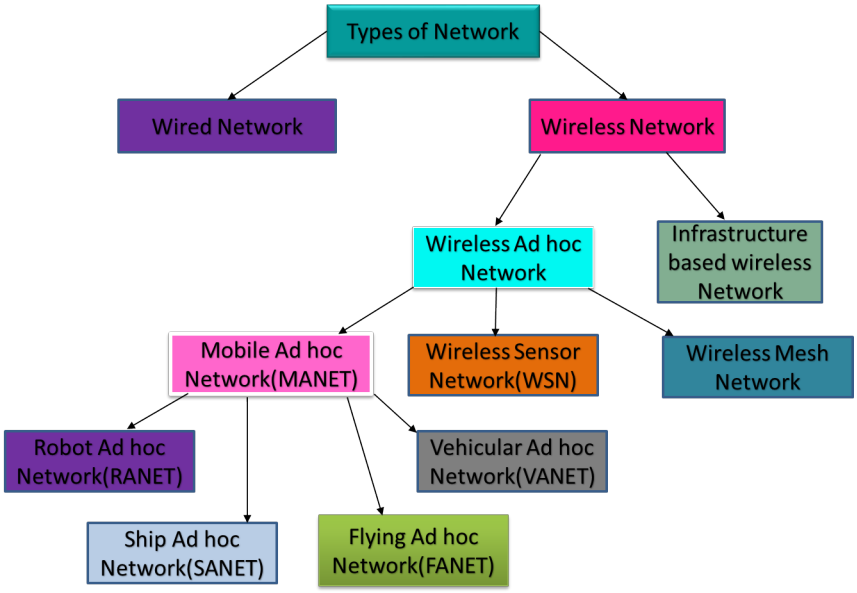


Figure 1: The relation between FANET and other network types.

FANET architecture can be classified into two main categories based on the arrangement and interaction of drones: Single-drone systems and Multi-drone systems. Another classification that classifies FANET architecture into centralized and decentralized FANET Architecture.

2.1.1 Single-Drone Systems

In this architecture, a standalone drone operates independently without direct communication or coordination with other drones [3]. The communication may rely on links to fixed infrastructure or other external sources. This form is less dynamic and lacks the collaborative capabilities associated with multi-drone systems.

2.1.2 Multi-Drone Systems

Multi-drone systems involve the collaboration and communication between multiple drones operating in the same airspace. These drones interact with each other to share information, coordinate tasks, and enhance overall system capabilities [4].

2.1.3 Centralized Architecture

A centralized FANET architecture is a type of network design for FANETs that consists of multiple drones and a ground control station (GCS) [5]. In this architecture, the GCS acts as the central server and the only controller of the network, while all the drones are directly connected to it. Therefore, any data communication between any two drones is carried out via the GCS. This architecture simplifies network management and control but also introduces a single point of failure and a high dependency on the GCS.

2.1.4 Decentralized Architecture

A decentralized FANET architecture is a network topology that does not rely on a central server or controller to manage the communication and coordination of FANETs [6]. FANETs are composed of drones that can dynamically form a network without any pre-existing infrastructure. A decentralized FANET architecture has several advantages over a centralized one, such as scalability, robustness, flexibility, and self-organization. However, it poses challenges like routing, synchronization, security, and consensus.

Figure 2: FANET architecture.

2.2 FANET Characteristics

FANET exhibits several characteristics that distinguish it as a specialized drone network:

1. **Ad-Hoc Networking:** FANET relies on ad-hoc networking principles, where drones dynamically form and maintain the communication network without needing a fixed infrastructure. The network is self-organizing, allowing drones to connect and disconnect as required.

2. **Decentralization:** FANET operates decentralized, with no central control. Each drone in the network can make independent decisions, contributing to the flexibility and adaptability of the overall system.

3. **Dynamic Network Topology:** The topology of the FANET is highly dynamic and constantly changing as drones move within the airspace. This dynamic nature allows the network to adapt to environmental changes and mission requirements.

4. **Collaborative Communication:** Drones in a FANET actively communicate and collaborate; they share information, coordinate actions, and may collectively work towards achieving mission objectives.FANET communication is illustrated in Figure 3.

5. **Scalability:** FANET is designed to be scalable to accommodate varying numbers of drones. The network should be able to handle the addition or removal of drones without significant disruption, making it suitable for diverse applications ranging from small-scale operations to larger missions.

6. **Real-Time Communication:** FANET requires real-time communication capabilities to support rapid decision-making, coordination, and timely exchange of information among drones. Low-latency communication is essential for ensuring the responsiveness of the network.

7. **Network Lifetime:** The duration in which FANET can operate without running out of energy or losing connectivity is critical. Maximizing network lifetime is essential for prolonged and reliable mission execution, especially in applications with extended operational duration.

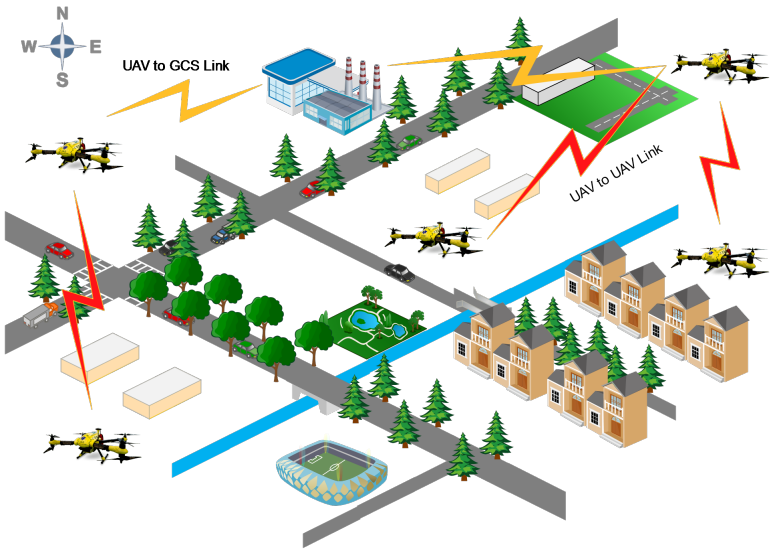


Figure 3: FANET communication.

2.3 FANET Design Considerations

When delving into the design of FANETs, several critical considerations must be taken into account to ensure their effective operation, reliability, and security. The key design considerations are described as follows:

1. Autonomous Drones: Drones in a FANET are capable of autonomous flight and operation. They can navigate, communicate, and make decisions independently, contributing to the agility and responsiveness of the network.

2. Drone Density: The number of drones in a given area influences network connectivity and interference. High drone density can enhance collaboration and data sharing but may also lead to increased interference and resource contention.

3. Drone Mobility: Drones' speed and direction impact the network topology's dynamic nature. Rapid drone position changes require adaptive routing protocols and dynamic network management to maintain effective communication.

4. Communication Range: Antenna type, power, and frequency band factors determine the maximum distance between two drones that can communicate. Communication range affects network connectivity and the efficiency of data exchange among drones.

5. Radio Propagation Model: The mathematical model describing how radio signals propagate in the air depends on environmental factors such as weather, terrain, and obstacles.

6. Localization: The ability of drones to determine their position and orientation is essential for navigation, coordination, and data fusion.
7. Power Consumption: The amount of energy drones consume for flying and communication directly impacts the network's lifetime and reliability.
8. Frequency Band: The choice of the frequency band for communication affects data rate, interference, and regulatory considerations.
9. Cost-Efficiency: The trade-off between the performance and cost of drones and their communication systems is a crucial consideration.
10. Versatility: The ability of drones to adapt to different scenarios and tasks adds versatility to FANETs. Versatile drones can be employed in various applications, making the network suitable for multiple missions.

2.4 FANET Applications

A FANET can be used for various applications, such as military surveillance, disaster management, environmental monitoring, and traffic control [7].

2.4.1 Disaster Management

FANET can be used for various applications in disaster management, such as data collection and rescue operations. For example, FANET can be deployed to monitor a wildfire scenario where mobile sensors on the ground are investigating the devastated area. The drones can collect the fire identification data from the sensors and send it to the base station, where it can be processed and analyzed to localize the fire and provide rescuing services to the firefighters trapped inside the fire [8].

2.4.2 Communication Infrastructure and Connectivity

Drones contribute to network maintenance by inspecting telecommunication towers, antennas, and critical components with equipped cameras and sensors, swiftly identifying issues, and facilitating proactive maintenance. Drones are instrumental in installing and deploying communication equipment, particularly in challenging or hazardous locations. They also serve as valuable signal strength and coverage analysis tools, flying over specific areas to assess signal quality and optimize network performance. Drones with communication relay capabilities act as flying base stations, offering temporary connectivity during events or emergencies[5].

2.4.3 Border Surveillance and Security

Drones with high-resolution cameras and sensors can conduct aerial surveillance over vast and challenging terrains, providing real-time intelligence on border activities. This capability aids border control agencies in detecting and tracking potential threats, including illegal border crossings, smuggling activities, or unauthorized intrusions. Moreover, drones are particularly effective in areas with difficult accessibility, such as rugged terrain or dense vegetation, where traditional surveillance methods may be limited [9]. Drones with thermal imaging and night vision capabilities enhance surveillance during low-light conditions. This is crucial for continuous monitoring, as illegal activities often occur under the cover of darkness.

2.4.4 Smart Agriculture

Drones play a vital role in smart agriculture by offering diverse applications to enhance efficiency and sustainability. Drones equipped with high-resolution cameras and sensors provide detailed aerial imagery for crop monitoring, enabling farmers to assess plant health, detect diseases, and optimize resource use. They contribute to precision agriculture by mapping fields and facilitating targeted fertilizers, pesticides, and irrigation applications. Drones automate tasks like planting, seeding, and crop spraying, promoting uniform growth and minimizing environmental impact [10]. These drones also assist in livestock monitoring, water management, and post-harvest assessments.

2.4.5 Healthcare

Drones have emerged as valuable tools in the healthcare sector, offering innovative solutions to enhance medical services, improve logistics, and provide timely interventions. In various applications, drones contribute to healthcare by overcoming geographical barriers, reducing delivery times, and enabling swift responses.
One notable application is transporting medical supplies, such as vaccines, blood samples, and medications. Furthermore, drones facilitate rapid and reliable delivery to remote or hard-to-reach areas, especially during emergencies or in regions with inadequate infrastructure. In addition, drones equipped with defibrillators can deliver life-saving devices to individuals experiencing cardiac arrest before traditional emergency services arrive [11].

2.5 FANETs Communication Models and Standards

A critical challenge within FANETs involves the development of a robust communication model to ensure dependable data transmission among drones and between drones and GCS. Addressing this challenge requires consideration of various communication models and standards tailored to the unique dynamics of FANETs. Among the potential approaches are the utilization of renowned protocols such as Ad hoc On-Demand Distance Vector (AODV) and Dynamic Source Routing (DSR) for routing, adherence to IEEE 802.11 standards for wireless communication, and the integration of specialized protocols like MAVLink and UAVCAN designed explicitly for unmanned aerial systems.

2.5.1 IEEE 802.11

IEEE 802.11 refers to standards for implementing wireless local area networking (WLAN) communication. Commonly known as WiFi, these standards are developed by the Institute of Electrical and Electronics Engineers (IEEE). It encompasses various amendments and extensions, such as 802.11a, 802.11b, 802.11g, 802.11n, 802.11ac, and more, each introducing improvements in data rates, modulation schemes, frequency bands, and other features [12].

2.5.2 MAVLink Protocol

MAVLink, or Micro Air Vehicle Link, is a lightweight communication protocol for unmanned systems like drones and robotics. It is an open-source framework for exchanging information between drones and their GCSs. The protocol is characterized by its efficiency, relying on packet-based communication to exchange data related to vehicle state, sensor information, and command inputs. MAVLink's extensibility allows developers to define custom messages, ensuring adaptability to diverse unmanned systems and their specific requirements. With a focus on compatibility and platform independence, MAVLink is widely implemented in open-source autopilots, fostering a standardized and interoperable communication environment for drones and their operators [13]. Figure 4 illustrates the MAVLink frame Vs. IEEE 802.11 frame.

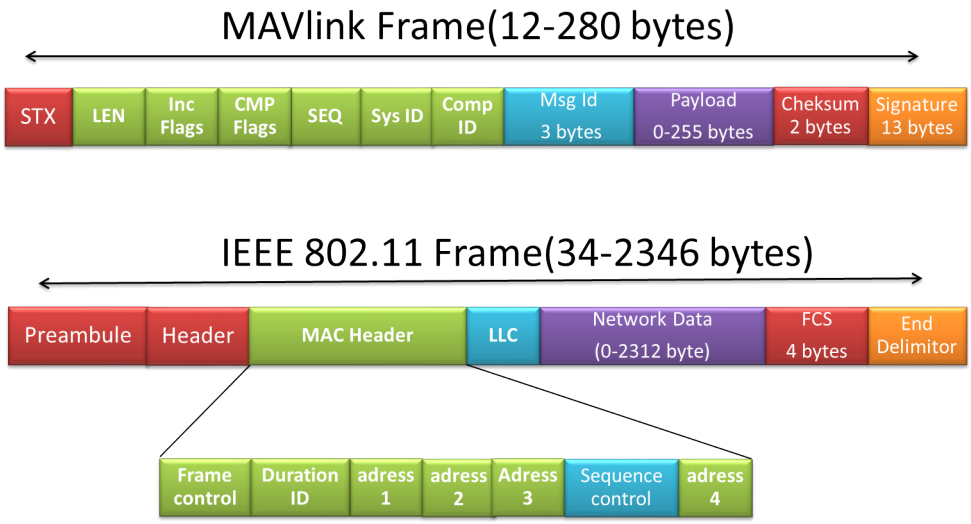


Figure 4: MAVLink frame Vs IEEE802.11 frame.

2.6 FANET Routing Protocols

Routing in FANET is critical due to the dynamic and mobile nature of drones and the changing network topologies. Various routing protocols have been adapted or designed to address the challenges specific to FANETs, these protocols are summarized as follows :

2.6.1 Ad Hoc On-Demand Distance Vector (AODV)

AODV is a reactive routing protocol commonly used in MANETs. It establishes routes between drones only when necessary, making it suitable for FANETs where network topology changes frequently [14]. AODV is designed to be scalable and adaptive to dynamic conditions.

2.6.2 Dynamic Source Routing (DSR)

DSR is another reactive protocol that allows drones to dynamically discover and maintain routes in the network [15]. It is well suited for FANETs due to its ability to adapt to rapidly changing topologies and rely on route caching for improved efficiency.

2.6.3 Optimized Link State Routing (OLSR)

OLSR is a proactive routing protocol that proactively maintains a routing table for all drones in the network [16]. It is designed to minimize control overhead and is suitable for drones with frequent and predictable movement, making it applicable to some FANET scenarios.

2.6.4 Adaptive DSDV (A-DSDV)

A-DSDV is an extension of the Destination-Sequenced Distance Vector (DSDV) routing protocol [14]. It adapts the traditional DSDV algorithm better to suit the mobility patterns of drones in FANETs, providing a more dynamic approach to route maintenance.

2.6.5 Zone Routing Protocol (ZRP)

ZRP is a hybrid routing protocol that combines the advantages of both proactive and reactive approaches. It divides the network into zones, employing proactive routing within zones and reactive routing between zones [15]. ZRP aims to strike a balance between adaptability and control overhead.

2.6.6 Prophet Routing Protocol

Prophet is a delay-tolerant routing protocol designed for networks with intermittent connectivity, making it suitable for FANETs. It utilizes historical encounter information to predict future contacts between drones, optimizing data forwarding strategies in scenarios with unpredictable connectivity [14].

2.6.7 Cluster-Based Routing

FANETs often leverage cluster-based routing approaches, where drones are grouped into clusters, and each set has a cluster head responsible for inter-cluster communication[16]. This helps in managing network scalability and improving routing efficiency.

2.6.8 Geographical Routing

Routing based on geographical information is relevant in FANETs where the location of drones is crucial. Geographical routing protocols use location information to make routing decisions, which can be advantageous in scenarios where drones must navigate specific geographic regions [15].

3 Security in FANETS, a General Overview

The security of FANETs is critical, considering the potential implications for safety and efficiency. FANETs, being decentralized and dynamic, present unique challenges in ensuring robust communication and safeguarding against malicious activities. In this section, we provide an overview of security requirements, types of attackers, potential attacks, and countermeasures essential for ensuring the safe and effective operation of FANETs.

3.1 Security Requirements in FANET

Exploring the security requirements within FANETs is a crucial area of research. It involves identifying specific security prerequisites that must be taken into account during the design and implementation of security measures for FANETs.

Confidentiality: is an essential aspect of FANET security, as it ensures that the data exchanged among the drones is protected from unauthorized access or disclosure. Confidentiality can be achieved by using encryption techniques, such as symmetric or asymmetric cryptography [17].
Confidentiality can also be enhanced by using authentication mechanisms, such as digital signatures or certificates, to verify the identity and integrity of the sender and receiver drones.

Integrity: is an essential aspect of FANET, as it ensures that the data exchanged among the drones is authentic, accurate, and consistent. Integrity can be achieved by using cryptographic techniques, such as digital signatures, message authentication codes, and hash functions, to verify the source and content of the messages. Integrity is essential for FANET, enabling the drones to coordinate actions, avoid collisions, share information, and perform collaborative tasks.

Availability : is a crucial aspect of FANET, ensuring that the network remains operational and accessible as needed. This is particularly vital for the seamless communication and operation of drones and ground stations, where uninterrupted connectivity is essential. Guaranteeing high Availability in FANET contributes to the network's reliability, supporting the continuous functioning of aerial and ground components.

Authenticity : revolves around verifying the identity of communication participants, ensuring that they are genuinely who they claim to be. The principle of authenticity plays a crucial role in preventing unauthorized entities from participating in the network [18]. Verifying the legitimacy of messages exchanged within the FANET ensures a secure and trustworthy communication environment, mitigating the risk of malicious entities infiltrating the network and contributing to the system's overall integrity.

Privacy and Anonymity: involves ensuring that sensitive information, such as the identities of drone operators or locations, is shielded from unauthorized access or disclosure [19]. Privacy protection in FANET aims to establish a secure environment where users can operate drones or communicate without the risk of their own data being compromised. This includes employing encryption techniques, secure communication protocols, and anonymity features to mitigate potential privacy threats and uphold the privacy rights of individuals participating in FANET activities.

Traceability: is essential for tracking and understanding the flow of data, communications, and activities within FANET. Traceability mechanisms help identify the source and destination of data packets, track communication paths, and monitor the actions of network nodes. This capability is valuable for security purposes, enabling the detection of anomalies, unauthorized access, or potential security breaches within the FANET.

3.2 Adversary Types in FANET

Various types of adversaries can pose security threats. Understanding these adversary types is crucial for designing effective security measures. Some common adversary types in FANET are defined as follows:

3.2.1 Actively (Active or Passive)

Active Attacker: An active attacker in FANET possesses the capability to alter, remove, or generate new messages actively. This involves direct interference with the network's communication to affect its performance.

Passive Attacker: A passive attacker, on the other hand, engages in eavesdropping, merely listening to the exchanged communications without directly harming the network[20].

3.2.2 Behaviourally (Malicious or Rational)

Malicious Attacker: A malicious attacker in FANET aims to execute destructive attacks that can cause damage to the network. Their primary objective is to disrupt or compromise the network's functionality through various methods.

Rational Attacker: A rational attacker, in contrast, seeks personal benefit from the attack. Their actions are more predictable than malicious attackers, as specific personal goals or gains drive them [21].

3.2.3 Locationally (Insider or Outsider)

Insider Attacker: An insider attacker in FANET is an authenticated network member who can perform severe attacks [22]. Due to their authorized status, they pose a significant threat to the network.

Outsider Attacker: An outsider attacker does not have direct participation privileges in the network. Their limited capabilities make them less dangerous than insider attackers with authenticated access.

3.2.4 Proprietarily (Global or Local)

Global Adversary: A global adversary in FANET controls a large area of radio stations deployed across the network. They can easily detect mobile entities within the covered area, enabling them to exert a significant influence.

Local Adversary: A local adversary controls fewer network entities than a global one, limiting their power to a smaller covered area [23]. Their capabilities are restricted compared to a global adversary.

3.2.5 Movably (Static or Dynamic)

Static Adversary: A static adversary's eavesdropping stations are fixed in specific network locations. They may have limitations regarding coverage and flexibility.

Dynamic Adversary: A dynamic adversary's eavesdropping stations are mobile and can move across the observed map. Their strength depends on the mechanisms and algorithms used, allowing them to adapt to changing network conditions [24].

3.2.6 Occasionally (Permanent or Temporal)

Permanent Observer: A permanent observer in FANET is more dangerous as they continuously gather data and eavesdrop on communications occurring at all times. Their consistent monitoring seriously threatens the network's security [20].

Temporal Observer: A temporal observer selectively eavesdrops at specific periods based on their interests, intentions, and benefits. Their monitoring activities are intermittent, making them less pervasive than permanent observers.

3.3 Potential Attacks in FANETs

Potential Attacks in FANETs encompass a wide array of security risks ranging from physical layer vulnerabilities to sophisticated cyber threats, necessitating a comprehensive understanding and mitigation strategy.

1. **Malicious Hardware/Software Attack:** A malicious hardware/software attack in FANET involves deliberately introducing compromised elements, such as drones or ground stations, intending to undermine the network's security and functionality. This attack may include tampering with firmware or software, injecting malware, or exploiting vulnerabilities to gain unauthorized control or access [25]. The impact ranges from compromising data integrity and unauthorized access to disrupting communication and navigation within the network.

2. **Wormhole Attack:** In this attack, the malicious drone, known as the wormhole, establishes a high-speed, covert communication link between two distant points in the network, creating a shortcut for data transmission [26]. As a result, the attacker can replay or selectively forward packets, potentially misleading the network and compromising its integrity. Wormhole attacks are also known as Selective Forwarding attacks.

3. **Grayhole Attack:** In a Grayhole attack, a malicious drone selectively drops or modifies data packets passing through it, behaving as a semi-cooperative participant. The grayhole attacker may selectively drop packets based on specific criteria, such as packet content, source, destination, or data types. This makes it challenging to detect the attack since not all packets are discarded, and the attack does not exhibit a complete denial of service [27].

4. **Blackhole Attack:** In this type of security threat, a rogue drone, termed the blackhole, deceitfully attracts incoming network traffic by advertising itself as having the most efficient route to the destination. However, instead of forwarding the packets to their intended recipients, the blackhole drone selectively drops or absorbs them, leading to a denial of service for legitimate drones [28].

5. **Sinkhole Attack:**A Sinkhole attack is a cybersecurity threat in which an attacker redirects legitimate network traffic to a compromised drone, the sinkhole. This malicious drone captures and potentially manipulates the diverted traffic for unauthorized purposes [28]. The attacker may use the sinkhole to eavesdrop on sensitive information, launch further attacks, or disrupt communication within the network.

6. **Sybil Attack:**A Sybil attack is a security threat in which a malicious actor impersonates multiple fake identities or drones within a network to manipulate or disrupt its regular operation. In FANETs, a Sybil attack can be particularly damaging as the attacker gains disproportionate influence or control over the network by having a more significant presence than their actual share [29].

7. **Bad-mouthing Attack:**A Bad-mouthing attack is a type of security threat in which a participant within a network maliciously spreads negative or false information about other drones or entities in the network. The attacker aims to undermine the reputation of specific drones by disseminating misleading or harmful data, which can lead to a loss of trust among network participants [30].

8. **Flooding Attack:** A Flooding attack is a type of cybersecurity threat where an attacker inundates a network, system, or service with an overwhelming volume of traffic, causing it to become unavailable or significantly slowing down its performance. The attack exploits the network's resources, such as bandwidth or processing capacity, by sending excessive requests or data packets [31]. Flooding the target system can lead to a denial of service (DoS) or distributed denial of service (DDoS) scenario.

9. **On-Off Attack:** In this attack, a malicious drone in the network alternates between periods of regular activity and prolonged periods of inactivity or sleep, consuming minimal power during active phases [32]. The goal is to deceive the network's energy management mechanisms, making it challenging for the network to accurately predict and manage the energy consumption of the malicious drone.

10. **Packet Injection Attack:** A Packet Injection attack is a cybersecurity threat where an attacker introduces unauthorized packets into a network, aiming to disrupt communication, manipulate data, or compromise the integrity of the targeted system. By sending malicious packets into the network infrastructure, the attacker can exploit vulnerabilities, potentially leading to data corruption, unauthorized access, or denial of service [33].

11. **Jamming Attack:** A jamming attack deliberately disrupts communication systems by emitting interference on the same frequencies as transmitting signals. Typically involving signal jammers, these attacks overwhelm and drown out legitimate signals, leading to the loss of connectivity or functionality in targeted devices [34]. Jamming attacks can impact a wide range of systems, from wireless networks and radio communications to GPS signals and radar systems.

12. **GPS Spoofing Attack:** GPS spoofing is a cyber-attack where the GPS signals received by a device are manipulated to deceive it about its actual location. In this attack, the attacker generates fake GPS signals that appear authentic to the targeted device. By broadcasting these counterfeit signals, the attacker can trick navigation systems, such as GPS-enabled drones, into believing they are in a different location. This can lead to malicious activities like misguiding drones, redirecting shipping routes, or disrupting location-based service [35].

13. **GPS Jamming Attack:** A GPS jamming attack is a deliberate interference with GPS signals to disrupt or block the reception of accurate location information by GPS-enabled devices. In this attack, the perpetrator employs signal jammers that emit radio frequency interference on the same frequencies used by GPS satellites. The interference overwhelms and drowns out legitimate GPS signals, causing navigation devices to lose accurate positioning information [36].

14. **Man in the Middle Attack:** A Man-in-the-Middle (MitM) attack is a cyber threat where an unauthorized third party intercepts and potentially alters communication between two parties without their knowledge. The attacker positions themselves between the communicating entities, secretly eavesdropping on or manipulating the exchanged data [25]. This attack can occur in various communication channels, including WiFi networks, MANET, VANET, and FANET.

15. **(ADS-B) Attack:** An ADS-B (Automatic Dependent Surveillance–Broadcast) attack refers to the manipulation or interference with the information transmitted by aircraft through the ADS-B system. ADS-B is a real-time surveillance technology used in aviation to track aircraft. An attack on ADS-B can involve broadcasting false position, altitude, or identification information, leading to potential risks such as unauthorized access to airspace, collision risks, or misinformation for air traffic control systems. By compromising the integrity of ADS-B data, attackers can create deceptive scenarios that threaten aviation safety [37].

16. **Sniffing Attack:** A sniffing attack, also known as packet sniffing or network sniffing, is a form of cyber attack where an unauthorized entity intercepts and monitors network traffic to capture sensitive information. Attackers can extract usernames, passwords, or other confidential data by analyzing data packets as they traverse a network [38]. Sniffing attacks often exploit unsecured or poorly configured networks, making them vulnerable to eavesdropping.

17. **Denial of Service (DoS) Attack:** A Denial of Service (DoS) attack is a malicious attempt to disrupt the normal functioning of a computer system, network, or service by overwhelming it with a flood of illegitimate requests or traffic. The primary goal of a DoS attack is to make the targeted system unavailable to users, causing a temporary or prolonged disruption of services [39]. Attackers may use various methods, such as flooding the network with traffic, exploiting vulnerabilities, or exhausting system resources. Distributed Denial of Service (DDoS) attacks involve multiple sources, making them more challenging to mitigate.

18. **Energy Drain Attack:** An Energy Drain attack is a type of cyber threat aimed at depleting the energy resources of a targeted system or device. In the context of FANETs, attackers may employ techniques to utilize energy resources inefficiently, leading to accelerated battery depletion inefficiently [40]. This attack can have significant consequences, especially when energy-efficient operation is critical for the device's functionality or longevity.

4 Countermeasures

Various existing countermeasures and ongoing research efforts in network security to mitigate the previous attacks in FANETs exist.

4.1 Authentication and Authorization

Authentication and authorization are crucial concepts in information security that work together to control access to resources and ensure data confidentiality, integrity, and availability. Authentication involves verifying the identity of drones or processes attempting to access a network. This process typically requires presenting credentials, such as passwords and cryptographic keys, verified against stored records to confirm the drone's identity [41]. Once authenticated, authorization determines the access privileges granted to the authenticated entity. Authorization specifies what actions or resources a drone can access based on its identity, role, or other attributes. Access control mechanisms, such as access control list (ACL) or role-based access control (RBAC), are commonly used to enforce authorization policies and manage permissions. Authentication and authorization mechanisms are critical in implementing security policies and protecting sensitive information from unauthorized access or misuse.

4.2 Encryption and Cryptography

Encryption and cryptography are fundamental to modern cybersecurity, protecting sensitive information and communications from unauthorized access and tampering. Encryption involves transforming plaintext data into ciphertext using algorithms and keys, rendering it unreadable without the corresponding decryption key [42]. Cryptography encompasses various techniques and principles to secure data transmission and storage, including cryptographic algorithms, protocols, and cryptographic key management. It is vital in ensuring confidentiality, integrity, and authenticity across various applications, such as secure communication, digital signatures, and data protection within FANETs. Encryption and cryptography are indispensable tools in safeguarding digital assets and privacy FANETs.

4.3 Certificate Revocation

Certificate revocation is used in critical infrastructure (PKI) systems to invalidate digital certificates before expiration. This process becomes necessary when a certificate must be revoked for various reasons, such as a compromise of the private key, a change in the certificate holder's status, or errors in the certificate issuance process [43]. Certificate revocation is essential for maintaining the integrity and security of PKI-based systems because it prevents using compromised or unauthorized certificates for malicious purposes. There are several methods for certificate revocation, including certificate revocation list (CRL) and online certificate status protocol (OCSP), which allow relying on parties to check the validity status of certificates issued by a certificate authority (CA) in real time. Revoked certificates are added to CRLs or flagged as such in OCSP responses, enabling relying parties to identify and reject no longer trusted certificates. Effective certificate revocation mechanisms are critical for ensuring the trustworthiness and reliability of digital certificates within FANETS.

4.4 Intrusion Detection Systems (IDSs)

Intrusion Detection Systems are specialized security mechanisms designed to monitor and analyze the network traffic and behavior within the FANET environment. These systems aim to detect unauthorized, malicious, or abnormal activities that may indicate potential security breaches or intrusions [44]. In a FANET context, IDS typically function by examining various parameters such as communication patterns, drone behavior, packet content, and network anomalies. They may utilize both signature-based and anomaly-based detection techniques:

4.4.1 Signature-based Detection

It relies on a database of known attack patterns or signatures. It compares incoming network traffic against these signatures to identify matches indicative of known attacks.

4.4.2 Anomaly-based Detection

It establishes a baseline of normal network behavior. It then flags deviations from this baseline as potential intrusions or threats. This method is beneficial for detecting novel or previously unseen attacks.

IDS in FANETs must address several challenges unique to the dynamic and decentralized nature of aerial ad-hoc networks.

These challenges include drone mobility, limited resources, varying network conditions, and potential communication disruptions.

4.5 Machine Learning and AI-based Solutions

Leveraging machine learning(ML) and AI techniques is crucial for detecting and responding to real-time network attacks. ML and AI-based solutions present promising avenues for bolstering security in FANETs. These advanced technologies can detect anomalies, analyze drone behavior, and predict potential attacks by leveraging historical data and real-time observations. By employing adaptive defense mechanisms and optimizing resource allocation, these systems can enhance the network's resilience against evolving threats while minimizing performance impact. Integrating these approaches into FANETs allows for proactive threat mitigation, distributed decision-making, and continuous improvement of security measures in dynamic aerial environments [45].

4.6 Trust Management

Trust management is vital in enhancing security and attack detection in FANETs. Trust management mechanisms help evaluate the reliability and trustworthiness of drones or devices. By assessing factors such as drone behavior, communication patterns, and reputation, trust management systems enable FANETs to make informed decisions regarding data transmission, routing, and collaboration [22].

Regarding attack detection, trust management can be leveraged to identify and mitigate malicious activities within the network. Drones with high trust levels are more likely to be considered reliable sources of information. In contrast, those with low trust levels may be closely monitored or excluded from critical network operations. Trust-based anomaly detection techniques can also detect deviations from normal behavior, flagging suspicious drones or activities that may indicate potential attacks.

Furthermore, trust management facilitates collaborative defense mechanisms in FANETs, where drones can share information and collectively respond to security threats. By establishing trust relationships and fostering cooperation among drones, FANETs can enhance their resilience against attacks, including jamming, spoofing, and drone misbehavior.

4.7 Deception Technologies

Deception technologies play a crucial role in enhancing the security and resilience of networked systems against cyber threats. Deception technologies within FANETS encompass a range of strategies and tools aimed at misleading adversaries and detecting malicious activities. These technologies strategically deploy decoys [46], honeypots [47], and honeytokens [48] within the network, creating a complex web of false targets and information. By enticing attackers to engage with these deceptive elements, security teams can gain valuable insights into their tactics, techniques, and objectives. Furthermore, deception technologies in FANETS help to divert attackers' attention away from critical assets and infrastructure, buying precious time for defenders to respond effectively. Integrating advanced analytics and ML algorithms, these deception mechanisms continuously adapt to evolving threats, ensuring proactive threat detection and response. Ultimately, in the dynamic and unpredictable environments of FANETS, deception technologies serve as essential components in the arsenal of defensive measures, bolstering the overall cybersecurity posture and safeguarding mission-critical operations.

These countermeasures represent just a subset of the extensive research efforts underway in cybersecurity to defend against FANET attacks.

5 Conclusion

This chapter has delved into the intricate world of Flying Ad-Hoc Networks, elucidating their architecture, characteristics, design considerations, applications, communication models, and routing protocols. Following this, we provided a comprehensive overview of the myriad security challenges inherent in FANETs, highlighting the potential attacks and the importance of addressing these issues for the seamless operation of such networks. In response to these challenges, we explored existing countermeasures employed in FANETs, showcasing the ongoing efforts to fortify their security posture. It becomes evident that while FANETs hold immense promise for various applications, their viability and effectiveness hinge upon robust security frameworks. In the next chapter, we will delve deeper into one of the existing countermeasures: trust management.

Chapter II: Trust Management in FANETs: A State of Art and Taxonomic Study

"Trust is the antidote that overcomes fear, and fear is the greatest inhibitor to a relationship that welcomes and nurtures new ideas."

– John Pepper

1 Introduction

Cryptographic methods, securing communication protocols, and certificate revocation are promising strategies that protect FANET from potential attacks. However, drones are vulnerable to cyber-attacks from malicious actors who can exploit their weaknesses, such as poor encryption, outdated firmware, or weak authentication. They can also take control of a drone and use it for various attacks, such as gray hole attacks, Sybil attacks, denial-of-service, or physical damage [49]. For instance, a hijacked drone can tamper with the communication between other drones and the ground station or collide with a critical infrastructure or a populated area. Hence, it is crucial to secure the drones from cyber-attacks and ensure their safe and reliable operation in a network environment. Trust management is one of the pivotal techniques in fortifying the reliability, integrity, privacy, security, and performance of Flying Ad-Hoc Networks (FANETs). This is achieved by systematically assessing the trustworthiness of network nodes and strategically choosing optimal partners for collaboration. The primary objective of trust management is to tackle the challenge of determining how entities within the network, including drones, can engage with a heightened level of confidence in the dependability and intentions of other entities [50]. The key contributions made in this chapter can be summarized as follows:

- Comprehensive Overview of State-of-the-Art Trust Management Techniques: We extensively examine the latest trust management techniques in Flying Ad-Hoc Networks (FANETs). This overview spans the initial proposal of integrating trust in FANETs to the present, providing a comprehensive understanding of the field's evolution.
- Innovative Classification of Trust Management Techniques: We introduce a pioneering classification system for trust management techniques in FANETs. This classification is based on distinct criteria such as design objectives, architecture, and performance. The identified categories include reasoning models, machine learning-based models, and blockchain-based schemes. This classification aims to clarify the nature and mechanisms of these techniques, which are particularly beneficial for new researchers entering the field.

- Comparative Analysis of Trust Management Techniques: We thoroughly compare and analyze the presented trust management techniques using various criteria. These include trust properties such as availability, integrity, context awareness, and computational overhead. Trust requirements such as effectiveness, robustness, and privacy protection.

It is crucial to highlight that this research is pioneering in the network security domain, filling a void in the existing literature. It complements and adds depth to other significant research endeavors, such as those presented in [51] and [52]. These previous works have offered valuable insights and guidance on trust management techniques tailored to diverse network environments. By contributing novel perspectives and insights, our research aims to enhance the collective understanding of trust management within the broader context of network studies. The remainder of this chapter is structured as follows: Section 2 introduces the concept of trust and its significance in network communication. Section 3 outlines the extended related works. Section 4 presents a novel classification of trust management techniques proposed for FANETs. In Section 5, we offer a comparative analysis of these techniques. Finally, Section 6 concludes the chapter by summarizing key insights and emphasizing the importance of robust trust management solutions.

2 Trust Management: A Crucial Security Aspect in FANET

This section presents the fundamental aspects of trust management in FANET.

2.1 Notion of Trust

Trust management is a concept that refers to the process of establishing, maintaining, and evaluating the trustworthiness of entities in a network [49]. Trust management can be applied to various domains, such as e-commerce, social networks, peer-to-peer systems, cloud computing, and cybersecurity. It involves defining trust policies, metrics, and mechanisms that specify how trust is established, measured, and updated among the entities. It also involves designing and implementing trust models, architectures, and protocols that enable the entities to communicate and cooperate securely and efficiently based on their trust levels.

2.2 Direct and Indirect Trust

Trust can be established in two ways: direct and indirect. Direct trust is based on the first-hand experience or observation of an entity's actions, while indirect trust is based on the recommendations or opinions of other trusted entities [53]. For example, in FANET, direct trust can be established by evaluating the behavior of neighboring drones. In contrast, indirect trust can be demonstrated by collecting feedback from other drones interacting with the target drone.

2.3 Trust Management Process

Trust management in FANETs entails numerous interrelated procedures for assessing and managing the trustworthiness of drones in the network. Figure 5 illustrates the six-step trust management process for FANETs: Information gathering, trust computation, trust aggregation, trust propagation, and decision making.

2.3.1 Information Gathering

The information-gathering step in trust management is collecting and analyzing data about the trustworthiness of different entities in FANET [54]. This step involves identifying the information used in trust evaluation, such as drone behavior, drone performance, and feedback from other drones. During the information-gathering phase, drone locations, speed, communication patterns, message delivery rates, transmission delay, drone energy, and other parameters are monitored. However, the information gathered can be used to evaluate the drone's reliability and competence to update the trust values accordingly.

2.3.2 Trust Computation

Trust computation is a crucial step in trust management; it involves applying mathematical models and algorithms to quantify the trustworthiness of a drone based on its behavior, reputation recommendations, or other factors. Several trust computation methods have been used, such as Fuzzy Logic, Dempster Shafer, Bayesian Inference, Policy-based Method, etc. Trust calculation may rely on historical data, encompassing factors like prior interactions, instances of successful collaboration, and the sustained consistency of behavior throughout time [55].

2.3.3 Trust Aggregation

Trust aggregation plays a crucial role in trust management within FANETs, mainly when dealing with multiple trust information sources. Trust aggregation aims to combine direct and indirect trust values from different sources into a single trust value that represents the overall trustworthiness of a target entity [56]. Trust aggregation can be performed using various methods, such as arithmetic mean, weighted mean, geometric mean, probabilistic methods, etc. The choice of the aggregation method depends on the characteristics of the trust model, the trust sources, and the application domain.

2.3.4 Trust Propagation

Once trust relationships are established, trust values can be propagated or transferred from one drone to another. This propagation can occur through various mechanisms, such as recommendations or direct experiences [21]. The idea is that if drone A trusts drone B and drone B trusts drone C, then there is a propagation of trust from A to C. Trust propagation often involves transitive trust relationships; for example, if A trusts B and B trusts C, then A may also somewhat trust C. The trust propagation degree depends on the rules, algorithms, or models used in the trust management system. Trust propagation is dynamic and may change over time based on the evolving relationships and interactions within FANET. New information, feedback, or events can influence the trust values and network.

2.3.5 Decision Making

The decision-making process is an important step that should be able to adapt dynamically to new information and updates [57]. Moreover, predefined trust policies and rules guide trust decisions and help to establish criteria for trust, specifying thresholds, conditions, or standards that drones must meet to be considered trustworthy. Once a decision is made, trust management systems may enforce it by regulating access to resources, granting permissions, or facilitating interactions based on the assessed trustworthiness of the involved drones.

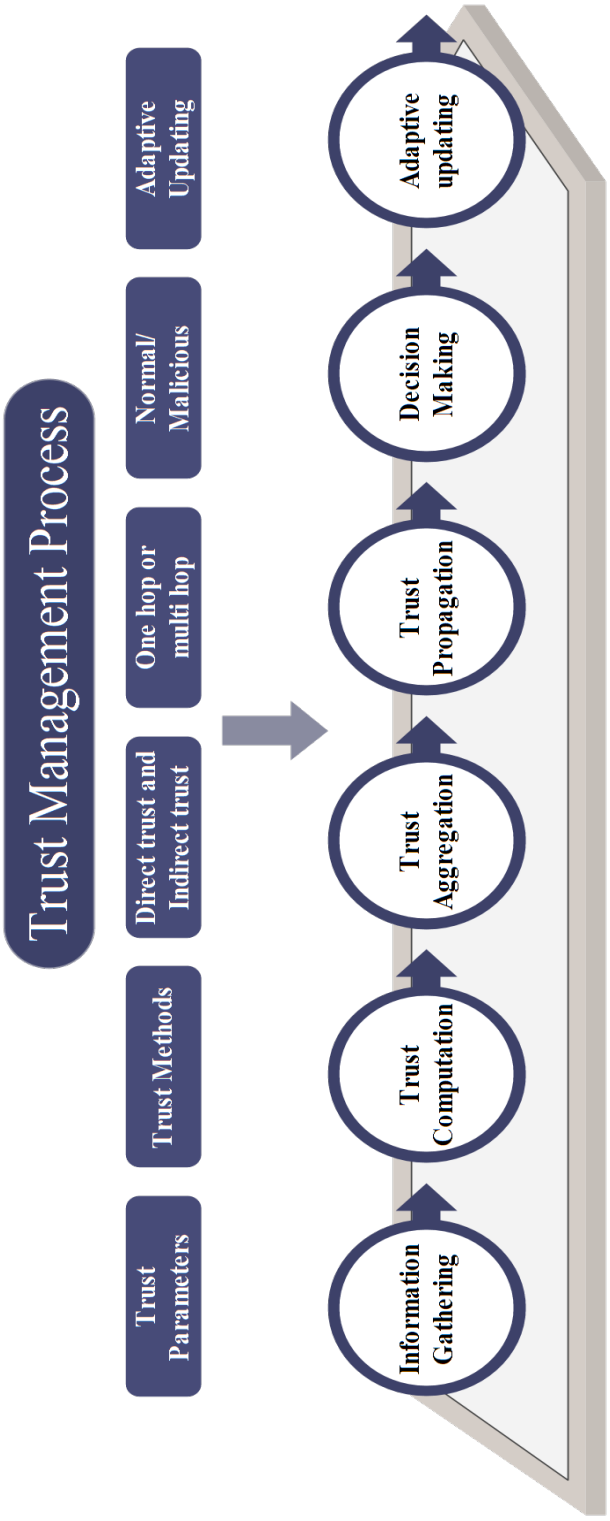


Figure 5: Trust management process

2.3.6 Adaptive Updating

Adaptive updating is the last step in the process, where the trust values of different drones are updated based on the feedback from previous interactions. Consequently, the system continuously monitors and analyzes recent actions to update the trust results. Subsequently, the trust management system adapts trust assessments to the specific context or scenario, allowing for more nuanced and situation-aware decision-making. Moreover, the system monitors the performance of trust decisions and adjusts thresholds based on the changing dynamics of the network, ensuring that the trust management process remains effective.

3 Extended Related Work

Various trust evaluation methods for insider attack detection, including reasoning and machine learning models, are available.

3.1 Reasoning Models based Trust Management Schemes

Reasoning evaluation models are used for trust evaluation in FANETS, including, but not limited to, Fuzzy Logic, Bayesian Inference, Game Theory, and Policy-based Methods.

Singh et al.[56] suggested a Fuzzy Logic for trust evaluation in FANET to deal with the behavioral unpredictability of drones. They used a multicriteria Fuzzy classification approach to classify nodes based on their behavior and performance in complicated environments. To distinguish between selfish and malevolent drones, they evaluated the trust value of each node using quality of service and social parameters (recommendation). However, the establishment of rules and the determination of membership functions could be challenging.

Bayesian Inference has been proposed by Barka et al. [58] in FANET; the model calculates probability to represent trust value. They integrate a blockchain-based solution to ensure the security and trust between drones and GCS. However, it confuses probability's randomness with subjectivity and trust uncertainty.

Zhang and Zhu [59] presented a Game Theory to defend against civil GPS signal spoofing for civilian drones. They offer a signaling game framework in which the GPS receiver may strategically discern the actual position when the attacker tries to deceive it with a forged and purposely designed signal. However, it is challenging to create game rules.

A critical application of drones involves simultaneous coordination with another ad hoc network operating on the ground, termed cooperative ad hoc networking. Sharma et al. introduce an ambient network framework designed to coordinate between ground-based and FANET; these networks engage in extensive data sharing, utilizing a Fuzzy Cognitive Map. The framework proposes a fault-tolerant and robust connectivity strategy, incorporating neural, Fuzzy, and genetic modules. The neural and decision system for guiding the aerial network is formed using a quaternion Kalman filter and its variant.

To counter a coordinated GPS-spoofing attack in FANETs, Bada et al.[35] provide a Policy-based distributed detection system. They determine the trust levels based on the burglary scene and the drones' locations during the attack using absolute power and carrier-to-noise density ratio. The trust model, which utilizes a Beta and Weibull distribution, is implemented to reduce the propagation of bad-mouthing attacks and categorize different signals.

Berka et al. [60] suggested an energy-efficient method for FANET that is reputation-aware. Their technique calculated trustworthiness by considering the number of legal and illegal drone interactions required to create trust with little energy and indirect trust values.

In [61], Berka et al. introduced UNION, a second model to distinguish between permitted and harmful drone activity. They offer a new context-aware, trust-based method to reduce produced error ratios while maintaining the appropriate security standards. Their approach builds inter-drone trust while estimating the present context regarding drone energy, movement pattern, and enqueued packets, ensuring complete context awareness in the overall honesty judgment.

Barka et al. [62] suggested FNDN (Flying Named Data Networking) as a communication architecture to prevent man-in-the-middle attacks. This design is built on the integration of trust mechanisms. When propagating data, their model system employs a trust management method to handle the network attack risk.

Bhargava et al. introduced a Kalman trust estimator (KATE) in [63]. KATE assesses drone misbehavior by integrating direct and indirect trust values. Kate investigates how previous trust values maintained on the Internet of Drones (IoDs) affect present trust levels.

Carlos et al. in [64] proposed UAVouch, a method for detecting and finding UAVs in a group. UAVouch employs a movement plausibility check and a public-key-based authentication method to detect intruders who stray from the group's expected path. Previous works apply reasoning models that lack intelligent and dynamic support for trust management, impacting factor selection and rule formulation.

3.2 Machine Learning-Based Trust Management Schemes

Previous models deal with several serious problems, including a lack of crucial evaluation data, the requirement for big data processing, the need for trust expression, and the expectation of automation. machine learning has been used in trust evaluation to solve these issues intelligently and automatically.

For example, SUN et al.[65] used supervised ML algorithms such as Linear Regression(LiR) and Kmeans for FANET to detect malicious drones. They proposed a Malicious Drones Detection Algorithm (MDA) based on supervised learning and clustering algorithms.

For a suitable drone selection, Khan et al.[66] employed the Support Vector Machine (SVM) algorithm for a reputation-based auction mechanism to simulate the interaction between outsourcing-interested business agents; they treated the trust evaluation problem as a classification problem.

Greco et al.[34] proposed a drone network jamming detection framework. It is built on a distributed method using supervised ML, namely Multi-layer Perceptron(MLP) and Decision Tree(DT). Their approach computes the characteristics of various preset metrics, such as throughput, and RSSI, which change during a jamming attack and may thus be utilized to identify it, given a reference data packet trace set. However, the crucial step is choosing the appropriate features and algorithms, and acquiring labeled datasets is also essential.

Khaista et al. al.[44] introduced a research investigation centered on an in-depth examination of IDSs employing ML. The study leverages the UNSW-NB 15 dataset to implement the cognitive lightweight logistic Regression (LoR)technique. Utilizing ML, the research endeavors to construct an Internet of Things (IoT)based drone network capable of detecting potential security intrusions. When a queuing and data traffic model is employed in this IoT-based drone network to implement DT, Random Forest (RF), XGBoost, AdaBoost, Bagging, and LoR.

Donpiti et al.[29] provided a method for detecting Sybil attacks using FANETs-based IoFT that uses the physical layer properties of the radio signals picked up by two ground nodes from the drones. A supervised ML technique is used, and various classifiers from the Weka workbench platform are tested. The experiment used the received signal strength difference (RSSD) and the time difference of arrival (TDoA).

Shrikant et al.[67] offered a method to find the network's adverse drones. The approach uses linear regression to determine a drone's reputation or trust value in the network. Then, based on Logistic Regression classification, the drone is classified as benign or malicious using the previously discovered trust value.

Some researchers have also proposed using deep learning(DL) for trust evaluation. For example, Rabie et al.[68] presented a real-time data analytics system based on DL for examining threats in FANET intrusion detection. The foundational framework of this system relies on Recurrent Neural Networks (RNN). Detecting anomalies involves collecting and utilizing data analytics to scrutinize network data. Within each FANET, an agent is assigned the responsibility of gathering data, with the expectation that the agent will document real-time data from the FANET.

Escorcia et al. [69] employed the Sea Turtle Foraging technique with Hybrid DL-based Intrusion Detection (STFA-HDLID) to detect and classify intrusions in the IOD environment. The initial step involves data pre-processing, specifically using min-max normalization to standardize incoming data. Furthermore, the Sea Turtle Foraging Algorithm (STFA) is the foundation for the feature selection process. A Deep Belief Network (DBN) is used with the Sparrow Search Optimization (SSO) method for classification.

To the best of our knowledge, there has been little research on applying federated learning to improve ML efficiency in FANET scenarios. However, for instance, Kanchan et al.[70] provided a federated technique based on group signatures that improves drone identity protection and greatly minimizes overheads by removing the requirement for crucial information sharing between drones.

Similarly, based on Federated Reinforcement Learning(FRL), Mowla Nishat et al.[36] designed a dynamic and innovative protective system to thwart jammer attacks. Their solution involved the design of an adaptable defense mechanism utilizing a model-free Q-learning process. An on-device federated jamming detection system oversees this process, and its adaptability is further enhanced by incorporating an adaptive exploration-exploitation epsilon-greedy strategy.

Yazdinejad et al. [71] proposed a FL for drone authentication paradigm for IoT networks. In the suggested model, drone authentication is done using a Deep Neural Network (DNN) architecture, and local drone optimization using stochastic gradient descent (SGD) is implemented.

For drone-assisted MEC Servers, Wang et al.[72] suggested SFAC, a secure FL system. They provide a blockchain-based collaborative learning architecture for drones that enables safe local model update sharing and contribution verification without needing a central curator.

Based on the methods analyzed in the previous paragraphs, it is evident that the ML schemes have significant limitations. While reinforcement learning offers a promising solution due to its dynamic model adjustment through environment interaction and support for context awareness, FL ensures privacy preservation and conserves drone energy. However, DL can analyze large volumes of network traffic and identify patterns and anomalies that indicate malicious activity.

4 Classification of Trust Management Techniques for FANET

In wireless sensor networks, the researchers [73] categorized trust management techniques into centralized and distributed trust schemes. Conversely, in vehicular ad hoc networks, they [74] classified these techniques into three distinct categories: (a) entity-based, (b) data-based, and (c) combined. An alternative classification emerged in the context of the Internet of Vehicles(IoVs) [52], which sorted trust techniques into two overarching categories: those leveraging artificial intelligence-enabled methods and those incorporating emerging technologies.

In the social IoT, the authors [75] introduced various techniques such as Weighted Sum, Belief Theory, Bayesian Inference, Fuzzy Logic, and Regression Analysis. Presenting a pioneering effort in securing FANETs, this section proposes a novel classification of trust management techniques explicitly tailored for FANETs. It is crucial to highlight that this research represents the first work in this domain. As depicted in Figure 6, the proposed classification divides trust management techniques into three main categories: Reasoning models, Machine Learning (ML) models, and Blockchain-based schemes.

4.1 Reasoning Models Based Trust Management Schemes

In FANET, reasoning models play a crucial role in trust management, enabling the analysis and decision-making processes concerning drone behavior and interactions. These strategies are instrumental in evaluating whether a drone can be deemed trustworthy, relying on available information and predefined criteria. Some typical reasoning models for trust management in FANET include:

4.1.1 Policy-based Method

The Policy-based Method is an approach where trust decisions are made based on predefined policies distributed across the network. This method involves the establishment of a set of rules, guidelines, or procedures that dictate the conditions in which trust is granted or denied to drones within the FANET[35].
In a Policy-based distributed trust model, each drone in the network adheres to these predefined policies when interacting with others. These policies could encompass various criteria, such as communication behavior, reliability, past performance, or adherence to security protocols.

4.1.2 Fuzzy Cognitive Map

A Fuzzy Cognitive Map is a graphical representation of the causal relationships between concepts in a complex system. It can be used to model the trustworthiness of drones in a network based on their interactions and feedback. A fuzzy Cognitive Map can capture the uncertainty and provide a way to reason and learn from data. A Fuzzy Cognitive Map can also be used to simulate the behavior of the network under different scenarios and to evaluate the impact of interventions or policies[76].

The basic idea behind a Fuzzy Cognitive Map is to represent a system as a network of interconnected concepts, where the relationships between concepts are characterized by weights that indicate the strength and nature of the connections.

These connections can be positive, negative, or neutral, reflecting the influence one concept has on another.

4.1.3 Game Theory

Game Theory is a mathematical framework that models interactions and decision-making among rational entities and players in strategic situations with conflicting or cooperative objectives. Game Theory can be applied to network security problems, where the agents are the attackers and the network's defenders, and their goals are to maximize their utility or payoff. Game Theory can help to analyze the optimal strategies for both sides, predict the outcomes of their interactions, and design mechanisms that can improve the security and efficiency of the network. Game Theory can also measure the network's security level by quantifying the agents' incentives and costs. Different game models can capture various aspects of network security problems, such as cooperative games, non-cooperative games, static games, dynamic games, complete information games, incomplete information games, etc. [77].

4.1.4 Information Entropy Model

Information Entropy is a concept that measures the uncertainty or unpredictability of a random variable [78]. It can quantify the trustworthiness of a drone based on its behavior and reputation. In this method, trust is treated as information entities possess about each other in a networked environment. This information could include historical action, recommendations from different entities, or any other factors influencing trust.

4.1.5 Fuzzy Logic

Fuzzy Logic is a form of logic that deals with uncertainty and imprecision. It allows for the representation and manipulation of vague concepts and linguistic terms, such as "high," "low," "good," "bad," etc. In trust management, Fuzzy Logic can be used to model and evaluate the trustworthiness of drones based on multiple criteria, such as reliability, capability, reputation, feedback, etc. One way to apply Fuzzy Logic in trust evaluation is to use Fuzzy rules to define the relationship between the input criteria and the output trust level. Fuzzy rules express the degree of implication between Fuzzy sets, which are collections of elements with a membership function ranging from 0 to 1. This process consists of four steps: fuzzification, rule evaluation, aggregation, and defuzzification [55].

4.1.6 Bayesian Inference

Bayesian Inference is a statistical method that utilizes Bayes' theorem to update probabilities based on new evidence or information. In the context of trust management for FANETs, Bayesian Inference is applied to model and update beliefs about the trustworthiness of drones within a network. It provides a formal framework for incorporating new evidence and adjusting trust assessments dynamically and probabilistically [79]. Bayes' theorem is used to update the prior beliefs based on the likelihood of the observed evidence. The likelihood function represents the probability of observing the evidence given a specific trustworthiness hypothesis.

4.2 Machine Learning-based Trust Management Schemes

ML-based trust management in FANET is a research topic that seeks to employ ML techniques to assess, estimate, and improve drone trustworthiness. FANET uses two types of ML approaches for trust management: centralized learning and federated learning. Each can be characterized as supervised, reinforcement, or deep learning.

4.2.1 Centralized ML

Centralized ML for trust management in FANET involves using a central authority or GCS to handle trust-related considerations in a network. This approach entails collecting data on drones' trustworthiness, extracting relevant features, and developing a centralized ML model trained on historical data to predict trust levels.

The model's predictions can be integrated into decision-making processes within the system, influencing how entities are prioritized or filtered. Continuous learning mechanisms and feedback loops ensure the model evolves based on new data and user feedback[80].

4.2.2 Federated ML

Federated ML for trust management in FANET is an approach that addresses trust-related considerations in a decentralized manner. In this framework, trust models are built collaboratively across multiple drones without centralizing data. Each drone maintains its local trust evaluation model based on neighbor interactions and experiences [36]. Federated ML can be applied to trust management as follows: First, drones individually collect and process data related to the trustworthiness of their neighbors, such as user ratings, transaction history, and behavior patterns. Second, using ML algorithms, local trust models are built on each drone, considering features like past behavior, interactions, and other relevant factors[70]. Thirdly, these local models remain on the drones, and model updates are performed locally. Periodically, drones share model updates or aggregated information with a central server or among each other. Finally, the central server collaboratively aggregates and integrates these model updates, creating a global federated trust model.

4.2.3 Supervised ML

Supervised ML for trust management in FANET involves training a model using labeled data to predict the trustworthiness of drones or make decisions based on trust-related features. The process begins with collecting a dataset that includes labeled examples of trust-related instances, such as user interactions or transaction histories. Relevant features are extracted, and an ML algorithm is chosen for model training, such as LiR[65], SVM [66], and LoR [29]. The model then learns patterns and relationships between features and trust labels, evaluated on separate validation and test sets. Once trained, the model is integrated into the trust management system, which predicts new entities' trustworthiness or informs decision-making processes. Continuous learning mechanisms ensure the model adapts to changes over time.

4.2.4 Reinforcement Learning Methods

Reinforcement learning(RL) methods for trust management in FANET involve training agents (GCSs) to make decisions in a dynamic environment (drones) by interacting with it and receiving feedback based on the consequences of their actions. Trust-related features and historical data form the state space in this context, while actions such as establishing connections or adjusting trust levels comprise the action space.

The RL model is trained using algorithms like Q-learning or SARSA when a policy is learned to maximize cumulative expected rewards associated with trust outcomes. These strategies for exploration and exploitation enable the agent to adapt trust dynamics, and continuous policy evaluation ensures alignment with current trust-related goals.[81].

4.2.5 Deep Learning Methods

Deep learning for trust management involves employing neural networks with multiple layers to model and understand intricate patterns within trust-related data. By representing data such as drone interactions and transaction histories, deep learning architectures like Deep Belief Network(DBN) [69], RNN [68], or Convolutional Neural Network(CNN)[82] can automatically learn complex features and dependencies. Supervised DL techniques are applied when labeled data is abundant, enabling the model to predict trust levels or make decisions based on learned patterns. Unsupervised DL methods such as Generative Adversarial Networks (GAN)[83] can capture latent representations in cases with limited labeled data.

4.3 Blockchain-based Trust Management Schemes

Blockchain is a decentralized ledger that records transactions among multiple parties in a verifiable and immutable way [84]. Blockchain-based trust management schemes in FANET can be classified into on-chain and off-chain. On-chain schemes store the trust information on the blockchain, while off-chain schemes use external sources or platforms to store and update the trust information. Both methods have advantages and disadvantages, depending on the application scenario, the network environment, and the system requirements. Some challenges and open issues that need to be addressed in blockchain-based trust management schemes are scalability, interoperability, and standardization [85].

4.4 Hybrid Models

A unique strategy for trust management in FANETs mixes ML models with blockchain [81] or reasoning models with blockchain [58]. Based on previous interactions, ML models may be used to learn drone behavior and reputation. In comparison, blockchain can validate and record drone transactions and trust values in a distributed ledger. The hybrid approach can combine the benefits of both strategies while overcoming their limitations.

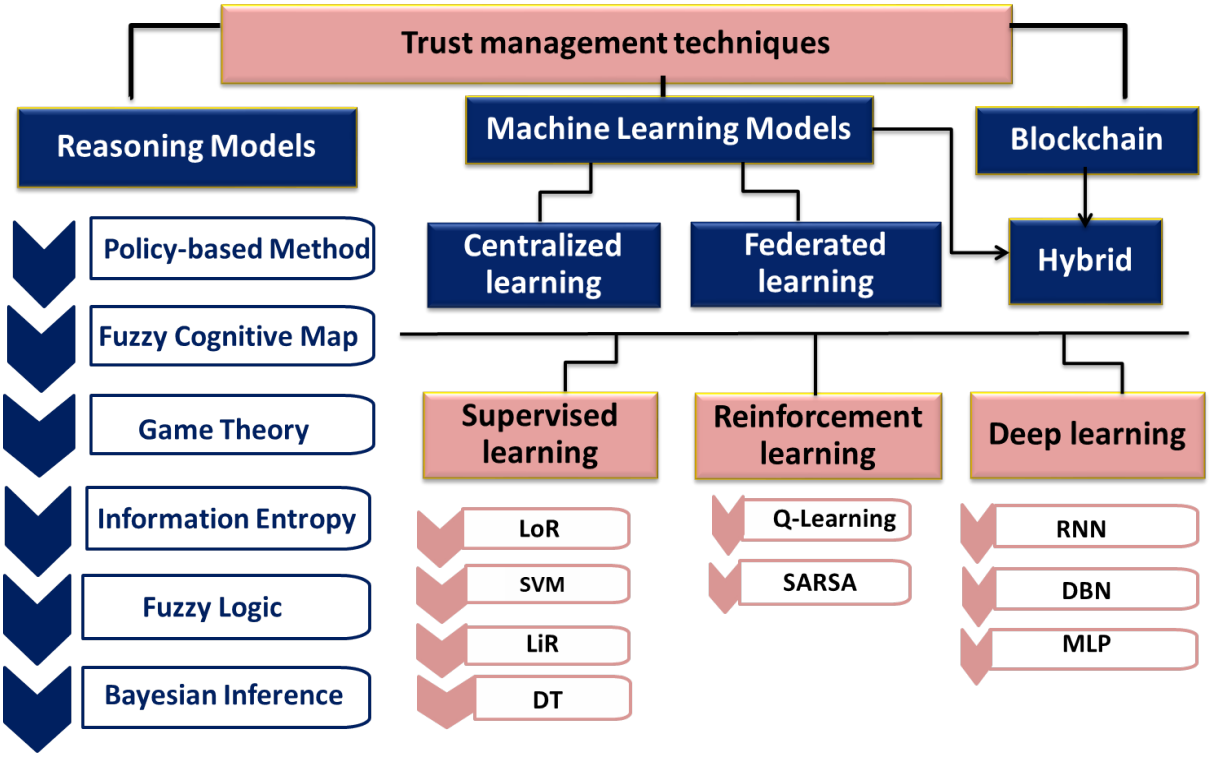


Figure 6: A novel classification of trust management techniques for FANETs.

5 Comparative Study

This section examines the trust management methods for FANETs discussed in the preceding Sections. These strategies are evaluated across various dimensions, including first-trust properties such as availability, integrity, context awareness, and computational overhead. Second, trust requirements such as effectiveness, robustness, and privacy protection are considered. The summarized findings of this comparison are presented in Table 1 and Table 2.

5.1 Trust Properties

A set of trust management properties are defined as follows:

5.1.1 Availability

refers to the assurance that a system or resource is accessible and operational when needed. Specifically, in trust management scenarios, availability focuses on ensuring that trustworthy services, resources, or entities are consistently accessible and operational. Trust management systems aim to establish and maintain trust relationships among entities in a network or system.

5.1.2 Integrity

Integrity, which refers to the assurance that information or data remains unaltered and trustworthy throughout its lifecycle, from creation to transmission and storage, is particularly crucial. This involves verifying that the messages exchanged between entities have not been modified during transmission, preventing unauthorized alterations.

5.1.3 Context Awareness

Context awareness in a trust management model refers to the system's ability to consider the specific circumstances, environment, and conditions in which trust evaluations are made. In the context of trust management, especially in dynamic systems like FANETs, context awareness plays a crucial role in ensuring that trust assessments are relevant and adaptive to the changing nature of the network [86].

5.1.4 Computational Overhead

a metric indicates how practical a specific approach is. The algorithm runs quicker when the computing overhead is low, suggesting its outstanding efficiency [87]. The computational overhead of a trust evaluation method's applied algorithms might indicate its efficiency. It denotes the amount of computing resources (e.g., CPU, memory, battery) used by FANET protocols and algorithms.

5.2 Comparison

Each suggested trust management scheme has its unique characteristics. To better comprehend them, a set of measurements is required. Based on our observations, we give a comparative table (Table 1) of the different strategies with different metrics, such as availability, integrity, context awareness, and computational overhead. We conclude the following:

- Reasoning models have low computational overhead, suitable for resource-constrained FANET nodes. However, they cannot adapt well to the changing environmental contexts in dynamic FANET topologies.
- ML methods Provide better context awareness and integrity than reasoning methods. However, DL methods like RNN have high context modeling power and computational costs. Moreover, RL and other ML methods offer a good trade-off between capabilities and efficiency. FL and FRL are particularly suitable as they can distribute training across FANET nodes.
- Blockchain methods Provide a distributed trust framework, but consensus mechanisms have a high overhead. It may not scale well for large dynamic networks like FANETs with constantly changing nodes. Moreover, it lacks sophisticated context-aware modeling compared to advanced ML techniques.

In terms of capabilities for trust management in dynamic FANET environments: ML methods like RL and FL perform best by learning from contexts and distributing training. They can accurately model complex trust relationships and dynamics within FANETs. Reasoning model techniques are too simplistic, while blockchain has significant scalability issues. Considering FANET constraints, ML techniques like federated reinforcement learning provide a good balance between capabilities, distributed learning abilities, and lower computation overhead than other alternatives.

Table 1: Comparison of existing techniques in FANETs in terms of trust properties

	Techniques	Availab-	Integrity	Context Awareness	Computat- Overhead
Reasoning Models (RM)	Policy-based Method [35]	✓	x	x	Low
	Fuzzy Cognitive Map[76]	✓	x	x	Low
	Game Theory [59]	✓	x	x	Low
	Information Entropy [78]	✓	x	✓	Low
	Fuzzy Logic [55]	✓	x	x	Low
	Bayesian inf [58]	✓	x	x	Low
Machine Learning	DL (RNN) [68]	✓	✓	x	High
	RL [81]	✓	✓	✓	Medium
	ML (LiR) [65]	✓	✓	x	Medium
	ML(LoR) [29]	✓	✓	x	Medium
	FRL [36]	✓	✓	✓	Low
	FL [70]	✓	✓	✓	Low
Block-chain	Blockchain [84]	✓	✓	x	High
	Blockchain [85]	✓	✓	x	High

5.3 Trust Requirements

A set of trust requirements is defined as follows:

5.3.1 Effectiveness

In the context of trust management, effectiveness involves the system's ability to achieve its intended objectives and deliver expected services to users and applications. It encompasses various performance indicators, including but not limited to accuracy, precision,recall, and F-score,in Table 2 these metrics are abbreviated respectively as (**Acc, Pre, Rec, F1s**) which collectively measure the accuracy and reliability of trust evaluations within the system[87].

5.3.2 Robustness

A robust trust management model performs well even in the presence of noisy or incomplete information, outliers, or adversarial attacks; robustness represents the resilience and stability of a trust evaluation system in the face of uncertainties, adversarial activities, or changing conditions[88].

5.3.3 Privacy Protection

The data used for schemes-based trust management includes private information. Privacy protection refers to the measures and mechanisms implemented to safeguard the sensitive information of entities involved in a system or network[89].

Table 2: Comparative analysis of trust requirements in existing FANET Techniques

	Techniques	Effectiveness				Robustness (Attack types)	Privacy Protection
		Acc	Pre	Rec	Fls		
Reasoning Models (RM)	policy-based method [35]	x	x	x	x	None	x
	Fuzzy cognitive map[76]	✓	x	x	x	None	x
	Game theory [59]	x	x	x	x	GPS spoofing	x
	Information entropy [78]	x	x	x	x	Emergent behavior	x
	Fuzzy Logic [55]	✓	x	x	x	Selfishness	x
	Bayesian inf [58]	x	x	x	x	Fake reports	✓
Machine Learning	DL (RNN) [68]	✓	✓	✓	✓	Intrusion detection	x
	RL [81]	x	x	x	x	None	✓
	ML (LIR) [65]	✓	x	x	x	Drop, Data replay	x
	ML(LR) [29]	✓	x	x	x	Sybil attack	x
	FRL [36]	✓	x	x	x	Jamming attack	✓
	FL [70]	x	x	x	x	DOS ,MIM	✓
Block-chain	Blockchain [84]	x	x	x	x	GNSS signal attacks	✓
	Blockchain [85]	x	x	x	x	Malicious drone	✓

5.4 Discussion

Table 2 represents a comparative analysis of different existing techniques for trust management in FANETs. Some key points about what the table is analyzing:

- Effectiveness measures how well the technique achieves its objectives in terms of accuracy, precision, and recall. Many machine learning and reasoning model-based techniques performed well, while game theory, fuzzy logic, and policy-based methods were less effective.
- Robustness refers to how well the technique withstands various attack types. Most techniques did not address attacks like GPS spoofing, emergent behavior, selfishness, fake reports, etc. Reinforcement learning, federated learning, and blockchain-based approaches provided better robustness against jamming, DOS, MIM, GNSS attacks, and malicious drones.

- Privacy protection indicates if the technique protects user privacy. Reasoning models, game theory, information entropy, and fuzzy logic-based methods did not provide privacy. Bayesian inference, federated learning, and blockchain approaches specifically addressed privacy through mechanisms like anonymity or encrypted communication.

In summary, machine learning (intense learning using RNN) and federated reinforcement learning techniques showed the best overall performance in effectiveness, robustness, and privacy protection. Blockchain-based distributed ledger approaches also demonstrated strong robustness and privacy, albeit with some limitations in effectiveness. Traditional techniques like fuzzy logic and game theory had loads. Future research can integrate privacy-preserving machine learning and blockchain to develop highly effective, robust, private trust management solutions for FANETs.

6 Conclusion

This chapter describes the concept of trust management in FANET, provides a detailed review of related works, and offers a comparative analysis of selected trust management schemes. In This chapter, we provide a classification system for trust management techniques. We thoroughly compare and analyze the presented trust management techniques using various criteria. These include trust properties such as availability, integrity, context awareness, and computational overhead. Trust requirements such as effectiveness, robustness, and privacy protection. We believe that this chapter will be helpful as a reference for researchers and practitioners interested in FANET trust management. As we conclude this literature review section, we seamlessly transition to the second part of our thesis which is the Scientific Contributions section. This segment encapsulates three distinct contributions from our research, specifically addressing the critical aspect of insider attack detection within FANET. These contributions represent our unique and valuable additions to the existing body of knowledge, demonstrating the practical implications of our work in enhancing the security and reliability of FANETs.

PART TWO: SCIENTIFIC CONTRIBUTIONS

Chapter III: FUBA: A Fuzzy-based UAV Behavior Analytics for Trust Management in FANETs

"You may be deceived if you trust too much, but you will live in torment if you don't trust enough."

– Frank Crane

1 Introduction

This chapter represents our first contribution to the field of FANET security. FANET is a network of unmanned aerial vehicles(UAV) or drones that work together to complete critical missions. However, various cyberattacks against drones have evolved, and their impact can be harmful, resulting in divesting consequences. As a result, it is critical to secure FANETs against both internal and external threats. In FANETs, drones can leave and rejoin the network at any moment, allowing attackers to hack a drone and mimic a valid one, resulting in insider attacks. Insiders utilize their privileged access to carry out illegal activities. Thus, they are undetected by external network security mechanisms (firewall and cryptographic technologies) [90].

As a result, guaranteeing secure and trustworthy communications in FANETs is vital and remains a concern. In the following, we present FUBA, a novel fuzzy-based drone behavior analytics system for trust management in FANETs. FUBA assesses drone trustworthiness using fuzzy logic methods, considering energy levels, weather conditions, signal strength, packet delivery ratios, and transmission delays.

The proposed model has various advantages, including more excellent performance in outdoor flying ad hoc networks, accurate characterization of drone behavior, subjective evaluation of drone activity, and the capacity to make confident judgments about network information sharing. The remainder of this chapter is organized as follows:

In Section 2, we explain the motivation. In Section 3, we discuss the network architecture. Section 4 describes the suggested fuzzy-based drone behavior analytics for trust management in FANETs (FUBA). Section 5 describes the full implementation of FUBA. Section 6 discusses the effects of RSSI and humidity on trust outcomes. In Section 7, we provide and analyze the experimental findings. Section 8 describes the practical elements and limits of FUBA. Finally, Section 9 summarizes the chapter and suggests potential future directions.

2 Motivation

Trust management is a powerful and effective strategy for preventing unexpected drone behavior and detecting malicious ones. It can increase the robustness and reliability of typical security approaches by ensuring that only trustworthy drones participate in network missions. However, trust is based on observation and recommendation, and several models have been developed for FANET to assess node trust, although they may result in uncertainty. Fuzzy logic is a widely used approach for describing and processing uncertain data, such as drone behaviors.
Few similar research employs the Received Signal Strength Indicator (RSSI) as an essential metric for trust evaluation, which performs better in indoor networks. However, in outdoor networks, the RSSI might be impacted by humidity, affecting trust results. Furthermore, the drone might be identified as uncooperative owing to unintentional misbehavior caused by low signal strength. The main challenge in this domain is designing an efficient analytical trust model for evaluating and understanding drone behavior in FANET under poor signal (RSSI) and bad weather conditions. Without this approach, there is no practical technique for distinguishing between legal and malicious drone activity in FANETs. While numerous trust models have been recently suggested, none have specifically addressed the influence of unfavorable weather conditions on the trust management process within FANET. The proposed work aims to address this gap using the fuzzy logic method to ascertain drone trust based on various parameters, such as energy (battery level), weather (humidity), signal strength (RSSI), Packet Delivery Ratio (PDR), and Transmission Delay (TD).

3 Network Architecture

FANET comprises three essential elements: drones, ground control stations (GCS), and communication links. Two types of communication exist within the system: drone-to-drone (D2D) and drone-to-infrastructure (D2I). In FANET, drones can join or leave the network anytime, but this adaptability also exposes the network to potential vulnerabilities. In such scenarios, a hacker could compromise a regular drone, transforming it into a malicious one [30]. The intruder then integrates into the network as a seemingly legitimate drone, posing a threat by potentially deleting or corrupting messages or tarnishing the reputation of trustworthy drones. This form of attack, termed an insider attack, poses a significant security risk to FANET.

Figure 7 illustrates the network model of FANET operating under the assumption of an insider attack.

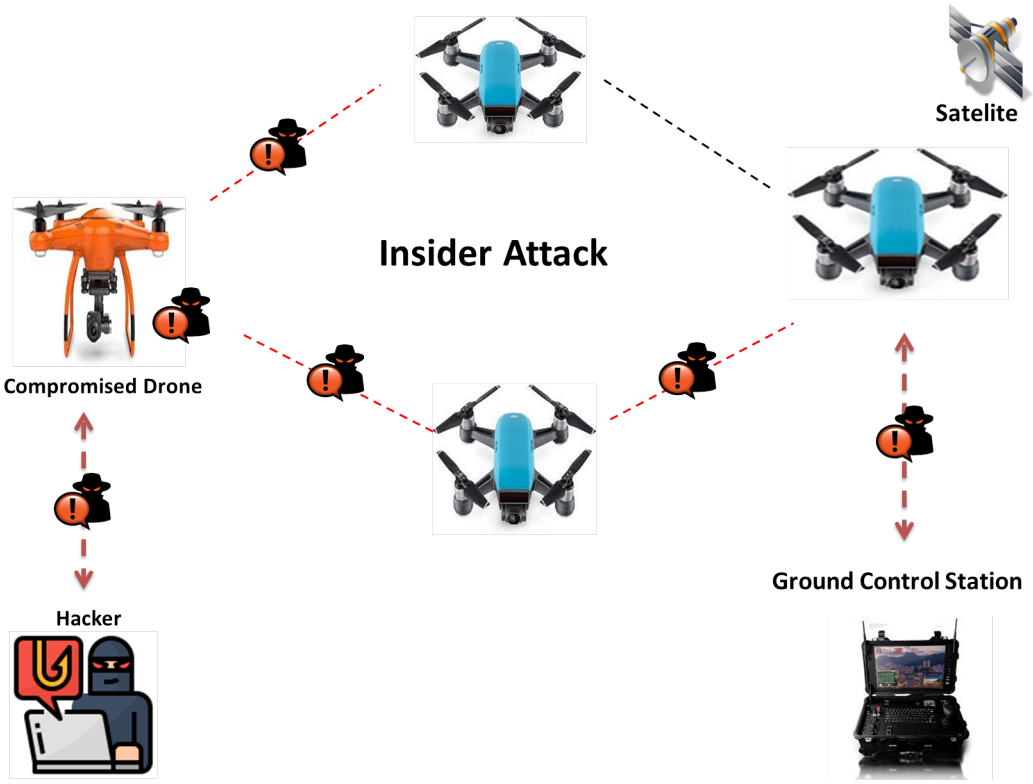


Figure 7: FANET model during an insider attack.

4 The Proposed FUBA Model

As seen in Figure 8. The suggested FUBA model has four steps: information gathering, trust score calculation, trust aggregation, and decision-making.

4.1 Information Gathering

In the proposed model, each drone gathers behavioral data from neighboring drones, encompassing both software and hardware performance over time. Unlike Singh et al.'s approach [91], which utilized a fuzzy logic method with four parameters, our model introduces an expanded set of five parameters: signal strength, drone energy levels, packet delivery ratio, transmission delay, and humidity. Figure 9 illustrates the specific parameters collected by each drone.

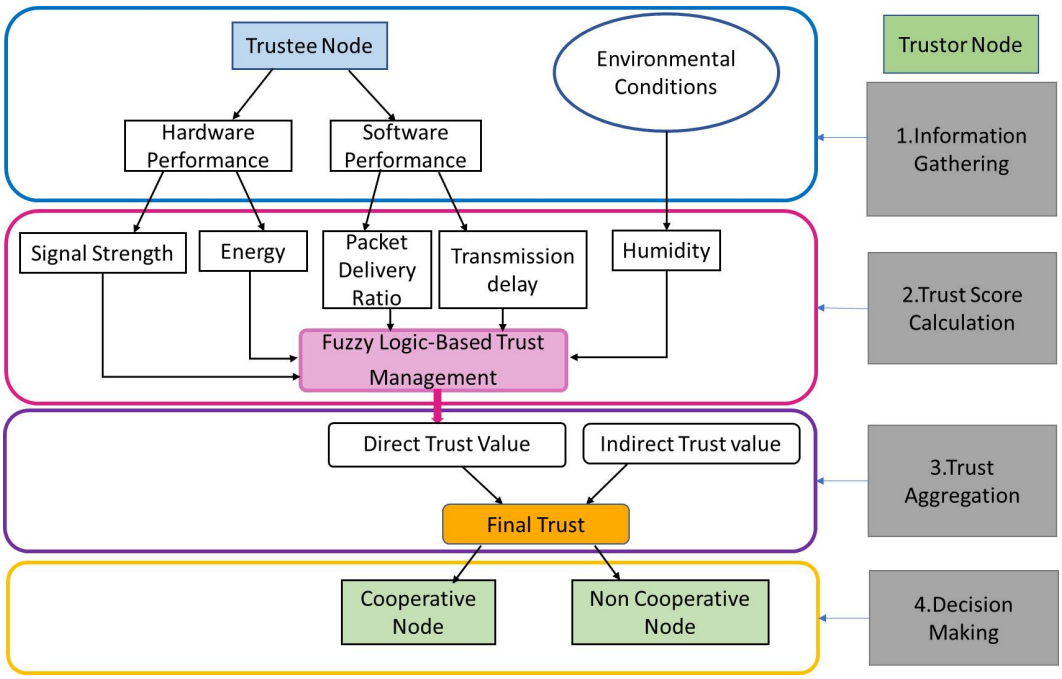


Figure 8: FUBA trust model.

Figure 9: The collected parameters in FANET.

4.1.1 Received Signal Strength Indicator (RSSI)

The drone can gauge the received signal power of its neighboring drone at a specific location and time. This measurement, expressed as the received signal strength indicator , is denoted in dBm, typically as a negative value [92].

The typical RSSI is more substantial than -50 dBm for optimal signal power, reaching values like -30 dBm. Signal power considered excellent or acceptable falls within the range of -50 to -70 dBm (e.g., -60 dBm). Conversely, signal power categorized as poor is characterized by an RSSI less than -70 dBm, exemplified by values like -90 dBm [56].

4.1.2 Node's Energy (Battery Level)

The energy level of a node stands out as a pivotal factor in drone operations. Hence, ensuring effective power management is crucial, encompassing practices such as wireless drone charging, solar drone charging, and integrating artificial intelligence technology [93]. These measures are imperative for the seamless continuity of applications [94].

The drone is deemed mission-ready when its battery level surpasses 50%. However, its ability to collaborate with neighboring drones becomes compromised when the energy level falls within 20% to 50%. If the battery level drops below 20%, there is a risk of the drone significantly impacting the overall network mission [56].

4.1.3 Packet Delivery Ratio (PDR)

The packet delivery ratio is correctly received packets to the total number of packets transmitted by the sender [95], as expressed in Equation (1).

$$PDR = \frac{\sum_{i=0}^{n} ReceivedP_i}{\sum_{i=0}^{n} SentP_i} \tag{1}$$

Here, $ReceivedP_i$ represents the number of correctly received packets, and $SentP_i$ denotes the number of packets transmitted by the sender. By findings from [56], it has been demonstrated that if the ratio of effectively sent packets is below 40%, the packet delivery ratio is low. The PDR is classified as a medium when this ratio falls within the range of 40% to 70%. Conversely, if the ratio surpasses 75%, the PDR is considered high.

4.1.4 Transmission Delay (TD)

TD denotes the duration for the drone sender to transmit packets across the link. The formula for TD is expressed as follows:

$$TD = \frac{L}{R} \tag{2}$$

Where L represents the length of the data packet, and R is the transmission rate measured in bits per second. The assessment of the transmission delay is categorized as small if its value falls below 0.61 ms, medium if the weight ranges between 0.96 ms and 1.47 ms, and significant if the value surpasses 1.47 ms, as outlined in [56].

4.1.5 Weather Condition

The drone is equipped with various sensors, such as rain sensors, wind direction sensors, wind speed sensors, air temperature sensors, and humidity sensors, continuously measuring and recording information about the current environmental conditions [96]. Any alterations in the weather conditions are promptly communicated to the ground control station.

When weather changes significantly affect one of the parameters utilized in the assessment, distinguishing between legitimate and malicious drone activities or discerning intentional from unintentional drone behavior becomes challenging. To tackle this challenge, humidity is an input parameter in the proposed Fuzzy Logic system. In this specific context, the authors have conducted an extensive investigation as outlined in reference [97], utilizing an empirical setup based on IEEE 802.11b/g. The experimentation involves two external radio connections of varying lengths, maintaining continuous data transfer. Surprisingly, the findings reveal that the shorter-distance link is more susceptible to adverse weather conditions, contrary to initial expectations. This susceptibility is attributed to the modulation strategy employed in that specific scenario. The conclusion drawn is that inclement weather can alter the propagation of the drone radio signal.

To assess the influence of temperature and humidity on RSSI values, the authors in [98] conducted measurements at a constant distance of 25 m under varying weather conditions during both summer and winter. The results indicate that temperature has a relatively minor impact on RSSI compared to humidity. Despite similar temperatures, RSSI values can vary significantly.

Notably, humidity substantially influences RSSI, with an observed decrease in RSSI values as humidity increases. This, in turn, directly impacts the path loss exponent. The overall conclusion is that humidity significantly impacts RSSI more than temperature.

4.2 Trust Score Calculation

After gathering essential data, each drone utilizes a fuzzy logic approach to compute the trust score of its neighbors. The system incorporates diverse input parameters such as RSSI, packet delivery ratio, transmission delay, energy, and humidity. Triangular and trapezoidal membership functions are employed for these input parameters to optimize performance. Following this, fuzzy rules are applied in the inference engine phase to yield a conclusive numerical value. This value serves as a direct assessment of trust for the neighboring node. Figure 10 illustrates the proposed trust management model configuration based on fuzzy logic.

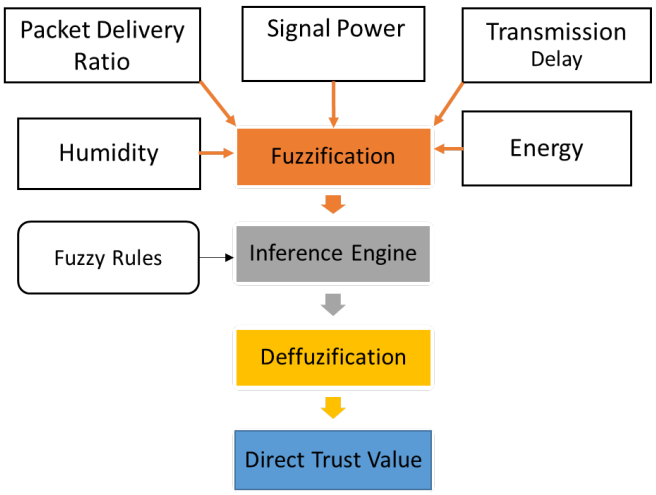


Figure 10: Structure of Fuzzy system.

4.3 Trust Aggregation

During this phase, the parameters α and β are established to consolidate the direct and indirect trust values. Typically, a FANET exhibits lower drone density and shorter link durations between communicating nodes. The values assigned to α and β are derived from these characteristics and are instrumental in determining the requisite trust value. Table 3 outlines the values of α and β corresponding to the trust state:

Table 3: Weight parameters

Direct trust	α	β	**Final Trust**
Bad/Good	1	0	Final Trust(i) = Direct trust(i)
Medium	0.5	0.5	Final Trust(i) =0.5 Direct trust(i) +0.5 Indirect trust(i)j

a) If the output characteristic value (trust) is categorized as either "Bad" or "Good," the confidence factors are set to $\alpha = 1$ and $\beta = 0$. This signifies that $Finaltrust(i)$ is equivalent to $Directtrust(i)$.

b) In the case of an output characteristic value being "Medium," the node solicits recommendations (indirect trust) from its neighboring nodes. Consequently, the final trust computation integrates direct and indirect trust values, as Figure 11 illustrates. The calculation for indirect trust is expressed as follows:

$$IndirectTrust(i) = \frac{1}{n}\sum_{j=1}^{n} DirectTrust(i)_j \tag{3}$$

Where n denotes the total number of drones in the network, (i) represents the index of the trustee drone, and (j) signifies the index of the trustor drone.

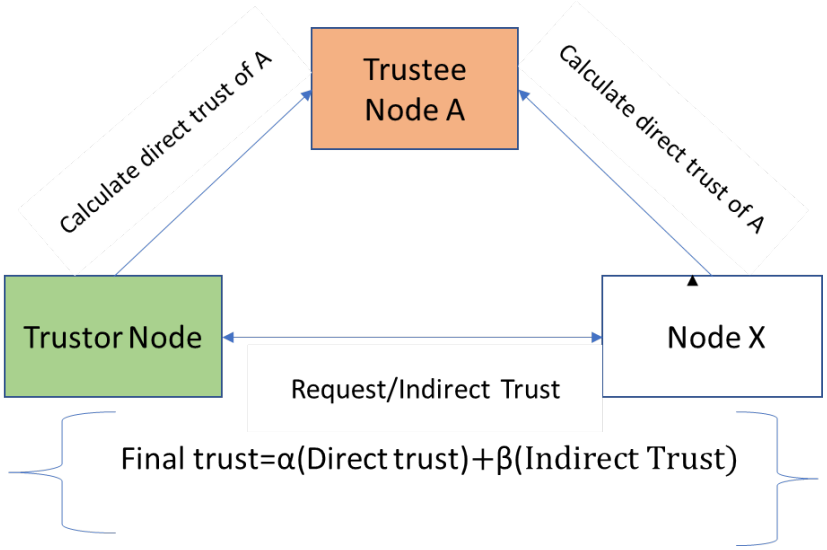


Figure 11: Trust aggregation.

4.4 Decision Making

The primary objective of the decision-making process is to address the following inquiries:

1. Is it secure to exchange information within the network?
2. Are the nodes inclined towards cooperation or not?

After scrutinizing a node's behavior and considering climate changes, the trustor node evaluates the trust score of its neighbors using a Fuzzy Logic-based trust management system. Subsequently, a decision module based on thresholds determines whether to cooperate with the node involved in the given operation. Specifically, the trust score of each node is compared to a threshold value to ascertain if the node is trustworthy or potentially malicious, as outlined below:

$$\begin{cases} if \quad Final \quad trust \quad > \quad 30\% \quad then \quad Trust \quad node \\ if \quad Final \quad trust \quad \leq \quad 30\% \quad then \quad Malicious \quad node \end{cases}$$

5 Implementation Details of the FUBA Model

Fuzzy logic is a computational approach that handles uncertain information by allowing for degrees of truth rather than rigid binary values. This section uses MATLAB to evaluate the proposed FUBA model. The fuzzy logic used to evaluate node behavior comprises three steps: fuzzification, inference engine, and defuzzification.

5.1 Step 1: Fuzzification

This step generates a membership function to determine how the numerical data correspond to a linguistic variable, using triangular and trapezoidal functions presented in Figures 12 and 13. Typically, a triangular membership function is defined using three parameters, namely, a, b, and c, as follows:

$$f(x,a,b,c) = \begin{Bmatrix} 0 & x \leq a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x \end{Bmatrix} \tag{4}$$

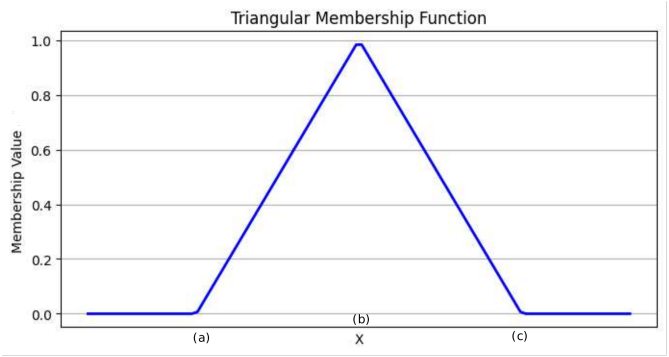


Figure 12: Triangular membership function.

The expression given in equation (4) can be written in a simple form using *min* and *max* functions as follows:

$$F(x, a, b, c) = \max\left(\min\left(\frac{x-a}{b-a}\right), \left(\frac{c-x}{c-b}\right), 0\right) \tag{5}$$

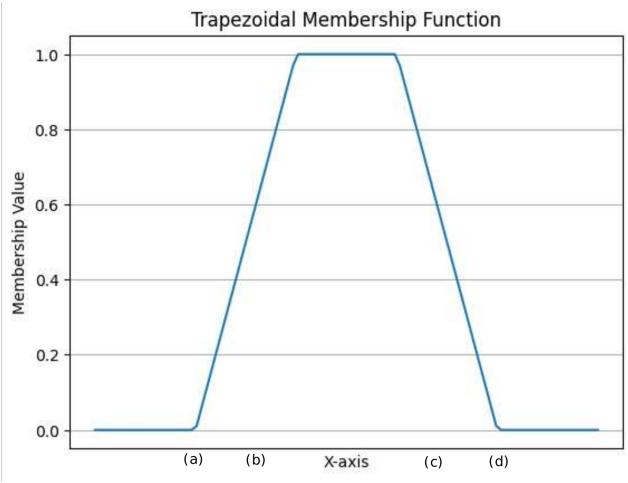
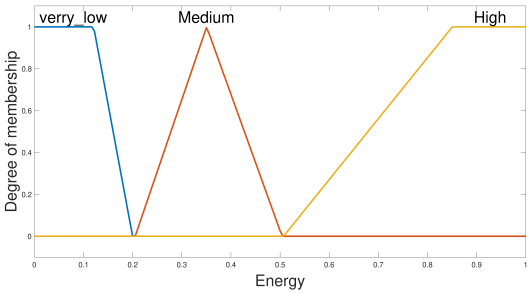
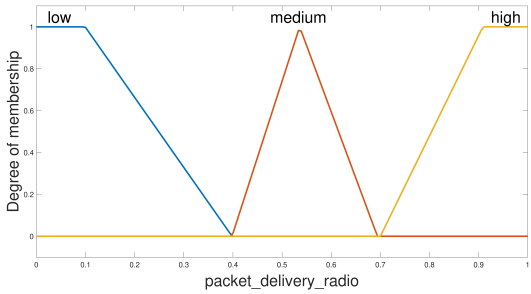


Figure 13: Trapezoidal membership function.

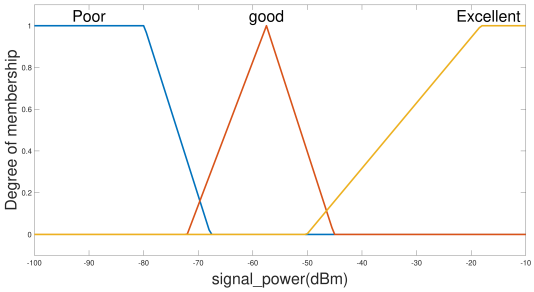
$$F(x, a, b, c, d) = \max\left(\min\left(\frac{x-a}{b-a}\right), 1, \left(\frac{d-x}{d-c}\right), 0\right) \tag{6}$$



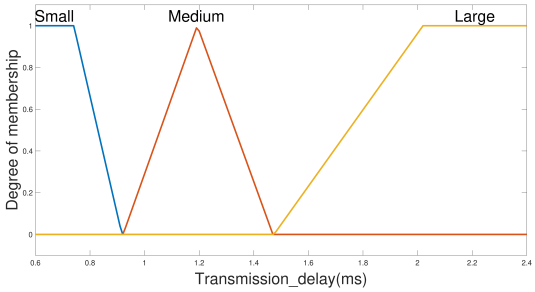
(a) Membership Functions of Energy



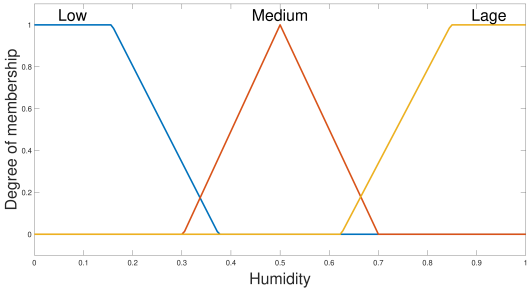
(b) Membership Functions of Packet Delivery Ratio



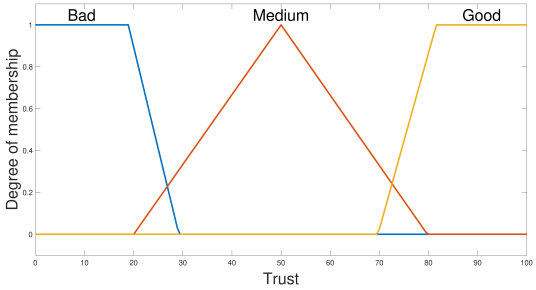
(c) Membership Functions of Signal Power



(d) Membership Functions of Transmission delay



(e) Membership Functions of Humidity



(f) Membership Functions of Trust Score

Figure 14: Membership functions of the different parameters.

Figure 14(a) illustrates the membership functions of Energy: A triangular membership function for the linguistic variable $Medium$ is defined by the triangle (x, 0.2, 0.35, 0.5). The trapezoidal membership function for the linguistic variable $High$ is determined by the trapezoid (x, 0.5, 0.85, 1, d).

As shown in Figure 14(a), the fuzzy variables of the energy are $VeryLow$, $Medium$, and $High$ and are analyzed from 0 to 1.

Figure14(b) shows the fuzzy variables for the packet delivery ratio, which are Low, $Medium$, and $High$, analyzed from 0 to 1.

Figure14(c) shows that the fuzzy variables of the signal power are $Poor$, $Good$, and $Excellent$. They are analyzed from -100 to -10 dBm.

Figure 14(d) shows the fuzzy variables of the transmission delay: *Small*, *Medium*, and *Large*. They are analyzed from 0.6 to 2.4 ms.

In Figure 14(e), the fuzzy variables of humidity (*Low*, *Medium*, and *Large*) are analysed from 0 to 1.

Figure 14(f) illustrates the output trust fuzzy variables that are *Bad*, *Medium*, and *Good* and are analyzed from 0 to 100.

5.2 Step 2: The Inference Engine

In this step, all the rules need to be defined in the proposed fuzzy logic model and then explain those that reflect realistic situations:

The first rule illustrates the worst-case scenario. In contrast, the second rule represents the best case. The third rule requires the system to consider the node trustworthy due to its low battery, implying that unintentional misbehavior is considered.

Rules 4 through 7 state that if all variables have low values except for one that has a positive value. Then, the node is considered untrustworthy with a wrong trust value.

Rule 8 requires the system to consider the node trustworthy because it has a weak RSSI due to the high humidity. This implies that the system takes into account unfavorable weather conditions. Table 4 illustrates the rules when humidity is low, while Table 5 shows the rules when humidity is high.

Table 4: Fuzzy rules with low humidity

R	**RSSI**	**PDR**	**Energy**	**TD**	**Output**
1	poor	Low	Very Low	Large	Bad
2	Excellent	High	High	Small	Good
3	Excellent	High	Very Low	Small	Good
4	Excellent	Low	Very Low	Large	Bad
5	poor	Large	Very Low	Large	Bad
6	poor	Low	High	Large	Bad
7	poor	Low	Very Low	Small	Bad

Table 5: Fuzzy rules with high humidity

R	RSSI	PDR	Energy	TD	Output
1	poor	Low	Very Low	Large	Bad
2	Excellent	High	High	Small	Good
3	Excellent	High	Very Low	Small	Good
4	Excellent	Low	Very Low	Large	Bad
5	poor	Large	Very Low	Large	Bad
6	poor	Low	High	Large	Bad
7	poor	Low	Very Low	Small	Bad
8	poor	High	High	Small	Good

5.3 Step 3: Defuzzification

Defuzzification is the pivotal stage within the fuzzy logic process, where the crisp output is derived from the fuzzy inference engine's fuzzy output. This involves translating the fuzzy set or linguistic term (*Bad*, *Medium*, and *Good*) into a single, definite value that can be understood and utilized for drone behavior evaluation. Various methods, such as center of gravity, bisector, and maxima, can be employed for defuzzification to convert the fuzzy output into a clear and actionable result [99]. In the proposed model, the centroid method (COG) is considered the most widely used technique and is depicted in Figure 15.

This method involves determining the center of gravity of the obtained polygon:

$$CG = \frac{\sum_{x}^{b} = af\,(x) \times x}{\sum_{x}^{b} = af\,(x)} \tag{7}$$

where $f(x)$ represents the aggregation of the membership functions while a and b represent the bounds of the obtained polygon.

This method calculates the output by determining the abscissa of the centroid located beneath the curve's surface. The selection of the defuzzification method exerts a significant impact on the final result of the fuzzy logic model. The center of gravity method is more flexible, considering the entire fuzzy output (trust) to calculate the trust result. The functions that determine the membership of the input and output parameters must be adjusted for each iteration of the fuzzy rule base [100]. The cut-off method is depicted in Figure 16.

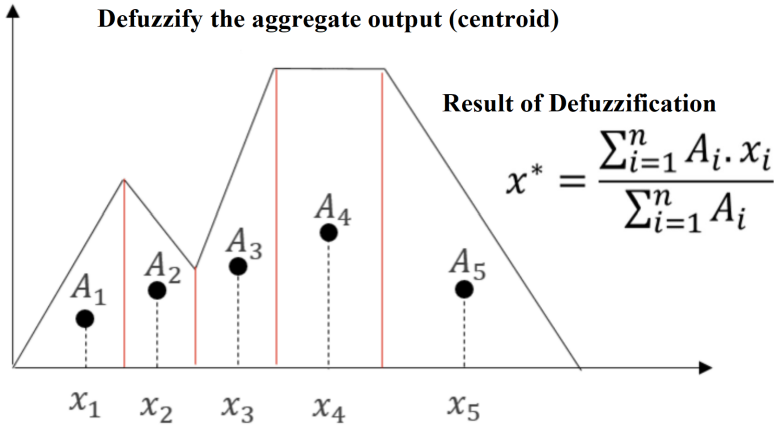


Figure 15: COG method.

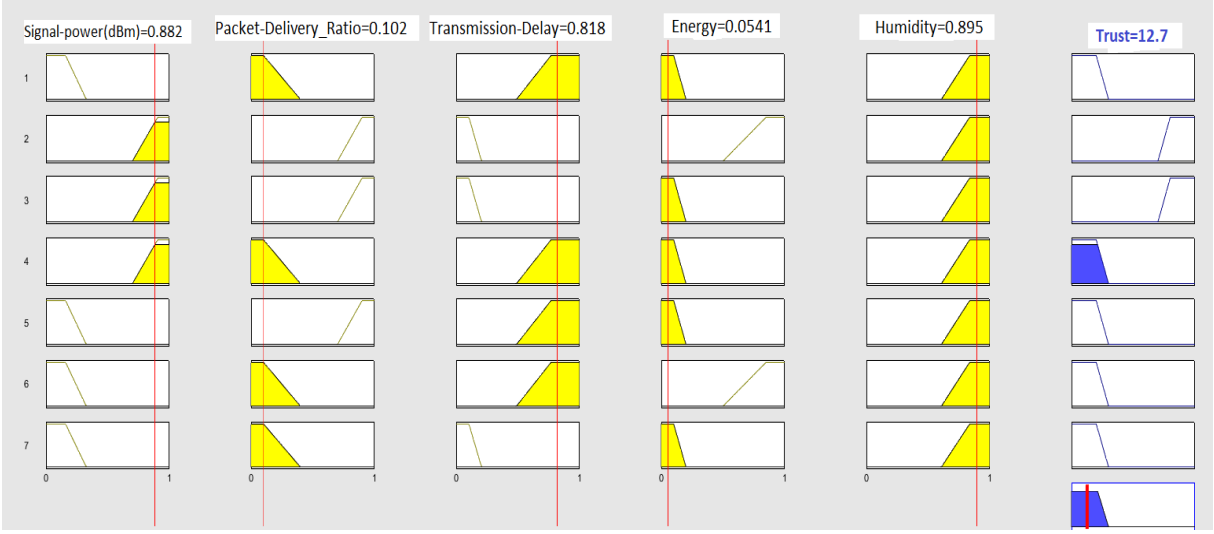


Figure 16: The cut-off method

6 Impact of RSSI and Humidity on Trust Result

MATLAB programs serve as the tools for assessing the performance of the proposed FUBA model. The fuzzy logic application is employed to analyze and comprehend node behavior in the presence of adverse weather conditions and weak signal strength (RSSI).

6.1 Impact of RSSI on Trust Result

The bar chart depicted in Figure 17 illustrates the trust outcomes for eight nodes in the network under varying levels of RSSI while adjusting other parameters (TD, Energy, PDR) to yield distinct trust values: high, medium, and low. Notably, the trust value exhibited a decline from 87% to 50% in nodes 1 and 3, while nodes 4, 6, and 7 experienced a decrease from 52% to 12%.

Conversely, nodes 2, 5, and 8 maintained a consistent trust level.So the proportion of trust notably increases when the RSSI is optimal.

This observation suggests that signal power (RSSI) is pivotal in evaluating drone performance within FANET. While several explanations may account for these results, it is crucial to emphasize the significant impact of signal power on trust values, especially in high humidity conditions. Consequently, it is advisable to eliminate the drone from the network to enhance network security.

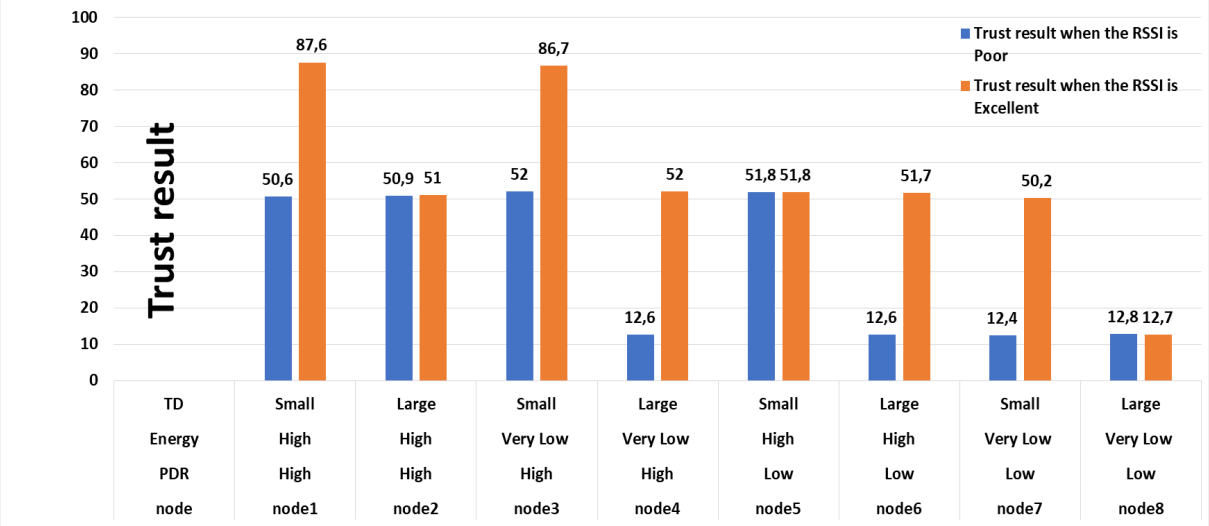


Figure 17: The impact of RSSI on trust result.

6.2 Impact of Humidity on Trust Result

Figure 18 provides a visualization of the trust outcomes for a network comprising 16 drones under varying humidity levels while adjusting other parameters (TD, Energy, PDR, RSSI) to generate high, medium, and low trust values. Node nine witnessed a decline in trust from 87.6% to 50.6%, while node 12 experienced a decrease from 51.2% to 12.6%. Conversely, the trust values for the remaining nodes remained constant.

A key observation derived from Figure 18 is the significant impact of humidity on trust outcomes within FANET. This underscores the necessity of incorporating considerations for climate change when designing a trust management system in FANET.

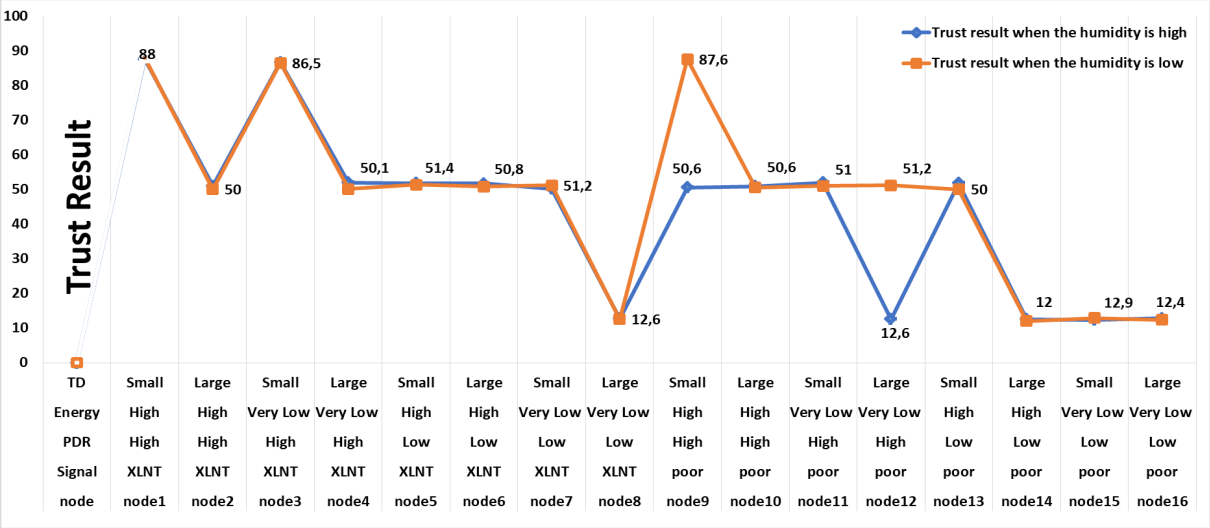


Figure 18: The impact of humidity on trust results.

7 Experimental Result and Discussion

In this section, we describe the simulation setup and the simulation results.

7.1 Simulation Setup

The effectiveness of the newly introduced FUBA system is evaluated through simulations conducted within the following communication frameworks:

- **OMNeT++**: This simulation library and framework, constructed using C++, is tailored for developing and testing intricate communication networks. Known for its extensibility, modularity, and component-based structure, OMNeT++ is widely embraced by researchers and engineers in the field [101].
- **INET Framework**: Integrated with OMNeT++, the INET Framework enriches the simulation environment with a diverse array of network models and protocols, enhancing the realism of communication scenarios [102].
- **AVENS (Aerial Vehicle Network Simulator):**Specifically designed for virtual experiments assessing the network coverage and interconnectivity of drones, AVENS integrates the XPlane Flight Simulator and the OMNeT++ network simulator [103].
- **XPlane Flight Simulator:** Contributes to simulation authenticity by enabling the modeling of real-world flight dynamics and interactions among drones [104].

The simulated network employs the IEEE 802.11 communication protocol for wireless drone interactions within 2500 m x 2500 m, replicating real-world operational conditions. The simulation duration is set to 3000 seconds, ensuring a comprehensive evaluation of network behavior and performance over an extended period. The simulations are conducted on a 64-bit PC running Windows 10, providing ample computational resources for precise and reliable results. Table 6 summarizes the simulation parameters utilized in the experiments.

The simulation scenario emulates a drone communication network with the FUBA system, starting with ten drones and a ground control station. Drones collaboratively share a wireless communication medium within the AVENS simulation framework. During the simulation, drones exchange messages and gather critical network parameters, including transmission delay, RSSI, packet delivery ratio, and node energy. The number of drones is incrementally increased from 10 to 200 to assess scalability and performance.

OMNeT++ records end-to-end delay across the network, allowing for a comprehensive understanding of the proposed method's behavior under varying drone parameters.

7.2 Simulation Results

To evaluate the effectiveness of FUBA, established models such as FNDN [62] and UNION [61] are used as benchmarks, providing a baseline for comparison based on their inherent characteristics. The simulations specifically focus on two critical parameters: false positive rate and end-to-end delay. The false positive rate quantifies instances where the system incorrectly identifies trustworthy nodes as untrustworthy [105]. In the simulation experiments, examples of false positives occur when the FUBA system erroneously categorizes a drone as a regular node despite not meeting the criteria for such classification.

Additionally, the analysis includes the end-to-end delay, which represents the time taken for data to travel from the source drone to the destination drone in the network [106]. In the context of the simulations, the end-to-end delay is measured as the time it takes for a message or packet to be transmitted from one drone (source) to another drone or ground control station (destination). This metric is crucial for evaluating real-time communication performance.

Table 6: Simulation parameters.

Simulation tools	OMNET++,Avens,Xplane10
Simulation area	2500 m × 2500 m
Node counts	10-200
Ping-transmission interval	10s
Ping-sleep period	10s
UDP-transmission interval	10ms
UDP-packet size	1000B
UDP application type name	UdP video stream SVR
UDP application video size	10MIB
MAC address assignment	auto
Ip process delay	10us
Mac Queue size	14
WLAN data rate	2MIB
Transmission frequency	2400HZ
Physical Tx Power	100mW
Power generation	100MW
Simple energy storage	0.05J
Energy generator sleep interval	Exponential(10s)
Mobility model	Random way-point
Ground control station mobility	Stationary mobility
Mobility update rate	2s
Wireless standard	IEEE 802.11
Simulation time	3000s
Operating platform	64bit windows10

The FNDN [62] is a recent monitor-based communication architecture that utilizes both direct and indirect trusts for Flying Named Data Networking. While assessing node behavior through these trust mechanisms, the UNION [61] model considers drone energy, mobility patterns, and enqueued packets. Both models incorporate direct and indirect trust to evaluate node behavior. A critical step in assessing the effectiveness and practicality of FUBA is to compare its proposed trust model with these two existing models.

Based on Figure 19, it is evident that the proposed FUBA model significantly diminishes the average end-to-end delay of data packets, particularly when compared to the UNION model in scenarios with high drone density. Specifically, with 50 drones, the FUBA model reduces delay exceeding 1.4 seconds, a contrast to the performance of the UNION model. In scenarios with a more prominent drone population of 100, the improvement is approximately 1.1 seconds. Notably, as the number of nodes surpasses 150, the mean end-to-end delay for both FNDN and the proposed solution becomes nearly identical.

The figure distinctly illustrates that the proposed FUBA model consistently exhibits the lowest end-to-end delay among the three models.

Figure 20 illustrates the false positive ratio for FUBA, FNDN, and UNION as a function of drone density. The false positive is determined by computing the trusted node using the FUZZY logic application if a node (i) is not compromised. The graph curves demonstrate a steady increase in calculated false positives for FNDN and UNION. However, it is noteworthy that no instances of false positives were generated at the inception of the simulation experiments. Consequently, the proposed solution exhibits a lower error ratio when comparing the proposed FUBA model with FNDN and UNION.

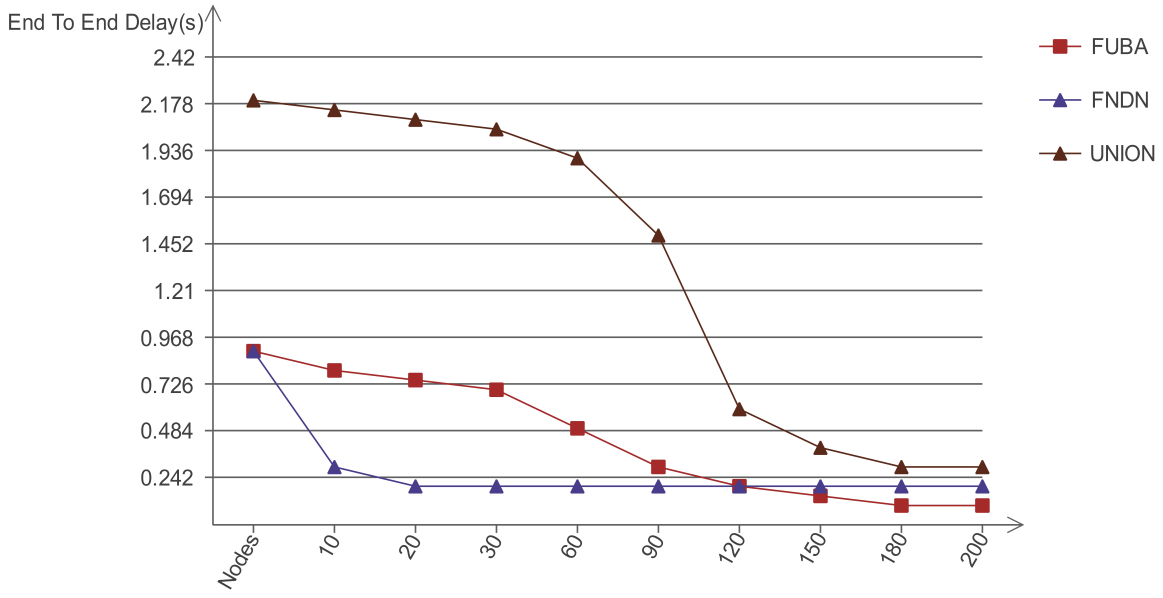


Figure 19: The average end-to-end delay versus the number of nodes.

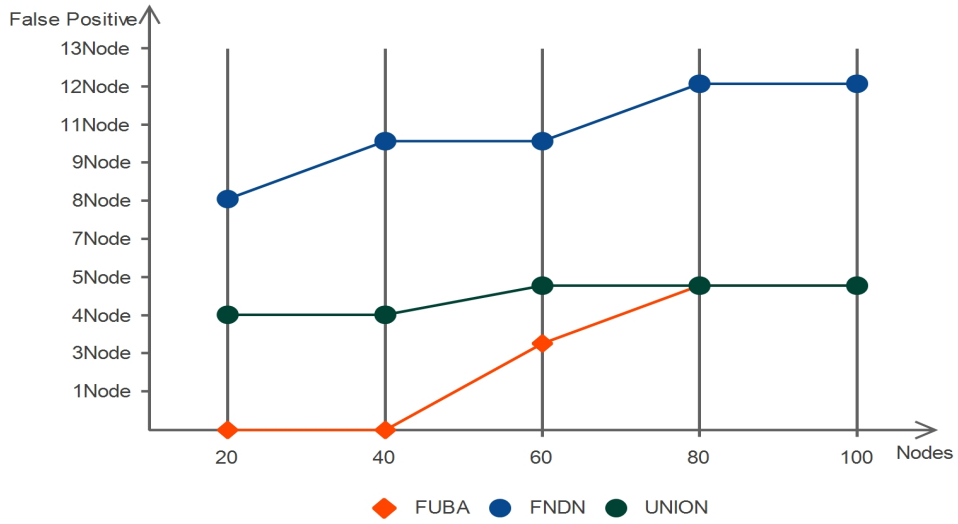


Figure 20: Number of false positives compared to FNDN and UNION.

8 Practical Aspects and Limitations of FUBA

The FUBA system introduces an innovative strategy for improving trust management in FANETs by integrating humidity as a novel parameter. This section delves into assessing the FUBA model's adaptability and scalability across diverse scenarios while acknowledging and examining the model's constraints and real-world applicability.

8.1 FUBA Generalizability and Applicability

FUBA demonstrates its versatility by being applicable under diverse weather conditions, as it dynamically adjusts its operational parameters to the surrounding environment. This includes incorporating fuzzy variables for humidity which is Low, Medium, and Large. This adaptive capability ensures that FUBA's trust assessment remains pertinent and practical across various environmental circumstances.

Moreover, integrating environmental parameters into trust management can be extended to various domains, including agricultural robotics, environmental monitoring, border surveillance, and disaster response. Conversely, the principles of behavior analysis, integrating environmental parameters, can serve as inspiration for a range of autonomous systems tackling complex external conditions. Examples include wildlife monitoring, pollution detection, and habitat preservation.

Lastly, ethical considerations about environmental data collection and the broader societal implications of incorporating natural parameters, such as airspace congestion and urban environments, into autonomous systems must be carefully addressed.

8.2 FUBA Scalability

To address the scalability of FUBA, each drone within the network assesses the reliability of its neighboring drones and transmits this information to the ground control station for decision-making support. This strategy effectively limits the spread of trust data and evenly distributes the computational workload, allowing for implementing a scalable FUBA system. Consequently, the demonstrated scalability of the proposed FUBA model is evident in its capability to handle an expanding number of drones.

8.3 FUBA Practicality and Feasibility

Employing FUBA in real-world applications shows potential in tackling the growing security issues linked to drone threats. Fuzzy logic emerges as a valuable asset for recognizing and dealing with nefarious drones that might be utilized for activities such as unauthorized surveillance, smuggling, or even acts of terrorism. Subsequently, we investigate putting the FUBA model into action and examine its tangible consequences in identifying malicious drones.

- The suggested FUBA system can detect irregularities in drone behavior by comparing the observed actions of a detected drone with predefined models of standard drone behavior. Should the drone's movements significantly deviate from the anticipated norm, the system is designed to trigger an alert and activate appropriate response measures.
- FUBA can integrate contextual information into its decision-making process, including local regulations, flight restrictions, and historical data. This enhances the system's capability to make more precise judgments regarding the legitimacy of a drone's presence. This feature ensures that drones engaged in harmless activities, such as those operated by hobbyists or for commercial purposes, are not erroneously identified as malicious.
- The FUBA model can continuously learn from newly acquired data, allowing it to adjust its rules and inference mechanisms dynamically. This adaptability enables FUBA to stay attuned to the evolving tactics employed by malicious drones over time.

- The FUBA system can seamlessly integrate with established aviation and security infrastructure, including air traffic control systems, airport security, and critical infrastructure protection. This integration enhances airspace security by leveraging the existing frameworks and contributing to a more comprehensive security infrastructure.
- FUBA can offer real-time monitoring and reporting of drone activities, assisting security personnel in making prompt and informed decisions to address potential threats effectively.

8.4 FUBA Practical Limitations

While FUBA exhibits effective trust management in FANETs during adverse weather conditions, the proposed approach does have several primary limitations:

- As the quantity of rules and fuzzy sets expands to model an extensive problem space, the complexity of fuzzy rule bases can intensify, posing challenges in terms of management. This complexity significantly impacts aspects such as debugging, updating, and interpretability.
- The efficacy of fuzzy systems is contingent on the quality of the input features they receive. In security, crucial features may be absent or contain noise, constraining detection capabilities. Additionally, when deploying a complete system, it is essential to integrate security considerations related to trust data exchange to ensure a comprehensive and secure implementation.

9 Conclusion

Trust management proves to be a practical approach to identifying insider threats. The key challenge lies in formulating a conceptual and analytical trust model tailored for FANET, capable of assessing and comprehending the behavior of nodes. In this chapter, we introduced FUBA, a Fuzzy-based drone behavior analytics system for trust management that relies on direct and indirect information. Unlike prior models, our proposed approach enhances network trustworthiness even in bad weather conditions and under poor RSSI. Notably, FUBA demonstrates efficacy in distinguishing between legitimate and malicious drone actions. Rigorous implementation and testing of the system were conducted through extensive simulation experiments using OMNeT++, Xplane, and Avens frameworks. Simulation results reveal the superiority of our model in comparison to existing ones, particularly in terms of average end-to-end delay and false positive ratio. Future endeavors could benefit from incorporating machine learning and blockchain technology to augment FUBA's capabilities. Leveraging ML for automated tuning of fuzzy logic rules and membership functions holds the potential for enhancing performance across diverse operating environments. Additionally, integrating blockchain-based distributed ledgers could secure the sharing of trust data and provide resilience against compromised nodes. As these technologies continue to evolve, their integration promises to create more resilient, secure, and efficient FANETs, thereby shaping the future of aerial communication and navigation.

In the next chapter, more advanced algorithms, such as deep learning and reinforcement learning, will be explored to detect insiders in FANET. Furthermore, the introduction of federated learning offers an exciting opportunity, particularly in scenarios where data privacy is of utmost importance.

Chapter IV: Federated Learning-based Intrusion Detection in FANETs

"If you spend more on coffee than IT security, you will be hacked."

– Richard Clarke

1 Introduction

The security of FANETs is crucial, mainly due to their deployment in critical and intelligent environments. FANETs, with their dynamic topology and resource-constrained drones, pose unique security challenges [53].

Jamming, flooding, selective forwarding, blackholes, wormholes, and Sybil attacks are some of the most typical attacks against FANETs, and they may all inflict considerable damage. However, developing a secure and reliable FANET is essential for the efficient functioning of intelligent environments. Various security mechanisms and protocols can be employed to mitigate these threats, such as encryption, authentication, intrusion detection systems , and secure routing protocols. On the other hand, techniques like trust management and deception systems can help identify and isolate malicious drones. As shown in Chapter 1, IDS plays a pivotal role in ensuring the safety of FANETs. It can be broadly categorized into two main types: signature-based and anomaly-based intrusion detection[107]. However, developing reliable, robust, and flexible IDS that support real-time, secure, and cost-effective data delivery has driven the advancement of more sophisticated ML and artificial intelligence technologies.

This chapter represents our second contribution in the field of network security; we proposed federated learning-based intrusion detection in FANETs(FLID).

This chapter provides a detailed overview of the network model, outlining its components and functionalities. Subsequently, we delve into the threat model, our proposed federated learning model-based intrusion detection (FLID) is presented, showcasing its mechanisms for detecting intrusions. We then focus on attack mitigation strategies, discussing approaches to thwarting potential threats to the system's integrity. Model preparation is addressed, highlighting the steps taken to prepare the model for experimentation and evaluation. We then discuss experimental results and comparisons. Finally, we conclude and offer reflections on the findings presented in this chapter.

2 Motivation

A predetermined collection of known malicious attack patterns is utilized in signature-based IDS. When encountering any of these identified patterns, the IDS detects an attack. On the other hand, anomaly-based IDS operates by pinpointing deviations from typical behavior to uncover malicious activities. Nonetheless, these traditional approaches encounter difficulties in swiftly expanding mobile communications networks, especially in more extensive networks where scaling and asynchrony present challenges. Researchers suggest employing machine learning for sophisticated and precise anomaly-based detection methods to tackle these issues. ML can enhance the self-learning capabilities of these systems and foster the development of more intelligent and adaptive security solutions for detecting attacks within FANET [108]. However, many ML and DL-based IDS tailored for FANET fall short in addressing critical data privacy concerns. This deficiency stems from using data, often containing sensitive information about interconnected drones, during model training phases. Consequently, safeguarding data privacy is paramount during the development of intrusion detection systems in this context [70].

Federated Learning has emerged as a novel ML approach. FL offers a promising avenue to mitigate privacy risks while fostering the creation of more resilient and precise IDS solutions for FANETs. By leveraging FL, data remains decentralized, allowing for collaborative model training without compromising the confidentiality of drone data.

A series of challenges demand attention to fortify FANET environments against diverse attacks and enhance the detection of intruders. Motivated by these challenges, the present chapter puts the subsequent contributions:

- Initially, we engineer three distinct deep neural network architectures namely, Multilayer Perceptron (MLP), Recurrent Neural Network with Long Short-Term Memory (RNN+LSTM), and Convolutional Neural Network (CNN) tailored explicitly for anomaly-based intrusion detection within FANET environments. Our models demonstrate proficiency in identifying three primary attack types: Flooding attacks, blackhole attacks, and selective forwarding attacks.

- Subsequently, we introduce a Federated Deep Learning approach for intrusion detection, empowering drones to undertake localized training on their respective datasets. This strategy upholds data privacy while harnessing individual drone capabilities effectively.
- Lastly, we conduct a comparative analysis among the three Federated Deep Learning models MLP, RNN+LSTM, and CNN to ascertain the most optimal choice for FANET environments.

3 Background

In this section, we define federated learning, the loss function, the three known types of deep neural networks, and finally, we define machine learning metrics.

3.1 Federated Learning

Federated Learning is an ML technique that allows multiple devices to train a shared ML model collaboratively without directly sharing their raw data [109]. Each instrument holds its local dataset, and instead of sending the data to a central server, it only sends updates or gradients computed from the local data. These updates are then aggregated by a central server in the network to create a global model representing all participating devices' collective knowledge. The main advantage of FL is privacy preservation, which is achieved by keeping the data locally on the devices. FL also offers computational efficiency by leveraging the processing power of the devices themselves, thereby reducing communication overhead and saving bandwidth.

3.2 Loss Function

A loss function, in the context of ML measures how well a model's predictions match the true values or labels of the training data. It quantifies the difference between the model's predicted output and target values. The loss function is crucial in training the model using decentralized data from multiple clients or devices in federated learning. The goal is to find a model that performs well across all clients while preserving the privacy of individual data. The loss function's specific choice depends on the learning task's nature and the desired objective [110].

3.3 Deep Neural Network Types for Anomaly Detection

The anomaly detection in our proposed is performed using different DL models, such as MLP, RNN, and CNN. These deep neural network types are artificial neural networks that have multiple layers between the input and output layers.

3.3.1 MLP(Multilayer Perceptron)

Multilayer Perceptron is a type of artificial neural network that consists of multiple layers of neurons connected by weighted links. MLP can learn nonlinear functions and perform classification or regression tasks [111].

The layers are typically categorized into three types:

Input Layer: This layer receives input data and passes it to the next layer without applying any transformations. The number of neurons in the input layer corresponds to the number of features in the input data.

Hidden Layers: These are intermediate layers between the input and output layers. Each neuron in a hidden layer computes a weighted sum of the inputs from the previous layer and applies an activation function to produce an output. The number of hidden layers and the number of neurons in each hidden layer are hyperparameters that can be adjusted based on the complexity of the problem and the available computational resources.

Output Layer: This layer produces the final output of the network. The number of neurons in the output layer depends on the nature of the task. For example, in a binary classification problem, there may be one neuron in the output layer to produce a single output indicating the probability of belonging to one of the classes. In a multi-class classification problem, there may be multiple neurons in the output layer, each corresponding to a different class.

Figure 21 describes the MLP architecture.

3.3.2 CNN (Convolutional Neural Network)

Convolutional Neural Network is a type of artificial neural network that uses convolutional layers to extract features from images or other types of data. CNN can perform anomaly detection, image recognition, or segmentation [113]. The key components and concepts of CNNs are described as follows:

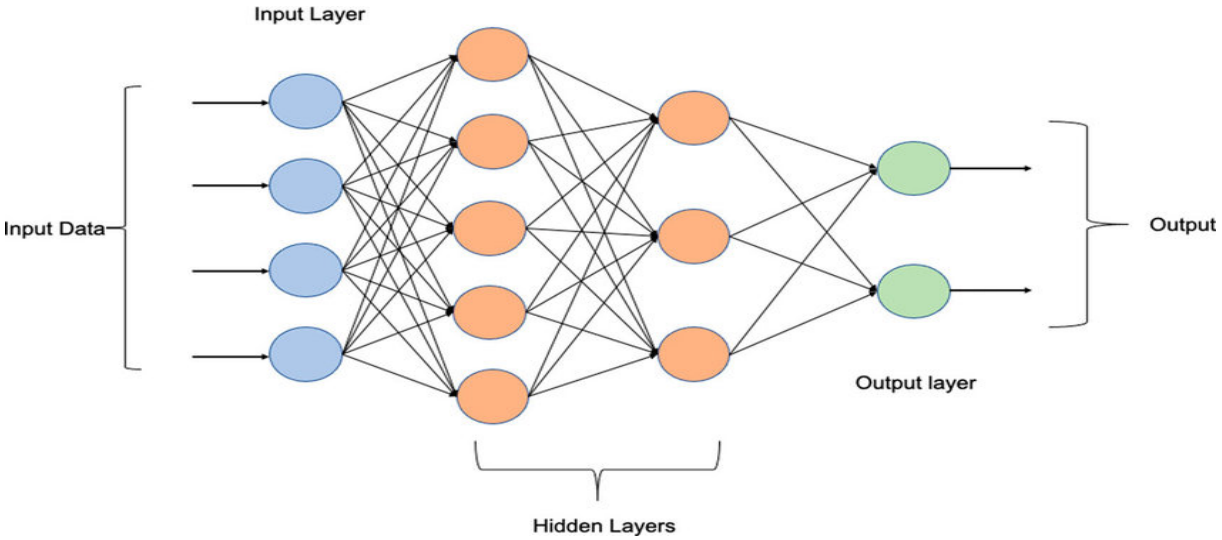


Figure 21: Multilayer Perceptron [112].

Convolutional Layers: The fundamental building blocks of CNNs are convolutional layers. Each layer consists of a set of learnable filters (also called kernels) that are convolved with the input data to produce feature maps. These filters detect various features such as edges, textures, and shapes at different spatial locations in the input image.

Activation Function: After the convolution operation, an activation function is applied element-wise to the feature maps to introduce nonlinearity into the network. Common activation functions used in CNNs include ReLU (Rectified Linear Unit), sigmoid, and tanh.

Pooling Layers: Pooling layers are used to downsample the feature maps and reduce their spatial dimensions, which helps to make the learned features more invariant to small translations and distortions in the input data. Max pooling and average pooling are two common types of pooling operations used in CNNs. Fully Connected Layers: In addition to convolutional and pooling layers, CNNs often include one or more fully connected layers at the end of the network. These layers take the high-level features learned by the convolutional layers and transform them into predictions or classifications. Fully connected layers are similar to those found in traditional neural networks.

Training: CNNs are typically trained using the backpropagation algorithm along with an optimization technique such as stochastic gradient descent (SGD) or its variants. During training, the network learns to adjust the weights of the filters and the parameters of the fully connected layers to minimize a predefined loss function.

Figure 22 describes the CNN basic model.

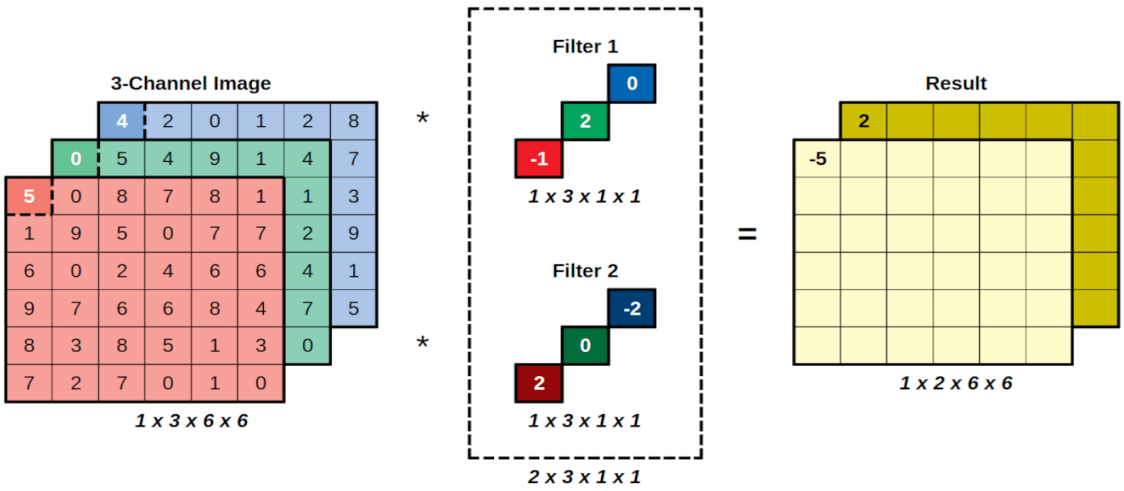


Figure 22: Convolutional Neural Network [114].

3.3.3 RNN (Recurrent Neural Network)

Recurrent Neural Network is type of artificial neural network that has feedback loops that allow it to process sequential data such as text or speech. RNNs can perform anomaly detection, machine translation, or speech recognition tasks [115].

Key components and concepts of recurrent neural networks include:

Recurrent Connections: RNNs contain recurrent connections that allow information to persist over time by feeding the output of a neuron back into its input. This recurrent structure enables RNNs to process sequences of arbitrary length and capture temporal dependencies in the data.

Hidden State: At each time step, an RNN computes an output based on the current input and the previous hidden state. The hidden state serves as the memory of the network, encoding information about previous inputs in the sequence. The hidden state is updated recursively at each time step, allowing the network to retain information over multiple time steps.

Training: RNNs are trained using backpropagation through time (BPTT), a variant of the backpropagation algorithm adapted for sequential data. During training, the network's parameters (weights and biases) are adjusted to minimize a loss function that measures the discrepancy between the predicted outputs and the ground truth.

Figure 23 describes the RNN basic model.

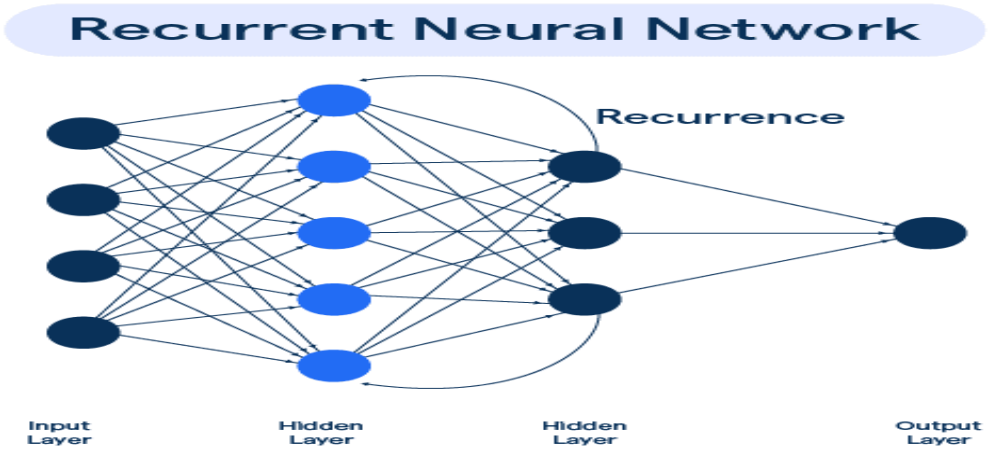


Figure 23: Recurrent Neural Network [116].

3.3.4 LSTM (Long Short-Term Memory)

is a variant of RNN that can handle long-term dependencies and avoid the problem of vanishing gradients.

Memory Cells: LSTM networks contain special memory cells that can store information over long periods of time. These memory cells have an internal state that can be updated over time based on the input data and the network's previous state.

Gates: LSTMs use a set of gates to control the flow of information into and out of the memory cells. These gates include the forget gate, input gate, and output gate, each of which regulates different aspects of the memory cell's behavior.

Forget Gate: Controls which information from the previous state should be discarded or forgotten.

Input Gate: Determines which new information from the current input should be stored in the memory cell.

Output Gate: Regulates the information that is output from the memory cell to the rest of the network.

Forget, Input, and Output Gates are implemented using sigmoid activation functions, which produce values between 0 and 1. The forget gate decides which information from the previous cell state should be retained (1) or forgotten (0). The input gate controls the flow of new information into the cell state, while the output gate regulates the flow of information from the cell state to the output.

Cell State: The internal state of an LSTM memory cell, also known as the cell state, serves as a conveyor belt that carries information across time steps. The cell state can be modified by the forget gate, input gate, and various activation functions to retain relevant information and discard irrelevant information over time.

Figure 24 describes the LSTM cell components.

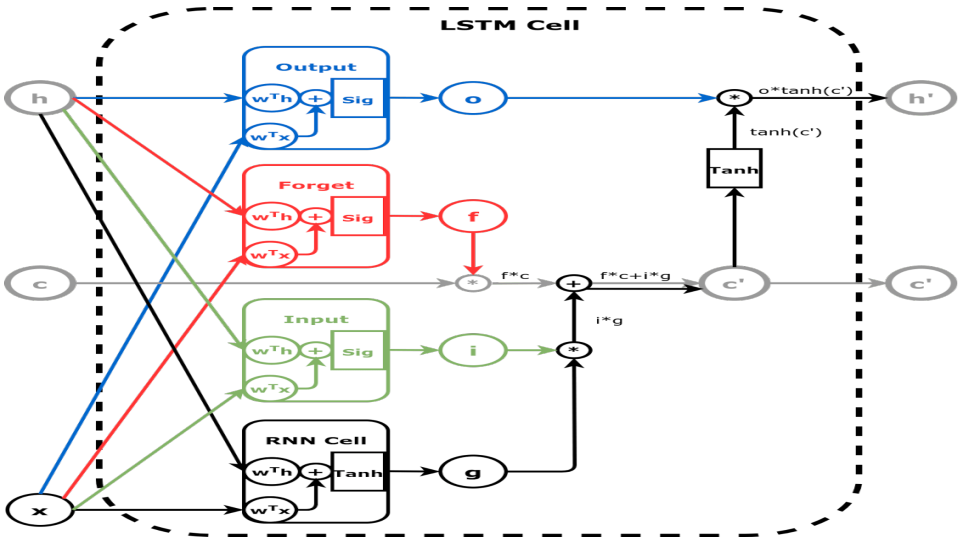


Figure 24: Long Short-Term Memory [114].

3.4 Machine Learning Metrics

Machine learning metrics are quantitative measures used to evaluate the performance of ML models. The most commonly used ML metrics are described as follows:

3.4.1 Accuracy

Accuracy represents the proportion of correctly classified instances among the total instances in a dataset. It is a measure of the overall correctness of the model.Accuracy is commonly used to evaluate the performance of classification models.

$$\text{Accuracy} = \frac{\text{Number of Correctly Classified Instances}}{\text{Total Number of Instances}} \tag{8}$$

3.4.2 Precision

Precision measures the proportion of correctly predicted positive cases (true positives) out of all instances predicted as positive (both true positives and false positives). Precision is necessary when the cost of false positives is high.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{9}$$

3.4.3 Recall

(also known as Sensitivity or True Positive Rate) Recall measures the proportion of correctly predicted positive cases (true positives) out of all actual positive cases (true positives and false negatives).Recall is essential when it is crucial to capture all positive instances.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{10}$$

3.4.4 F1 Score

The F1 score is the harmonic mean of precision and recall. It balances precision and recall, giving equal weight to both metrics. The F1 score is instrumental when seeking a balance between precision and recall.

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

4 Network Model

The network model used in our proposition is shown in Figure 25 and contains the following components:

4.1 Mission Area

The mission area of drones varies depending on the target domain. Drones may carry cameras to survey vast areas and other sensors to detect anomalies, while some transport payloads. Their findings are compiled into unified situational maps in real-time, allowing for precise targeting of rescue efforts during disasters. Infrastructure inspection missions that previously took days can be completed in hours by a drone swarm meticulously scanning for defects from multiple vantage points. Drones also enhance security operations through dynamic distributed surveillance of borders/facilities. Commercial applications involving deliveries will also be scaled up using drone networks with self-organizing traffic management.

4.2 Drone Layer

In an intelligent environment, sensor devices are randomly located on the ground, and multiple drones fly in the sky to provide wireless services for them using frequency division multiple access (FDMA)[117]. These drones are deployed in large numbers and distributed throughout the mission to provide complete coverage of the entire area, which is used to collect real-time data about the environment. Drones are specifically utilized to aggregate sensor data and transmit them to the ground control station. These devices act as a bridge between the sensors and the MEC servers and ensure that the data are transmitted reliably and securely.

4.3 MEC Servers Layer(Control Layer)

The control layer is crucial in managing and coordinating various aspects of drone operations. It is represented by the ground control station, the critical equipment, and the management center in FANET, which includes hardware and software components and facilitates human control of drones. The GCS includes Multi-access Edge Computing (MEC) servers. MEC systems are advantageous regarding data access, synchronization, storage, processing sensor data in real-time, and improving decision-making, navigation, and mission planning [118].

4.4 Cloud Layer

Cloud computing plays a vital role in FANET applications. It allows the accessible collection and management of large amounts of data from MEC servers. These data can be accessed live from any location, allowing live monitoring and end-to-end connectivity among all stakeholders involved [119]. Cloud computing also provides services, such as field maps, cloud storage, and others, that enhance the efficiency and productivity of FANET applications. Final management services use these cloud computing services to obtain all the necessary information to make final decisions.

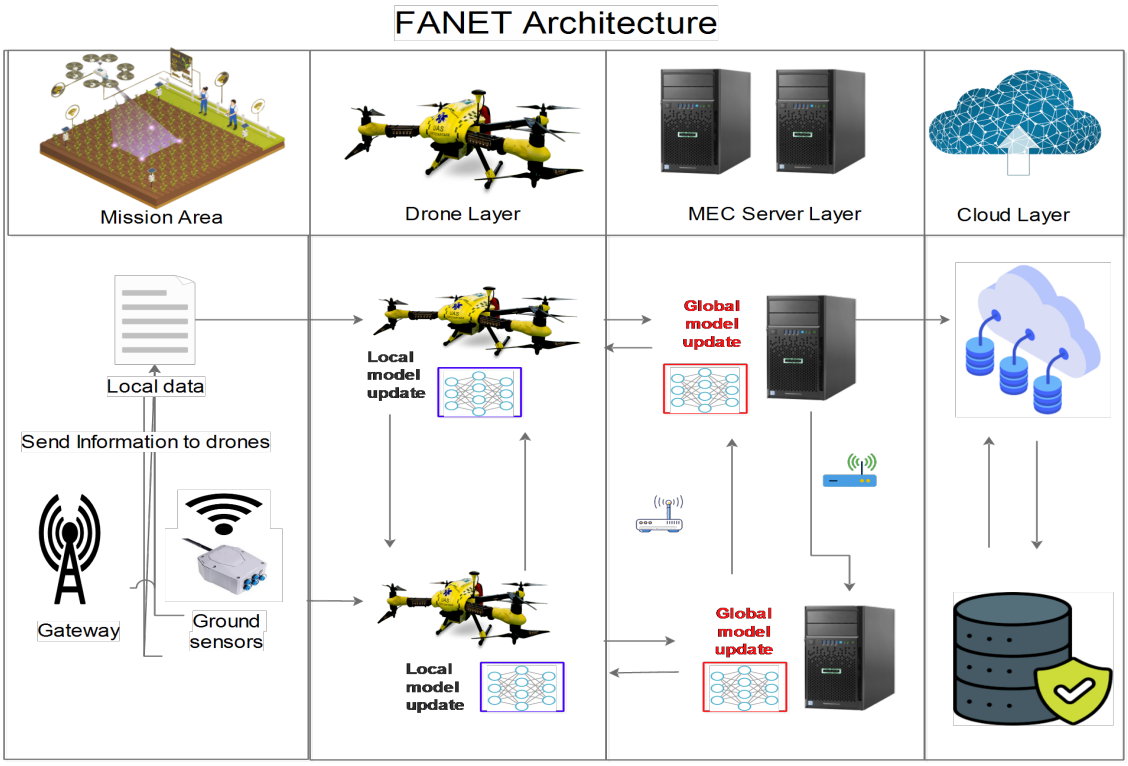


Figure 25: FANET architecture.

5 Threat Model

In our contribution, we assume that the drones have been compromised and hacked, enabling them to launch the following attacks.

5.1 Flooding Attacks

As shown in Figure 26, in this attack, a rogue drone inundates a network, system, or service with an overwhelming traffic volume, causing it to become unavailable or significantly slowing down its performance. The attack exploits the network's resources, such as bandwidth or processing capacity, by sending excessive requests [31].

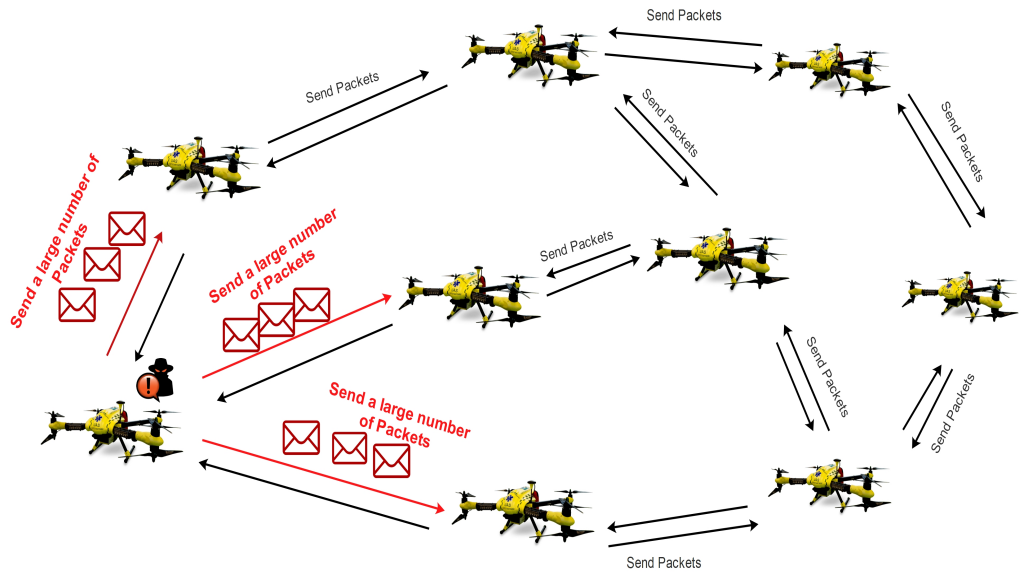


Figure 26: Flooding attacks.

5.2 Blackhole Attacks

As shown in Figure 27, in this attack, a malicious drone, termed the blackhole, deceitfully attracts incoming network traffic by advertising itself as having the most efficient route to the destination. However, instead of forwarding the packets to their intended recipients, the blackhole drone selectively drops or absorbs them, leading to a denial of service for legitimate drones [28].

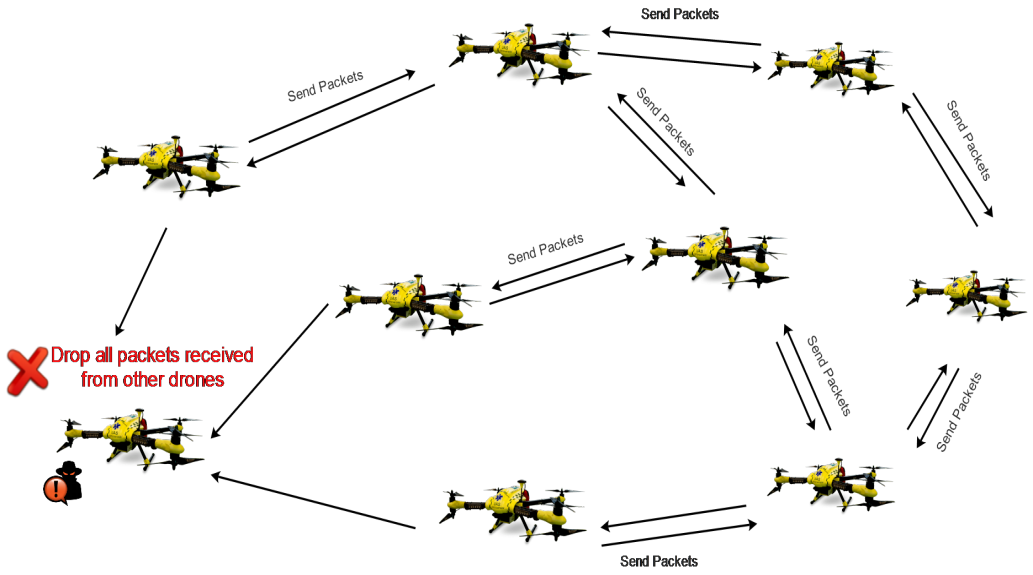


Figure 27: Blackhole attacks.

5.3 Selective Forwarding Attacks

As shown in Figure 28, in this attack, the malicious drone, known as the wormhole, establishes a high-speed, covert communication link between two distant points in the network, creating a shortcut for data transmission [26]. As a result, the attacker can replay or selectively forward packets, potentially misleading the network and compromising its integrity. Selective Forwarding attacks are also known as Wormhole attacks.

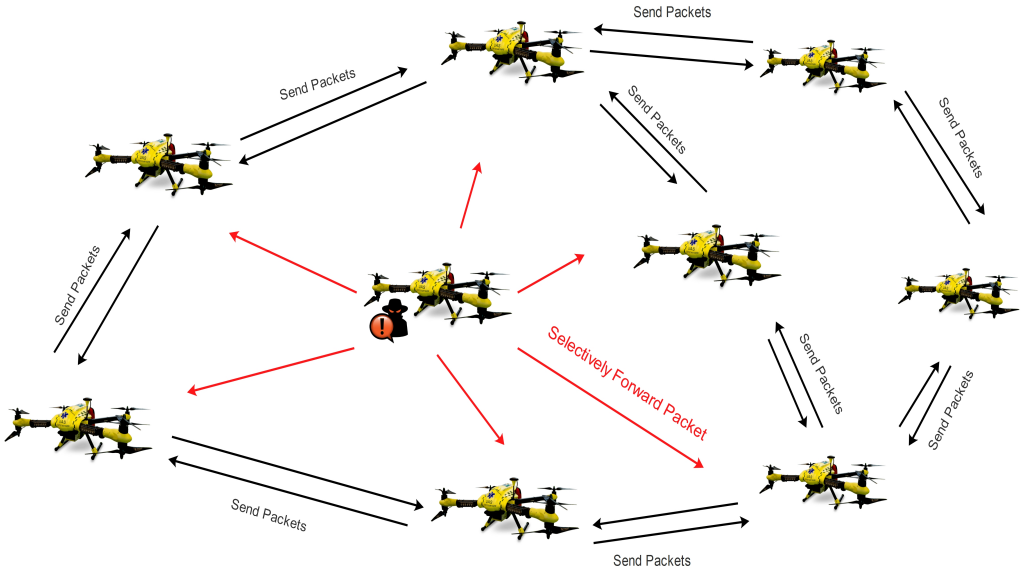


Figure 28: Selective Forwarding attacks.

5.4 Assumptions

Our assumptions are as follows:

- We presume the existence of a charging station (CH) within the mission area, utilized by drones for battery recharging purposes.
- MEC servers are considered uncompromised and trustworthy, serving as FL aggregators for network security in the FANET system.
- No malicious sensors are present; the sensor devices lack resilience. While sensor devices may initially possess vulnerabilities upon release by manufacturers, they are assumed to remain uncontaminated and uninfected during their initial deployment. Furthermore, these sensors cannot withstand or recover from adverse conditions.

- FL systems incorporate privacy-preserving techniques, such as Secure Multiparty Computation(SMC), which securely transmits ML parameters from drones to MEC servers.

6 The Proposed FL-based Intrusion Detection (FLID)

Figure 29 illustrates our federated deep learning (FLID) model for intrusion detection within FANET environments. This approach effectively categorizes network traffic into four classes: standard, selective forwarding, blackhole, and flooding attacks. FL is a decentralized ML technique facilitating collaborative model training across multiple devices without raw data exchange.

Our methodology incorporates three neural network architectures tailored for anomaly detection: MLP, RNN+LSTM and CNN.

We evaluate our method on a publicly available dataset that contains realistic network scenarios with different types of attacks. The proposed FL-based intrusion detection (FLID) has six primary phases: Global Model Initialization, Model Generic Request, Model Response, Local Training and Analysis, Model Aggregation, Refined Model and Parameter Update.

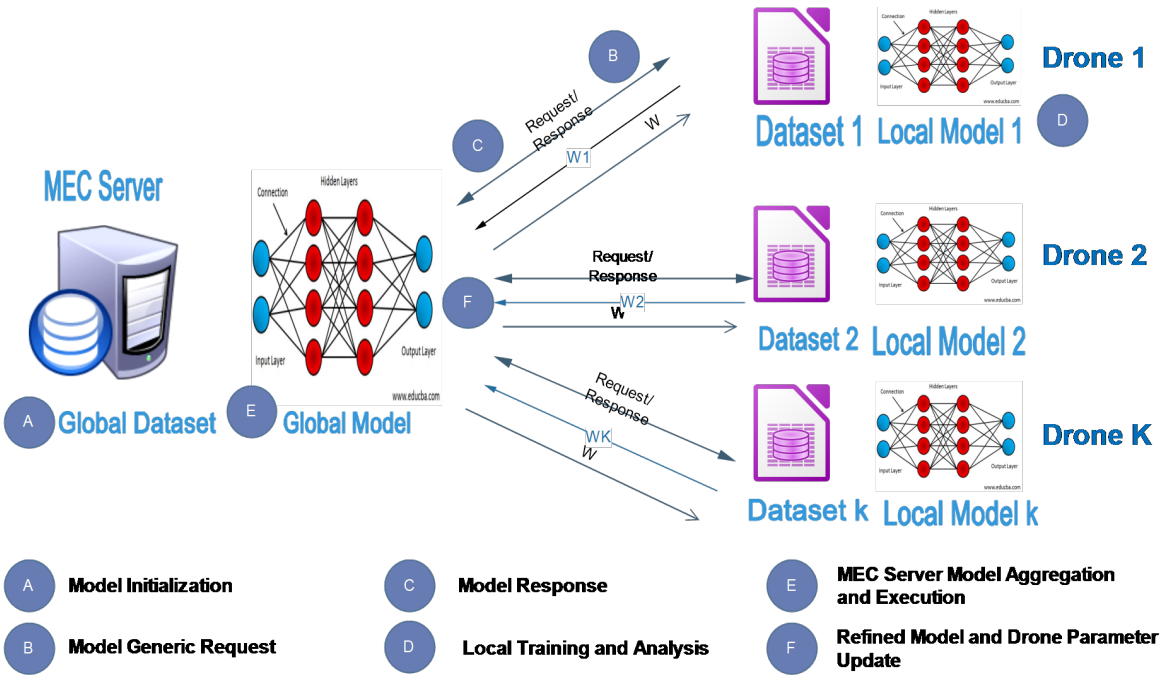


Figure 29: FL-based IDS for FANETs.

6.1 Global Model Initialization

The process begins by initializing the model on the server. Initialization involves setting up the model parameters before training or learning occurs. Two common approaches to initializing the model parameters are random and initialization from a previously saved checkpoint. In our case, we utilize random initialization. Random initialization involves assigning random values to the model's parameters. This ensures that each training round in FL begins with a different set of initial parameter values. This prevents biases that could be introduced if the same initial parameters were used consistently.

6.2 Model Generic Request

The drone k requests the MEC server to participate in the FL task. The request typically includes the drone's ID, the data it can contribute to the task, and the model architecture and hyperparameters.

6.3 Model Response

The MEC server responds to the drone's request by returning the initial model parameters for training. Along with the initial model parameters, the answer may include additional instructions. These instructions can specify the number of local training iterations the drone should perform and the portion of data (denoted as p_k) that should be used for training.

6.4 Local Training and Analysis

Every individual drone indexed as k undergoes training of the local model utilizing its specific local dataset denoted as p_k, subsequently adjusting the model parameters by the outcomes of this local training phase. The resultant parameter updates are then transmitted to the MEC server, which undergoes aggregation to construct a refreshed global model.

The dataset is divided into B batches for each local epoch, with each drone randomly opting for a subset b of these batches for training. This deliberate selection process injects variability and diversity into the training process.

The drone adjusts the local weights denoted as w based on the training data within the chosen batches using the following formula:

$$w = w - \eta \triangle (w; b) \tag{12}$$

For each mini-batch b ,$\triangle (w; b)$ represents the change in the weight w. The learning rate (η) is a hyperparameter that controls the size of the weight updates. $\eta \triangle (w; b)$ is subtracted from the w to compute the new weight for the local model. This means that the new weight w is obtained by subtracting the product of the learning rate (η) and the change in the weights $\triangle (w; b)$ from the current weight w. Finally, the drone securely transmits the updated model to the MEC server utilizing Secure Multiparty Computation, safeguarding the privacy of its local data.

6.5 MEC Server Model Aggregation and Execution

The MEC server consolidates the model updates it receives from the drones, generating a revised global model. Subsequently, this updated global model is disseminated back to the drones, serving as their initial reference point for the subsequent round of local training.

After each iteration m, drone k updates w_{m+1}^k. Leveraging the federated averaging technique, the MEC server computes a weighted average of the local updates received from individual drones. As w_{m+1}^k arrives at the MEC server, local drone updates are integrated to refine the global model in the following manner:

$$w_{m+1}^k = \sum_{k=1}^{k} \frac{\mathrm{p}k}{\mathrm{p}} w_{m+1}^k \tag{13}$$

w_{m+1} is the global weight after m rounds over all the p data points.

The loss function of drone k on the local portion dataset p_k is defined as follows:

$$L_k(w) = \frac{1}{\mathrm{p}k} \sum_{i=1}^{k} f_i(w) \tag{14}$$

Where

$f_i(w) = f(x_i, y_i, w)$

$f_i(w)$ is a loss function of prediction for input x_i to an expected output y_i with weight vectors w. Based on the above formulation, the following is how the global loss function minimization is derived:

$$MinL(w) = \sum_{k=1}^{k} \frac{\mathrm{p}k}{p} l_k w \tag{15}$$

Where k is the Number of participants drones in the current learning round and $L_k(w)$ is the local objective function of the kth drone.

6.6 Refined the Model and Drone Parameter Update

Following the aggregation process (step 5), we acquire a model trained on the data contributed by all participating drones. Nevertheless, it is imperative to recognize that this model undergoes only a brief training period. It is essential to continually iterate through this process to attain a fully trained model capable of performing optimally across all drone data.

At this step, the drone can download the global weight W, signifying the model parameters derived from the aggregation. This global weight is valuable during the drone's local training procedure.

By leveraging the globally ratified update, which encapsulates the collective knowledge gleaned from all drones, individual local drones can augment their learning capabilities. The details of our FL algorithm are explained in Algorithm 1.

Algorithm 1 Federated learning algorithm based IDS

MEC Server:
Models parameter ← values
Model aggregation and execution
ForEach drone-client$_k$
$w_{m+1}(k) \leftarrow Drone - client(k, w)$
$w_{m+1}(k) \leftarrow \sum_{k=1}^{k} \frac{\mathrm{p}k}{\mathrm{p}} w_{m+1}^{k}$
Refined the model
Function Drone(k,w)
Local training and analysis
Split p_k into b batch of size B
ForEach Local-epoch
Forbatch b
$w = w - \eta \triangle (w; b)$
return w
End Function

7 Attack Mitigation

After training and deploying the FL model, the MEC servers can classify incoming traffic into predefined categories, which include selective forwarding, blackhole attacks, and flooding attacks. These incidents may trigger alarms or alerts indicating a security threat and automatic blocklisting. Our system is built to respond rapidly and take practical actions to prevent or address security breaches.

7.1 Selective Forwarding Attack Mitigation

For traffic classified as a selective forwarding attack, mitigation strategies may include:

Packet Filtering: Identify and filter out malicious packets or those exhibiting selective forwarding behavior.

Dynamic Routing: Modify routing paths dynamically to bypass compromised drones.

Authentication and Authorization: Strengthen authentication and authorization mechanisms to prevent unauthorized access to the network.

7.2 Blackhole Attack Mitigation

When traffic is identified as a blackhole attack, mitigation measures can involve:

Route Reconfiguration: Adjust network routing to avoid or isolate malicious drones.

Rate Limiting: Restrict traffic to the malicious drone to minimize its impact.

7.3 Flooding Attack Mitigation

In the case of flooding attacks, mitigation steps may include:

Traffic Shaping: Implement traffic shaping to limit the rate of incoming traffic to prevent network congestion.

Access Control Lists: Configure ACLs to filter and block traffic from known malicious drones.

8 Model Preparation

This section describes the dataset used for our proposed FL-based intrusion detection. We also detail the data preprocessing and discuss overfitting mitigation.

8.1 Dataset Description

To the best of our knowledge, there is no dedicated dataset specifically tailored to drone attack scenarios. Consequently, several datasets are commonly used in drone network research, including the RF dataset [120], the CICIDS2017 dataset [121], and the UNSW-NB15 dataset [115]. Additionally, our model incorporates data from the WSN-BFSF dataset [122], chosen for its recent introduction in 2023, adding a novel dimension to our analysis. Moreover, our drones have diverse sensors to augment our data collection and analysis efforts.

The WSN-BFSF dataset serves a specific purpose in IDS within wireless sensor networks, encompassing three primary routing attack types: flooding, blackhole, and selective forwarding. These attack scenarios are pivotal for evaluating the efficacy of our proposed model. Our experimental setup utilizes the WSN-BFSF dataset, comprising 312106 instances, each delineated by 16 distinct features.

8.2 Data Preprocessing

Data preprocessing is an essential step in preparing data for model training. It involves cleaning, transforming, and organizing the data to ensure it is in a suitable format for further processing. First, to handle the categorical feature 'Class,' we employed label encoding. Second, we normalized skewed data using the Standard Scaler, which is necessary for DL methods. The Standard Scaler is typically used to perform feature scaling by transforming the data with a mean (average) of 0 and a standard deviation of 1. Third, we computed the correlation coefficient between each feature and the target variable to measure the strength and direction of the linear relationship between them. The parts were then sorted based on their correlation coefficients with the target variable, considering positive and negative values. Finally, we selected the top-ranked features based on the correlation coefficients. If two or more elements were strongly correlated, one might need to be removed to avoid redundancy and improve model interpretability. The feature's importance is depicted in Figure 30.

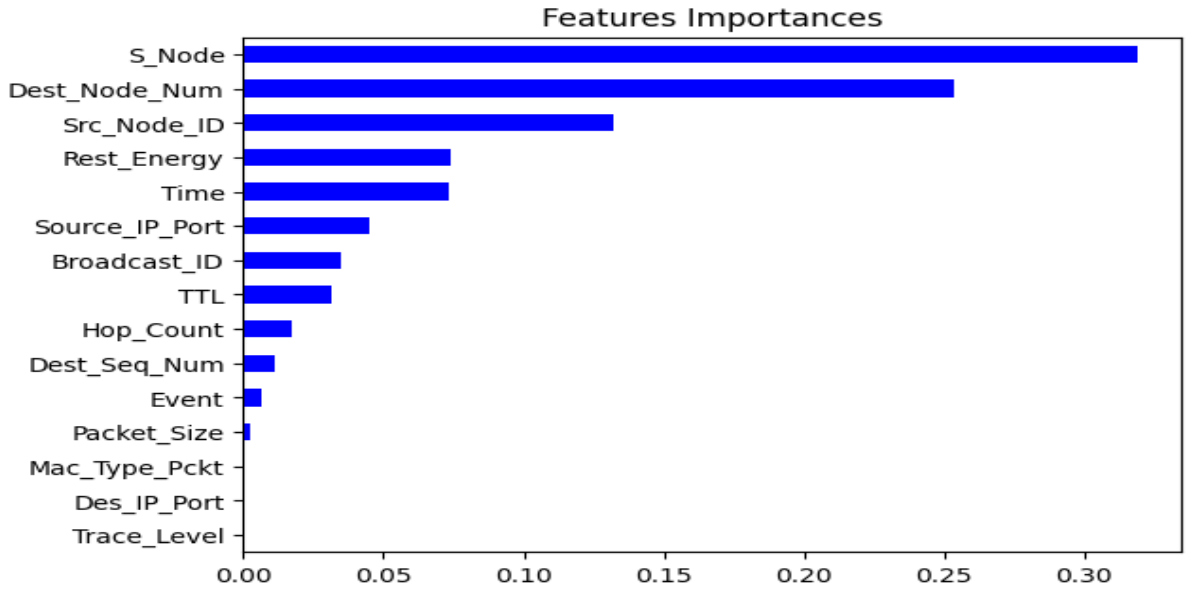


Figure 30: Features importance.

8.3 Overfiting Mitigation

To mitigate the overfitting, we utilized two methods in our model: stratified K-Fold cross-validation and Early stopping. These methods are briefly explained below:
- Stratified K-Fold cross-validation is a method that splits the data into K folds in our model (k=5), where each fold contains approximately the same proportion of classes as the original data. Then, one fold is used as the validation set and the rest as the training set.
- Early stopping is a method that monitors the validation loss during the training process and stops the training when the loss stops decreasing or increases. In our model (Patience=3, monitor=val loss). This prevents the model from overfitting to the training data and losing its ability to generalize to new data.

9 Experimental Setup

The experimentation was conducted utilizing a Jupyter Notebook environment on an HP Notebook powered by an Intel Core i7 10th generation processor and 8 GB of RAM. Python served as the primary programming language for implementation, while our model was realized through established libraries. Specifically, numerical computations were handled using Numpy, data manipulation with pandas, ML functionalities with sci-kit-learn, and the training and testing of the FL model were facilitated by Keras and TensorFlow libraries.

The FL process encompassed 20 rounds (R) of communication among drones. At each local node, a batch size of 32 (B) and four local epochs was employed for training, with a local learning rate (LR) set at 0.001 to regulate the pace of parameter updates.

Before model training, the dataset underwent preprocessing, resulting in 312106 instances, each characterized by eight features. The dataset was then partitioned into 80% for training and 20% for testing, with four distinct classes: Normal, Flooding attack, Blackhole attack, and Selective Forwarding attack.

10 Experimental Results

This section outlines the optimal federated deep neural network model for anomaly detection. We then delve into the performance evaluation of our proposed approach.

10.1 The Optimal Federated Deep Neural Network Model for Anomaly Detection

10.1.1 Flooding Attack

Figure 31 shows that the best model is RNN+LSTM with federated training. It achieves a perfect precision, recall, and F1 score of 0.99. CNN and MLP are also good models, with F1 scores of 0.97 and 0.98, respectively. But RNN performs slightly better.

10.1.2 Blackhole Attack

Figure 32 shows that the best model is RNN+LSTM with federated training. It has the highest precision of 0.99, recall of 0.96, and F1 score of 0.98 MLP has the next-best performance, with an F1 score of 0.85. CNN performs poorly on this attack, with an F1 score of only 0.54.

10.1.3 Selective Forwarding Attack

As shown in Figure 33, the RNN+LSTM and MLP have comparable performance with F1 scores of 0.96 and 0.93, respectively. RNN+LSTM has a slightly higher recall of 0.96 compared to MLP's 0.91. CNN performs the worst for this attack, with an F1 score of only 0.65.

In summary, for all three attacks, RNN+LSTM with federated training consistently shows the best overall performance in achieving high precision, recall, and F1 scores. CNN seems to be the weaker model. MLP is a close second-best model after

RNN+LSTM. So, based on this analysis, RNN+LSTM would be recommended as the best model for detecting these cyber attacks.

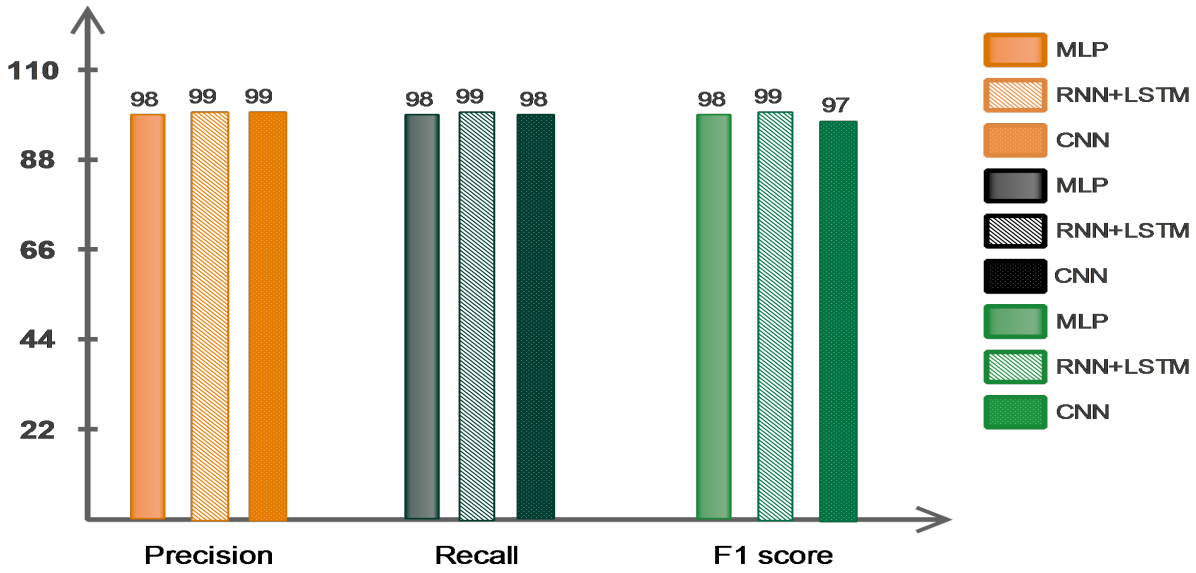


Figure 31: Models performance for Flooding attacks.

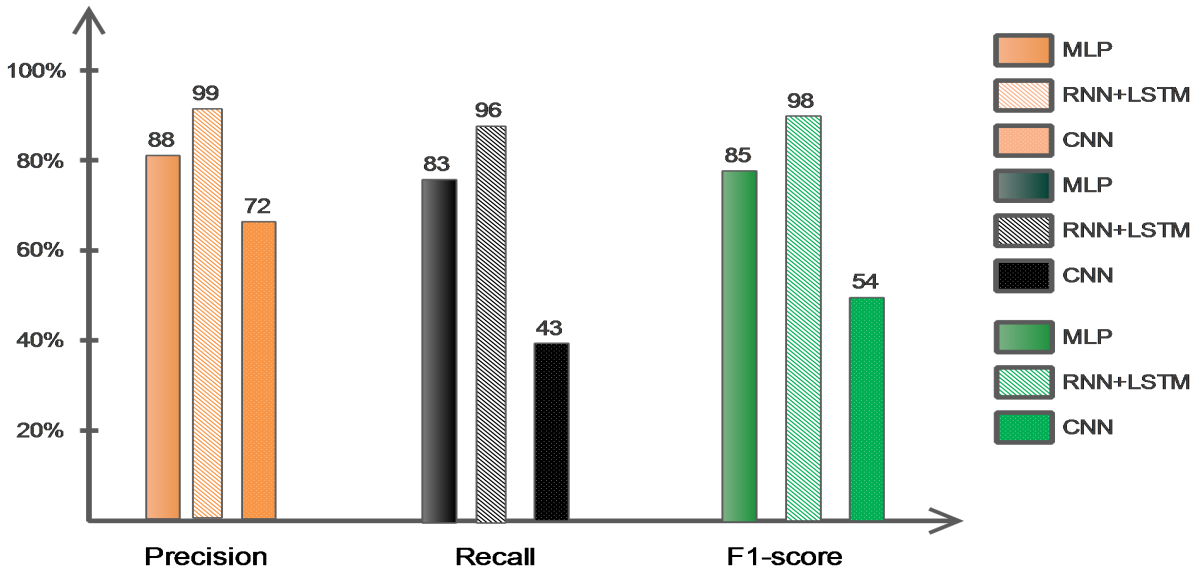


Figure 32: Models performance for Blackhole attacks.

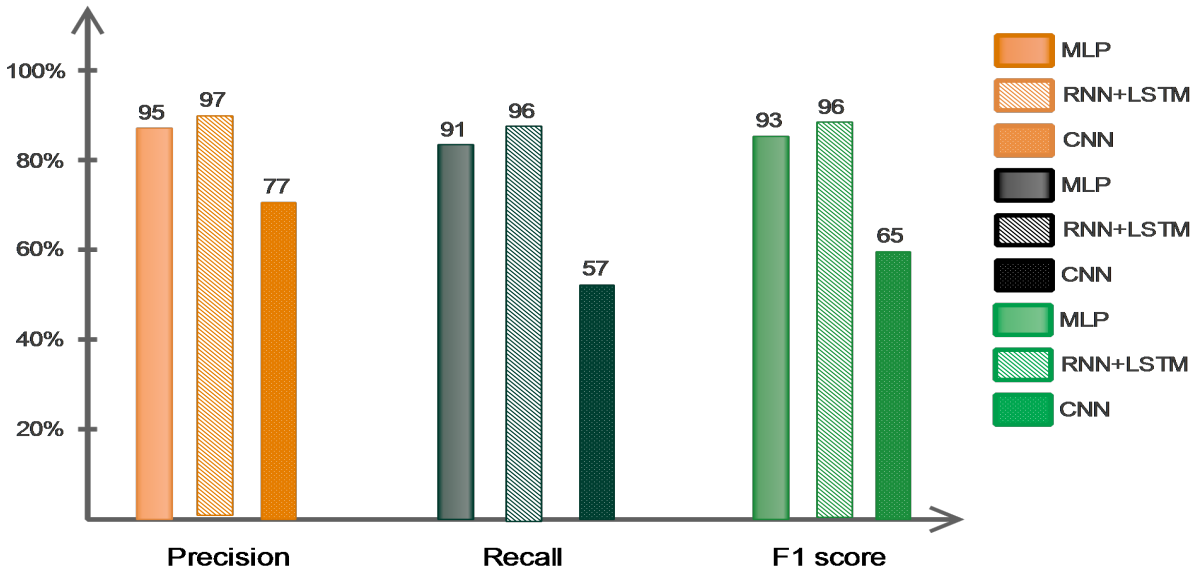


Figure 33: Models performance for Selective Forwarding attacks.

10.2 Performance Evaluation of FLID

The analysis of different models under Federated Learning (FL) based on Table 7 reveals distinctive performance patterns:

The RNN+LSTM model consistently displayed robust performance improvements with each additional communication round on the WSN-BFSF dataset. Its accuracy exhibited a steady upward trend, surpassing 99% after 25 rounds. This trend suggests that RNN+LSTM excels in learning representations from federated distributed data sequentially.

In contrast, while Federated MLP initially showed significant accuracy improvements between rounds 1-5, its progress slowed down after ten rounds, stabilizing around 98%. This suggests the MLP architecture might have quickly reached its learning limit from the decentralized data.

Federated CNN began with the lowest accuracy among the models and gradually improved with more communication rounds, but it plateaued below 97%. This indicates that the CNN architecture's capability to extract meaningful features across devices is comparatively constrained compared to RNNs and MLPs in a federated learning setup.

Table 7: FL performance

Accuracy	Federated RNN+LSTM	Federated MLP	Federated CNN
Accuracy in First Round	94.23 %	94.03%	93.75%
Accuracy after 5 rounds	95.12 %	95.71%	94.87%
Accuracy after 10 rounds	96.79 %	96.24%	95.64%
Accuracy after 15 rounds	98.23 %	97.06%	95.79%
Accuracy after 20 rounds	99.13 %	97.81%	95.84%
Accuracy after 25 rounds	99.38 %	98.41%	96.45%

11 Performance Comparison

This section outlines the classification accuracies of three distinct types of deep neural networks (MLP, RNN+LSTM, and CNN) in FL and centralized learning. Next, we present the comparative results of ROC curves and AUC Values in centralized and federated learning.

11.1 Performance Assessment: ROC Curves and AUC Values Comparison in Centralized and Federated Learning

ROC curve: stands for "Receiver Operating Characteristic curve." It's a graphical plot that illustrates the diagnostic ability of a multiclassifier system as its discrimination threshold is varied. The curve is created by plotting the true positive rate (sensitivity) against the false positive rate (1-specificity) at various threshold settings.

AUC: stands for Area Under the Curve It's a scalar value that quantifies the overall performance of a multiclassification model based on its ROC curve. AUC ranges from 0 to 1, where a higher value indicates better discrimination ability of the model. An AUC of 0.5 suggests no discrimination, meaning the model performs as well as random chance, while an AUC of 1 suggests perfect discrimination.

Figure 41 illustrates notable distinctions between the ROC curves and the corresponding AUC values of centralized and federated MLP, RNN+LSTM, and CNN.

11.1.1 MLP

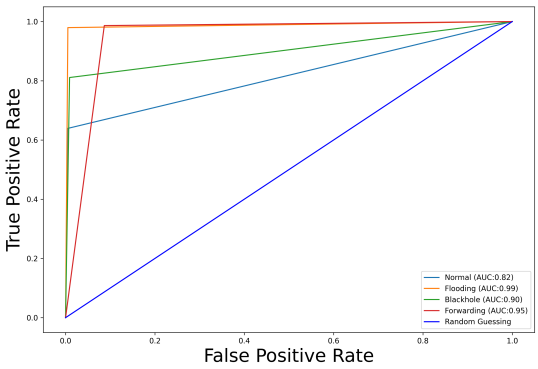
In the case of centralized MLP, it exhibits high AUC values for flooding (AUC = 99%), selective forwarding(AUC=95%), and blackhole (AUC = 90%, signifying its capability to classify these traffic types accurately. However, it demonstrates relatively lower AUC values for normal traffic (AUC = 82%), indicating some challenges distinguishing these types of attacks. Conversely, federated MLP showcases improvements in AUC values across all traffic categories. It excels particularly in flooding (AUC = 99%) and attains perfect classification in the case of normal traffic (AUC = 97%). Federated MLP also maintains high AUC values for selective forwarding (AUC = 95%) and blackhole (AUC = 91%). These findings underscore federated MLP superiority in classifying various attack types, notably in scenarios involving flooding, selective forwarding, and blackhole attacks.

11.1.2 RNN+LSTM

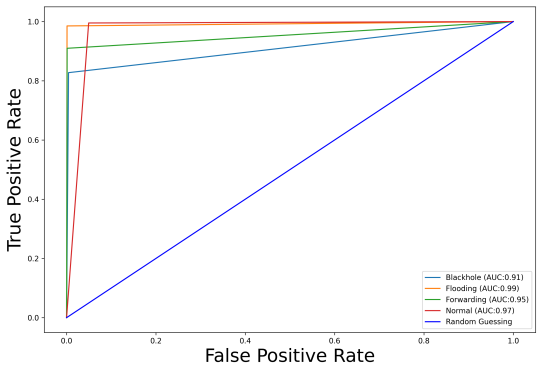
The case of the centralized RNN+LSTM model demonstrates high AUC values across several traffic categories. Notably, it achieves AUC values of 99% for flooding, 96% for selective forwarding, and 97% for blackhole attacks, indicating strong performance in identifying these types of attacks. However, the model exhibits relatively lower AUC values for normal traffic, with an AUC of 86%. Conversely, the federated RNN+LSTM model showcases notable improvements in AUC values across all traffic categories. It mainly achieves perfect classification in flooding attacks with an AUC of 100% and near-perfect classification in normal traffic with an AUC of 99%. The federated RNN+LSTM model also maintains high AUC values for selective forwarding (AUC = 99%) and blackhole attacks (AUC = 98%). These results underscore the superiority of the federated RNN+LSTM approach in accurately classifying flooding, selective forwarding, and blackhole attacks.

11.1.3 CNN

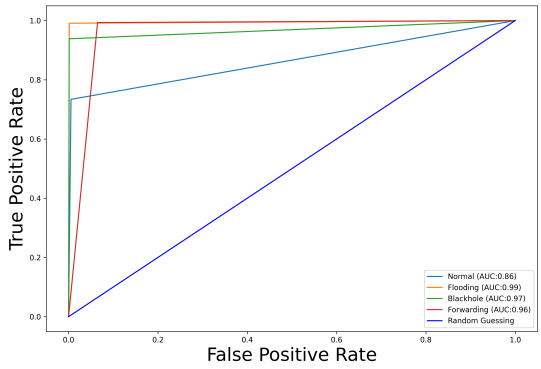
The centralized CNN model showcases exceptional performance in identifying flooding attacks, achieving a perfect AUC of 100%. However, regarding selective forwarding attacks, the model demonstrates slightly lower performance, with an AUC of 89%. This suggests the model may face challenges distinguishing selective forwarding attacks from normal network behavior. Furthermore, for blackhole attacks, the AUC drops to 74%, indicating a notable performance decrease. Similarly, the model struggles to distinguish normal traffic, as evidenced by the AUC of 65%. Conversely, while federated CNN demonstrates substantial improvements in AUC values across several traffic categories, it's important to note that its performance varies across different attack types. Notably, the model identifies flooding attacks with an outstanding AUC of 99%; surprisingly, the AUC for normal traffic is 90%, indicating strong performance in distinguishing normal network behavior. However, regarding selective forwarding and blackhole attacks, the model's performance diminishes, with AUC values of 78% and 71%, respectively. The findings still highlight the superiority of the federated approach in classifying flooding attacks.



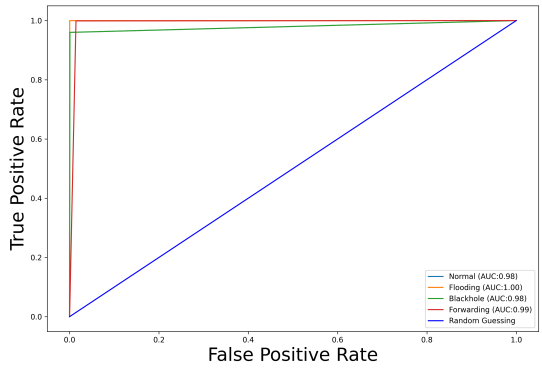
(a) ROC curve using Centralized MLP



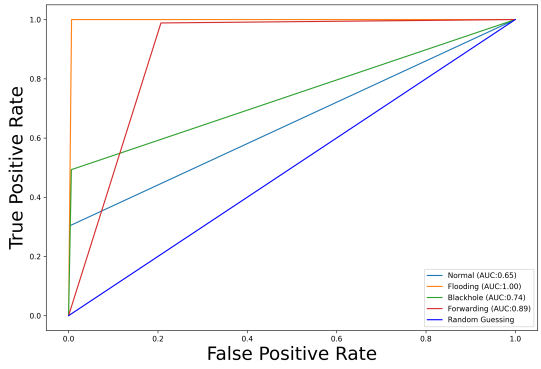
(b) ROC curve using Federated MLP



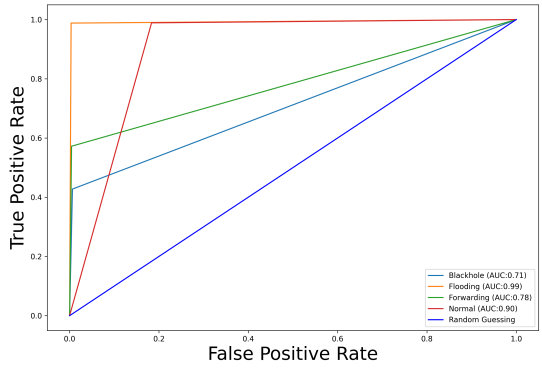
(c) ROC curve using Centralized RNN+LSTM



(d) ROC curve using Federated RNN+LSTM



(e) ROC curve using Centralized CNN



(f) ROC curve using Federated CNN

Figure 34: ROC curve and AUC values using different models in centralized and federated learning.

12 Conclusion

Integrating FANET technology with intelligent sensors and devices has led to several innovative applications. However, this convergence has exposed FANETs to many potential security threats, jeopardizing the network's integrity and reliability. Over the past few years, a surge of interest has been observed in leveraging ML-based intrusion detection for FANETs, and one up-and-coming solution is FL. In this chapter, we introduce a novel FL model tailored explicitly for intrusion detection within FANETs, named FLID. Our innovative approach allows drones within the network to engage in localized model training on their data while safeguarding sensitive information privacy and harnessing each drone's unique capabilities. In practical terms, our FLID model exhibits the ability to effectively identify and thwart three distinct network attack types, including flooding, blackhole, and selective forwarding attacks. In our experiments, we analyzed the efficacy of various deep neural network architectures for our intrusion detection system: multilayer perceptron, convolutional neural networks , and recurrent neural networks with long-short-term memory. In particular, the RNN + LSTM model demonstrated exceptional performance, achieving the highest accuracy and outstanding precision, recall, and F1 score. Based on the RNN+LSTM model, the upcoming chapter will explore additional advancements in FL by integrating another ML algorithm and leveraging its unique capabilities to enhance model performance and adaptability in distributed environments.

Chapter V: Reinforcement Learning-based Drone Client Selection for Efficient Federated Learning-based IDS in FANETs

"Digital freedom stops where that of users begins. Nowadays, digital evolution must no longer be offered to customers in a trade-off between privacy and security. Privacy is not for sale; it's a valuable asset to protect."

– Stephane Nappo

1 Introduction

Intrusion Detection Systems emerge as a promising solution in FANET security, enabling monitoring of network traffic and identifying suspicious activities or potential attacks. Leveraging ML-based-IDS technology in FANETs enhances situational awareness and allows proactive defense measures against emerging threats. The dynamic topology and resource limitations inherent in FANETs necessitate adaptive and efficient IDS frameworks tailored to the unique challenges of these networks. Robust IDS deployment empowers FANET operators to swiftly respond to security incidents, safeguard critical assets, and ensure uninterrupted operation.

FL offers a promising approach to enhance IDS capabilities within FANETs by enabling collaborative learning among distributed drones while preserving data privacy. Unlike traditional centralized IDS, where all data is collected and analyzed in a central repository, FL allows individual drones to train local models using their data and share only model updates with a central server or among neighboring drones. Improving FL-based IDS solutions in FANETs is paramount for several reasons. The inherent dynamism of FANETs, combined with drones' varying computational capabilities, poses notable challenges in optimizing the FL process. Consequently, selecting the most appropriate drone for each FL operation is critical. This chapter presents a comprehensive solution to enhance the model proposed in Chapter IV. It leverages FL's benefits for IDS while integrating RL's context awareness. The primary focus will be addressing communication and resource-constrained issues within the dynamic FANET environment. The rest of this chapter is organized as follows: Section 2 presents the motivation behind the research. Section 3 provides the background of the proposition. Section 4 outlines the network model and IDS requirements. Following that, Section 5 introduces the proposed model. Section 6 describes the experimental setup, while Section 7 presents the experimental results. Section 8 conducts a performance comparison. Finally, Section 9 concludes the chapter.

2 Motivation

When applying ML for network intrusion detection, it is crucial to select appropriate features and algorithms and effectively combine them to achieve optimal performance. FL is a promising solution for IDS in FANETs, and diverse schemes have been introduced for the Internet of Things (IoT) [123]-[124] and vehicular ad hoc networks (VANETs) [109]-[125]. However, implementing FL in dynamic and resource-constrained FANETs introduces notable challenges. Two primary obstacles emerge in the context of FL for IDS within FANETs.

Firstly, communication challenges arise due to the high mobility and dynamic topology of participating drones, potentially causing transmission issues between the server and drones during FL parameter exchange. Consequently, the precise positioning of each drone becomes crucial for selecting suitable participants in the FL process.

Secondly, drones' resource-constrained natures present a significant problem. Drones exhibit heterogeneous capabilities, including varying levels of computing power, constrained by limited battery and storage capacity that may fluctuate over time. This heterogeneity necessitates an intelligent selection mechanism to determine which drones are most suitable for participation, leading to challenges in model synchronization.

Motivated by these challenges, we aim to address the issues of communication and resource constraints. The contributions outlined in this chapter are as follows:

- Our proposal introduces a novel FL approach for IDS within FANETs, utilizing a fusion of FL and reinforcement learning algorithms. This combination aims to meet the security requirements of the FANET environment, encompassing aspects such as privacy protection, effectiveness, robustness, and context awareness.
- We advocate using the Q-learning algorithm to tackle the drone-client selection challenge. This framework empowers MEC servers to learn from drone behavior, enabling them to make informed decisions regarding drone-client selections based on location, power capacity, and storage availability.
- We employ the Epsilon Greedy policy, which balances exploration and exploitation. This policy enables MEC servers to make informed decisions when selecting suitable drones for the FL process, thus enhancing the efficiency and effectiveness of the overall system.

3 Background

In this Section, we will discuss the background of our proposed system. Firstly, we will provide an overview of the RL paradigm. Next, we will introduce the Markov Decision Process (MDP) and describe the Q-learning algorithm. Lastly, we will explore the epsilon greedy policy.

3.1 Reinforcement Learning

Reinforcement Learning (RL) is one of the most pivotal algorithms that significantly contribute to the advancement of artificial intelligence. RL is considered one of the three fundamental ML paradigms, where an intelligent agent learns to interact with a given environment to maximize a cumulative reward [126]. The agent utilizes the MDP to explore the environment by taking action and receiving feedback in the form of rewards. The agent then uses this feedback to update behavior and enhance decision-making. Reinforcement learning encompasses four key components: policy, reward function, value function, and environmental model.

3.2 Markov Decision Process

Markov Decision Process is a mathematical framework used to model decision-making problems and aid the agent in probabilistically controlling the process. MDPs are characterized by states, actions, transition probabilities, and rewards. The state space is denoted by S and represents a finite set of possible conditions. The action space is denoted by A and represents a finite set of potential actions. The transition probability from the current state S to the next state S' after taking action A is represented by P. The immediate reward the environment provides for taking action denotes a.Generally, an MDP is defined by the tuple (S, A, P, R), which captures these four parameters [127].

As shown in Figure 36, the agent observes its current state S_t in the environment at each time step before taking action A_t. The environment rewards the agent R_t; the next state is S_{t+1}. The main goal of the agent is to select a policy π that maximizes the accumulated rewards from the environment described by :

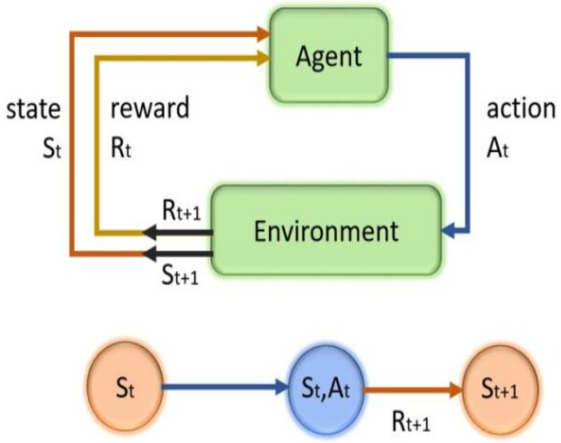


Figure 35: Markov Decision Process [128].

$$max|\sum_{t=0}^{T}\delta R_t(S_t, \pi S_t)| \tag{16}$$

where $\delta \in [0, 1]$ is the discount factor. When the state transition probabilities are known in advance, a Bellman equation known as the Q-function is created utilizing the discounted reward to determine the following action A_t. The Q-function is characterized by:

$$Q(S_t, A_t) = Q(S_t, A_t) + \alpha(R_{t+1} + \gamma max_a(Q(S_{t+1}, a) - Q(S_t, A_t))) \tag{17}$$

Where α is the learning rate.

3.3 Q-Learning Algorithm

Q-learning is an RL algorithm that teaches agents how to make decisions in an environment to maximize cumulative rewards. It is beneficial when the agent has limited environmental knowledge and must learn through trial and error [129]. It achieves this by iteratively updating an action-value function, known as the Q-function, based on observed rewards and state transitions. One critical advantage of Q-learning is its ability to learn from exploratory actions outside the current policy, allowing for greater exploration of the environment. This feature is handy when the agent needs to gather more information about the environment to make better decisions. Another advantage of Q-learning is its model-free nature, which does not require an accurate environmental model. By iteratively updating the Q-function, Q-learning lets the agent gradually improve its action-selection policy and converge to the optimal policy π^* that maximizes long-term cumulative rewards.

3.4 Epsilon Greedy Policy

In RL, the epsilon-greedy policy is commonly used to balance exploration and exploitation. The policy determines whether the agent should select the action with the highest expected reward (exploitation) or explore new actions that may lead to higher rewards in the long run. This policy is commonly employed in the Q-learning algorithm and other similar algorithms [130]. Figure 37 explains how Epsilon Greedy Policy works.

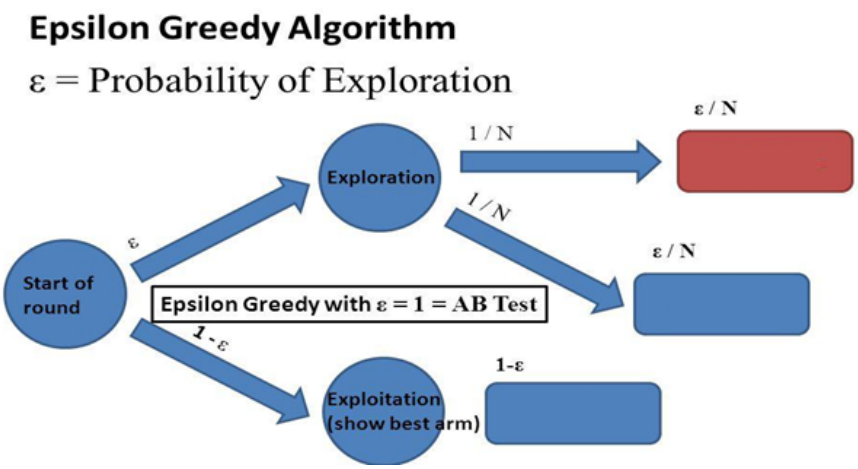


Figure 36: Epsilon greedy policy [131].

4 System Model

In this Section, we outline the network model and IDS requirements.

4.1 Network Model

The network model is represented in Figure 38 and contains the following components:

4.1.1 FANET Infrastructure

The FANET consists of a fleet of drones equipped with sensors and communication modules. Drones operate decentralized, forming an ad hoc network to communicate with each other and with ground stations. They can process data locally and perform basic intrusion detection tasks onboard.

4.1.2 Multi Access Edge Computing (MEC) Servers

MEC servers are strategically positioned within the FANET coverage area, typically at the fixed ground station. These servers are equipped with powerful computational resources and storage capacity. MEC servers act as aggregation points for data collected by drones, providing additional processing capabilities for more advanced intrusion detection algorithms.

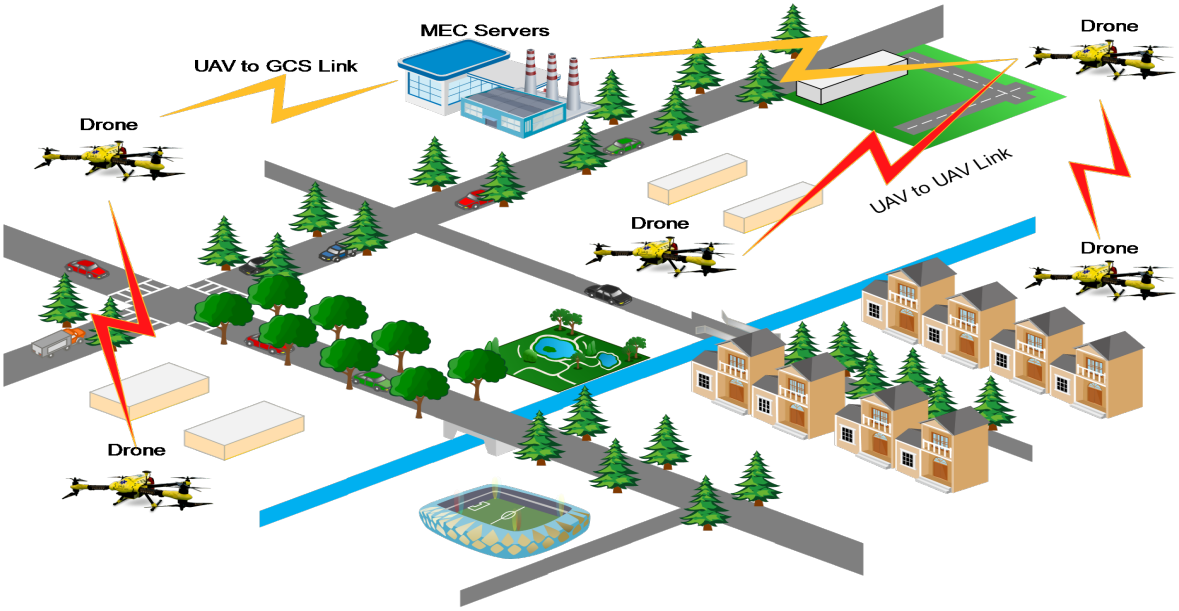


Figure 37: Network model.

4.2 IDS Requirements

A set of IDS requirements is defined as follows:

4.2.1 Effectiveness

In the context of IDS for FANETs, effectiveness involves the system's ability to achieve its intended objectives and deliver expected services to users and applications. It encompasses various performance indicators, including but not limited to recall, precision, accuracy, and F1-score [87].

4.2.2 Robustness

A robust IDS performs well even in the presence of noisy or incomplete information, outliers, or adversarial attacks; robustness represents the resilience and stability of an IDS in the face of uncertainties, adversarial activities, or changing conditions[88].

4.2.3 Privacy Protection

The data used for schemes-based IDS include private information. Privacy protection refers to the measures and mechanisms implemented to safeguard the sensitive information of drones involved in FANETs.

4.2.4 Context Awareness

Context awareness in IDS refers to the system's ability to consider the specific circumstances, environment, and conditions in which IDS are made, especially in dynamic systems like FANETs.

5 Proposed Model

Our scheme focuses on applying FL for IDS in FANET. However, drones have heterogeneous capabilities and a dynamic topology, and selecting the appropriate drone for model training poses challenges. As a result, we employ an RL approach to select the drones that will participate intelligently in the FL process. This selection phase ensures optimal resource allocation and participation of drones with desirable characteristics. As shown in Figure 39, the proposed model contains three main steps: RL-based- drone client selection, local training for anomaly detection, and finally, federated averaging and decision-making.

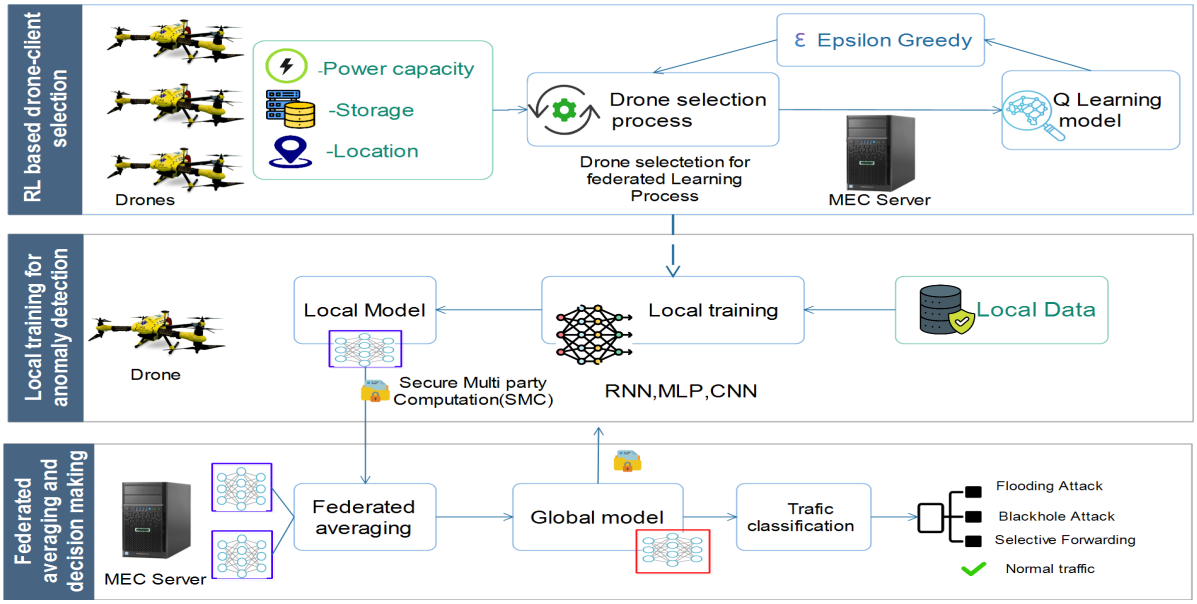


Figure 38: RL for Efficient FL-based IDS in FANETs

5.1 RL based-Drone Client Selection

In our proposal, the MEC server(Agent) utilizes the Markov Decision Process to explore the environment by taking action and receiving feedback as rewards. The MEC servers are regarded as multiple agents responsible for observing the drones and aiming to maximize the cumulative expected reward. The MDP, denoted as S, A, P, and r, can be described as follows: S represents the state space, A represents the action space, P represents the state transition probability and r represents the reward function.

5.1.1 State Space

The state space represents all possible states that the drone can be in. The state space, denoted as S, consists of all possible combinations of power capacity, storage availability, and location values that the drone can occupy at the time t.

5.1.1.1 Power Capacity

The drone k has a restricted rechargeable battery to store energy. Positive integers are used to represent stored energy where $Emax$ is an acronym for the drone's battery's maximum capacity, and the drone battery level at the start of the time slot is shown by the variable $E_k(t)$ where

$$E_k\left(t\right) = \{0, E1, ..., E_{max}\} \tag{18}$$

5.1.1.2 Storage Availability

We divide the storage availability into discrete levels, such as low, medium, and high. Positive integers are used to represent storage availability where Mmax is an acronym for the drone's memory card maximum availability, and the drone memory space at the t time slot is shown by the variable$M_k(t)$ where

$$M_k\left(t\right) = \{0, M1, ..., M_{max}\} \tag{19}$$

5.1.1.3 Location

Represent the location for drone k at time slot t as discrete regions or grid cells identified by

$$\theta_k\left(t\right) = \{x_k, y_k, z_k\} \tag{20}$$

Figure 40 shows that The mission area is divided into $n+1$ identical subareas, and the MEC servers are located in the area x_0. Where

$$X \in \{x_0, x_1, x_2, ...x_n\} \tag{21}$$

From the geographical coordinates x,y, and z, we can specify which subarea the drone is located. The state at any given time can be a combination of the devices' power capacity, storage availability, and location.

$$S_k(t) = \{\theta_k(t), E_k(t), M_k(t)\} \tag{22}$$

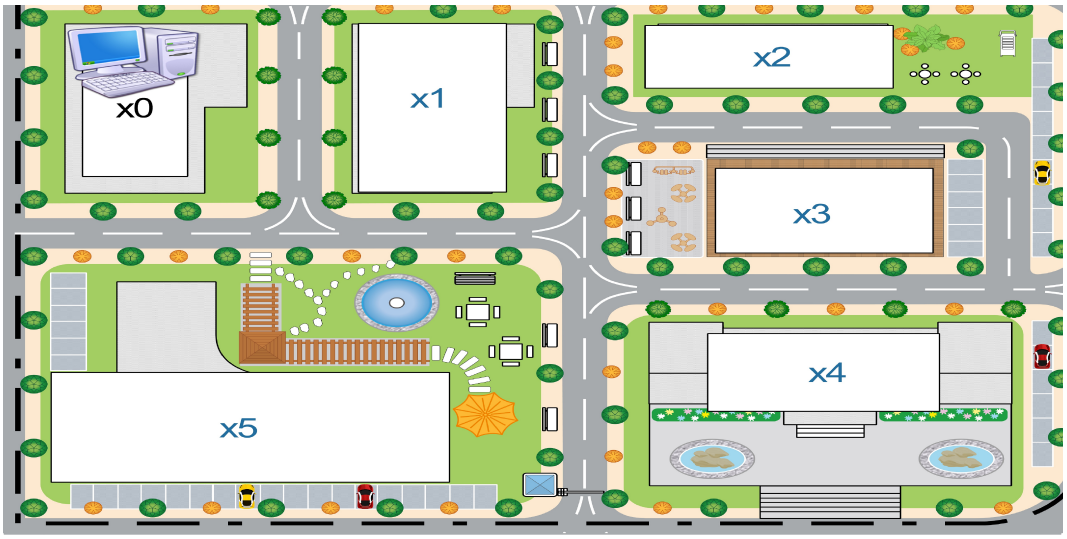


Figure 39: Subareas.

5.1.2 Action Space

At t time slot, each MEC server selects the action according to the observed state $S_k(t)$. $A_t = \{a_t, a_{t+1}, ...\}$ is a sequence of actions from t to ∞.

Drone-client Selection: Choose a drone from the available options based on their characteristics for FL participation.

Task Allocation: Allocate computational tasks to the selected drone. The action space consists of selecting a drone and deciding how to allocate tasks for the FL process.

5.1.3 State Transition Probability

Given the selected drone and assigned tasks, the transition probabilities indicate the likelihood of transitioning to different states based on the drone's power capacity, storage availability, and location. Drone going from state s_t to state s_{t+1} during one transition step is denoted by $p\left(S_{t+1}|S_t, a_t\right)$. In our scenario, we assume that the drone moves at a constant speed. As shown in Figure 41, the drone has three options at each time slot: traveling from one subarea to another, moving within the same subarea, or exiting the entire mission region.

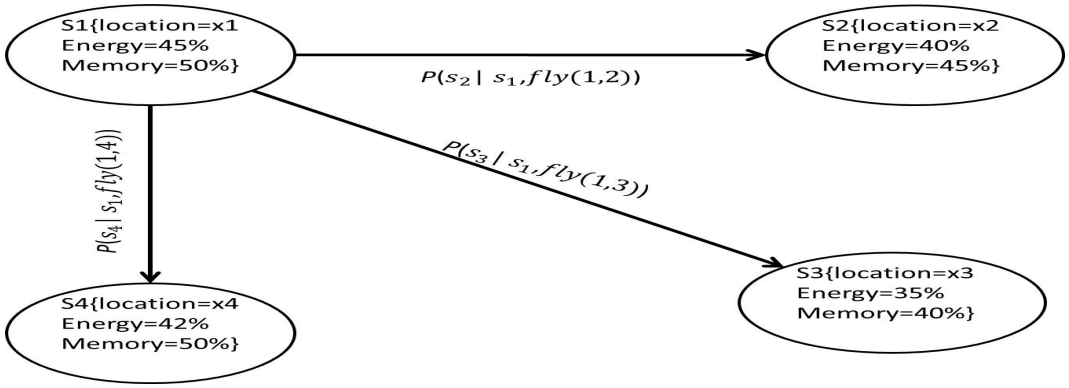


Figure 40: States of drone.

5.1.4 Reward Function

In our model, the reward function captures the desired properties of the selected drones. The received reward, $R_t = \{S_t, a_t, s_{t+1}\}$ after transitioning from state s_t to state s_{t+1} due to action a_t. The reward function can assign positive rewards for high power capacity, storage availability, and proximity to the agent's location. For example:

The positive reward for high power capacity and storage availability.

Negative reward for low power capacity and storage availability.

Negative reward for devices located far away from the agent's current location.

The rewards can be scaled or weighted to reflect the relative importance of each property.

By integrating the defined state space, action space, transition probabilities, and reward function, we can construct an MDP. The agent can then utilize RL algorithms to learn an optimal policy π^* that maximizes the expected cumulative rewards over time.

Figure 42 illustrates how all MEC servers (agents) continuously update their policies

through iterative interactions with the network environment. This allows for the selection of drones based on factors such as power capacity, storage availability, and location.

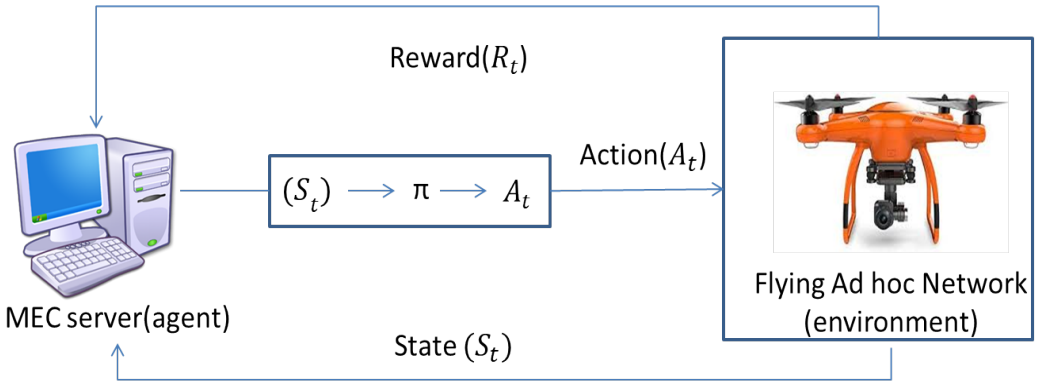


Figure 41: Reinforcement learning-based drone-client selection.

In Figure 42, we employ the Q-learning with epsilon greedy policy to solve MDP.

5.1.5 Q-learning Algorithm for Solving MDP Based on Power Capacity, Storage Availability, and Location

In our model, we employ Q-learning for managing large-scale decision-making as follows:

5.1.5.1 Initialize Q-Table

The MEC server creates a Q-table to store the action values for each state-action pair of drone k at a specific location and time. The dimensions of the Q-table will depend on the discretized state and action spaces.

5.1.5.2 Initialize Hyperparameters

α, γ, ϵ represent the hyperparameters of the Q-learning algorithm used in our scheme where:

Learning Rate (α): Determines the weight given to the new information when updating the Q-values [132].

Discount Factor (γ): Controls the importance of future rewards compared to immediate rewards where $\gamma \in (0, 1)$[126].

Exploration Rate (ϵ): The epsilon greedy controls the trade-off between exploration and exploitation [133] used in model-free RL.

5.1.5.3 Training Loop

In our model, the MEC server (agent) initializes the drone's current state. The agent then enters a loop until convergence or a predetermined number of episodes is reached. Within each episode, the agent selects an action using an epsilon-greedy policy, which involves generating a random number between 0 and 1.
If the generated random number is less than epsilon, a random action is selected for exploration. Otherwise, the action with the highest Q-value for the current state is chosen for exploitation, considering the current state and the Q-table that stores the estimated Q-values for state-action pairs.
After choosing an action, the MEC server performs it on the environment and observes the next state, the reward received, and whether the episode is terminated. This information is crucial for updating the Q-value of the current state-action pair. The Q-value update is performed using the Bellman equation[134] depicted in (23). That combines the immediate reward with the maximum Q-value of the following state, discounted by a factor that accounts for the importance of future rewards.

$$Q\left(S_t, A_t\right) = Q\left(S_t, A_t\right) + \alpha\left(R_{t+1} + \gamma max_a\left(Q\left(S_{t+1}, a\right) - Q\left(S_t, A_t\right)\right)\right) \tag{23}$$

By iteratively performing these steps, the agent learns and refines its Q-values over time, continuously improving its decision-making process. The convergence of the algorithm occurs when the Q-values stabilize, indicating that the agent has learned an optimal policy π^* for the given task.

5.1.5.4 Inference

Given a trained Q-table, actions are chosen based on the highest Q-value for the current state using a greedy policy. The Q-learning algorithm iterates for a given number of iterations. In each iteration, the agent selects an action (drone-client selection) based on the epsilon-greedy policy, simulates the environment to obtain the reward and next state, and updates the Q-table accordingly. The reward function determines the reward based on the properties of other conditions, such as power capacity, storage availability, and location. This Q-learning algorithm enables the agent to learn the optimal device selection policy by updating the Q-values during training. The details of our Q-learning are explained in Algorithm 2.

Algorithm 2 Epsilon greedy policy-based drone-client selection

Data: Learning rate $\alpha \leftarrow 0.1$
Discount factor $\gamma \leftarrow 0.9$
Exploration vs exploitation trade-off $\epsilon \leftarrow 0.1$
Result: A Q-table containing$Q(S_t, A_t$ defining optimal policy π^*
Function$SELECTACTION\,(Q, S_t, \epsilon)$
$n \leftarrow$uniform random number between 0 and 1;
if $n < \epsilon$ then $A \leftarrow$ random action from the action space;
Else $A \leftarrow maxQ\,(S_t, .)$
End
return selected action A_t;
End
initialization: Initialize $Q(s_t, At)$, arbitrarily except that $Q(\text{terminal}, \cdot) \leftarrow 0$
ForEach(episode)
Initialize S;
ForEach(step of episode)
$A_t \leftarrow SELECTACTION\,(Q, S_t, \epsilon)$
Take action A_t, then
observe reward R_t,and next state S_{t+1}
$Q(S, A) \leftarrow Q(S_t, A_t) + \alpha[R_t + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)]$
$S_t \leftarrow S_{t+1}$

5.2 Local Training for Anomaly Detection

Before local training, MEC servers deploy a global IDS with initial parameters. Subsequently, these servers distribute the initial model parameters to drones chosen explicitly within the FANET. Each drone conducts local training leveraging its collected data while upholding stringent data privacy measures. Furthermore, every drone iteratively updates the model parameters based on its locally acquired training data. As described in the previous chapter, they employ sophisticated deep learning (DL) algorithms such as RNN+LSTM, MLP, and CNN. The drones meticulously analyze the collected data to detect potential intrusions or abnormal behaviors. Consequently, the intrusion detection model is refined using locally sourced data without compromising data privacy or security, as no raw data is shared with other drones or centralized servers.

The drone continuously monitors network traffic and analyzes it using the updated global intrusion detection model. When anomalies or potential security threats are detected, the drone may trigger alerts, initiate countermeasures, or notify relevant stakeholders.

5.3 Federated Averaging and Decision-Making

Periodically, drones communicate with nearby MEC servers to share their updated model parameters. Using Federated Averaging to compute a global model update, MEC servers aggregate these parameters.As shown in Figure 43 Federated Averaging typically involves averaging the model parameters from all participating drones with appropriate weighting to account for differences in data distribution and model convergence.

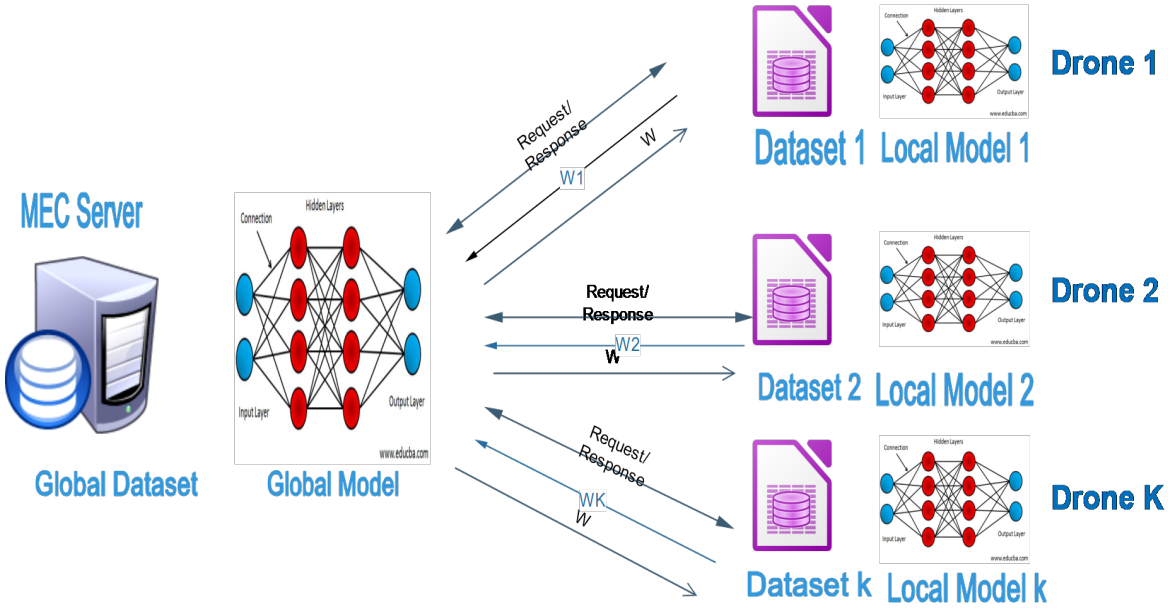


Figure 42: Federated averaging process.

5.3.1 Global Model Update

The MEC server updates the global intrusion detection model using the aggregated parameters obtained through Federated Averaging. The updated global model reflects the collective knowledge learned from all participating drones.

5.3.2 Decision-Making

After training and optimizing the IDS, the drone continuously analyzes network traffic in real time using the model. When suspicious activities or potential intrusions are detected, the drone can take immediate action, such as alerting nearby drones, adjusting its flight path for closer inspection, or transmitting relevant data to ground control stations. Table 9 summarizes the notations used in this proposition.

Table 8: Notation and its meaning

Notation	Meaning
S_t	State space
A_t	Action space
$p\left(S_{t+1}\middle\vert S_t, a_t\right).$	State transition probability
R_t	Reward function
K	Number of drones
$E_k\left(t\right)$	Drone energy at time t
E_{max}	Maximum Energy
$M_k\left(t\right)$	Memory space at the t time slot
M_{max}	Maximum storage capacity
$\theta_k\left(t\right)$	Location for drone k at time slot t
α	Learning Rate for Q-learning
γ	Discount Factor
ϵ	Exploration Rate
P	Global dataset
P_k	Dataset of drone k
b	Subset of the batches
w	Local weight
η	Learning rate for DL
$\triangle\left(w; b\right)$	The change in the weights from the current weight w
m	Number of rounds in federated training
w_{m+1}	Global weight after m rounds
$f_i(m)$	Represents a loss function

6 Experimental Setup

The state representation comprises three essential components: drones' location, battery levels, and storage capacities. This problem is approached as an RL challenge, where a MEC server acts as the agent, aiming to learn a policy for optimal drone selection to maximize cumulative rewards. We trained the agent for the drone selection task by utilizing the Q-learning algorithm with an epsilon-greedy policy. The Q-values were updated according to the observed rewards using the Q-learning update rule. The environment was crafted to simulate drone operations within a 1000-unit area. Each drone's state representation included its current location within the region, battery level (ranging from 0 to 100), and storage capacity. The action space entailed selecting one of the 20 drones within the FANET.

6.1 Training Procedure

To facilitate the training of the RL agent, we conducted a total of 20,000 episodes. This figure was determined based on initial experiments, demonstrating convergence patterns within this range. We implemented an epsilon-greedy policy to balance exploration and exploitation throughout training. The exploration rate, epsilon, diminished over episodes, commencing from an initial value of 1.0. Our decay strategy was tailored to diminish quest as the agent accrued experience. We monitored the total negative rewards for each episode to gauge learning progression. These negative rewards denoted penalties or costs linked to drone selection action.

7 Experimental Results and Comparison

7.1 Comparison Between Related Works and Our Proposed Model

The table compares several IDS proposed for FANETs across several criteria: Effectiveness, Robustness against different attack types, Privacy, and Context Awareness.

Effectiveness: It seems that all models listed are effective to some degree; this suggests that they provide reasonable performance in accurately identifying and classifying network intrusions.

Robustness (Attack types): The robustness of the models varies depending on the attack types considered. Some models are effective against specific attacks like DoS/DDoS, while others are designed to handle a broader range of intrusion attacks. For instance, models utilizing ML and DL techniques are remarkably robust against intrusion attacks, including DoS, DDoS, and others.

Privacy: Privacy concerns are addressed differently across models. Some models explicitly consider privacy as a criterion, while others do not. For example, Federated RL and FL models focus on privacy preservation, whereas others may not prioritize this aspect.

Context Awareness: Context awareness, which refers to the ability of the IDS to adapt its detection capabilities based on the context of the network environment, is not explicitly mentioned for most models. However, our proposed model uses a Q-learning algorithm, which offers a promising solution for drone-client selection due to its dynamic model adjustment through interaction with the environment and support for context

awareness.

Based on the table, the proposed model shows competitiveness across all evaluated criteria. In summary, through fair Comparison against related work, this evaluation demonstrates the effectiveness and practical benefits of the proposed model over existing solutions for the FANET security problem.

Table 9: Comparison between related works and our proposed model

Techniques	**Effectiveness**				**Robustness (Attack types)**	**Privacy**	**Context Aware-**
	Acc	Pre	Rec	Fls			
ML (LR) [44]	x	x	x	x	DoS/DDoS	x	x
ML (RF,DT,SVM) [136]	✓	x	x	x	DoS attacks	x	x
ML(LR,DT,KNN)[137]	✓	✓	✓	✓	DoS, DDoS	x	x
DL (RNN) [68]	✓	✓	✓	✓	Intrusion attacks	x	x
DL(LSTM) [138]	✓	✓	✓	✓	Intrusion attacks	x	x
DL (DBN) [69]	✓	x	x	✓	Intrusion attacks	x	x
Deep RL [139]	✓	✓	✓	✓	Intrusion attacks	x	✓
Deep RL [140]	x	x	x	x	Intrusion attacks	x	✓
Blockchain-DL [141]	✓	✓	✓	✓	DDoS, Port scan	✓	x
Federated RL [142]	✓	x	x	x	Jamming attack	✓	✓
FL [72]	x	x	x	x	Privacy Leakage	✓	x
FL (CNN+LSTM) [135]	✓	✓	✓	✓	DDoS, spoofing	✓	x
Our proposed model RL+FL	✓	✓	✓	✓	Flood-, Blackhole Selective forward	✓	✓

8 Conclusion

our study underscores the importance of selecting appropriate features and algorithms when applying ML for IDS in FANETs. While FL emerges as a promising solution for IDS within FANETs, the inherent communication challenges and resource constraints drones face require careful consideration in selecting the most suitable drone for FL operations.

In this chapter, we have endeavored to enhance our previous model (FLID) by incorporating a sophisticated drone-client selection technique based on Q-learning algorithms. Through rigorous experimentation, we have demonstrated the efficacy of Q-learning in solving the drone-client selection problem.

Moreover, our comparative analysis with three other models has consistently revealed that our proposed model outperforms competitors across all evaluated metrics.

Furthermore, our model exhibits commendable performance in ensuring network security, with notable strengths in privacy protection, effectiveness, robustness, and context awareness.

Future research endeavors may explore avenues for further refining and extending the proposed model, addressing more attacks, identifying limitations, and delving into emerging trends such as adaptive learning strategies.

General Conclusion

Integrating FANET technology with intelligent sensors and devices has ushered in a wave of innovative applications. However, this amalgamation has also exposed FANETs to many security threats, compromising the network's reliability and privacy. This underscores the urgent need to fortify FANETs against internal and external threats. The fluid nature of drone participation presents a prime opportunity for malicious actors to compromise drones, assuming false identities and perpetrating insider attacks. These insiders exploit their privileged access to execute illicit activities, including inserting malicious hardware or software into the drone, compromising its functionality, or allowing unauthorized control. On the other hand, data privacy is paramount in FANET. Indeed, drones can gather sensitive information from the environment through various sensors and technologies. Therefore, Curious actors and hackers can extract sensitive information and disrupt network privacy.

Consequently, cryptographic methods, securing communication protocols, and certificate revocation are promising strategies for protecting FANETs from potential attacks. However, the existing solutions have their main limitations which can obstruct their effectiveness in addressing specific challenges.

This thesis has extensively addressed these challenges, particularly in detecting insider attacks. We aim to develop models tailored to drones' unique characteristics, marked by their high velocity, dynamic topology, and energy limitations.

In our first contribution, we adopted a trust management technique leveraging fuzzy logic to devise a novel solution named FUBA, which stands for a Fuzzy logic-based drone behavior analytics system. Unlike previous approaches, FUBA operates robustly even in adverse weather conditions and amidst poor signal strength (RSSI), effectively distinguishing between legitimate and malicious drone behaviors.

In our second contribution, we introduced FLID, an IDS tailored explicitly for FANETs, integrating ML techniques such as DL and FL. FLID empowers drones within the network to conduct localized model training on their data while upholding data privacy and capitalizing on each drone's unique capabilities. Notably, our FLID model demonstrates efficacy in detecting and thwarting various network attack types, including Flooding, Blackhole, and Selective Forwarding attacks.

While FL holds promise for IDS within FANETs, drones' communication challenges and resource limitations require careful consideration when selecting suitable drones for FL operations. Our third contribution advanced our previous work by incorporating a sophisticated drone-client selection technique based on RL algorithms, specifically Q-learning, using an epsilon-greedy policy. This enhancement bolsters network security and data privacy and enhances context awareness within FANETs.

Our contributions hold substantial implications for real-world applications within the realm of FANETs. The deployment of our novel solutions offers tangible benefits beyond the academic sphere. In practical terms, these advancements ensure the uninterrupted operation of critical services across diverse domains such as surveillance, disaster management, precision agriculture, and infrastructure inspection.

However, as we look to the future, integrating blockchain technology into federated learning for attack detection in FANETs is a crucial area for further research. Moreover, developing dedicated datasets for IDS in FANETs is imperative to support advancements in this field. In addition, creating a comprehensive dataset for frequent attacks in FANETs is within the scope of my agenda. In conclusion, this thesis has made significant strides in enhancing the security posture of FANETs, tackling the challenges posed by insider attacks through innovative solutions such as FUBA and FLID. As we chart the course for future research, integrating blockchain technology and developing dedicated datasets are pivotal to fortifying the resilience of FANETs against evolving threats. With these advancements, we are poised to not only safeguard the security and privacy of FANETs but also to unleash their full potential in revolutionizing various domains.

References

[1] Ilker Bekmezci, Ozgur Koray Sahingoz, and Şamil Temel. Flying ad-hoc networks (fanets): A survey. *Ad Hoc Networks*, 11(3):1254–1270, 2013.

[2] Haythem Bany Salameh, Rakan Al-Maaitah, Haitham Al-Obiedollah, and Ahmad Al-Ajlouni. Energy-efficient power-controlled resource allocation for mimo-based cognitive-enabled b5g/6g indoor-flying networks. *IEEE Access*, 10:106828–106840, 2022.

[3] Akhtar Saeed, Mikail Erdem, Ammar Saleem, Ozgur Gurbuz, and Mustafa Alper Akkas. Joint resource allocation for terahertz band drone communications. *IEEE Transactions on Vehicular Technology*, 2024.

[4] Pengfei Zhu, Jiayu Zheng, Dawei Du, Longyin Wen, Yiming Sun, and Qinghua Hu. Multi-drone-based single object tracking with agent sharing network. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(10):4058–4070, 2020.

[5] Muhammad Asghar Khan, Alamgir Safi, Ijaz Mansoor Qureshi, and Inam Ullah Khan. Flying ad-hoc networks (fanets): A review of communication architectures, and routing protocols. In *2017 First international conference on latest trends in electrical engineering and computing technologies (INTELLECT)*, pages 1–9. IEEE, 2017.

[6] Muhammad Asghar Khan, Ijaz Mansoor Qureshi, and Fahimullah Khanzada. A hybrid communication scheme for efficient and low-cost deployment of future flying ad-hoc network (fanet). *Drones*, 3(1):16, 2019.

[7] Sihem Benfriha and Nabila Labraoui. Insiders detection in the uncertain iod using fuzzy logic. In *2022 International Arab Conference on Information Technology (ACIT)*, pages 1–6. IEEE, 2022.

[8] Giuseppe Faraci, Santi Agatino Rizzo, and Giovanni Schembra. Green edge intelligence for smart management of a fanet in disaster-recovery scenarios. *IEEE Transactions on Vehicular Technology*, 72(3):3819–3831, 2023.

[9] Naser Hossein Motlagh, Miloud Bagaa, and Tarik Taleb. Uav-based iot platform: A crowd surveillance use case. *IEEE Communications Magazine*, 55(2):128–134, 2017.

[10] Adi Prabowo Budiyanta, Mardikasari Wirawan, Wardana Pranoto, Kusumoaji, et al.

Drone-assisted climate smart agriculture (dacsa): The design of the groundwork flow data for drone operations. *EAI Endorsed Transactions on Scalable Information Systems*, 11, 2024.

[11] Hanif Ullah, Nithya Gopalakrishnan Nair, Adrian Moore, Chris Nugent, Paul Muschamp, and Maria Cuevas. 5g communication: An overview of vehicle-to-everything, drones, and healthcare use-cases. *IEEE Access*, 7:37251–37268, 2019.

[12] Rui Du, Haocheng Hua, Hailiang Xie, Xianxin Song, Zhonghao Lyu, Mengshi Hu, Yan Xin, Stephen McCann, Michael Montemurro, Tony Xiao Han, et al. An overview on ieee 802.11 bf: Wlan sensing. *IEEE Communications Surveys & Tutorials*, 2024.

[13] Sukhrob Atoev, Ki-Ryong Kwon, Suk-Hwan Lee, and Kwang-Seok Moon. Data analysis of the mavlink communication protocol. In *2017 International Conference on Information Science and Communications Technologies (ICISCT)*, pages 1–3. IEEE, 2017.

[14] Moazzam Ali, Adil Idress, and Jawwad Ibrahim. Fanet:communication architecture and routing protocols a review. *International Journal of Computer Science & Network Security*, 24 (5):181–190, 2024.

[15] Omar Sami Oubbati, Abderrahmane Lakas, Fen Zhou, Mesut Güneş, and Mohamed Bachir Yagoubi. A survey on position-based routing protocols for flying ad hoc networks (fanets). *Vehicular Communications*, 10:29–56, 2017.

[16] Omer T Abdulhae, Jit Singh Mandeep, and Mt Islam. Cluster-based routing protocols for flying ad hoc networks (fanets). *IEEE Access*, 10:32981–33004, 2022.

[17] Se-Ra Oh and Young-Gab Kim. Security requirements analysis for the iot. In *2017 International Conference on Platform Technology and Service (PlatCon)*, pages 1–6. IEEE, 2017.

[18] Shantanu Pal, Michael Hitchens, Tahiry Rabehaja, and Subhas Mukhopadhyay. Security requirements for the internet of things: A systematic approach. *Sensors*, 20(20):5897, 2020.

[19] PanJun Sun. Security and privacy protection in cloud computing: Discussions and challenges. *Journal of Network and Computer Applications*, 160:102642, 2020.

[20] Messaoud Babaghayou. *Safety-oriented Identity And Location Preservation In Internet Of Vehicles.* PhD thesis, Université Abou Bekr Belkaid - Tlemcen-Algeria, 2021.

[21] Nabila Labraoui, Makhlouf Aliouat, and Jonathan Petit. Rahim: Robust adaptive approach based on hierarchical monitoring providing trust aggregation for wireless... *Journal of universal computer science*, 17(11):1550–1571, 2011.

[22] Sihem Benfriha, Nabila Labraoui, Haythem Bany Salameh, and Hafida Saidi. A survey on trust management in flying ad hoc networks: Challenges, classifications, and analysis. In *2023 Tenth International Conference on Software Defined Systems (SDS)*, pages 107–114. IEEE, 2023.

[23] Mauro Tropea, Mattia Giovanni Spina, Abderrahmane Lakas, Panagiotis Sarigiannidis, and

Floriano De Rango. Secgpsr: A secure gpsr protocol for fanet against sybil and gray hole attacks. *IEEE Access*, 2024.

[24] Isha Pali, Ruhul Amin, and Mohammad Abdussami. Autonomous vehicle security: Current survey and future research challenges. *Security and Privacy*, 7(3):e367, 2024.

[25] Safia Lateef, Muhammad Rizwan, and Muhammad Abul Hassan. Security threats in flying ad hoc network (fanet). *Computational Intelligence for Unmanned Aerial Vehicles Communication Networks*, pages 73–96, 2022.

[26] Shanyao Ren, Dongyu Li, Qinglei Hu, Yizhong Liu, and Jianwei Liu. An improved security olsr protocol against black hole attack based on fanet. In *2022 13th Asian Control Conference (ASCC)*, pages 383–388. IEEE, 2022.

[27] Mauro Tropea, Mattia Giovanni Spina, Abderrahmane Lakas, Panagiotis Sarigiannidis, and Floriano De Rango. Secgpsr: A secure gpsr protocol for fanet against sybil and gray hole attacks. 2023.

[28] Ozlem Ceviz, Pinar Sadioglu, and Sevil Sen. Analysis of routing attacks in fanets. In *International Conference on Ad Hoc Networks*, pages 3–17. Springer, 2021.

[29] Donpiti Chulerttiyawong and Abbas Jamalipour. Sybil attack detection in internet of flying things-ioft: A machine learning approach. *IEEE Internet of Things Journal*, 2023.

[30] Kuldeep Singh and Anil Kumar Verma. Tbcs: A trust-based clustering scheme for secure communication in flying ad-hoc networks. *Wireless Personal Communications*, 114:3173–3196, 2020.

[31] Mehdi Hosseinzadeh, Saqib Ali, Husham Jawad Ahmad, Faisal Alanazi, Mohammad Sadegh Yousefpoor, Efat Yousefpoor, Omed Hassan Ahmed, Amir Masoud Rahmani, and Sang-Woong Lee. A novel q-learning-based secure routing scheme with a robust defensive system against wormhole attacks in flying ad hoc networks. volume 49, page 100826. Elsevier, 2024.

[32] Zhao Beiying, Ji Weifeng, Weng Jiang, Sun Yan, Li Yingqi, and Wu Xuan. Trusted routing protocol for flying ad hoc networks. *Journal of Frontiers of Computer Science & Technology*, 15(12):2304, 2021.

[33] Cong Pu, Imtiaz Ahmed, Evan Allen, and Kim-Kwang Raymond Choo. A stochastic packet forwarding algorithm in flying ad hoc networks: Design, analysis, and evaluation. *IEEE Access*, 9:162614–162632, 2021.

[34] Claudia Greco, Pasquale Pace, Stefano Basagni, and Giancarlo Fortino. Jamming detection at the edge of drone networks using multi-layer perceptrons and decision trees. *Applied Soft Computing*, 111:107806, 2021.

[35] Mousaab Bada, Djallel Eddine Boubiche, Nasreddine Lagraa, Chaker Abdelaziz Kerrache, Muhammad Imran, and Muhammad Shoaib. A policy-based solution for the detection of

colluding gps-spoofing attacks in fanets. *Transportation Research Part A: Policy and Practice*, 149:300–318, 2021.

[36] Nishat Mowla, Nguyen H= Tran, Inshil Doh, and Kijoon Chae. Afrl: Adaptive federated reinforcement learning for intelligent jamming defense in fanet. *Journal of Communications and Networks*, 22(3):244–258, 2020.

[37] Vedadatta Gouripeddi. Improvement of security in uas communication and navigation using ads-b. 2016.

[38] Ozlem Ceviz, Pinar Sadioglu, and Sevil Sen. A survey of security in uavs and fanets: Issues, threats, analysis of attacks, and solutions. *arXiv preprint arXiv:2306.14281*, 2023.

[39] Khaista Rahman, Muhammad Adnan Aziz, Ahsan Ullah Kashif, and Tanweer Ahmad Cheema. Detection of security attacks using intrusion detection system for uav networks: A survey. In *Big Data Analytics and Computational Intelligence for Cybersecurity*, pages 109–123. Springer, 2022.

[40] Omer Abdulhae, Jit Singh Mandeep, and Mt Islam. Cluster-based routing protocols for flying ad hoc networks (fanets). *IEEE Access*, 10:32981–33004, 2022.

[41] Saeed Ullah Jan, Irshad Ahmed Abbasi, Fahad Algarni, and Adnan Shahid Khan. A verifiably secure ecc based authentication scheme for securing iod using fanet. *IEEE Access*, 10: 95321–95343, 2022.

[42] Sergio Arnosti, Rayner Pires, and Kalinka RLJC Branco. Evaluation of cryptography applied to broadcast storm mitigation algorithms in fanets. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1368–1377. IEEE, 2017.

[43] Girraj Kumar Verma, BB Singh, Neeraj Kumar, and Debiao He. Cb-ps: An efficient short-certificate-based proxy signature scheme for uavs. *IEEE Systems Journal*, 14(1):621–632, 2019.

[44] Khaista Rahman, Muhammad Adnan Aziz, Nighat Usman, Tayybah Kiren, Tanweer Ahmad Cheema, Hina Shoukat, Tarandeep Kaur Bhatia, Asrin Abdollahi, Ahthasham Sajid, et al. Cognitive lightweight logistic regression-based ids for iot-enabled fanet to detect cyberattacks. *Mobile Information Systems*, 2023.

[45] Sadoon Hussain, Sami Ahmed, Abida Thasin, and Redhwan MA Saad. Ai-enabled ant-routing protocol to secure communication in flying networks. *Applied Computational Intelligence and Soft Computing*, 2022.

[46] Stephan Frisbie and Mohamed Younis. Ai-enabled jammer deception using decoy packets. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pages 5013–5018. IEEE, 2022.

[47] Rajasoundaran Soundararajan, Maheswar Rajagopal, Akila Muthuramalingam, Eklas Hossain,

and Jaime Lloret. Interleaved honeypot-framing model with secure mac policies for wireless sensor networks. *Sensors*, 22(20):8046, 2022.

[48] Nilin Prabahker, Ghanshyam S Bopche, and Michael Arock. Generation and deployment of honeytokens in relational databases for cyber deception. page 104032. Elsevier, 2024.

[49] Nabila Labraoui, Mourad Gueroui, and Larbi Sekhri. A risk-aware reputation-based trust management in wireless sensor networks. *Wireless Personal Communications*, 87:1037–1055, 2016.

[50] Muruganandam, Arokia Renjit, and Sendhil Kumar. A survey: Comparative study of security methods and trust management solutions in manet. In *2019 Fifth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)*, volume 1, pages 125–131. IEEE, 2019.

[51] Faizan Ullah, Abdu Salam, Farhan Amin, Izaz Ahmad Khan, Jamal Ahmed, Shamzash Alam Zaib, and Gyu Sang Choi. Deep trust: A novel framework for dynamic trust and reputation management in the internet of things (iot) based networks. *IEEE Access*, 2024.

[52] Amal Hbaieb, Samiha Ayed, and Lamia Chaari. A survey of trust management in the internet of vehicles. *Computer Networks*, 203:108558, 2022.

[53] Sihem Benfriha, Nabila Labraoui, Radjaa Bensaid, Haythem Bany Salameh, and Hafida Saidi. Fuba: A fuzzy-based unmanned aerial vehicle behaviour analytics for trust management in flying ad-hoc networks. *IET Networks*, 2023.

[54] Zhengru Fang, Jingjing Wang, Yong Ren, Zhu Han, H Vincent Poor, and Lajos Hanzo. Age of information in energy harvesting aided massive multiple access networks. *IEEE Journal on Selected Areas in Communications*, 40(5):1441–1456, 2022.

[55] Kuldeep Singh and Anil Kumar Verma. A fuzzy-based trust model for flying ad hoc networks (fanets). *International Journal of Communication Systems*, 31(6):e3517, 2018.

[56] Kuldeep Singh and Anil Kumar Verma. Fctm: A novel fuzzy classification trust model for enhancing reliability in flying ad hoc networks (fanets). *Ad Hoc Sens. Wirel. Networks*, 40 (1-2):23–47, 2018.

[57] Joydeep Kundu, Sahabul Alam, and Chandan Koner. Tcsfanet: Trusted communication scheme for fanet system. In *2022 International Conference on Machine Learning, Computer Systems and Security (MLCSS)*, pages 353–357. IEEE, 2022.

[58] Ezedin Barka, Chaker Abdelaziz Kerrache, Hadjer Benkraouda, Khaled Shuaib, Farhan Ahmad, and Fatih Kurugollu. Towards a trusted unmanned aerial system using blockchain for the protection of critical infrastructure. *Transactions on Emerging Telecommunications Technologies*, 33(8):e3706, 2022.

[59] Tao Zhang and Quanyan Zhu. Strategic defense against deceptive civilian gps spoofing of unmanned aerial vehicles. In *Decision and Game Theory for Security: 8th International*

Conference, GameSec 2017, Vienna, Austria, October 23-25, 2017, Proceedings, pages 213–233. Springer, 2017.

[60] Chaker Abdelaziz Kerrache, Ezedin Barka, Nasreddine Lagraa, and Abderrahmane Lakas. Reputation-aware energy-efficient solution for fanet monitoring. In *2017 10th IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 1–6. IEEE, 2017.

[61] Ezedin Barka, Chaker Abdelaziz Kerrache, Nasreddine Lagraa, Abderrahmane Lakas, Carlos T Calafate, and Juan-Carlos Cano. Union: A trust model distinguishing intentional and unintentional misbehavior in inter-uav communication. *Journal of Advanced Transportation*, 2018.

[62] Ezedin Barka, Chaker Abdelaziz Kerrache, Rasheed Hussain, Nasreddine Lagraa, Abderrahmane Lakas, and Safdar Hussain Bouk. A trusted lightweight communication strategy for flying named data networking. *Sensors*, 18(8):2683, 2018.

[63] Arpita Bhargava and Shekhar Verma. Kate: Kalman trust estimator for internet of drones. *Computer Communications*, 160:388–401, 2020.

[64] Carlos Felipe Emygdio de Melo, Tulio Dapper e Silva, Felipe Boeira, Jorgito Matiuzzi Stocchero, Alexey Vinel, Mikael Asplund, and Edison Pignaton de Freitas. Uavouch: a secure identity and location validation scheme for uav-networks. *IEEE Access*, 9:82930–82946, 2021.

[65] Shanshan Sun, Zuchao Ma, Liang Liu, Hang Gao, and Jianfei Peng. Detection of malicious nodes in drone ad-hoc network based on supervised learning and clustering algorithms. In *2020 16th International Conference on Mobility, Sensing and Networking (MSN)*, pages 145–152. IEEE, 2020.

[66] Amjad Saeed Khan, Gaojie Chen, Yogachandran Rahulamathavan, Gan Zheng, Basil Assadhan, and Sangarapillai Lambotharan. Trusted uav network coverage using blockchain, machine learning, and auction mechanisms. *IEEE Access*, 8:118219–118234, 2020.

[67] Shrikant Tangade, R Arun Kumaar, S Malavika, S Monisha, and Farooque Azam. Detection of malicious nodes in flying ad-hoc network with supervised machine learning. In *2022 Third International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE)*, pages 1–5. IEEE, 2022.

[68] Rabie A Ramadan, Abdel-Hamid Emara, Mohammed Al-Sarem, and Mohamed Elhamahmy. Internet of drones intrusion detection using deep learning. *Electronics*, 10(21):2633, 2021.

[69] José Escorcia-Gutierrez, Margarita Gamarra, Esmeide Leal, Natasha Madera, Carlos Soto, Romany F Mansour, Meshal Alharbi, Ahmed Alkhayyat, and Deepak Gupta. Sea turtle foraging algorithm with hybrid deep learning-based intrusion detection for the internet of drones environment. *Computers and Electrical Engineering*, 108:108704, 2023.

[70] Sneha Kanchan and Bong Jun Choi. An efficient and privacy-preserving federated learning

scheme for flying ad hoc networks. In *ICC 2022-IEEE International Conference on Communications*, pages 1–6. IEEE, 2022.

[71] Abbas Yazdinejad, Reza M Parizi, Ali Dehghantanha, and Hadis Karimipour. Federated learning for drone authentication. *Ad Hoc Networks*, 120:102574, 2021.

[72] Yuntao Wang, Zhou Su, Ning Zhang, and Abderrahim Benslimane. Learning in the air: Secure federated learning for uav-assisted crowdsensing. *IEEE Transactions on network science and engineering*, 8(2):1055–1069, 2020.

[73] Amit Kumar Gautam and Rakesh Kumar. A comprehensive study on key management, authentication and trust management techniques in wireless sensor networks. *SN Applied Sciences*, 3(1):50, 2021.

[74] Seyed Ahmad Soleymani, Abdul Hanan Abdullah, Wan Haslina Hassan, Mohammad Hossein Anisi, Shidrokh Goudarzi, Mir Ali Rezazadeh Baee, and Satria Mandala. Trust management in vehicular ad hoc network: a systematic review. *EURASIP Journal on Wireless Communications and Networking*, pages 1–22, 2015.

[75] Rajanpreet Kaur Chahal, Neeraj Kumar, and Shalini Batra. Trust management in social internet of things: A taxonomy, open issues, and challenges. *Computer Communications*, 150: 13–46, 2020.

[76] Vishal Sharma and Rajesh Kumar. G-fanet: an ambient network formation between ground and flying ad hoc networks. *Telecommunication Systems*, 65(1):31–54, 2017.

[77] Sankardas Roy, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Vivek Shandilya, and Qishi Wu. A survey of game theory as applied to network security. In *2010 43rd Hawaii International Conference on System Sciences*, pages 1–10. IEEE, 2010.

[78] Qiang Liu, Ming He, Daqin Xu, Ning Ding, and Yong Wang. A mechanism for recognizing and suppressing the emergent behavior of uav swarm. *Mathematical Problems in Engineering*, 2018.

[79] Hao Wang and Dit-Yan Yeung. A survey on bayesian deep learning. *ACM computing surveys (csur)*, 53(5):1–37, 2020.

[80] Georgios Drainakis, Konstantinos V Katsaros, Panagiotis Pantazopoulos, Vasilis Sourlas, and Angelos Amditis. Federated vs. centralized machine learning under privacy-elastic users: A comparative analysis. In *2020 IEEE 19th International Symposium on Network Computing and Applications (NCA)*, pages 1–8. IEEE, 2020.

[81] Ismaeel Al Ridhawi, Ouns Bouachir, Moayad Aloqaily, and Azzedine Boukerche. Design guidelines for cooperative uav-supported services and applications. *ACM Computing Surveys (CSUR)*, 54(9):1–35, 2021.

[82] Moez Krichen. Convolutional neural networks: A survey. *Computers*, 12(8):151, 2023.

[83] Haythem Bany Salameh, Rami Mohawesh, Sumbal Maqsood, Yaser Jararweh, and Nuha

Alshuqayran. Generative adversarial network (gan) in social network: Introduction, applications, challenges and future directions. In *2023 Tenth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 1–7, 2023. doi: 10.1109/SNAMS60348.2023.10375461.

[84] Rui Han, Lin Bai, Jianwei Liu, and Peng Chen. Blockchain-based gnss spoofing detection for multiple uav systems. *Journal of Communications and Information Networks*, 4(2):81–88, 2019.

[85] Yawen Tan, Jiajia Liu, and Nei Kato. Blockchain-based key management for heterogeneous flying ad hoc network. *IEEE Transactions on Industrial Informatics*, 17(11):7629–7638, 2020.

[86] Angela Carrera-Rivera, Felix Larrinaga, and Ganix Lasa. Context-awareness for the design of smart-product service systems: Literature review. *Computers in Industry*, 142:103730, 2022.

[87] Joseph Migga Kizza, Wheeler Kizza, and Wheeler. *Guide to computer network security*, volume 8. Springer, 2013.

[88] Andrew Lockhart. *Network Security Hacks: Tips & Tools for Protecting Your Privacy.* " O'Reilly Media, Inc.", 2006.

[89] Bo Liu, Ming Ding, Sina Shaham, Wenny Rahayu, Farhad Farokhi, and Zihuai Lin. When machine learning meets privacy: A survey and outlook. *ACM Computing Surveys (CSUR)*, 54 (2):1–36, 2021.

[90] Fangfang Yuan, Yanan Cao, Yanmin Shang, Yanbing Liu, Jianlong Tan, and Binxing Fang. Insider threat detection with deep neural network. In *Computational Science–ICCS 2018: 18th International Conference, Wuxi, China, June 11–13, 2018, Proceedings, Part I 18*, pages 43–54. Springer, 2018.

[91] Kuldeep Singh and Anil Kumar Verma. A trust model for effective cooperation in flying ad hoc networks using genetic algorithm. In *2018 International Conference on Communication and Signal Processing (ICCSP)*, pages 0491–0495. IEEE, 2018.

[92] Yingying Chen and Jie Yang. Defending against identity-based attacks in wireless networks. In *Handbook on Securing Cyber-Physical Critical Infrastructure*, pages 191–222. Elsevier Inc., 2012.

[93] Yun Ji, Xiaoxiong Zhong, Zhoubin Kou, Sheng Zhang, Hangfan Li, and Yuanyuan Yang. Efficiency-boosting federated learning in wireless networks: A long-term perspective. *IEEE Transactions on Vehicular Technology*, 2023.

[94] Thalita Ayass, Thiago Coqueiro, Tássio Carvalho, José Jailton, Jasmine Araújo, and Renato Francês. Unmanned aerial vehicle with handover management fuzzy system for 5g networks: challenges and perspectives. *Intell Robot*, 2(1):20–36, 2022.

[95] Khaeel Ullah Khan and Ramesh. Effect on packet delivery ratio (pdr) & throughput in

wireless sensor networks due to black hole attack. *Int. J. Innov. Technol. Explore. Eng*, 8 (12S):428–432, 2019.

[96] Jiaxing Wang, Lin Bai, Jianrui Chen, and Jingjing Wang. Starling flocks-inspired resource allocation for isac-aided green ad hoc networks. *IEEE Transactions on Green Communications and Networking*, 7(1):444–454, 2023.

[97] Diana Bri, Miguel Garcia-Pineda, Jaime Lloret, and Francisco Ramos. Performance analysis of weather's impact on outdoor ieee 802.11 b/g links using network management parameters. *Mobile Networks and Applications*, 21:603–619, 2016.

[98] Sinan Kurt and Bulent Tavli. Path-loss modeling for wireless sensor networks: A review of models and comparative evaluations. *IEEE Antennas and Propagation Magazine*, 59(1):18–37, 2017.

[99] Rao and Saraf. Study of defuzzification methods of fuzzy logic controller for speed control of a dc motor. In *Proceedings of the international conference on power electronics, drives, and energy systems for industrial growth*, volume 2, pages 782–787. IEEE, 1996.

[100] Kavitha Thangaraj and Dejey Dharma. Optimized fuzzy system dependent trust score for mobile adhoc network. *Wireless Personal Communications*, 117:3255–3269, 2021.

[101] Jappreet Singh Gill, Mahdi Saeedi Velashani, Jordan Wolf, Jonathan Kenney, Mohsen Riahi Manesh, and Naima Kaabouch. Simulation testbeds and frameworks for uav performance evaluation. In *2021 ieee international conference on electro information technology (eit)*, pages 335–341. IEEE, 2021.

[102] Antonio Virdis and Michael Kirsche. Recent advances in network simulation. *EAI/Springer Innovations in Communication and Computing*, 2019.

[103] Emerson Alberto Marconato, Mariana Rodrigues, Rayner de Melo Pires, Daniel Fernando Pigatto, Alex Roschildt Pinto, Kalinka RLJC Branco, et al. Avens-a novel flying ad hoc network simulator with automatic code generation for unmanned aircraft system. 2017.

[104] Richard Garcia and Laura Barnes. Multi-uav simulator utilizing x-plane. In *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*, pages 393–406. Springer, 2010.

[105] Alberto Pliego Marugán, Ana María Peco Chacón, and Fausto Pedro García Márquez. Reliability analysis of detecting false alarms that employ neural networks: A real case study on wind turbines. *Reliability Engineering & System Safety*, 191:106574, 2019.

[106] Maha Alaslani, Faisal Nawab, and Basem Shihada. Blockchain in iot systems: End-to-end delay evaluation. *IEEE Internet of Things Journal*, 6(5):8332–8344, 2019.

[107] Dinesh Chowdary Attota, Viraaji Mothukuri, Reza M Parizi, and Seyedamin Pouriyeh. An ensemble multi-view federated learning intrusion detection for iot. *IEEE Access*, 9: 117734–117745, 2021.

[108] Zhuo Chen, Na Lv, Pengfei Liu, Yu Fang, Kun Chen, and Wu Pan. Intrusion detection for wireless edge networks based on federated learning. *IEEE Access*, 8:217463–217472, 2020.

[109] Monika Arya, Hanumat Sastry, Bhupesh Kumar Dewangan, Mohammad Khalid Imam Rahmani, Surbhi Bhatia, Abdul Wahab Muzaffar, and Mariyam Aysha Bivi. Intruder detection in vanet data streams using federated learning for smart city environments. *Electronics*, 12(4):894, 2023.

[110] Juan Miguel García-Gómez and Salvador Tortajada. Definition of loss functions for learning from imbalanced data to minimize evaluation metrics. In *Data Mining in Clinical Medicine*, pages 19–37. Springer, 2014.

[111] Marius-Constantin Popescu, Valentina E Balas, Liliana Perescu-Popescu, and Nikos Mastorakis. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems*, 8(7):579–588, 2009.

[112] Kishore NG. Multi-layer perceptron. https://www.linkedin.com/pulse/multi-layer-perceptron-kishore-ng-wvrcc/, Accessed: 2024-02-21.

[113] Jianxin Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5(23):495, 2017.

[114] github. Convolutional neural network. https://github.com/topics/cnn-lstm-models?o=desc&s=updated, Accessed: 2024-02-21.

[115] Yadala Sucharitha, Pundru Chandra Shaker Reddy, and G Suryanarayana. Network intrusion detection of drones using recurrent neural networks. *Drone Technology: Future Trends and Practical Applications*, pages 375–392, 2023.

[116] BotPenguin. Recurrent neural network. https://botpenguin.com/glossary/recurrent-neural-network, Accessed: 2024-02-21.

[117] Sukhrob Atoev, Oh-Heum Kwon, Suk-Hwan Lee, and Ki-Ryong Kwon. An efficient sc-fdm modulation technique for a uav communication link. *Electronics*, 7(12):352, 2018.

[118] Pawani Porambage, Jude Okwuibe, Madhusanka Liyanage, Mika Ylianttila, and Tarik Taleb. Survey on multi-access edge computing for internet of things realization. *IEEE Communications Surveys & Tutorials*, 20(4):2961–2991, 2018.

[119] Anis Koubâa and Basit Qureshi. Dronetrack: Cloud-based real-time object tracking using unmanned aerial vehicles over the internet. *IEEE Access*, 6:13810–13824, 2018.

[120] Saria Allahham, Mohammad Al-Sa'd, Abdulla Al-Ali, Amr Mohamed, Tamer Khattab, and Aiman Erbad. Dronerf dataset: A dataset of drones for rf-based detection, classification and identification. *Data in brief*, 26:104313, 2019.

[121] Ranjit Panigrahi and Samarjeet Borah. A detailed analysis of cicids2017 dataset for designing

intrusion detection systems. *International Journal of Engineering & Technology*, 7(3.24): 479–482, 2018.

[122] Murat Dener, Celil Okur, Samed Al, and Abdullah Orman. Wsn-bfsf: A new dataset for attacks detection in wireless sensor networks. *IEEE Internet of Things Journal*, 2023.

[123] Othmane Friha, Mohamed Amine Ferrag, Lei Shu, Leandros Maglaras, Kim-Kwang Raymond Choo, and Mehdi Nafaa. Felids: Federated learning-based intrusion detection system for agricultural internet of things. *Journal of Parallel and Distributed Computing*, 165:17–31, 2022.

[124] Viraaji Mothukuri, Prachi Khare, Reza M Parizi, Seyedamin Pouriyeh, Ali Dehghantanha, and Gautam Srivastava. Federated-learning-based anomaly detection for iot security attacks. *IEEE Internet of Things Journal*, 9(4):2545–2554, 2021.

[125] Amal Hbaieb, Samiha Ayed, and Lamia Chaari. Federated learning based ids approach for the iov. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*, pages 1–6, 2022.

[126] Daoyi Dong, Chunlin Chen, Hanxiong Li, and Tzyh-Jong Tarn. Quantum reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 38(5): 1207–1220, 2008.

[127] Nicole Bäuerle and Anna Jaśkiewicz. Markov decision processes with risk-sensitive criteria: an overview. *Mathematical Methods of Operations Research*, 99(1):141–178, 2024.

[128] Erwin Daniel Lopez Zapata. Components present in the finite markov decision process and its function in the agent-environment interaction. https://www.researchgate.net/publication/331769570_Towards_using_multi-agents_systems_for_assisting_undergraduate_STEM_students_learning/figures, Accessed: 2024-02-27.

[129] Beakcheol Jang, Myeonghwi Kim, Gaspard Harerimana, and Jong Wook Kim. Q-learning algorithms: A comprehensive classification and applications. *IEEE access*, 7:133653–133667, 2019.

[130] Chris Dann, Yishay Mansour, Mehryar Mohri, Ayush Sekhari, and Karthik Sridharan. Guarantees for epsilon-greedy reinforcement learning with function approximation. In *International conference on machine learning*, pages 4666–4689. PMLR, 2022.

[131] Rana singh. Introduction reinforcement learning, with epsilon-greedy(bandit game)algorithm. https://ranasinghiitkgp.medium.com/introduction-reinforcement-learning-with-epsilon-greedy-bandit-game-algorithm-73c22c1646b3, Accessed: 2024-02-29.

[132] Qinglai Wei, Frank L Lewis, Qiuye Sun, Pengfei Yan, and Ruizhuo Song. Discrete-time deterministic q-learning: A novel convergence analysis. *IEEE transactions on cybernetics*, 47 (5):1224–1237, 2016.

[133] M Gimelfarb, S Sanner, and CG Lee. ϵ-bmc: a bayesian ensemble approach to epsilon-greedy exploration in model-free reinforcement learning. *arXiv preprint arXiv:2007.00869*, 2020.

[134] Yingjie Fei, Zhuoran Yang, Yudong Chen, and Zhaoran Wang. Exponential bellman equation and improved regret bounds for risk-sensitive reinforcement learning. *Advances in Neural Information Processing Systems*, 34:20436–20446, 2021.

[135] Ernest Ntizikira, Wang Lei, Fahad Alblehai, Kiran Saleem, and Muhammad Ali Lodhi. Secure and privacy-preserving intrusion detection and prevention in the internet of unmanned aerial vehicles. *Sensors*, 23(19):8077, 2023.

[136] Said Ouiazzane, Fatimazahra BarramoU, and Malika Addou. Towards a multi-agent based network intrusion detection system for a fleet of drones. *International Journal of Advanced Computer Science and Applications*, 11(10), 2020.

[137] Rakesh Shrestha, Atefeh Omidkar, Sajjad Ahmadi Roudi, Robert Abbas, and Shiho Kim. Machine-learning-enabled intrusion detection system for cellular connected uav networks. *Electronics*, 10(13):1549, 2021.

[138] Abdulrahman M Abdulghani, Mokhles M Abdulghani, Wilbur L Walters, and Khalid H Abed. Improving intrusion detection in uav communication using an lstm-smote classification method. *Journal of Cybersecurity (2579-0072)*, 4(4), 2022.

[139] Omar Bouhamed, Ouns Bouachir, Moayad Aloqaily, and Ismaeel Al Ridhawi. Lightweight ids for uav networks: A periodic deep reinforcement learning-based approach. In *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 1032–1037. IEEE, 2021.

[140] Jing Tao, Ting Han, and Ruidong Li. Deep-reinforcement-learning-based intrusion detection in aerial computing networks. *IEEE Network*, 35(4):66–72, 2021.

[141] Arash Heidari, Nima Jafari Navimipour, and Mehmet Unal. A secure intrusion detection platform using blockchain and radial basis function neural networks for internet of drones. *IEEE Internet of Things Journal*, 2023.

[142] Nishat I Mowla, Nguyen H Tran, Inshil Doh, and Kijoon Chae. Federated learning-based cognitive detection of jamming attack in flying ad-hoc network. *IEEE Access*, 8:4338–4350, 2019.

IET Networks

Special issue
Call for Papers

Be Seen. Be Cited.
Submit your work to a new IET special issue

Connect with researchers and experts in your field and share knowledge.

Be part of the latest research trends, faster.

Read more

The Institution of Engineering and Technology

Received: 1 March 2023 | Revised: 7 September 2023 | Accepted: 31 October 2023

IET Networks

DOI: 10.1049/ntw2.12108

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH

FUBA: A fuzzy-based unmanned aerial vehicle behaviour analytics for trust management in flying ad-hoc networks

Sihem Benfriha[1] | Nabila Labraoui[2] | Radjaa Bensaid[1] | Haythem Bany Salameh[3,4,5] | Hafida Saidi[1]

[1]STIC Laboratory, University of Abou Bekr Belkaid, Chetouane, Tlemcen, Algeria

[2]LRIT Laboratory, University of Abou Bekr Belkaid, Chetouane, Tlemcen, Algeria

[3]Artificial Intelligence Research Center, Al Ain University, Al Ain, UAE

[4]Telecommunication Engineering Department, Yarmouk University, Irbid, Jordan

[5]College of Engineering, Staffordshire University, Stoke-in City, UK

Correspondence

Sihem Benfriha.
Email: benfriha.sihem@univ-tlemcen.dz

Abstract

Flying Ad-Hoc Network (FANET) is a promising ad hoc networking paradigm that can offer new added value services in military and civilian applications. Typically, it incorporates a group of Unmanned Aerial Vehicles (UAVs), known as drones that collaborate and cooperate to accomplish several missions without human intervention. However, UAV communications are prone to various attacks and detecting malicious nodes is essential for efficient FANET operation. Trust management is an effective method that plays a significant role in the prediction and recognition of intrusions in FANETs. Specifically, evaluating node behaviour remains an important issue in this domain. For this purpose, the authors suggest using fuzzy logic, one of the most commonly used methods for trust computation, which classifies nodes based on multiple criteria to handle complex environments. In addition, the Received Signal Strength Indication (RSSI) is an important parameter that can be used in fuzzy logic to evaluate a drone's behaviour. However, in outdoor flying networks, the RSSI can be seriously influenced by the humidity of the air, which can dramatically impact the accuracy of the trust results. FUBA, a fuzzy-based UAV behaviour analytics is presented for trust management in FANETs. By considering humidity as a new parameter, FUBA can identify insider threats and increase the overall network's trustworthiness under bad weather conditions. It is capable of performing well in outdoor flying networks. The simulation results indicate that the proposed model significantly outperforms FNDN and UNION in terms of the average end-to-end delay and the false positive ratio.

KEYWORDS

computer network security, fuzzy logic, mobile ad hoc networks

1 | INTRODUCTION

Our world has changed and is still evolving due to rapidly developing technology in sensors, communications, and networking over the past few decades [1]. Unmanned Aerial Vehicles (UAVs) have been proposed for a multitude of applications in both military and civilian domains, encompassing ad hoc networks, search and rescue missions, electronic operations in hostile zones, ground target identification and tracking, automated forest fire surveillance, wind energy generation [2], and a host of other possibilities. Furthermore, flying ad hoc networks (FANETs), a revolutionary concept, comprise a group of UAVs that cooperate to perform some crucial missions [3]. However, many cyberattacks against UAVs have emerged since 2007 [4], and their impact can be dangerous with divesting effects. Therefore, it is essential to protect FANETs from insider and outsider attacks. In FANETs, drones can leave and rejoin the network anytime, creating an opportunity for attackers to compromise a node and impersonate a legitimate one, leading to insider attacks. Insiders use their trusted access to carry out illicit actions. As a result, they are undetectable by external network security

protocols (intrusion detection, firewalls, and cryptographic methods) [5]. Consequently, ensuring secure and reliable communications in FANETs is critical and continues to be an issue.

In this context, trust management is an effective and attractive technique to prevent unexpected node actions and detect malicious nodes [6]. It can improve the robustness and reliability of standard security techniques by guaranteeing that only trustworthy nodes cooperate in network missions. Nevertheless, trust depends on observation and recommendation, and some models have been proposed for FANET to calculate the trust of the drone, but they may lead to uncertainty [7]. Fuzzy logic is a popular method for representing and manipulating uncertain data, such as node behaviours. Few related works use the RSSI as an important parameter for trust evaluation, and this performs better in indoor networks. However, in outdoor networks, the RSSI can be influenced by humidity and thus impact the trust results. In addition, the drone can be detected as non-cooperative due to unintentional misbehaviour related to poor signal strength (RSSI). The main challenge in this domain is designing an efficient analytical trust model for evaluating and understanding node behaviour in FANET under poor signal (RSSI) and bad weather conditions. Without this model, there will be no effective strategy to distinguish between legitimate and malicious drone activities in FANETs. Although several trust models have recently been proposed, none have yet focused on the impact of bad weather conditions on the trust management process in FANET. The proposed work aims to address this gap using the fuzzy logic method to determine the trust of a drone based on several parameters, such as energy (battery level), weather (humidity), signal strength (RSSI), packet delivery ratio (PDR) and transmission delay (TD). Specifically, this article introduces a novel fuzzy-based UAV behaviour analytics system named FUBA for trust management in FANETs. FUBA utilises fuzzy logic methodology to assess drone trustworthiness by considering various factors such as energy levels, weather conditions, signal strength, packet delivery ratios, and transmission delays. The proposed model offers several advantages: superior performance in outdoor flying networks, effective characterisation of node behaviour, subjective evaluation of node behaviour, and the ability to make confident decisions regarding network information exchange. To comprehensively evaluate the model's performance, we implement and rigorously test the system through extensive simulation experiments conducted using the Omnet++, Xplane, and Avens frameworks. The simulation results demonstrate that our model outperforms existing ones in terms of average end-to-end delay and false positive ratio. Furthermore, we analyse the influence of RSSI and humidity on trust results through Matlab simulations, shedding light on prevailing challenges and open issues in this domain.

The rest of this paper is listed as follows: Section 2 offers an overview of the related works on trust management in FANETs. Section 3 introduces the proposed fuzzy-based UAV behaviour analytics for trust management in FANETs (FUBA). Section 4 detailed the practical aspects and limitations of FUBA. Section 5 provides the detailed implementation of FUBA. Section 6 explains the impact of RSSI and humidity on trust results. Section 7 reports and discusses the experimental results. Finally, Section 8 concludes the paper and provides possible future directions.

2 | RELATED WORK

Since UAV networks appeared, several trust models have been implemented to strengthen the trust management systems in FANET. Most of them were initially proposed for Mobile Ad hoc Networks(MANETs) [8]. The recent research on trust-based solutions is presented below:

Berka et al. [9] proposed a new energy-efficient scheme for FANET that is reputation-aware. Their approach computed the trustworthiness by considering the count of both legal and illegal node interactions to establish trust with low energy, considering the indirect trust values. However, when there is no interaction between the trustor and the trustee, the findings impact the system's accuracy. To differentiate between legitimate and malicious drone activities, the authors in ref. [8] presented a second model referred to as UNION. To eliminate man-in-the-middle threats, Barka et al. [10] proposed a comprehensive communication architecture named FNDN (Flying Named Data Networking). This architecture revolves around the integration of trust mechanisms. When propagating data, their model system uses a trust management strategy to address the network attack concern. FNDN utilises inter-UAV trust to decide whether to verify the authenticity of data for a specific node. In ref. [11], the authors suggested a new trust scheme named BUAS. BUAS is based on a blockchain technology-based inter-UAV trust evaluation method.

The Bayesian inference method was used to calculate the probability of the message's trustworthiness. Singh and Verma described a fuzzy-based trust model in ref. [12] that addresses the trustworthiness of the FANET node. A fuzzy classification has been implemented, and the quality of services and social parameters are considered to calculate trust values. They also proposed a weightage-based method that uses the genetic algorithm [13] to ascertain the trust values by simultaneously optimising the weights assigned to different parameters. In ref. [14], the authors proposed a trust-based clustering scheme using the first model to select a trustworthy cluster head that can add new nodes to the network. Zhou and Wang in ref. [15] proposed a K-means ++ clustering algorithm. This model determines the optimal number of clusters and integrates a trust value using the Bayesian model to identify malicious nodes for exclusion from the cluster selection process. Jena et al. [16] provided a methodology for filtering erroneous event messages produced by the network using event-based reputation. The impact of the node's location on detecting the genuineness of the event is considered in this model. Bhargava et al. proposed a Kalman trust estimator (KATE) in ref. [17]. KATE checks drones' misbehaviour by combining direct and indirect trust values. Kate considers the impact of historical trust values stored on the Internet on current trust values. Carlos et al. in ref. [18] suggested and assessed UAVouch, a

system for identifying and locating UAVs in a group. UAVouch uses a movement plausibility check and a public-key-based authentication system to identify intruders who deviate from the group's anticipated trajectory.

In summary, several trust management solutions for FANETs have been proposed in recent years to enhance network security. Still, none have considered weather conditions' impact on node behaviour and trust computation. Therefore, this paper incorporates humidity as a novel parameter in the proposed study to demonstrate the influence of humidity on both RSSI and trust outcomes.

3 | THE PROPOSED FUBA MODEL

This section presents the network architecture and the detailed proposed FUBA model that considers the effects of climate change and poor signal strength. The major idea is to protect the network from insider attacks and differentiate between legitimate drone actions and malicious activities.

3.1 | Network architecture

FANET comprises three fundamental components: nodes (drones), ground control stations (GCS), and communication links. There are two types of communication, UAV to UAV (U2U) and UAV to Infrastructure (U2I). In FANET, the nodes can leave and join the network anytime, but this flexibility can also make the network vulnerable to attacks. During this, the hacker can compromise a normal drone and convert it into a malicious drone [14]. The intruder participates in the network as a legitimate node and potentially deletes or corrupts messages or damages the reputation of trustworthy nodes. This type of attack, known as an insider attack, can be a significant threat to the security of FANET. Figure 1 shows the network model of FANET operating under the assumption of an insider attack.

As illustrated in Figure 2, the proposed FUBA model is based on four steps: information gathering, trust score calculation, trust aggregation, and decision-making.

3.2 | Information gathering

In the proposed FUBA model, every node collects behaviour information about its neighbours, including their software and hardware performance over time. The fuzzy logic method has been used by Singh et al. [13] with four parameters. The proposed model employs five parameters: signal strength, the drone's energy, packet delivery ratio, transmission delay, and humidity. The parameters collected by each drone are shown in Figure 3.

3.2.1 | Received signal strength indicator (RSSI)

The drone can measure the received signal power of its neighbour at a specific location and time. The number obtained,

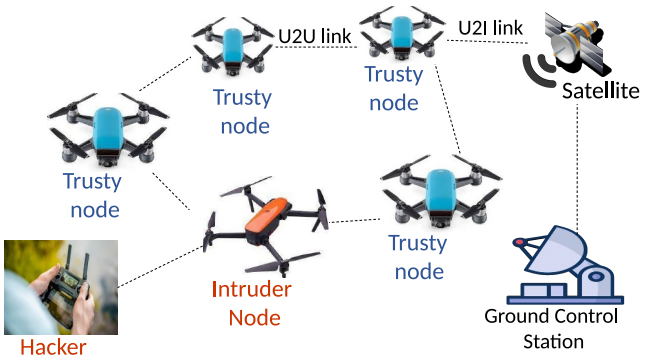


FIGURE 1 FANET model during an Insider attack.

known as the Received Signal Strength Indicator (RSSI) is given in dBm, which is typically a negative value [19]. The typical RSSI for most excellent signal power is greater than −50 dBm (e.g. −30 dBm). Good or acceptable signal power has RSSI ranging from −50 to −70 dBm, (e.g. −60 dBm). Poor signal power has RSSI less than −70 dbm (e.g. −90 dBm) [20].

3.2.2 | Node's energy (battery level)

The node's energy is one of the most critical factors to consider in a drone; therefore, effective power management, including wireless drone charging, solar drone charging, and even the utilisation of artificial intelligence technology [21], are required for the continuity of the applications [22]. The node can accomplish the mission when its battery level exceeds 50%. It can hardly collaborate with its neighbours when the energy level is between 20% and 50%. If the battery level falls below 20%, the node may degrade the network mission [20].

3.2.3 | Packet delivery ratio (PDR)

The packet delivery ratio is the proportion of correctly received packets to the total number of packets transmitted by the sender, represented by Equation (1) [23].

$$PDR = \frac{\sum_{i=0}^{n} ReceivedP_i}{\sum_{i=0}^{n} SentP_i} \quad (1)$$

where $ReceivedP_i$ and $SentP_i$, respectively, denote the number of correctly received packets and the number of packets transmitted by the sender. In ref. [20], it has been illustrated that if the ratio of packets sent effectively is less than 40%, then the PDR is low; if it is between 40% and 70%, then the PDR is medium; and if it is greater than 75%, then the PDR is considered high.

3.2.4 | Transmission delay (TD)

TD represents the drone sender's time to transmit the packets over the link. The following formula of TD is given as follows:

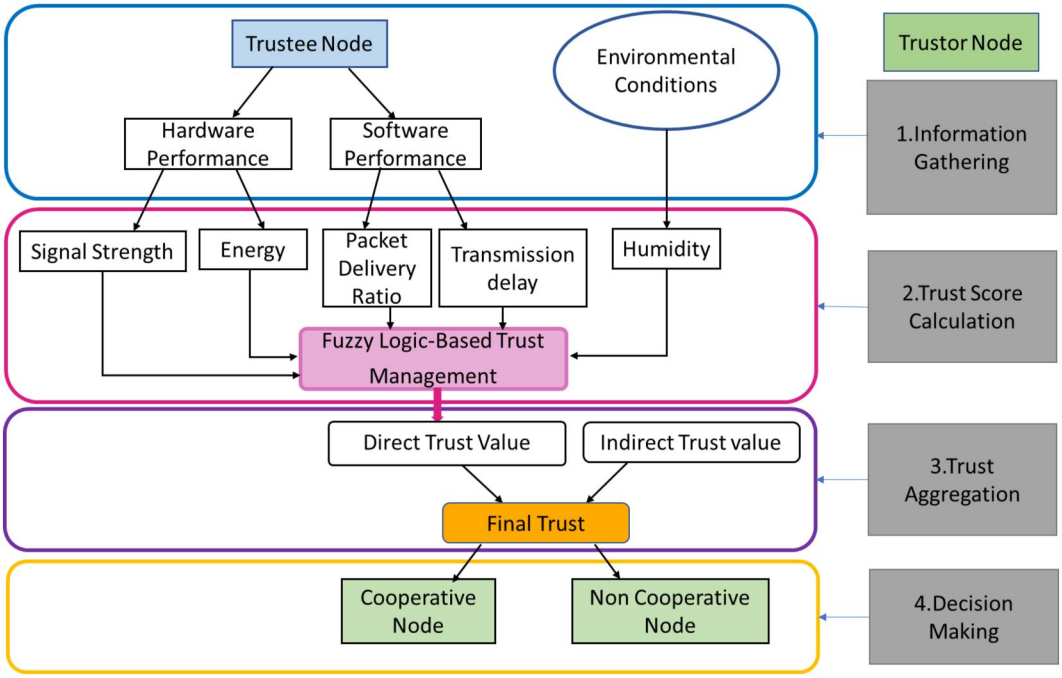


FIGURE 2 FUBA trust model.

FIGURE 3 The collected parameters in FANET.

$$TD = \frac{L}{R} \tag{2}$$

where L is the length of the data packet, and R is the transmission rate (bits per second). The transmission delay is judged small if its value is lower than 0.61 ms; medium if the value is between 0.96 and 1.47 ms; and large if the value exceeds 1.47 ms [20].

3.2.5 | Weather condition

The drone is equipped with many sensors that continuously measure and record information about the current environmental conditions [24], including rain sensors, wind direction sensors, wind speed sensors, air temperature, and humidity sensors. Any change in weather may be sent to the ground control station. If weather conditions significantly impact one of the collected parameters used in the assessment, it becomes challenging to differentiate between legitimate and malicious drone activities or discern intentional from unintentional drone behaviour. To address this issue, the humidity is used as an input parameter in the proposed Fuzzy Logic system.

In this particular context, extensive investigation has been carried out by the authors in ref. [25], focusing on an empirical setup based on IEEE 802.11b/g. The experimentation involves two external radio connections of varying lengths that maintain a continuous data transfer process. The findings indicate that, contrary to expectations, the shorter-distance link is found to be more susceptible to adverse weather conditions. This is attributed to the modulation strategy utilised in that specific scenario. It can be concluded that bad weather conditions may alter the UAV radio signal Propagation.

To analyse the influence of temperature and humidity on RSSI values, the authors in ref. [26] conducted measurements at a constant distance of 25 m under varying weather conditions during summer and winter. The measurement results indicate that the temperature has a relatively minor impact on RSSI compared to humidity because the RSSI values can significantly vary even under similar temperatures. It can be noticed that humidity has a significant influence on RSSI. When the humidity increases, the RSSI values decrease, thereby directly affecting the path loss exponent. It can be concluded that humidity has a greater impact on RSSI than temperature.

3.3 | Trust score calculation

After collecting the necessary information, each drone deploys a fuzzy logic method to calculate its neighbour's trust score.

The proposed system considers various input parameters, including the received signal strength indicator, packet delivery ratio, transmission delay, energy, and humidity. Triangular and trapezoidal membership functions of the input parameters are adopted to enhance the performance. Subsequently, fuzzy rules are employed within the inference engine phase to generate a final numerical value as an outcome. This resulting value signifies the direct trust assessment of the neighbouring node. The configuration of the proposed trust management model based on fuzzy logic is depicted in Figure 4.

3.4 | Trust aggregation

In this phase, the values of α and β are defined to aggregate the direct and indirect trust values. Generally, a FANET is characterised by lower node density and a small link duration between two communicating nodes. The values of α and β are determined based on these two facts, which are used to obtain the needed trust value. Table 1 represents the value of α and β according to the trust state:

a) If the output characteristic value (trust) is "Bad" or "Good," then the confidence factors $\alpha = 1$ and $\beta = 0$. This means that the $Finaltrust(i) = Directtrust(i)$.

b) If the output characteristic value is "Medium," the node requests the recommendations (indirect trust) to its neighbour nodes. Therefore, the final trust computation combines both direct and indirect trust values as shown in Figure 5. The indirect trust is given as follows:

$$IndirectTrust(i) = \frac{1}{n}\sum_{j=1}^{n} DirectTrust(i)_j \qquad (3)$$

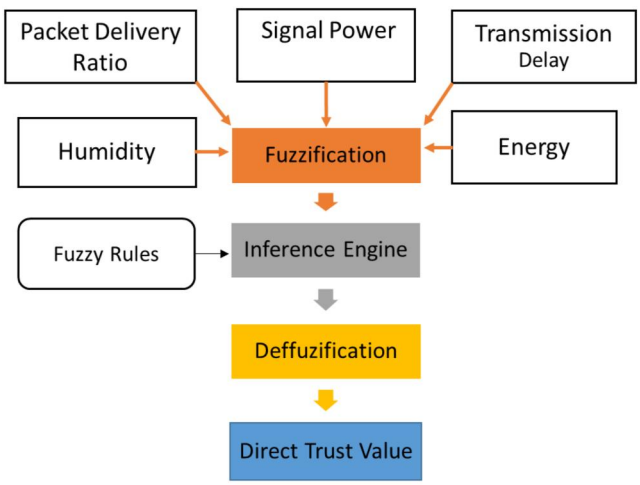


FIGURE 4 Structure of fuzzy system.

TABLE 1 Weight parameters.

Direct trust	α	β	Final trust
Bad/good	1	0	Final Trust(i) = direct trust(i)
Medium	0.5	0.5	Final Trust(i) = 0.5 direct trust(i) + 0.5 indirect trust(i)j

where n represents the total number of drones in the network, (i) is the index of the trustee drone, and (j) is the index of the trustor drone.

3.5 | Decision making

The main objective of the decision-making process is to respond to the following questions.

1. Is it confident to exchange information in the network?
2. Are the nodes interested in cooperating or not?

After analysing a node's behaviour and considering the climate changes, the trustor node estimates the trust score of its neighbours using Fuzzy Logic-based trust management. Subsequently, a threshold-based decision module decides whether to cooperate with the node involved in the considered operation. Specifically, the trust score of each node is then compared to a threshold value to determine if the node is trusted or malicious as follows:

$$\begin{cases} if \quad Final \quad trust \quad > \quad 30\% \quad then \quad Trust \quad node \\ if \quad Final \quad trust \quad \leq \quad 30\% \quad then \quad Malicious \quad node \end{cases}$$

4 | PRACTICAL ASPECTS AND LIMITATIONS OF FUBA

The FUBA system presents an innovative approach to enhancing trust management in FANETs by incorporating humidity as a new parameter. This section explores the generalisability and scalability of the FUBA model under various scenarios while also addressing the model's limitations and practicality.

4.1 | FUBA generalisability, applicability, and scalability

The FUBA is applicable under any weather conditions, as FUBA attempts to adapt its operating parameters according to the surrounding operating environment, which includes the incorporation of fuzzy variables of humidity, namely, Low, Medium, and Large. This adaptability ensures that FUBA's trust assessment remains relevant and effective across various environmental circumstances. Furthermore, the concept of integrating environmental parameters into trust management can be adapted to various contexts, such as agricultural robotics, environmental monitoring, border surveillance, and disaster response. On the other hand, the principles of

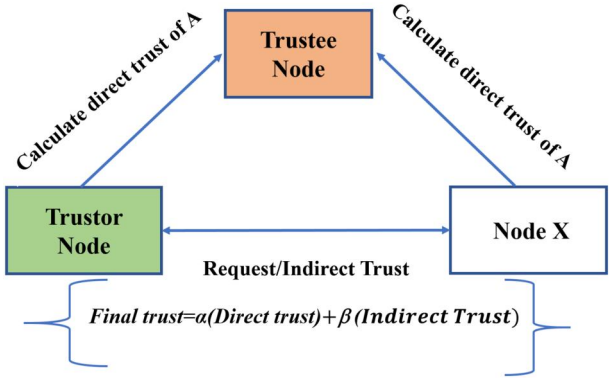


FIGURE 5 Trust aggregation.

behaviour analysis that incorporate environmental parameters could inspire diverse autonomous systems that face complex external conditions, such as wildlife monitoring, pollution detection, and habitat preservation. Finally, ethical considerations related to environmental data collection and the broader social implications of integrating natural parameters into autonomous systems such as airspace congestion, urban environments, or emergency response situations are required.

To tackle FUBA's scalability, every drone in the network evaluates the trustworthiness of its adjacent drones and subsequently relays this information to the ground control station to facilitate decision-making. This approach efficiently restricts the dissemination of trust data and evenly distributes the computational load, enabling the deployment of scalable FUBA. Thus, the scalability of the proposed FUBA model is evident in its ability to accommodate a growing number of drones.

4.2 | FUBA practicality and feasibility

Utilising FUBA in practical settings holds promise in addressing the increasing security challenges associated with drone-related threats. Fuzzy logic can be a valuable tool for identifying and responding to malicious drones, which can be used for unauthorised surveillance, smuggling, or even acts of terrorism. In what follows, we explore the implementation process of the FUBA model and its practical implications in identifying malicious drones.

- The proposed FUBA can identify anomalies in drone behaviour by comparing the detected drone's actions to predefined models of normal drone behaviour. If a drone's actions deviate significantly from the expected behaviour, the system can raise an alert and initiate appropriate response measures.
- FUBA can incorporate contextual information, such as local regulations, flight restrictions, and historical data, to make more accurate decisions about the legitimacy of a drone's presence. This ensures that harmless drones, such as hobbyists or commercial drones, are not mistakenly flagged as malicious.
- FUBA model can continuously learn from new data and adjust its rules and inference mechanisms to adapt to evolving tactics used by malicious drones.
- FUBA system can be integrated with existing aviation and security infrastructure, such as air traffic control systems, airport security, and critical infrastructure protection, to improve overall airspace security.
- FUBA can provide real-time monitoring and reporting of drone activities, helping security personnel make timely and informed decisions to mitigate potential threats.

4.3 | FUBA practical limitations

While FUBA demonstrates effective trust management in FANET under poor weather conditions, the proposed approach has the following main limitations:

- As the number of rules and fuzzy sets increases to model a large problem space, fuzzy rule bases can become very complex and difficult to manage. This affects issues such as debugging, updating, and interpretability.
- Fuzzy systems are only as good as the input features they are provided. Critical security features may be missing or noisy, limiting detection capabilities. Furthermore, security considerations around trust data exchange must be incorporated into a full system deployment.

In summary, the proposed model offers a practical and effective approach to improving security in the real world. By leveraging its ability to handle uncertain and imprecise data, the FUBA model can contribute to the development of robust and adaptive systems that safeguard against the misuse of drones for malicious purposes.

5 | IMPLEMENTATION DETAILS OF THE FUBA MODEL

Fuzzy logic is a computational approach that handles uncertain information by allowing for degrees of truth rather than rigid binary values. This section uses MATLAB to evaluate the proposed FUBA model. The fuzzy logic used to evaluate node behaviour comprises three steps: fuzzification, inference engine, and defuzzification.

5.1 | Step 1: Fuzzification

In this step, a membership function is generated to determine the degree to which the numerical data correspond to a linguistic variable, using triangular and trapezoidal functions presented in Figures 6 and 7. Typically, a triangular membership function is defined using three parameters, namely, a, b, and c, as follows:

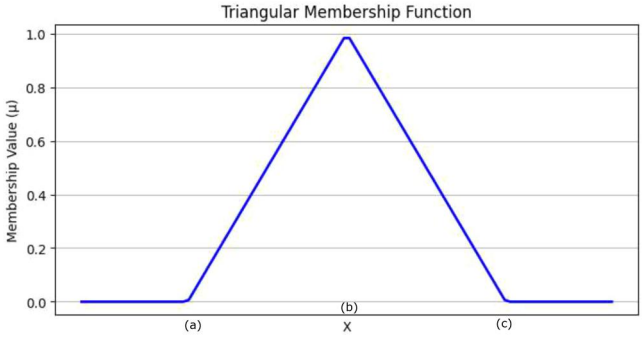


FIGURE 6 Triangular membership function.

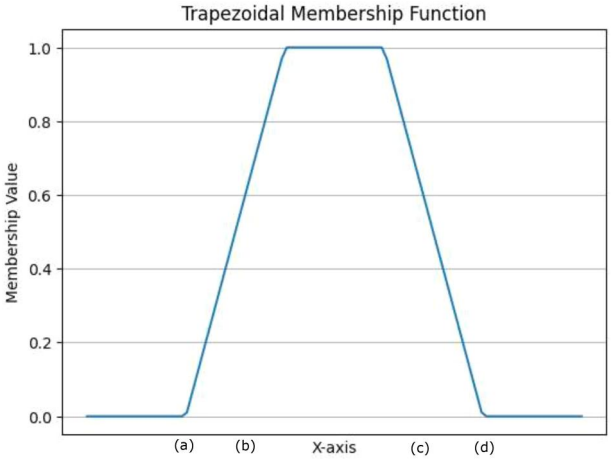


FIGURE 7 Trapezoidal membership function.

$$f(x,a,b,c) = \begin{cases} 0 & x \le a \\ \dfrac{x-a}{b-a} & a \le x \le b \\ \dfrac{c-x}{c-b} & b \le x \le c \\ 0 & c \le x \end{cases} \tag{4}$$

The expression given in Equation (4) can be written in a simple form using min and max functions as follows:

$$F(x,a,b,c) = \max\left(\min\left(\frac{x-a}{b-a}\right), \left(\frac{c-x}{c-b}\right), 0\right) \tag{5}$$

$$F(x,a,b,c,d) = \max\left(\min\left(\frac{x-a}{b-a}\right), 1, \left(\frac{d-x}{d-c}\right), 0\right) \tag{6}$$

Figure 8a illustrates the membership functions of Energy: A triangular membership function for the linguistic variable *Medium* is defined by the triangle (x, 0.2, 0.35, 0.5). The trapezoidal membership function for the linguistic variable *High* is defined by the trapezoid (x, 0.5, 0.85, 1, d).

As shown in Figure 8a, the fuzzy variables of the energy are *VeryLow*, *Medium*, and *High* and are analysed from 0 to 1. Figure 8b shows the fuzzy variables for the packet delivery ratio, which are *Low*, *Medium*, and *High*, analysed from 0 to 1. Figure 8c shows that the fuzzy variables of the signal power are *Poor*, *Good*, and *Excellent*. They are analysed from −100 to −10 dBm. Figure 8d shows the fuzzy variables of the transmission delay: *Small*, *Medium*, and *Large*. They are analysed from 0.6 to 2.4 ms. In Figure 8e, the fuzzy variables of humidity (*Low*, *Medium*, and *Large*) are analysed from 0 to 1. Figure 8f illustrates the output trust fuzzy variables that are *Bad*, *Medium*, and *Good* and are analysed from 0 to 100.

5.2 | Step 2: The inference engine

In this step, all the rules need to be defined in the proposed fuzzy logic model and then explain those that reflect realistic situations:

The first rule illustrates the worst-case scenario. While the second rule represents the best case. The third rule requires the system to consider the node as trustworthy due to its low battery, implying that unintentional misbehaviour is considered.

Rules 4 through 7 state that if all variables have low values except for one that has a positive value. Then, the node is considered untrustworthy with a bad trust value.

Rule 8 requires the system to consider the node as trustworthy because it has a weak RSSI due to the high humidity. This implies that the system takes into account unfavourable weather conditions. Table 2 illustrates the rules when humidity is low, while Table 3 shows the rules when humidity is high.

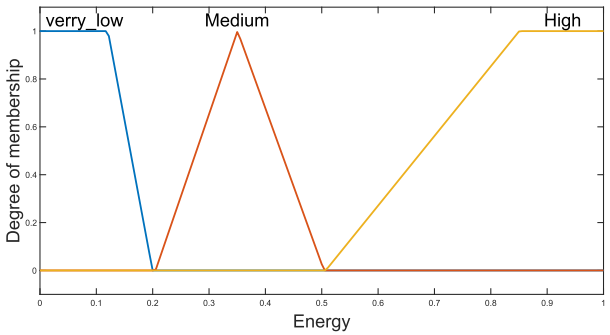
5.3 | Step 3: Defuzzification

Defuzzification is the pivotal stage within the fuzzy logic process, where the crisp output is derived from the fuzzy output generated by the fuzzy inference engine. This involves translating the fuzzy set or linguistic term (*Bad*, *Medium*, and *Good*) into a single, definite value that can be understood and utilised for drone behaviour evaluation. Various methods, such as centre of gravity, bisector, and maxima, can be employed for defuzzification to convert the fuzzy output into a clear and actionable result [27]. In the proposed model, the centroid method (COG) is considered, which is the most widely used technique and is depicted in Figure 9.

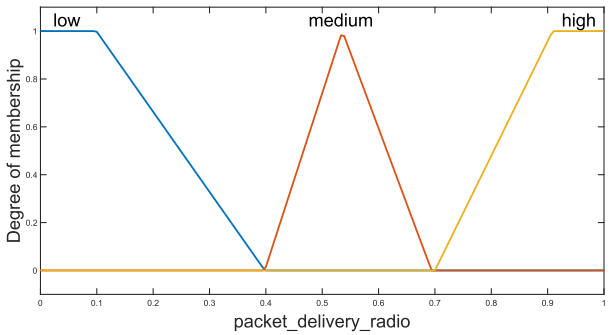
This method involves determining the centre of gravity of the obtained polygon:

$$CG = \frac{\sum_x^b = af(x) \times x}{\sum_x^b = af(x)} \tag{7}$$

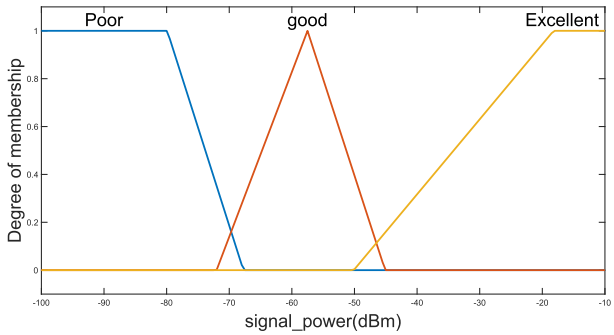
where $f(x)$ represents the aggregation of the membership functions while a and b represent the bounds of the obtained polygon.



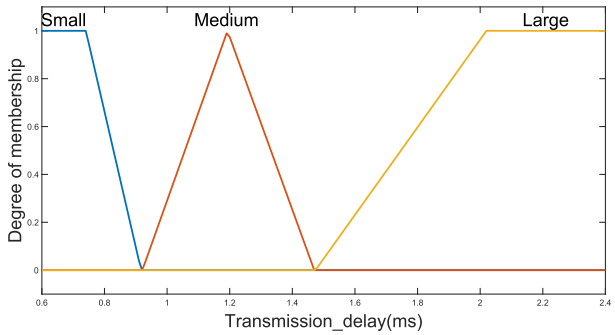
(a) Membership Functions of Energy



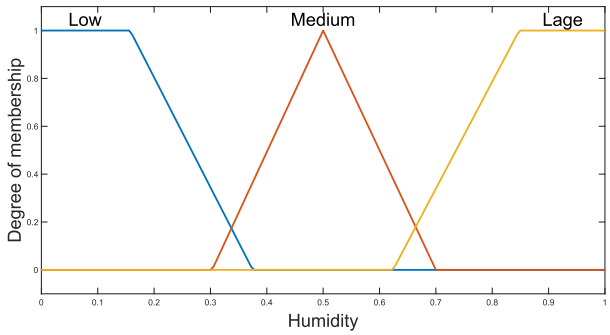
(b) Membership Functions of Packet Delivery Ratio



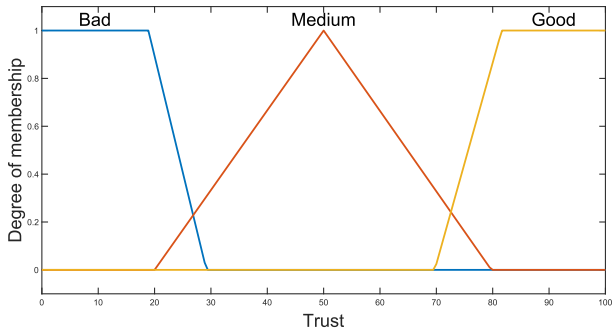
(c) Membership Functions of Signal Power



(d) Membership Functions of Transmission delay



(e) Membership Functions of Humidity



(f) Membership Functions of Trust Score

FIGURE 8 Membership functions of the different parameters.

TABLE 2 Fuzzy rules with low humidity.

R	RSSI	PDR	Energy	TD	Output
1	Poor	Low	Very low	Large	Bad
2	Excellent	High	High	Small	Good
3	Excellent	High	Very low	Small	Good
4	Excellent	Low	Very low	Large	Bad
5	Poor	Large	Very low	Large	Bad
6	Poor	Low	High	Large	Bad
7	Poor	Low	Very low	Small	Bad

This method calculates the output by determining the abscissa of the centroid located beneath the curve's surface. The selection of the defuzzification method exerts a significant impact on the final result of the fuzzy logic model. The centre of gravity method is more flexible, as it considers the entire fuzzy output (trust) to calculate the trust result.

The functions that determine the membership of the input and output parameters must be adjusted for each iteration of the fuzzy rule base [28]. The cut-off method is depicted in Figure 10.

6 | IMPACT OF RSSI AND HUMIDITY ON TRUST RESULT

In this section, MATLAB programs are used to evaluate the performance of the proposed FUBA model. The fuzzy logic application is used for evaluating and understanding the node

behaviour under the impact of bad weather conditions and poor signal strength (RSSI).

6.1 | Impact of RSSI on trust result

The bar chart in Figure 11 illustrates the trust result of 8 nodes in the network under high and low RSSI while varying the other parameters (TD, Energy, PDR) to obtain several trust values: *high*, *medium*, and *low*. The trust value dropped from 87% to 50% in nodes 1 and 3; the trust value decreased from 52% to 12% in nodes 4, 6, and 7. This figure shows that the trust level in nodes 2, 5, and 8 remained constant. However, the proportion of trust increases significantly when the RSSI is excellent. There are several possible explanations for this result, but it is essential to note that signal power (RSSI) plays a vital role in assessing drone performance in FANET. It can be concluded that trust values decrease when signal power decreases due to high humidity. Consequently, it is advisable to eliminate the node from the network to enhance network security.

TABLE 3 Fuzzy rules with high humidity.

R	RSSI	PDR	Energy	TD	Output
1	Poor	Low	Very low	Large	Bad
2	Excellent	High	High	Small	Good
3	Excellent	High	Very low	Small	Good
4	Excellent	Low	Very low	Large	Bad
5	Poor	Large	Very low	Large	Bad
6	Poor	Low	High	Large	Bad
7	Poor	Low	Very low	Small	Bad
8	Poor	High	High	Small	Good

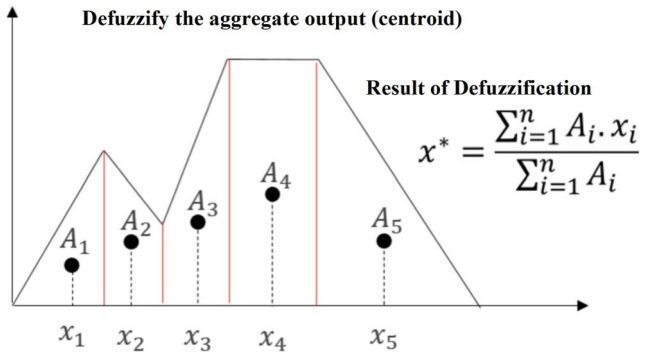


FIGURE 9 COG method.

6.2 | Impacts of humidity on trust result

Figure 12 illustrates the trust result for a network with 16 drones under high and low humidity while varying the other parameters (TD, Energy, PDR, RSSI) to obtain *high*, *medium*, and *low* trust values. The trust value reduced from 87.6% to 50.6% in node 9 and from 51.2% to 12.6% in node 12, then the trust values for the remaining nodes remained constant.

The most notable conclusion that can be drawn from Figure 12 is that the humidity significantly impacts the trust results in FANET. For this reason, climate change should be considered when designing a trust management system in FANET.

7 | EXPERIMENTAL RESULT AND DISCUSSION

7.1 | Simulation setup

The effectiveness of the newly introduced FUBA system is assessed through the following communication frameworks:

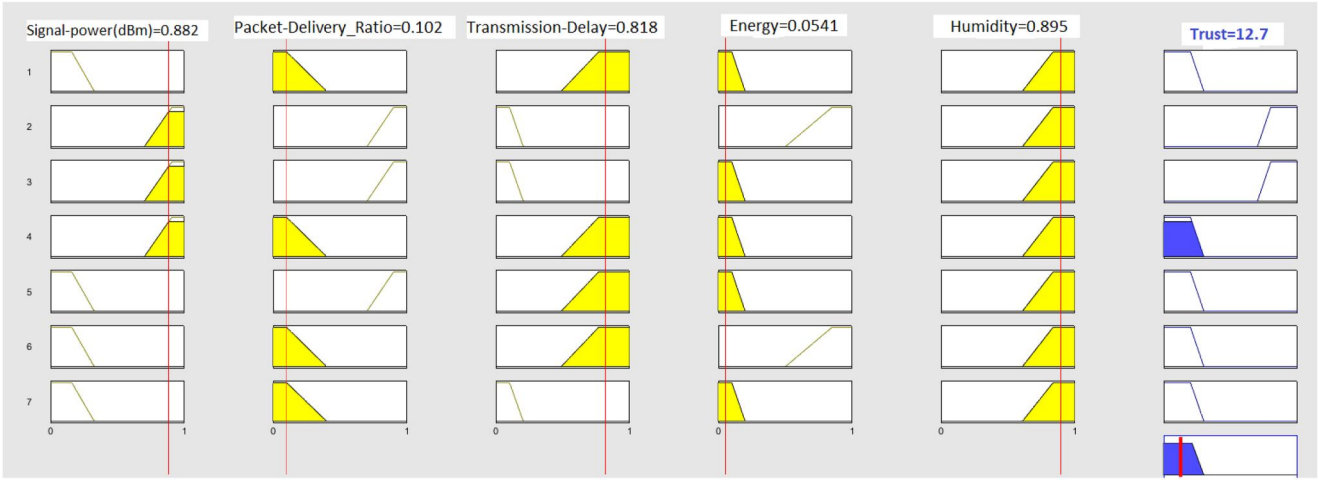


FIGURE 10 Cut-off method to combine the rules.

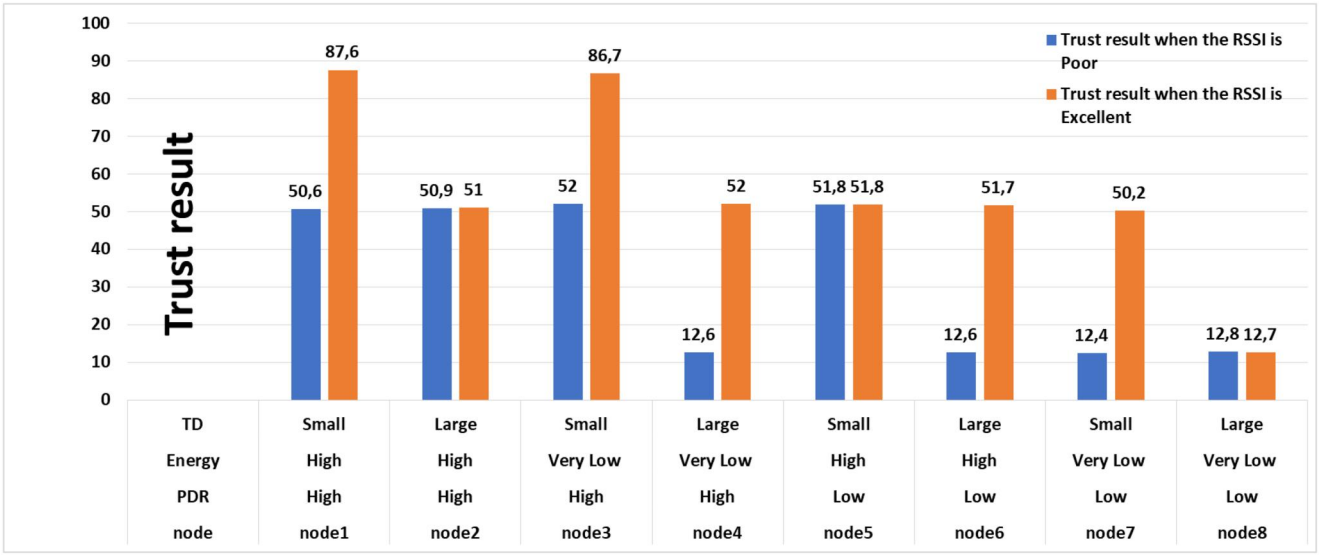


FIGURE 11 The impact of RSSI on trust result.

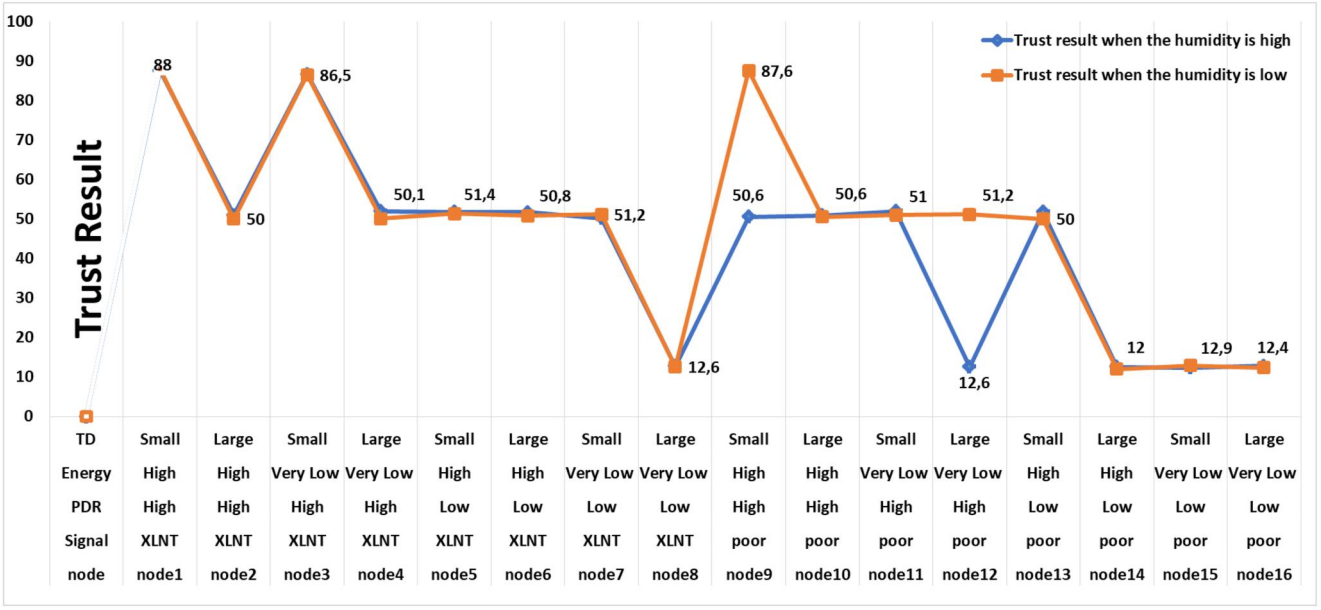


FIGURE 12 The impact of humidity on trust results.

- **OMNeT++**: It is a powerful simulation library and framework designed for building and testing complex communication networks. It is built using C++ and is highly extensible, modular, and component-based, making it a popular choice for researchers and engineers in the field [29].
- **INET Framework**: The INET Framework is integrated with OMNeT++ and provides a rich set of network models and protocols, facilitating realistic simulations of communication scenarios [30].
- **AVENS (Aerial Vehicle Network Simulator):** AVENS is designed primarily to establish a simulation testing environment that is specifically designed to conduct virtual experiments focused on evaluating the network coverage and interconnectivity of UAVs engaged in collaborative flights or coexisting within the same airspace. The integration strategy for AVENS revolves around incorporating both the XPlane Flight Simulator and the OMNeT++ network simulator [31].
- **XPlane Flight Simulator:** The XPlane Flight Simulator significantly contributes to the authenticity of the simulations by enabling the modelling of real-world flight dynamics and interactions among UAVs [32].

The simulated network uses the IEEE 802.11 communication protocol for wireless interactions among UAVs. The

network area is defined as a space of 2500 m × 2500 m, accurately reflecting real-world operational conditions. To ensure a comprehensive evaluation, the simulation time is set to 3000 s, enabling observation of network behaviour and performance over an extended duration. The simulations are executed on a 64-bit PC running Windows 10. This platform offers the necessary computational resources to conduct simulations, thus effectively ensuring reliable and precise results. Table 4 summarises the different simulation parameters used in the simulation experiments.

In the simulation experiment, a scenario is designed to emulate a UAV communication network with the proposed FUBA system. The scenario initiates with 10 UAVs and a ground control station. The UAVs collaborate to share a wireless communication medium within the AVENS simulation framework. During the simulation, UAVs exchange messages and collect critical network parameters, including transmission delay, received signal strength indicator (RSSI), packet delivery ratio, and node energy. To assess scalability and performance, the number of UAVs is systematically increased from 10 to 200 in steps. The increments are chosen to comprehensively understand the proposed method's behaviour across a wide range of UAV quantities. In OMNET ++, the recording module is configured to track end-to-end delay across the network by specifying the appropriate recording intervals and enabling scalar data collection.

TABLE 4 Simulation parameters.

Simulation tools	OMNET++, Avens, Xplane10
Simulation area	2500 m × 2500 m
Node counts	10–200
Ping-transmission interval	10 s
Ping-sleep period	10 s
UDP-transmission interval	10 ms
UDP-packet size	1000 B
UDP application type name	UdP video stream SVR
UDP application video size	10 MIB
MAC address assignment	Auto
Ip process delay	10 μs
Mac Queue size	14
WLAN data rate	2 MIB
Transmission frequency	2400 Hz
Physical Tx power	100 mW
Power generation	100 MW
Simple energy storage	0.05 J
Energy generator sleep interval	Exponential (10 s)
Mobility model	Random way-point
Ground control station mobility	Stationary mobility
Mobility update rate	2 s
Wireless standard	IEEE 802.11
Simulation time	3000 s
Operating platform	64-bit Windows 10

7.2 | Simulation results

To assess the performance of FUBA, the well-established FNDN [10] and UNION [8] models are utilised as reference points. These models provide a baseline for comparison based on their inherent characteristics. Specifically, the conducted simulations focus on two key parameters: false positive rate and end-to-end delay. The false positive rate quantifies instances where the system incorrectly identifies trustworthy nodes as untrustworthy [33]. In the context of the simulation experiments, several instances of false positives relate to situations in which the FUBA system erroneously categorises a drone as a regular node despite not meeting the criteria for such classification. Furthermore, the end-to-end delay is analysed, which reflects the time taken for the data to travel from the source to the destination node in the network [34]. In the context of the conducted simulations, the end-to-end delay could be measured as the time it takes for a message or packet to be transmitted from one UAV (source) to another UAV or the ground control station (destination). It can be a critical metric for assessing real-time communication performance.

The FNDN [10] is a recent monitor-based communication architecture that uses both direct and indirect trusts for Flying Named Data Networking. Nevertheless, the UNION [8] model considers the UAV energy, mobility patterns, and enqueued packets while employing both direct and indirect trust to assess node behaviour. Comparing the proposed trust model with these two trust models is an essential step in evaluating FUBA's effectiveness and practicality.

Based on Figure 13, it can be observed that the proposed FUBA model has a significant impact on reducing the average end-to-end delay of data packets in comparison to the UNION model in high-density scenarios. Specifically, when there are 50 drones, the FUBA model reduces the delay by more than 1.4 s, unlike UNION, and in large-density scenarios with 100 drones, the enhancement is roughly 1.1 s. When the number of nodes exceeds 150, the mean end-to-end delay for FNDN and the proposed solution is nearly the same. The figure shows that the proposed FUBA model consistently results in the lowest end-to-end delay across the three models.

The false positive ratio for FUBA, FNDN and UNION as a function of the density of the UAV is shown in Figure 14. The false positive can be obtained by calculating the trusted node using the FUZZY logic application if a node (i) is not compromised. The graph curves show that for both FNDN and UNION, the calculated false positive steadily increases; however, at the beginning of the simulation experiments, no false positive instances were generated during this process.

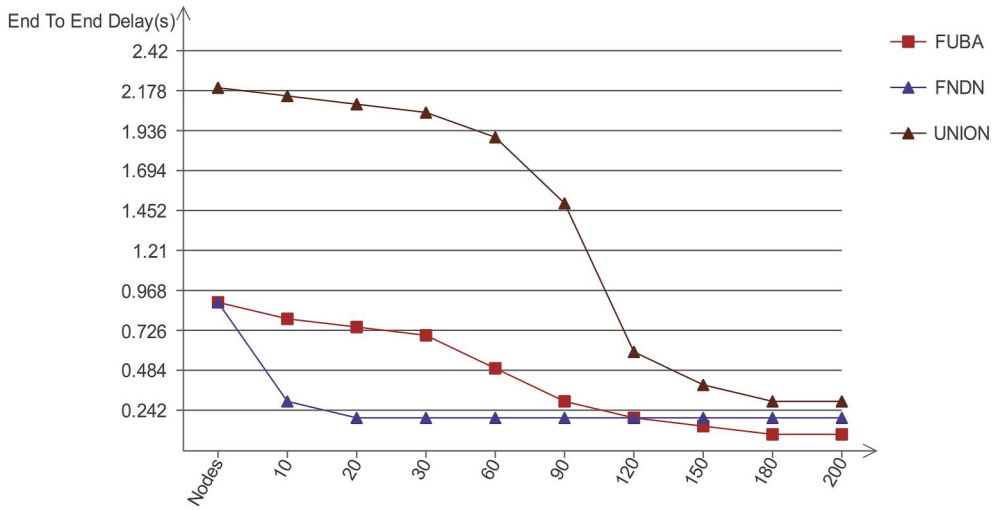


FIGURE 13 The average end-to-end delay versus the number of nodes.

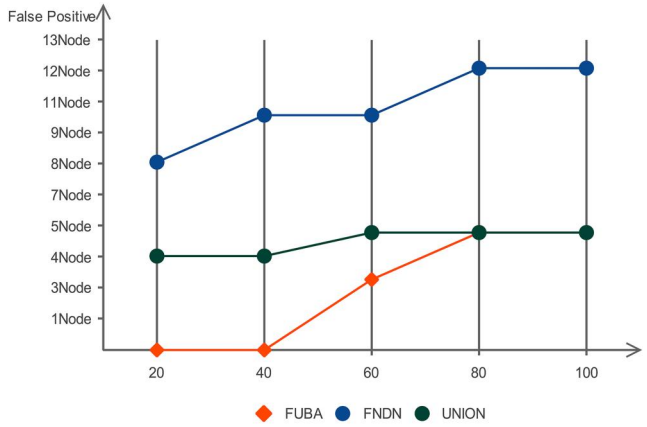


FIGURE 14 Number of false positives compared to FNDN and UNION.

Therefore, comparing the proposed model with FNDN and UNION, the proposed solution has a lower error ratio.

8 | CONCLUSIONS AND FUTURE WORK

Trust management is an effective method of detecting insider threats. The main challenge in this domain is designing a conceptual and analytical trust model for FANET that can evaluate and understand the behaviour of nodes. In the context of this article, FUBA, a Fuzzy-based UAV behaviour analytics for trust management based on direct and indirect information was introduced. Contrary to previous models, the proposed model increases the trustworthiness of the network in bad weather conditions and poor signal strength (RSSI). Furthermore, FUBA has the ability to effectively distinguish between legitimate drone actions and malicious ones. In future work, it would be valuable to incorporate machine learning (ML) and blockchain technology to enhance FUBA's capabilities. Automated tuning of the fuzzy logic rules and membership functions via machine learning may improve performance across diverse operating environments. Future research could explore more sophisticated algorithms, such as deep learning and reinforcement learning, to extract deeper insights from complex FANET data. Furthermore, federated learning presents an exciting opportunity for FANETs, where data privacy is of paramount importance. On the other hand, blockchain-based distributed ledgers could secure the sharing of trust data while providing resilience against compromised nodes. Furthermore, it can facilitate transparent and auditable trust records, reducing the reliance on centralised authorities. As these technologies continue to evolve, their integration offers the potential to create more resilient, secure, and efficient FANETs, shaping the future of aerial communication and navigation.

AUTHOR CONTRIBUTIONS

Sihem Benfriha: Conceptualization; data curation; formal analysis; investigation; methodology; project administration; resources; software; validation; visualization; writing – original draft; writing – review & editing. **Nabila Labraoui**: Conceptualization; formal analysis; investigation; project administration; supervision; validation; writing – original draft; writing – review & editing. **Radjaa Bensaid**: Formal analysis; methodology; resources. **Haythem Bany Salameh**: Formal analysis; funding acquisition; investigation; writing – review & editing. **Hafida Saidi**: Formal analysis; writing – review & editing.

CONFLICT OF INTEREST STATEMENT

We have no conflict of interest with anyone on the IET Networks Journal staff. We state that the paper is original and will not be submitted elsewhere until a decision is made by the IET Networks Journal.

DATA AVAILABILITY STATEMENT

Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

PERMISSION TO REPRODUCE MATERIALS FROM OTHER SOURCES

None.

ORCID

Sihem Benfriha https://orcid.org/0000-0002-4669-6053

REFERENCES

1. Saidi, H., et al.: DSMAC: privacy-aware decentralized self-management of data access control based on blockchain for health data. IEEE Access. 10, 101011–101028 (2022). https://doi.org/10.1109/access.2022.3207803
2. Fang, Z., et al.: Age of information in energy harvesting aided massive multiple access networks. IEEE J. Sel. Area. Commun. 40(5), 1441–1456 (2022). https://doi.org/10.1109/jsac.2022.3143252
3. Benfriha, S., Labraoui, N.: Insiders detection in the uncertain IoD using fuzzy logic. In: Proceedings of the International Conference on Information Technology (ACIT), pp. 1–6. IEEE (2022)
4. Fotouhi, A., et al.: Survey on UAV cellular communications: practical aspects, standardization advancements, regulation, and security challenges. IEEE Commun. Surv. Tutorial. 21(4), 3417–3442 (2019). https://doi.org/10.1109/comst.2019.2906228
5. Yuan, F., et al.: Insider threat detection with deep neural network. In: Proceedings of the International Conference in Computational Science–ICCS 2018, Wuxi, China, June 11–13, Part I 18, pp. 43–54. Springer International Publishing (2018)
6. Labraoui, N., Gueroui, M., Sekhri, L.: A risk-aware reputation-based trust management in wireless sensor networks. Wireless Pers. Commun. 87(3), 1037–1055 (2016). https://doi.org/10.1007/s11277-015-2636-3
7. Wu, G., et al.: A fuzzy-based trust management in WSNs. J. Internet Serv. Inf. Secur. 3(3/4), 124–135 (2013)
8. Barka, E., et al.: UNION: a trust model distinguishing intentional and UNIntentional misbehavior in inter-UAV communication. J. Adv. Transport. 2018, 1–12 (2018). https://doi.org/10.1155/2018/7475357
9. Kerrache, C.A., et al.: September. Reputation-aware energy-efficient solution for FANET monitoring. In: Proceedings of the International Conference on IFIP Wireless and Mobile Networking Conference (WMNC), pp. 1–6. IEEE (2017)
10. Barka, E., et al.: A trusted lightweight communication strategy for flying named data networking. Sensors. 18(8), 2683 (2018). https://doi.org/10.3390/s18082683
11. Barka, E., et al.: Towards a trusted unmanned aerial system using blockchain for the protection of critical infrastructure. Trans. Emerg. Telecommun. Technol. 33(8), e3706 (2022). https://doi.org/10.1002/ett.3706
12. Singh, K., Verma, A.K.: A fuzzy-based trust model for flying ad hoc networks (FANETs). Int. J. Commun. Syst. 31(6), e3517 (2018). https://doi.org/10.1002/dac.3517
13. Singh, K., Verma, A.K.: A trust model for effective cooperation in flying ad hoc networks using genetic algorithm. In: Proceedings of Interantional Conference Communication and Signal Processing (ICCSP), pp. 491–495. IEEE (2018)
14. Singh, K. and Verma, A.K.: TBCS: a trust-based clustering scheme for secure communication in flying ad-hoc networks. Wireless Pers. Commun., 202(4), 3173–3196 (2020). https://doi.org/10.1007/s11277-020-07523-8
15. Zhou, J., Wang, Z.: Security clustering algorithm based on integrated trust value for unmanned aerial vehicles network. KSII Trans. Internet Inf. Syst. (TIIS). 14(4), 1773–1795 (2020)
16. Jena, K.K., et al.: A trust based false message detection model for multi-unmanned aerial vehicle network. In: Proceedings of the International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 324–329. IEEE (2019)
17. Bhargava, A., Verma, S.: Kate: Kalman trust estimator for internet of drones. Comput. Commun. 160, 388–401 (2020). https://doi.org/10.1016/j.comcom.2020.04.027
18. DeMelo, C.F., et al.: UAVouch: a secure identity and location validation scheme for UAV networks. IEEE Access. 9, 82930–82946 (2021). https://doi.org/10.1109/access.2021.3087084
19. Chen, Y., Yang, J.: Defending against identity-based attacks in wireless networks. In: Handbook on Securing Cyber-Physical Critical Infrastructure, pp. 191–222. Elsevier Inc (2012)
20. Singh, K., Verma, A.K.: FCTM: a novel fuzzy classification trust model for enhancing reliability in flying ad hoc networks (FANETs). Ad Hoc Sens. Wirel. Networks. 40(1-2), 23–47 (2018)
21. Ji, Y., et al.: Efficiency-boosting federated learning in wireless networks: a long-term perspective. IEEE Trans. Veh. Technol. 72(7), 9434–9447 (2023). https://doi.org/10.1109/tvt.2023.3250273
22. Ayass, T., et al.: Unmanned aerial vehicle with handover management fuzzy system for 5G networks: challenges and perspectives. Intell. Robot. 2(1), 20–36 (2022). https://doi.org/10.20517/ir.2021.07
23. Khan, M.K.U., Ramesh, K.S.: Effect on packet delivery ratio (PDR) & throughput in wireless sensor networks due to black hole attack. Int. J. Innov. Technol. Explor. Eng. 8(12S), 428–432 (2019)
24. Wang, J., et al.: Starling flocks-inspired resource allocation for ISAC-aided green ad hoc networks. IEEE Trans. Green Commun. Netw. 7(1), 444–454 (2023). https://doi.org/10.1109/tgcn.2023.3234165
25. Bri, D., et al.: Performance analysis of weather's impact on outdoor IEEE 802.11 b/g links using network management parameters. Mobile Network. Appl. 21(4), 603–619 (2016). https://doi.org/10.1007/s11036-016-0758-9
26. Kurt, S., Tavli, B.: Path-loss modeling for wireless sensor networks: a review of models and comparative evaluations. IEEE Antenn. Propag. Mag. 59(1), 18–37 (2017). https://doi.org/10.1109/map.2016.2630035
27. Rao, D.H., Saraf, S.S.: Study of defuzzification methods of fuzzy logic controller for speed control of a DC motor. In: Proceedings of the International Conference on Power Electronics, Drives and Energy Systems for Industrial Growth, Vol. 2, pp. 782–787. IEEE (1996)
28. Thangaraj, K., Dharma, D.: Optimized fuzzy system dependent trust score for mobile AdHoc network. Wireless Pers. Commun. 117(4), 3255–3269 (2021). https://doi.org/10.1007/s11277-020-07984-x
29. Gill, J.S., et al.: Simulation testbeds and frameworks for UAV performance evaluation. In: Proceedings of the International Conference on Electro Information Technology (IT), pp. 335–341. IEEE (2021)
30. Virdis, A., Kirsche, M.: Recent advances in network simulation. EAI/Springer Innovations in Communication and Computing (2019)
31. Marconato, E.A., et al.: Avens-a Novel Flying Ad Hoc Network Simulator with Automatic Code Generation for Unmanned Aircraft System (2017)
32. Garcia, R., Barnes, L.: Multi-UAV simulator utilizing X-plane. In: Selected Papers from the 2nd International Symposium on UAVs, Reno, Nevada, pp. 393–406. Springer Netherlands (2009–2010)
33. Marugán, A.P., Chacón, A.M., Márquez, F.P.: Reliability analysis of detecting false alarms that employ neural networks: a real case study on wind turbines. Reliab. Eng. Syst. Saf. 191, 106574 (2019). https://doi.org/10.1016/j.ress.2019.106574
34. Alaslani, M., Nawab, F., Shihada, B.: Blockchain in IoT systems: end-to-end delay evaluation. IEEE Internet Things J. 6(5), 8332–8344 (2019). https://doi.org/10.1109/jiot.2019.2917226

How to cite this article: Benfriha, S., et al.: FUBA: a fuzzy-based unmanned aerial vehicle behaviour analytics for trust management in flying ad-hoc networks. IET Netw. 1–13 (2023). https://doi.org/10.1049/ntw2.12108